Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informačních technologií

Extrakce témat z nestrukturovaného textu

Diplomová práce

Autor: David Illner
Studijní obor: Informační management

Vedoucí práce: Ing. Martina Husáková, Ph.D.

Hradec Králové Duben 2024

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 18.04.2024

Poděkování:
Děkuji vedoucí diplomové práce paní Ing. Martině Husákové, Ph.D., za její čas, metodické vedení práce, trpělivost a pomoc při výběru tématu i jeho samotném zpracování.

Anotace

Předmětem diplomové práce je vytvoření modelu pro extrakci témat z nestrukturovaného textu. Byla využita platforma KNIME a veřejně dostupný dataset, který obsahuje pouze abstrakty vědeckých článků. V teoretické části práce je popsána analýza přirozené řeči a vysvětlena důležitost předzpracování textu, aby bylo nadále možné s přirozenou řečí pracovat v prostředí informačních technologií. Pro předzpracování textu jsou použity metody odstranění stop slov, stemming a tokenizace. Pro extrakci témat je použita latentní Dirichletova alokace. Výsledky jsou zobrazeny graficky ve word cloud grafu. Model je poté upraven pro dosažení nejlepších výsledků a následně ohodnocen s využitím sémantické koherence. Průběh vytvoření modelu je detailně popsán včetně všech použitých nastavení v platformě KNIME.

Klíčová slova: Extrakce témat, latentní direchletova alokace, LDA, KNIME

Annotation

Title: Extraction of topics from unstructured text

The subject of this master thesis is to create a model for topic extraction from unstructured text using the KNIME platform and a publicly available dataset containing only abstracts of scientific articles. The analysis of natural language and the importance of pre-processing text with this language to be able to continue working with it in an information technology environment. Word stemming, stemming and tokenization methods are used for text preprocessing. For topic extraction Latent Dirichlet allocation is used. The results are displayed graphically in word cloud graph. The model is then adjusted for best results, and finally it is then ranked according to semantic coherence. This model creation process is described in detail including all the settings used in the KNIME platform.

Key words: Topic extraction, latent direchlet allocation, LDA, KNIME

Obsah

1 Úvod

Extrakce témat je jednou z klíčových metod pro třídění textových informací, kterých je dnes plný internet. Pro lepší přehled, navigaci a třídění těchto textů je třeba toto téma posouvat dále, neboť skrze lidskou práci není možné zvládnout zpracování takového množství informací, které se denně nahrávají na internet. Jak je patrné, účelem této metody je ulehčení práce.

První zmínky o extrakci témat lze vystopovat již od roku 1990, kdy se objevil předek dnešních metod pro extrakci nebo detekci témat. Úvodní kapitoly teoretické části práce jsou zaměřeny na přirozenou lidskou řeč a její analýzu. Je důležité ukázat, že přirozená řeč obsahuje velkou míru různých nuancí a detailů, které jsou sice pro dospělého člověka triviální, ale pokud by počítač pracoval jen s touto formou přirozené řeči, nedospěl by k žádným výsledkům. [1]

Na tuto část navazuje problematika předzpracování textu. Tímto způsobem je přirozená řeč upravena do podoby, se kterou si počítač už dokáže poradit. Dále je zde popsána i metoda zvaná latentní Dirichletova alokace (LDA), která je již zaběhlým standardem v daném okruhu.

Po vysvětlení všech důležitých informací následuje praktická část, která je vytvořena v open-source platformě KNIME. Je zde vytvořen model pro extrakci témat podle předem popsaných metod a popsáno i následné upravování modelu, aby byly zajištěny co nejlepší výsledky. Model je pak nadále vyhodnocen. V závěru jsou shrnuty výsledky a navržena doporučení, která by mohla při zpracování zlepšit výsledky.

2 Cíl práce

Cílem této diplomové práce je vytvoření modelu pro extrakce témat z nestrukturovaného textu. Praktický přiklad, který je představen ve druhé části práce, popisuje vytvoření modelu v platformě KNIME s pomocí metod a způsobů určených k předzpracování textu spolu s metodami pro získávání informací z textu.

Je představena analýza textu, kde je poukázáno na přirozenou řeč, kterou počítač nedokáže zpracovat. Dále je pozornost věnována metodám pro zpracování textu. Základní metodou pro extrakci témat je latentní Dirichletova alokace, která je zde vysvětlena a také použita v praktické části. Každá důležitá metoda pro zpracování textu je detailně vysvětlena ve své vlastní kapitole.

Po této části následuje praktická část, v níž jsou teoretické poznatky využity. V praktické části je popsán a ilustrován průběh vytvoření modelu pro extrakci témat. Následně je model upravován a dále opakovaně hodnocen podle jeho výsledků.

3 Základy textové analýzy

Textová analýza vznikla už v začátcích 13. století, kdy dominikánských mnich Hugh z kláštera svaté Cher s ostatními mnichy vytvořili první konkordanci bible. Konkordancí se v literatuře rozumí abecední seznam slov a odkazy na stránky, kde tato slova lze v knize najít. Další náznaky se objevily v Evropě v 17. století, kdy probíhalo inkviziční studování novin. Poté pak na začátku 18. století, kdy vznikla první kvantitativní studie, v níž církev sledovala symboliku a ideologické obsahy písní, které nebyly v souladu morálkou a hodnotami tehdejší církve. [1]

Rozmach textové analýzy nastal až ve 20. století, kdy vědci ze sociálních věd vytvořili spektrum technik, které dodnes pomáhají analyzovat text. Jedná se o techniky, které fungují díky lidskému porozumění textu po zasazení do kontextu, nebo o techniky, které fungují na základě statistických metod, jako je klasifikace a diskriminace.

Ve světě textové analýzy existuje šest základních přístupů: analýza konverzace (conversation analysis), diskurzivní analýza (analysis of discourse position), kritická diskurzivní analýza (critical discourse analysis, CDA), obsahová analýza, foucaultovská analýza a analýza sociální informace. Tyto metodiky jsou postaveny na různých logických strategiích, mají jiné základy a filozofické podněty. Tyto metodiky také používají různé nástroje. Každá z těchto metodologií má jisté klady a zápory. Je důležité vědět, jakou informaci je z textu potřeba vydolovat, a získat o ní znalosti pro další práci. [1]

3.1 Analýza konverzace

Tato analýza se zabývá běžnou konverzaci v životě lidí. Sleduje, jak lidé komunikují a jak společně diskutují o významu konverzace, která je zaobalená ve větším kontextu. V tomto ohledu se tato analýza zaměřuje na to, jak je jazyk využit pragmaticky v konverzaci. Tento proces je poněkud schován za oponou. Avšak pokud dojde v dané konverzaci k neshodě v situaci, ke které se konverzace váže, poté dochází k pragmatickému využití jazyka v podobě techniky zvané střídání replik mluvčích (turn-taking). [1]

Tato technika dostala svůj název při spolupráci několika vědců v 60. letech 19. století, kdy Harvey Sacks, Emanuel Schegloff a Gail Jefferson vydali článek v časopise Language s názvem *A Simplest Systematics for the Organization of Turn-Taking for Conversation*. [2] Tento článek přinesl výše zmíněný koncept turn-taking neboli

takzvané střídání replik mluvčích. Lze poukázat na to, že existují různé způsoby střídání, například každý řečník může mít dvě minuty na svůj proslov a pořadí řečníků je předem zvoleno. [2]

Existují formy proslovu, které fungují takto, ale v běžné konverzaci by to nemohlo fungovat. Sacks představuje tento koncept skrze myšlenku konverzace mezi čtyřmi jedinci A, B, C, D. Co se například stane, když B položí A otázku, pokud předpokládáme, že v takovém systému mají účastníci A, B, C a D každý možnost mluvit a v tomto pořadí? Nyní musí B počkat, až C a D domluví, a teprve potom může A odpovědět. Ale co když se A ptá i C a D? Co když například D neslyší dotaz B? Je zřejmé, že systém s předem přiděleným časem nebude pro dialog fungovat. Toto „střídání" je tedy místně řízené, interakčně kontrolované a spravované stranami, které jsou v konverzaci přítomné. [2] Model autoři popsali pomocí dvou komponent, které tuto konverzaci korigují, a několika pravidel [2]:

1. **Konstrukční složka odbočky** (turn constructional component). Tento pojem určuje tvar a rozsah možných odboček tím, že specifikuje ostře ohraničenou množinu jednotek, z nichž lze odbočky skládat. Odbočku si lze představit jako obyčejnou větu, frázi nebo slovo, kde je možné udělat v rozhovoru odbočku. Tyto konstrukční složky se vyznačují předvídatelností svého uzavření. Konec této jednotky určuje místo, kde dojde k odbočce a změně mluvčího.
2. **Otočné konstrukční celky** (turn constructional unit). Toto je vnímáno jako bod možného dokončení. Jedná se o místo, kde může začít vlastní rozhovor.

Autoři také dále specifikovali několik dalších oblastí analýzy konverzace [2]:

1. **Oprava** – neboli opravné strategie, které účastníci používají k řešení problémů s mluvením, nasloucháním a porozuměním.
2. **Akce v interakci** – konstatování něčeho považovat za konání něčeho. Například pozdrav nebo pozvání funguje jako nápověda pro jistou akci nebo jako její kompletní náhrada. Pozdrav pro podání ruky nebo pozvání pro gestikulování, aby zvaný vstoupil.
3. **Sekvencováni akcí** – činnosti probíhají v sekvencích během diskuse, vytvoření otázky určuje vhodnou a předpokládanou pozici pro odpověď.

3.2 Diskurzivní analýza

Jedná se o typ textové analýzy, který pomáhá rekonstrukci komunikace a interakcí, které vedly k produkci určitého textu. Tímto způsobem je možné získat náhled a lépe porozumět autorovi a jeho myšlenkovému pochodu. Zakládá se na porozumění rolí, které autor nebo autoři zastávají ve spojení se sociálním prostorem, ve kterém se vyskytují nebo vyskytovali. Ve zkratce se jedná o přístup rekonstrukce komunikace, ať už v psané, nebo verbální podobě, k analýze textu, která se opírá o intepretaci textu člověkem. [1]

3.3 Kritická diskurzivní analýza

Kritická diskurzivní analýza (CDA) je analýza, která používá koncept intertextuality. Tento koncept intertextuality popisuje lidskou schopnost propůjčování znalosti nebo parafrázování z diskurzů. Diskurz je v ostatních tématech chápán jako způsob mluvy, ale v této analýze je chápán jako mluva nebo styl psaní, které jsou ovlivněny sociálním prostředím. Zkoumá tedy, jaký dopad má sociální prostředí na dominantní diskurzy. Dominantním diskurzem je zde myšlen způsob mluvení a jednání k danému tématu v konverzaci, které odráží ideologii těch, kteří drží v dané společnosti moc. Jednoduše si toto lze představit na přikladu komunistické vlády, která razila diskurz komunistického manifestu (dominantní diskurz) a vylučovala demokratický diskurz západu. [1]

3.4 Obsahová analýza

Tato analýza, jak již název naznačuje, se spíše zabývá obsahem textu nežli propojením textu a jeho vazbami na sociální a historický kontext. Je klasicky definována jako metoda objektivního, kvantitativního a systematického popisu zjevného obsahu komunikace. [1]

3.5 Foucaultovská analýza

Analýza nese pojmenování po filozofovi a historikovi Michaelu Foucaultovi, který přišel s odlišnou konceptualizací intertextuality oproti CDA. Než se zabývat vlivem externích diskurzů, je v této analýze třeba se podívat na význam textu, který vzniká při diskurzu, se kterým vede dialog. Znamená to, že ve foucaltovské analýze je třeba chápat, že diskurz je ovlivněn společností a danou historickou časovou dobou. Může se jednat o explicitní nebo implicitní zapojení. [1]

Tato analýza zkoumá, jak je sociální svět vyjádřený prostřednictvím jazyka ovlivňován různými zdroji moci. Tedy přesněji zkoumá, jak je diskurz tvořen sociálními skupinami, které jsou v mocenské pozici. Snaží se pochopit, jak jednotlivci vnímají svět. [4]

3.6 Analýza sociálních informací

Tento typ analýzy se zabývá textem prostřednictvím praktického poznání autora. Texty napsané autorem mohou dát analytikovi informaci o sociální realitě, ve které se autor textu nacházel, ale tyto informace jsou omezeny samotnými znalostmi autora textu. Tyto informace jsou také zaujaté, neboť prochází skrze autorův vlastní filtr, kterým se dívá na diskurz v sociálním prostředí. [1]

4 Základy dolování z textu

Dolování z textu je postup, který požaduje velké množství odborných znalostí a zahrnuje interakci člověka se sbírkou dokumentů. Při práci s těmito dokumenty se používá množství různých analytických nástrojů. Dolování textu má obdobný cíl jako v případě dolování dat, a to hledání zajímavých vzorů k získání relevantních informací z dat. V disciplíně dolování dat z textu jsou zdroji na rozdíl od dolování dat kolekce dokumentů, kde zdroje jsou formálně upravené databáze, ovšem zde jsou hledány vzory v textových dokumentech v dané kolekci. [5]

V dolování dat z textu je důležitá součást předzpracování dat, která pomáhá textový dokument přenést do podoby, která je pro počítač nejoptimálnější a zároveň srozumitelná. Při předzpracování textu dochází k transformaci nestrukturovaných dat uložených v dokumentu na takzvaný meziformát, který je organizovanější, ale stále představuje problém pro většinu systémů, které dolují data. Tato disciplína také vyniká ve využívání metod a technik z oblasti vyhledávání informací, extrakce informací a počítačové lingvistiky s využitím korpusů (tj. rozsáhlých souborů textů daného jazyka, které jsou organizovány pro vyhledávání). [5]

Zmíněná kolekce dokumentů je libovolné seskupení textových dokumentů. V této kolekci může být několik tisíc až několik milionů dokumentů. Tyto kolekce jsou rozděleny do dvou skupin, na statické nebo dynamické. Statická kolekce dokumentů je charakterizována tím, že počáteční počet dokumentů zůstává s průběhem času nezměněn. Dynamická kolekce dokumentů je přesným opakem statické kolekce dokumentů, zde se v průběhu času objevují nové nebo aktualizované dokumenty. [5]

Základní jednotkou kolekce dokumentů je dokument. Dokument je volně definován jako jednotka diskrétních textových dat v dané kolekci, která má svůj reálný protějšek jako obchodní zpráva, výzkumná práce, rukopis a podobně. Dokument může existovat současně v několika kolekcích bez problémů. Ačkoliv je dokument nazván jako nestrukturovaný objekt, z jiné perspektivy je možné ho vidět jako strukturovaný. Například z lingvistické perspektivy může dokument mít hlubokou syntaktickou strukturu a sémantiku, ale tato struktura není vždy hned poznat a je spíše ukrytá v obsahu textu. Dále jsou tu typografické elementy jako interpunkce, velká a malá písmena, čísla a speciální znaky, které společně s tabulkami, hvězdičkami a sloupci mohou sloužit jako jednoduchý značkovací jazyk. [5]

Vlastnosti těchto dokumentů jsou použity k dolování dat z textu, neboť se v dokumentech nachází nespočet dat, jako jsou slova, fráze, věty, typografické elementy. Algoritmy na dolování dat z textu jsou založeny na reprezentaci pomocí těchto vlastností dokumentu. Toho je možné dosáhnout pomocí dvou pravidel. Zaprvé je důležité při dolování dat z textu dosáhnout určité přesnosti objemu obsahu a sémantiky. Zadruhé je to identifikování vlastností dokumentu, které jsou, co se týče efektivity a praktičnosti, nejoptimálnější pro vyhledávání vzorů. K tomu je použito několik vlastností dokumentu, ale zde jsou vybrány čtyři, které jsou nejpoužívanější v této disciplíně [5]:

1. **Znaky** – jedná se o individuální písmena, číslice, speciální znaky a mezery, které pak vytváří samotný obsah textu. Také se používají i poziční informace znaků (bigram, trigram).
2. **Slova** – jsou přímo vybrána z dokumentu, představují základní úroveň sémantiky daného dokumentu. Reprezentace dokumentu pomocí slov už má jistou optimalizaci oproti reprezentaci pomocí znaků. Jsou vytvořeny z jednotlivých podmnožin vlastností, které jsou vybrány podle kritérií jako stop slova, symbolické znaky a číslice.
3. **Termíny** – jedná se o slova, která jsou vytažena přímo z korpusu daného dokumentu. Jednotlivé termíny jsou poté normalizovány a podrobeny vyhodnocení. Většinou porovnáním s externími slovníky a poté jsou vybrány takzvané kandidátní termíny. Tato podmnožina termínů jasně definuje daný dokument.
4. **Koncepty** – vlastnosti dokumentu, které jsou vybrány po pečlivé provedené analýze, ať už se jedná o manuální, statistickou, či kategorizační metodu. Tyto vlastnosti pak reprezentují celý dokument.

Z těchto vypsaných vlastností jsou nejvíce používané termíny a koncepty, protože dokážou nejlépe vyjádřit a popsat dokument. Jejich vyhodnocení je na podobné úrovni, ale svými výsledky daleko převyšují reprezentaci pomocí slov a znaků. [5]

4.1 Selekce dat

Jedním z důležitých kroků, od něhož se poté odvíjí všechny výsledky práce s daty a dokumenty, je selekce dat. Po výběru těch správných dat je potřeba takzvaného odběru vzorků dat. Ovšem i při takovém výběru dat je možné, že dojde k zaujatosti,

a je třeba se ujistit, že i o tyto zaujatosti je postaráno. Proto se doporučuje do výběru připojit i sociálního výzkumníka, aby se těchto zaujatostí zbavil. [1]

Toto je velmi důležitý krok, protože už zde dochází k propojení hypotézy z teoretických znalostí s datasety z reálného světa. Dataset je sbírka dat, která odpovídá jedné nebo více databázovým tabulkám se záznamy, jež se vztahují k tématu, které je zkoumáno. Každý projekt začne selekcí dat. Tento výběr dat, pokud bude správný, zvyšuje pravděpodobnost dosažení předem určených cílů. Při výběru dat lze vybrat takzvaný reprezentativní případ, který bude představovat populaci. Je možný také náhodný výběr případů, ale tento způsob výběru není brán jako optimální strategie, protože typický průměrný případ není vždy ten, který obsahuje množství informací, jež je potřeba. Pokud je správně vybrán dataset, který reprezentuje populaci, lze po dokončení výzkumu na tomto vzorku provádět závěry na celou populaci. [1]

4.2 Výběr vzorků dat

Poté, co je vybrán správný dataset, dojde k použití výběru vzorků na daném datasetu. Tento odběr vzorků funguje tak, že vytváří reprezentativní vzorky populace, ze kterých byly odebrány. Ideálním vzorkem je pravděpodobnostní vzorek, který pomáhá skrze statistické závěry vyvodit zobecnění výsledků na celou populaci. Při pravděpodobnostním výběru je šance každé složky na zařazení do vzorku známá a nikdy není nulová. Existují i případy, kde má každý případ stejnou šanci být vybrán do vzorku (prostý náhodný výběr). Dvěma hlavními zásadami tohoto argumentu je, že se eliminují výběrová zkreslení a je možné předpovědět, že vzorek bude pravděpodobně reprezentativní pro populaci, z níž byl vybrán. [6]

Ve výběru vzorků z dat se také používají další možné techniky, jako je například výčet. Při této technice je důležité vybrat, co bude zvoleno jako jednotka analýzy. Analytik může používat vzorky článků z celé historie vydaných novin, ale jako jednotku může používat počet slov nebo počet opakovaných slov. Každý výčet může mít očíslované jednotky analýzy nebo mohou být vypsány v seznamu. Pokud jsou tyto předešlé kroky splněny, lze přejít k další technice, kterou je systematický odběr vzorků. Zde vybíráme jako vzorek každou k-tou jednotku analýzy z výčtu. Interval k je konstantní, proto je zde možnost vytvoření zaujatosti skrze tuto konstantu. Ovšem má i své výhody, jako je lehké na uvedení do výzkumu, a není třeba předešlých znalostí statistiky, velikost vzorku nemusí být předem známa, výběr konstanty k pro použití je

rychlý a nekomplexní, ve vzorcích mohou být nesrovnalosti a různé nerovnosti v čase, prostoru a frekvenci a vliv na výběr bude minimální. [6]

Další technikou je stratifikovaný výběr vzorků, kde je výběr vzorků z vrstev rozdělených, tak aby se žádná vrstva nepřekrývala. Stratifikovaný výběr lze provést i na velkém souboru dat pomocí běžných programů pro statistickou analýzu, jako jsou SPSS a SAS. Pokud jsou porovnávány různé skupiny nebo vrstvy napříč zájmovými charakteristikami, je tato strategie výběru vzorků vhodná. [6]

Zajímavou technikou výběru reprezentativních vzorků je výběr s proměnlivou pravděpodobností. Tato technika pomáhá proporcionálně vzorkovat datasety, které mají odlišnou velikost nebo důležitost. [1]

Velmi oblíbenou technikou vzorkování je výběr vzorků sněhovou koulí (snowball sampling) v kvalitativních datech. Jedná se o iterativní proces, kdy na začátku existuje malý vzorek a poté je opakovaně vzorkováno z populace na základě kritérii, dokud není dosaženo limitu počtu vzorků. [1]

Poslední technikou je účelový výběr vzorků. Toto je technika, která spíše pokládá otázky místo vzorkování pomocí pravděpodobnosti. K výběru dat a vzorků lze dojít pomocí otázky, na kterou je výzkum zaměřen, a poté zredukováním textu, který není relevantní pro danou otázku. Tato technika se zdá velmi obyčejná, a proto je málokdy považována za kategorii sama o sobě, ale je použita téměř vždy. [1]

4.3 Tokenizace

Proces, který rozděluje text na takzvané tokeny. Tokeny mohou být jednotlivá slova nebo fráze, ale stále si zachovají stejný význam. Během tohoto procesu mohou být odstraněna různá interpunkční znaménka. To může vypadat na první pohled velmi triviálně, ale například u interpunkčního znaku tečky je třeba rozlišit mezi akronymem, jako je U.S.A, a zkratkou, jako je Ing., protože tečka u těchto příkladů je využita jinak. Při tokenizaci je důležité se zbavit těchto teček. Ovšem v příkladech uvedených výše je třeba ji ponechat kvůli jejich významu, pokud by byla odstraněna, tak slova ztratí svůj význam. Toto pravidlo platí i pro čísla s desetinnou tečkou a také pro tečku v datech. [5]

Tokenizace existuje ve třech možných variantách. První je „whitespace tokenization“ neboli tokenizace pomocí mezer. Text je rozdělen na token kdykoliv se mezi slovy objeví mezera v informatice známé jako „whitespace“. Další je za pomocí slovníku,

kde je použit slovník specifický pro daný jazyk. Tokeny jsou zde rozděleny podle toho, jestli se slova nacházejí ve slovníku nebo ne. Poslední variantou je „subword tokenization“, tedy tokenizace pomocí podslov. Slovo, které je vytvořeno z více slov, je dále rozděleno na minimální počet slov tak, aby si stále zachovala význam. Jedná se o strojové učení pod dohledem člověka. [7]

Je důležité zmínit, že tokenizace je velmi závislá na jazyku, ve kterém je prováděna. Každý jazyk má speciální případy zkratek a kontrakcí (zkrácení tvaru slov vypouštěním sousedních slabik). To samé platí pro apostrof a pomlčku. Lze tomuto problému předejít, ale je třeba mít seznam těchto slov, aby byla tokenizace spolehlivá. [1]

V procesu tokenizace se mohou objevit i procesy, které text normalizují, jako je změna na malá písmena nebo na takzvaný truecasing (výběr toho správného písmena a jeho změna na velké) nebo také odstranění HTML značek, pokud je text získán pomocí web scrapingu. Web scraping je proces, při kterém jsou extrahovány informace z webu do vhodnějšího formátu, například do tabulky v Excelu. [1]

4.4 Odstranění stop slov

Stop slova (funkční slova) jsou slova, která mají vysokou frekvenci, ale nepřidávají žádnou informační hodnotu pro analýzu. Také nepřináší žádnou sémantiku do textu. Jedná se slovní druhy, jako jsou předložky a zájmena. Odstranění je řešeno seznamy stop slov pro daný jazyk, které už jsou předem vytvořeny. Jazyky s velkou populací rodilých mluvčích nebo ty jazyky, které jsou velmi ve světě využívané, už mají takové seznamy veřejně dostupné. [5]

Pokud takový seznam neexistuje, je možné využít jednu z vlastností stop slov, a to je jejich velká frekvence. Lze tedy zjistit počet výskytů N a poté vytvořit seznam, který bude fungovat jako seznam kandidátních stop slov. Po pečlivé analýze těchto kandidátních slov a konzultací s rodilým mluvčím lze tento seznam kandidátů použít jako filtr stop slov pro daný jazyk. [1]

Tento proces je velmi důležitý při vyhledávání informací, detekci a extrakci témat, ale nemá své využití v jiných procesech, jako je například klasifikace. Přesto se používá, protože díky redukci stop slov, které podle své definice mají velkou frekvenci, je možné zmenšit velikost modelů při dolování dat z textu. [8]

4.5 Převedení na kořen slova

V každém jazyce jsou slova, která si jsou podobná, ale přesto mají různé formy. Může se stát, že tato slova mají jiný význam, ale přesto se jedná o stejná nebo podobná slova. K vyřešení tohoto problému pomáhá proces převedení na kořen slova, tzv. stemming neboli stemování. Jde o proces, který používá pravidla k odstranění přípony a předpony. Tímto způsobem lze získat kořen slova. Kořeny slov se dále používají k vyhledávání informací (informational retrieval, IR). Pomocí procesu stemmingu je vytvořen indexovaný seznam s kořeny slov. Tento indexovaný seznam kořenů není určen pro čtení nebo kontrolu uživateli, protože je těžko pochopitelný a nečitelný pro běžného uživatele. Například pro anglický jazyk existuje už předem připravený stemmer nazvaný The Porter (Porter Stemmer), který představuje ve svém článku [8] Kavita Ganesan. Podle pravidel anglického jazyka odebírá přípony a předpony bez použití slovníku. Tento algoritmus už byl několikrát ověřen a je považován za velmi přesný. Ovšem i tak se mohou vyskytnou problémy a výsledek nemusí mít validní převedení na kořenovou formu slova. V tabulce 1 je naznačeno, jak takový proces funguje odstraněním přípony a předpony v češtině. [8]

Tabulka 1 Příklad stemování; zdroj: vlastní zpracování

Originální slovo	Převedení na kořen slova
Hra	Hra
Vý**hra**	Hra
Hravý	Hra
Hračka	Hra
Vy**hra**ný	Hra

4.6 Převedení na základní podobu slova

Alternativou pro proces převedení na kořen slova je převedení na základní podobu slova, tzv. proces lemmatizace neboli lemmatization. Stemování a lemmatizace se mohou jevit jako podobné procesy, neboť se oba snaží najít kořen slova, ale odlišují se v přístupu. Proces lemmetizace neodstraňuje přípony a předpony, ale místo toho slovo transformuje na jeho základní tvar, v lingvistice se tento základní tvar nazývá lemma. Například slovo *lepší* je transformováno na slovo *dobrý*. Tímto způsobem je zaručen validní výstup tohoto procesu. Často se pro tento proces používá slovník WordNet.

Způsob používaný pro mapování základních tvarů slov také popisuje ve svém článku [8] Kavita Ganesan. Slovník WordNet zahrnuje většinu podstatných jmen, sloves a přídavných jmen. Nejnovější verze WordNet 3.1 má slova organizována se synonymy v takzvaných synsetech. Pro český jazyk jsou zde k dispozici slovníky nebo nástroje zvané lemmatizátory, jako je Ajka, Majka nebo Morče. Díky procesu lemmetizace je výsledek čitelný i pro člověka, a tak ho i člověk může zkontrolovat. I tento proces však má své zápory a jedním z nich je skutečnost, že se oproti procesu stemming jedná o výpočetně náročnější proces. [8]

Příklad procesu lemmetizace lze vidět v tabulce 2 na větě „Na vyzvání svého předsedy jsme odešli.“

Tabulka 2 Příklad procesu lemmatizace; zdroj: vlastní zpracování

Originální slovo	Po procesu lemmetizace
Na	Na
vyzvání	vyzvání
svého	svůj
předsedy	Předseda
jsme	být
odešli	odejít

4.7 Odstranění šumu

Noise removal neboli odstranění šumu je proces předzpracování dat pro účel dolování dat z textu, při kterém dochází k odstranění částí textu, které ztěžují analýzu nebo nejsou pro ni přínosné. Jsou různé způsoby odstranění šumu, např. odstranění znaků, číslic, interpunkce, speciálních znaků, zdrojového kódu, hlaviček dokumentů nebo i odstranění formátování HTML kódu. Odstranění šumu patří mezi nejdůležitější části v předzpracování textu, ale tento proces je velmi závislý na znalosti domény, odkud text pochází. Mohlo by totiž dojít k odstranění důležitých částí, které jsou v obecné struktuře dokumentu nevýznamné, ale v dané doméně mají své místo a význam. Například v oblasti sociálních sítí se využívá takzvaný hashtag, který je symbolizován znakem #. Za tento znak mřížky uvádí uživatelé slova, znaky nebo termíny, které charakterizují příspěvek, proto vše kromě toho by mohlo být považováno za šum a být odstraněno. [8]

4.8 Změna na malá písmena

Pomocí změny na malá písmena lze změnit celý text, aby měl konzistentní formát. Jedná se o jeden z nejvíce používaných aktivit při předzpracování textu. Počítač může vnímat výskyt slov např. „Kanada“ a „kanada“ jako dvě různá slova a jejich vzácnost výskytu v textu by mohla mít za následek chybný výsledek později vytvořených modelů, například modelu pro detekci témat. Tento problém může nastat, pokud je velikost kolekce dokumentů malá. Tento proces pomůže i při vyhledávání informací. Například při vyhledávání „česko“ nejsou nalezeny žádné výsledky, protože informace v textu je obsažena v podobě „Česko“. Je ovšem důležité být dobře seznámen s doménou, ze které je dokument, protože změna na malá písmena nemusí být vždy ta správná volba. [8]

5 Zpracování přirozené jazyka

Natural language processing (dále jen NLP) je proces, který dává počítači schopnost porozumět lidské řeči. Podle autorů článku [9] je lidská řeč pro počítač plná nejasností, ale pokud půjde tuto překážku překonat a počítač dokáže porozumět lidské řeči, nastane podle některých názorů druhá industriální revoluce. První takový náznak se objevil v roce 2019, kdy firma Google vydala svůj první jazykový model BERT (Bidirectional Encoder Representations from Transformers). Revoluce zde proběhla v tom, že text už nebyl zkoumán zprava doleva nebo naopak, ale v transformátorech, které mají kodér a dekodér. S jejich pomocí BERT dokázal určit kontext textu pomocí techniky Masked LM, která predikuje slova jejich maskováním. [10] Poté přišel rozmach díky firmě OpenAI, která na svět uvedla svou službu ve verzi ve formě GPT-3, jež později přešla v nám známý ChatGPT. Služba ChatGPT je založena na umělé inteligenci, která byla trénována na velkém množství dat, aby dokázala předpovědět, jak na sebe slova navazují. Na začátku jsou tyto předpovědi nahodilé, ale díky metodě zpětné vazby od člověka (reinforcement learning from human feedback, RLHF) je program trénován a poté jemně dolaďován, aby správně fungoval. [11]

Toho všeho bylo dosaženo díky rozmachu oblasti NLP. Je to potřeba, protože naše lidská řeč se vyvinula organicky za účelem porozumění a předávání znalostí a nápadů, kdežto programovací jazyk byl vytvořen, aby člověk mohl předávat počítači instrukce, které chce, aby splnil. [9] NLP je proces, při kterém dojde ke spojení informatiky a lingvistiky skrze techniky předzpracování textu, jež byly zmíněny výše. Výstupem je zpracovaný výsledek, kterému porozumí počítač. [9]

Vzhledem k tomu, že je lidská řeč ve formě textu a počítač a jeho algoritmy řeší problémy na základě čísel 0 a 1, je třeba text transformovat do čísel neboli takzvaně text vektorizovat. Nyní je namístě ukázat techniky NLP, které se používají pro vektorizaci textu kromě již výše zmíněných technik pro předzpracování textu, jež jsou již považovány za základní praktiky pro práci s textem.

5.1 TF-IDF

Term frequency-inverse document frequency (TF-IDF) je statistická míra, která hodnotí relevantnost slova pro dokument v kolekci daných dokumentů. Metoda byla vytvořena pro automatickou analýzu textu a vyhledávání informací. [12] Tato

hodnota je vyhodnocena pomocí dvou metrik, první říká, kolikrát se slovo v dokumentu vyskytuje, a druhá zjišťuje inverzní frekvenci dokumentu daného slova v souboru dokumentů. První metrika této metody je zřejmá, ale je důležité vysvětlit druhou metriku vyhodnocení.

Inverzní frekvence dokumentu značí, jak časté nebo vzácné je slovo v celém souboru dokumentů. Tuto metriku lze vypočítat pomocí celkového počtu dokumentů, který vydělíme počtem dokumentů, které slovo obsahují. Výsledek je poté zlogaritmován, aby bylo získáno skóre pro toto slovo. [12] Tyto metriky jsou poté vynásobeny mezi sebou. Čím více se blíží nule, tím je slovo častější a má tedy menší skóre. Často se jedná o nějaké stop-slovo.

5.2 Part-of-speech tagging

Proces part-of-speech tagging značkuje neboli provádí tagování (tagging) části přirozeného jazyka a kategorizuje jej podle korpusu. [13] Identifikuje v českém jazyce slovní druhy jako podstatná nebo přídavná jména a slovesa. Tento proces se zdá poněkud jednoduchý, ale opět je třeba nahlížet na to z pohledu stroje, který o organickém jazyce nic neví. Některá slova mohou mít několik významů a záleží na kontextu a na tom, jak je daná věta, ve které se slovo nachází, postavena. Níže v tabulce 3 je uveden příklad na slovu „kolem“, které může mít více významů.

Tabulka 3 Příklad part-of-speech tagging; zdroj: vlastní zpracování

Použití slova „kolem“	Slovní druh
Sejdeme se **kolem** šesté.	Příslovce
Šel po ulici s **kolem.**	Podstatné jméno
Lidé se shromáždili **kolem** radnice.	Předložka

Ovšem tento proces má několik nástrojů na to, jak ulehčit práci při tagování jednotlivých slov podle kontextu. Protože jedno slovo může představovat více slovních druhů, existuje set těchto tagů (tagset) pro rozeznání slovních druhů. Nejznámějším a nejpoužívanějším je tagset zvaný UPenn TreeBank tagset, který má dohromady 45 tagů a pomocí 3 znaků určuje slovní druh nebo jiné gramatické kategorie. Díky tomuto korpusu je přesnost procesu na 97 %, což je srovnatelné s tím, kdyby tyto tagy přiděloval sám člověk. [13]

Ovšem tato přesnost závisí plně na korpusu, který tagovací technika bude využívat. V dnešní době je part-of-speech tagging spíše ponecháno na strojovém učením nebo hlubokém učení skrze neuronové sítě. Dosahují tak podobné přesnosti, a to 94 %. Je toto sice o něco méně, ale toto negativum vyvažuje rychlost zpracování. [14]

V dnešní době jsou již tyto techniky založeny na takzvaném deep learning, což je počítačová metoda zpracování dat způsobem, který se podobá činnosti lidského mozku. Metoda je založena na umělé neuronové síti, neurony se zde jako v mozku dokážou učit, zevšeobecňovat a extrahovat a reprezentovat závislosti v datech, které nejsou zřejmé. Tyto neurony spolu spolupracují a vytvářejí síť, která je podobná lidskému mozku. Nejčastější technikou jsou nyní rekurentní neuronové sítě. [15]

5.3 Rozpoznávání pojmenovaných entit

Rozpoznávání pojmenovaných entit (named entity recognition, dále jen NER) se poprvé objevilo na konferenci Sixth Message Understanding v roce 1999. Důvodem byla potřeba z nestrukturovaného textu extrahovat důležité informace, jako jsou jména, názvy společností a lokalit, i informace v podobě čísel, jako je čas a datum. Mnozí už považují tuto problematiku za vyřešenou, protože NER algoritmy mají přesnost přes 95 %. [16]

V roce 2020 bylo uloženo 64,2 zettabytů na internetu. V následujících letech až do roku 2025 se předpokládá, že celosvětová tvorba dat vzroste na více než 180 zettabytů. Bohužel investice do personálu pro správu a řízení těchto dat tak rychle nerostou. V této oblasti by mohl pomoci NER k identifikaci sémantiky v nestrukturovaných textech. [17]

Pro správné rozpoznání entit existuje už od roku 1999 několik definic. Jedna z hlavních definic se objevila v roce 2000, a to, že se jedná o vlastní podstatné jméno, které slouží jako název pro něco nebo někoho. [18] Další definice pohlíží na NER jako na úkol zařadit neznámé objekty do známých hierarchií, které jsou předmětem zájmu a které jsou užitečné pro řešení určitého problému. [19] Experti, kteří se o tuto problematiku zajímají, přišli se čtyřmi kritérii, která berou v potaz předešlé různorodé definice NER [17]:

1. **Gramatická kategorie** – definuje pojmenované entity jako vlastní jména nebo obecná jména, protože tato gramatická kategorie označuje bytosti a jedinečné skutečnosti.

2. **Rigidní označení** – jasné označení nebo popis dané zkušenosti, která se nemění a zůstává v čase stejná.

3. **Jedinečná identifikace** – je vlastně referent toho, na co odkazuje, je zde potřeba předchozí znalost reference od přijímače zprávy. Například H_2O je jedinečná chemická identifikace vody, ale podstatné jméno voda už dále není identifikátorem, pokud nejsou rozlišovány různé druhy vody jako perlivá a neperlivá.

4. **Oblast použití** – jedná se o účel a oblast, kde se daná entita pohybuje. Tyto oblasti jsou již předem určeny a jsou přiřazeny pojmenovaným entitám jako první krok.

Jak je vidět z kritérií výše, některá mají velmi slabou definici a sama o sobě nejsou dostatečná pro identifikaci pojmenované entity, ale dohromady tvoří celek, který má za úkol vyhledat entity v textu a správně je klasifikovat. [17]

Existuje několik způsobů, jak rozpoznat entity v textu. V dnešní době je stěžejní rozdělení na metody, které používají strojové učení, a ty, které ho nepoužívají. Způsoby NER, které ho nepoužívají, jsou založeny na různých pravidlech pomocí slovníků nebo skrze ontologii. Ontologie se spíše využívá u strukturovaného nebo semi-strukturovaného textu. [20]

Na druhé straně existují už zaběhlejší způsoby rozpoznání pojmenovaných entit [21]:

1. **Na základě pravidel** – tato metoda potřebuje velkou iniciativu a hodnocení ze strany člověka v podobě úprav pravidel pro různé varianty textu a řeči, které nebyly v trénovací anotaci.

2. **Systémy založené na slovníku** – používají slovník s rozsáhlou slovní zásobou a synonymy k určení entit.

3. **Systémy založené na strojovém učení s dohledem** – tato metoda používá strojové učení, ale jsou zde třeba zásahy člověka, protože člověk předává modelu text, který už byl anotován.

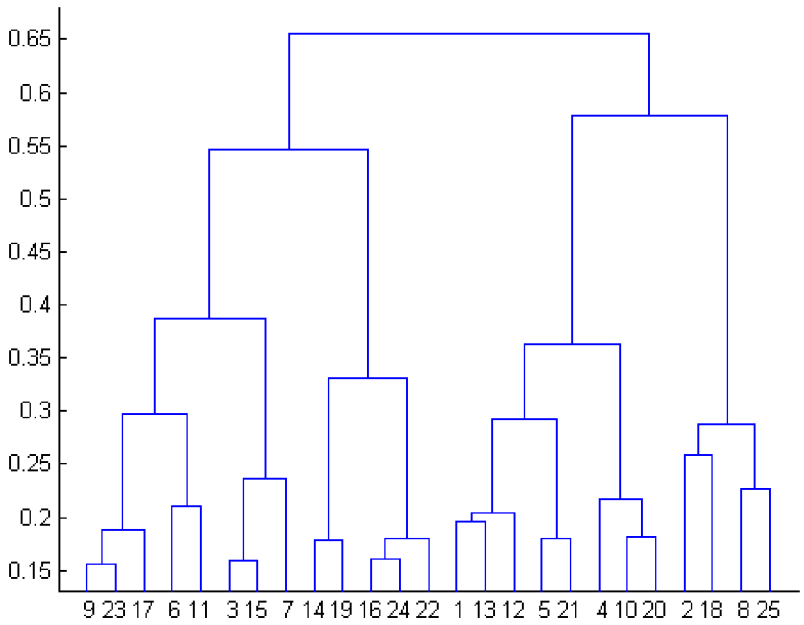
5.4 Shlukování

Tato metoda má za úkol rozdělit nestrukturovaná data, text nebo dokument do takzvaných shluků (cluster). Každý tento shluk dat má podobné vlastnosti a každý

shluk se nějak liší od ostatních shluků. Cílem je tedy vytvořit skupiny s podobnými vlastnostmi, a roztřídit tak data. V případech shlukování musí nejdříve dojít k vektorizaci a každý vektor musí mít přiřazenou nějakou váhu, která udává jeho důležitost v dokumentu. K tomu lze dojít právě díky metodě TF-IDF, která byla popsána výše. Základní metoda shlukování má dva typy: tvrdou a měkkou. Druhý typ má speciální případ shlukování, a to shlukování dokumentů. [22]

1. **Tvrdá metoda** – je známá jako hard, typu shluková, kde každý dokument nebo datový bod je přiřazen jen do jednoho shluku a neexistuje žádný průnik.
2. **Měkká metoda** – neboli soft je opak tvrdé, kdy dokument nebo datový bod se může objevit ve více shlucích.
    2.1. **Rozdělení na oddíly** – tento způsob shlukuje dokumenty nebo datové body do předem určeného množství shluků.
    2.2. **Hierarchické** – jak název napovídá, tento typ shlukování tvoří hierarchii v podobě dendrogramu.

Mezi nejpopulárnější algoritmy shlukování patří právě hierarchický způsob a algoritmus nejbližších sousedů. Hierarchický způsob vytvoří pomocí algoritmu takzvaný dendrogram. K němu dojde tak, že na začátku má každý datový bod nebo dokument svůj vlastní shluk. Poté se dva nejbližší shluky spojí do jednoho a takto algoritmus pokračuje, až existuje jen jeden shluk, v tomto případě se algoritmus ukončí a výsledkem je dendrogram. [22]



Obrázek 1 Ukázka dendrogramu; zdroj: [22]

Na obrázku 1 je vidět dendrogram, kde byl využit přístup bottom-up a naopak lze použít i přístup top-down. Na základě několika kritérií vzdálenosti mezi dvěma datovými body pomocí euklidovské matematiky jsou pak tyto shluky nebo datové body spojeny s nejbližším shlukem dohromady. Jediné negativum tohoto přístupu je jeho časová náročnost k vytvoření shluků. [22]

Pokud ovšem je potřeba rychlost a přesnost, existuje druhý přístup s názvem k-means. Jedná se o iterativní přístup shlukování, který hledá lokální maxima pro každou iteraci postupu. Na začátku metoda k-means přiřadí body do několika shluků náhodně, poté provádí výpočet geometrického středu a přiřadí daný bod nebo dokument do nejbližšího shluku. Tento postup se opakuje, dokud není změna umístění bodu nebo dokumentu do jiného shluku ani pro jeden datový bod nebo dokument. Stejně jako hierarchický přístup má i tato metoda svá negativa. Má lineární časovou složitost, a proto je lepší pro použití na větším množstvím dat. Hlavní problém je ovšem to, že tato metoda je citlivá na počáteční výběr shluků a může spíše hledat optimum nežli geotermické vzdálenosti mezi body. [22]

6 Detekce témat

Další technikou zpracování přirozeného textu je detekce témat, která je hlavní náplní této práce. Websterův naučný slovník definuje téma jako předmět diskurzu nebo jeho části. [24] Ovšem v detekci témat je definice poněkud odlišná. Téma je definováno jako pravděpodobnostní rozdělení nad pevně daným slovníkem. [25] Každé téma je tedy mix slov ze slovníku. Jak bylo již výše uvedeno, množství dat, které bylo dostupné v roce 2020, mělo objem 64,2 zettabytů a tento trend bude v následujících letech jen růst. [17] Proto je potřeba mít nástroje, které pomohou získávat informace mnohem rychleji a efektivněji, aby bylo možné dohnat tento technologický pokrok.

První zmínka o detekci témat se objevila v roce 1990, kdy vznikl předchůdce této techniky s názvem latentní sémantické indexování, metoda, která automaticky indexuje a vyhledávání relevantní termíny. Později se objevila metoda s názvem pravděpodobnostní latentní sémantická analýza, která na rozdíl od předešlé metody, jak už název napovídá, je pravděpodobnostní metodou a vytváří pravděpodobnostní model. [25]

Tyto předešlé metody daly za vznik metodě latentní Dirichletovy alokace, na které je založeno několik pravděpodobnostních modelů. Jedná se o tříúrovňový bayesovský model, v němž je každá úroveň modelovaná jako konečná množina nad základní množinou témat. Tato metoda je více vysvětlena v následujících kapitolách a také použita v praktické části. [27]

Dále jsou popsány jednotlivé kroky detekce témat, které jsou v praktické části práce použity a důkladné vysvětlení metod pro detekci, které jsou uvedeny výše.

6.1 Bag of words

Při zpracování přirozeného jazyka je dokument obvykle reprezentován pomocí bag of words (dále jen BoW) modelu, což je matice slov a dokumentů, kde existuje N počet dokumentů a V počet slov. Tudíž se vytvoří matice N x V. [25]

Ovšem tato metoda není limitována jen na dokumenty. BoW je víceúčelový model, který lze použít i na klasifikaci obrázků, kde klasifikuje obrázek podle rysů na každém z nich. Přesto je známý hlavně v metodách klasifikace textu, kde jsou klasifikátory vytvářeny pomocí slov a výskytu četnosti v textu. [28]

Každá instance dokumentu nebo obrázku je nestrukturovaný set pro BoW, v případě dokumentů jsou to slova. Pro klasifikaci textu je vypočtena váha slov. To jsou jednotlivé četnosti výskytu slova v dokumentu. Je třeba zmínit, že BoW se nezajímá o správnou gramatiku nebo o správný pravopis slova. Pokud se objeví pravopisně chybné slovo, BoW k němu bude přistupovat jako k odlišnému slovu a započítá jeho výskyt samostatně. [28]

Než přijde na řadu samotný přístup a počítání četností, musí se text tokenizovat, jak bylo vysvětleno v části o předzpracování textu. Poté je každé slovo jako token osamoceno a může se počítat jeho četnost. Po provedení součtu četností všech použitých slov lze vytvořit histogram pro grafické znázornění. Ovšem pokud se zde neodstranily stop slova, pak mohou mít na konci součtu největší četnosti stop slova, která nepředávají žádnou informaci. Jestliže k této situaci dojde, lze použít metodu TF-IDF k normalizování výsledku. [28]

Dalším problémem, kterému čelí BoW, je takzvané prokletí dimenzionality. Jde o problém, že se stoupajícím množstvím slov stoupá také velikost korpusu. Tudíž roste i počet parametrů, na kterých by se trénoval model. Také je zde problém, že pro BoW pořadí slov nemá smysl, a tudíž se při použití BoW může ztratit kontext a sentiment daného textu. [29]

V dnešní době už díky strojovému učení vznikají možná vylepšení této zaběhlé metody a jedním z nich je model AEBoW (Attribute of Network Extended to BoW). BoW je rozšířené o síť, která místo váhy slova pomocí metody TF-IDF uchovává informaci o váze atributu uzlu sítě k ostatním a tímto uchovává kontext a sentiment, díky propojením uzlů v síti. [30]

6.2 Klasifikace témat

Klasifikace témat funguje na základě strojového učení pod dohledem člověka. Analýza v klasifikaci témat spočívá v tom, že témata už jsou předem vybrána a model je podle nich trénován, aby správně přiřazoval témata v textu, na kterém se učí. Na rozdíl od vytváření témat je při modelování témat k přípravě potřeba více času, ale tato analýza zvládá přesnější určení a klasifikaci témat. [31]

Existuje několik způsobů, jak docílit této klasifikace [31]:

1. **Na základě pravidel.** Jako každá metoda, tak i tato metoda může docílit požadavků skrze předem určená pravidla pro klasifikaci. Funguje na základě

vzorů a predikcí jednotlivých témat. Predikce je zde dané téma a vzor jsou slova, které spadají pod určité téma.

2. **Strojové učení.** Model se učí skrze testovací datasety, aby sám automaticky dokázal určit téma. Zde přichází na řadu převedení textu na vektor, protože počítač přirozenému jazyku nerozumí. Nejčastěji vektorizace probíhá pomocí bag of words metody.
3. **Naivní Bayes.** Zde je důležitá korelace pravděpodobnosti výskytu slov v textu s pravděpodobností, že text je o daném tématu. Naivní se nazývá, protože předpokládáme nezávislost mezi tématy.
4. **Hluboké učení.** Učení probíhá skrze neurální sítě. Je třeba více trénování na datasetech pro tento model, aby dokázal klasifikovat lépe než ostatní zaběhlé algoritmy.
5. **Hybridní systémy.** Využívají kombinaci výše zmíněných metod pro klasifikaci témat.

Klasifikace témat a detekce témat jsou podobné metody zpracování přirozeného jazyka, ale liší se v cílech. Zatímco detekce témat automaticky vyhledává témata, klasifikace témat naopak má témata již vybrána a model je pak trénován tak, aby co nejrychleji a nejpřesněji dokázal klasifikovat. Používá se v situacích, kdy je známo, že dokumenty budou patřit do specifických kategorií. [31]

6.3 Modelování témat

Modelování témat se v dnešní době používá jako jedna ze základních úloh ke kompresi informací, kterými je svět v dnešní době zahlcen. Dokáže tisíce dokumentů skloubit do krátkého popisu, který obsahuje jejich základní vlastnosti, tedy témata. Tato akce už laicky definuje modelování témat, ale správná definice uvádí, že tematický model je neřízený matematický model, jenž na vstupu přijme množinu dokumentů D a vrací množinu dokumentů D a množinu témat T, která reprezentuje obsah množiny D výstižně a souvisle. [32]

První zmínka k modelování témat, jak už bylo napsáno v předchozích kapitolách, se objevila v roce 1990 za pomoci latentní sémantické analýzy. Poté v roce 1999 se už objevilo modelování témat jako kategorie sama o sobě, a to s příchodem pravděpodobnostní latentního sémantického indexování. V této metodě jsou témata

popsána jako faktory. První modely témat se objevily na přelomu roku 2000, konkrétně Dirichletova multinominální směs (DMM) a v roce 2003 latentní Dirichletova alokace (LDA), na kterou se tato práce zaměřuje. Obě metody používají Dirichletovo rozdělení. Po úspěchu LDA, která se stala zaběhlým standardem v této problematice, se začaly objevovat její varianty, které opravovaly nebo vylepšovaly její nedokonalosti. Jedním z takových modelů je Hierarchical Dirichlet Process (dále jen HDP), model, jenž nemá parametr témat *k*, který vybere člověk před spuštěním jako LDA, protože není známo, kolik témat se může v textu nacházet. Druhou variantou je korelační detekce témat, kde přidává model možnost korelace témat, kterou LDA nepoužívá ve své analýze. Dále je zde dynamický model témat, který dokáže zjišťovat témata textu v čase, jak se vytvářejí. [32]

V dnešní době už prošla detekce témat velkými změnami kvůli změně dokumentů, které jsou zkoumány. Rozsah použití je od dnešních příspěvků na sociálních sítí až po hledání témat v literatuře 19. století. Je možné je použít i v prediktivních modelech, kdy vypomáhají ke zvýšení prediktivní síly a lepší pochopení výsledků. I když LDA vznikla před 20 lety, tato metoda se stále používá jako základ pro detekci témat a jsou k ní přidávány další matematické metody. Například faktorizace nezáporných matic, která pomáhá snížit dimenze a odstranit šum, nebo modely založené na grafech, kde slova tvoří jednotlivé uzly a jejich postupné opakování v textu přidává danému uzlu váhu. Úzce spojená slova značí, že text se zaobírá daným tématem. Tato metoda dokáže najít v textu jak obsáhlá témata, tak i lokální témata, která nejsou tak objemná. [32]

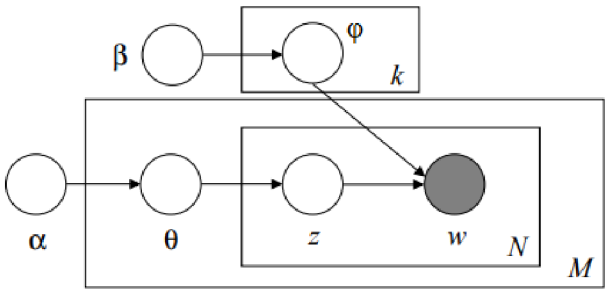
Důležitým milníkem ve vývoji detekce témat byl vznik Word2Vec v roce 2013. [32] Tato metoda se váže k BoW, také tvoří vektor z textu, aby počítač dokázal pracovat i s přirozeným jazykem, ale dokáže jednu věc mnohem lépe, umožňuje totiž, aby podobná slova měla podobný rozměr, a pomáhá tak vnést kontext do nestrukturovaného textu. Slova, která jsou si podobná, mají asociaci mezi sebou tvořenou tak, že jsou v prostoru vyobrazeny blíže k sobě. Díky tomuto vznikl model zvaný Lda2Vec, který používá LDA a Word2Vec. Tento model zvládá jak velké objemy nestrukturovaného textu, tak i ty s menším obsahem. Každé téma je vyobrazeno v prostoru jako vektor, vypočtený součtem pravděpodobností každého dokumentů, který k danému tématu patří. [33]

Je vidět, že vývoj modelování témat byl zapříčiněn množstvím informací, jež jsou dnes dostupné, a také tím, jaký je to typ informací, ze kterých je třeba získat témata. Stále ovšem přetrvává LDA v detekci témat a na ní se zakládají další metodologie pro detekci témat. Proto je LDA podrobněji popsána v další kapitole.

6.3.1 Latentní Dirichletova alokace

Tento generativní pravděpodobnostní model se prvně objevil v roce 2003. Autoři v té době stavěli na předešlých metodách a odborných pracích, jako je TF-IDF metoda, latentní sémantické indexování a pravděpodobnostní latentní sémantické indexování. Předešlé metody v základu spočívají na předpokladu fungování bag of words metody. To znamená, že pořadí slov je zanedbatelné, kvůli zaměnitelnosti v teorii pravděpodobnosti, kde nezáleží na pořadí prvků. Zde se ale jedná o slova i o pořadí dokumentů. Nejedná se zde ovšem o nezávislost mezi danými prvky. [34]

Princip LDA se samozřejmě odvíjí od De Finettiho teorie, kde tyto vyměnitelné prvky jsou podmíněně nezávislé ve vztahu k nějaké latentní proměnné. Tudíž je tato vyměnitelná posloupnost směs posloupností nezávislých náhodných veličin. [35] Je tedy třeba dbát skrze model i na vyměnitelnost u slov i dokumentů. [34]

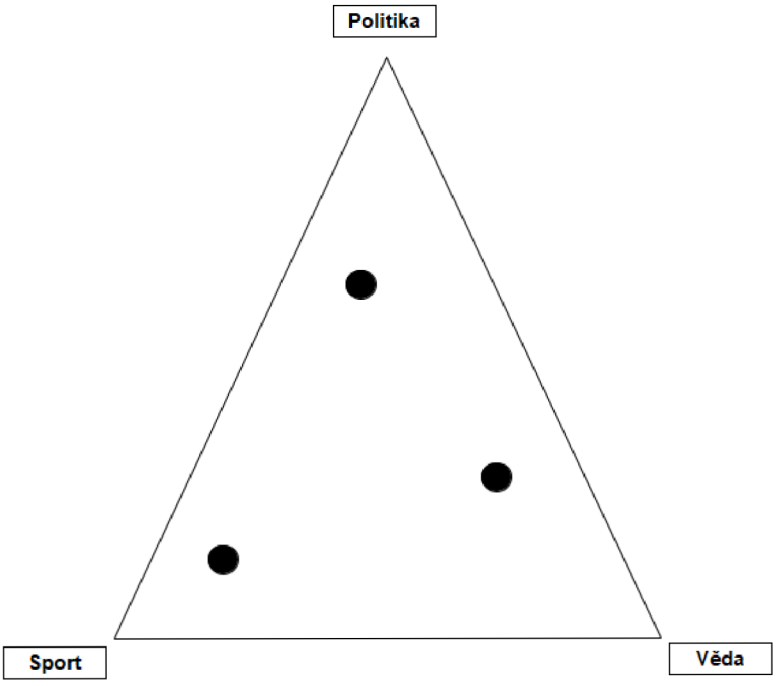


Obrázek 2 Latentní Dirichletova alokace; zdroj: [34]

LDA je tedy generativní pravděpodobnostní model korpusu, kde dokumenty jsou náhodné směsi latentních témat a kde každé téma je charakterizováno rozdělením slov. LDA má tři vrstvy reprezentace. První vrstva je korpus, tedy kolekce dokumentů M, kde se nachází parametry α a β, které jsou vybrány při generování korpusu. Proměnná θ je na úrovni dokumentu a z každého dokumentu je vybrána jen jednou. Poslední jsou proměnné *z* a *w*, jsou na úrovni slov a jsou vybrány jednou pro každé

slovo v daném dokumentu. Na obrázku 2 lze vidět grafickou reprezentaci LDA, zde je vidět šedě zbarvená proměnná *w*, protože jen tuto proměnnou lze pozorovat, ostatní proměnné jsou latentní. [34]

- M – Počet dokumentů
- N – Počet slov v dokumentu
- k – Počet témat
- α – Parametr pro rozdělení témat na dokument
- β – Parametr pro rozdělení slov podle tématu
- φ – Rozložení slov pro téma k
- θ – Rozdělení témat pro dokument *i*
- z – Téma pro *j-té* slovo v dokumentu *i*
- w – specifické slovo z dokumentu *i*



Obrázek 3 Zobrazení Dirichletova rozdělení; zdroj: vlastní zpracování

Parametry θ a φ mají multinominální rozdělení, zatímco parametry α a β mají Dirichletovo rozdělení. Dirichletovo rozdělení Dir(α) je spojité vícerozměrné

rozdělení pravděpodobnosti parametrizovaných vektorem α kladných reálných čísel. Je to vícerozměrné zobecnění rozdělení beta. Vektor α může nabývat *k* možných výsledků nebo kategorií, tedy $\alpha = \{\alpha_1, \alpha_2, \alpha_3 \ldots \alpha_k\}$, je to vektor kladných reálných čísel. Určení těchto kategorií určuje, v kolika dimenzích bude bod zobrazen v Dirichletově rozdělení. Například pro témata politika, sport a věda tedy jsou tři dimenze v rozdělení několika dokumentů D, které jsou v prostoru zobrazeny na obrázku 3 pro ilustraci. Dále jsou na obrázku 3 zobrazeny dokumenty jako černé body, které zobrazují, z kolika procent témat se daný dokument skládá a kde byl klasifikován pomocí LDA. Také záleží na výběru α, například α = 1 znamená, že vzorky jsou v prostoru rozloženy rovnoměrněji, α > 1 znamená, že vzorky se shromažďují uprostřed, a α < 1 znamená, že vzorky směřují do rohů. [36]

Výběr těchto parametrů záleží na daném korpusu dokumentů $D = \{w_1, w_2, w_3 \ldots w_M\}$. Hledají se takové parametry, které maximalizují logaritmickou pravděpodobnost. [34]

$$\ell(\alpha, \beta) = \sum_{d=1}^{M} \log p(w_d|\alpha, \beta).$$

Ačkoliv nelze zjistit horní hranici, lze zjistit nižší hranici a tu maximalizovat pro LDA. Horní hranice je nezjistitelná kvůli vazbě mezi θ a β při sčítání nad latentními hodnotami témat. Lze ji ovšem odvodit například pomocí Gibbsova vzorkování. [37] Tudíž malá hodnota α přidá méně témat k dokumentu, a naopak velká hodnota α bude mít opačný efekt. Pro malou hodnotu β použije méně slov pro modelování tématu, a naopak pro velkou hodnotu β použije více slov. Jediný parametr, který zadává člověk, je parametr k, tedy počet témat, kolik má LDA detekovat. [31]

LDA předpokládá, že každý dokument je vygenerován výběrem témat z každého dokumentu a následným výběrem slov z každého vybraného tématu. Toto vysvětluje generativní stránku modelu, kdy LDA dokáže generovat nové dokumenty, které ovšem nejsou pro člověka čitelné, ale jsou čitelné pro počítač. [37]

Generativní proces má tento postup [38]:

1. Vzorek θ z Dirichletova rozdělení $\theta_i \sim \mathrm{Dir}(\alpha)$ pro i od 1 do M
    - Náhodný výběr tématu ze vzorků

2. Vzorek φ z Dirichletova rozdělení $\varphi k \sim \mathrm{Dir}(\beta)$ pro k od 1 do K
    - Náhodný výběr slova, které popisuje dané téma
3. Vzorek $z_{ij} \sim$ multinomického rozdělení (θ_i) a vzorek $w_{ij} \sim$ multinomického rozdělení (φ_{zij}) pro i od 1 do M a pro j od 1 do N:
    - Postup je opakován, dokud nedojde k poslednímu tématu

Pravděpodobnost, že LDA vygeneruje podobný dokument s tématy jako již existuje je následující.

$$P(w, z, \theta, \varphi, \alpha, \beta) = \prod_{J=1}^{M} P(\theta_j; \alpha) \prod_{i=1}^{k} P(\varphi_i; \beta) \prod_{t=1}^{N} P(z_{j,z}|\theta_j) P(w_{j,t}|\varphi z_{j,t})$$

První dvě pravděpodobnosti mají Dirichletovo rozdělení a poslední dvě pravděpodobnosti mají multinomiální rozdělení. První součin členů udává pravděpodobnost témat na dokument, druhý součin pravděpodobnost slov na téma, poslední součin má pravděpodobnost témat na dokument a slov na téma. Tento zápis pravděpodobnosti generuje text popsaný v krocích, které jsou uvedeny výše. Pravděpodobnost $P(\theta_j; \alpha)$ udává pravděpodobnost vybrání tématu k danému dokumentu v Dirichletově rozdělení. S tím se pojí pravděpodobnost $P(z_{j,z}|\theta_j)$, ta udává pravděpodobnost vybrání tématu k danému dokumentu, která je naopak multinominální. Totéž existuje i pro slova v pravděpodobnosti $P(\varphi_i; \beta)$, která udává pravděpodobnost vybrání slova k danému tématu v Dirichletově rozdělení, a pravděpodobnost $P(w_{j,t}|\varphi z_{j,t})$, která udává pravděpodobnost výběru slova pro dané téma v multinominálním rozdělení. Takto LDA generuje kolekci dokumentů a porovnává je mezi sebou. Dokument je vybrán na základě toho, zda se správně nachází v rozdělení témat a témata jsou správně umístěna v rozdělení slov. Důležitý je zde přechod Dirichletova rozdělení na multinominální rozdělení. Multinominální rozdělení se zde používá ke generování slov, protože na rozdíl od Dirichletova rozdělení popisuje rozdělení pravděpodobnosti nad konečnou množinou diskrétních výsledků (slov ve slovníku). Hlavní myšlenkou zde je, že slova jsou vybírána z distribuce daného tématu v dokumentu. Tento hierarchický postup, který je graficky vyobrazen na obrázku 2, je důležitý pro generativní proces LDA. [39]

Vzhledem k tomu, že se jedná o generativní model, je třeba latentních proměnných, aby se z modelu mohl stát diskriminační model pro klasifikaci. Zde se nejvíce používá již zmíněné Gibbsonovo vzorkování. Je to simulační nástroj pro získávání vzorků z nenormalizované společné funkce hustoty rozdělení. Tato metoda je potřeba, protože metoda LDA má inferenční problém, jak bylo výše uvedeno. Nelze totiž odhadnout latentní proměnné a hyperparametry α a β, lze odhadnout pouze dolní hranici – v této časti nastupuje Gibbsonovo vzorkování, které problém řeší. Nutno podotknout, že vyřešit tento problém není nemožné, ale vysoce výpočetně náročné. [40]

Pro vyřešení latentních proměnných je třeba vypočítat následují rovnici.

$$p(\theta, \varphi, z|w, \alpha, \beta) = \frac{p(\theta, \varphi, z, w|\alpha, \beta)}{p(w|\alpha, \beta)}$$

Pravděpodobnost $p(w|\alpha, \beta)$ je zde ta problémová část, normalizace tohoto prvku nelze přesně vypočítat. Proto je třeba použít Gibbsonovo vzorkování, tento algoritmus je z rodiny Markovových řetězců Monte Carlo. Tyto Markovovy řetězce jsou přímo vytvořeny, aby se po několika iteracích přibližovaly do nejvíce možné podoby jako vzorek. Tento řetězec je přímo vytvořen na vzorkování z podmíněné pravděpodobnosti posteriorního rozdělení, tedy rozdělení latentních proměnných. V každé iteraci je vzorkováno z, θ, φ, zatímco ostatní proměnné jsou zafixované. Teoreticky je tento způsob také náročný, protože nelze předpovědět, kolik iterací bude třeba k dosažení výsledku, ovšem v praxi má tento algoritmus velmi dobré výsledky. [40]

Implantace Gibbsonova vzorkování v LDA je vcelku jednoduchá. Je důležité nastavit požadované proměnné $n_{d,k}$ a $n_{k,w}$ a náhodně je spustit.

- $n_{d,k}$ – počet slov v tématu k v dokumentu D
- $n_{k,w}$ – počet kolikrát je slovo w přiřazeno do tématu k

V každé smyčce je vybráno slovo pro každé téma v korpusu. Gibbsonovo vzorkování jde zpět po krocích LDA a místo přiřazení slov k tématu dělá přesný opak a přiřazuje témata ke slovu a hledá tu nejlepší shodu relativně k ostatním proměnným, které jsou fixované. Algoritmus se ptá: Jak dominantní je téma k v dokumentu D? Kolikrát bylo

téma použito v dokumentu D? Jak pravděpodobné je slovo pro téma k? Kolikrát bylo slovo *w* přiřazeno pod téma k? Po skončení algoritmu jsou počty uložené v proměnných $n_{d,k}$ a $n_{k,w}$ a ty jsou pak použity na výpočet latentních proměnných θ_d a φ_k. [40]

6.3.2 Evaluace modelu

Poslední částí je evaluace modelu. Je možné posoudit správnost modelu na první pohled jen pomocí lidské intuice, zda daná slova patří do jednoho tématu nebo ne a jestli spolu souvisí. Ovšem je třeba robustnější metody pro ověření správnosti modelu. Nejpoužívanější je zde koherence témat. Tato metrika hodnotí, jak je téma podpořeno referenčním korpusem. Referenční korpus obsahuje slova s kontextem a skrze pravděpodobnostní výsledky pak určuje koherentní skóre tématu. Koherence témat, přímá témata vytvořená LDA a referenční korpus skrze čtyři metody určí koherentní skóre *c* pro dané téma. To se skládá ze segmentace, výpočtu pravděpodobnosti, potvrzení opatření a agregace. [41]

Segmentace rozdělí slova *w* v tématu *k* na páry, kde slovo na druhé pozici bude udávat správnost slova na první pozici v páru. Je to přípravný krok pro počítání pravděpodobnosti. Slova z tématu lze promíchávat a vytvořit různé páry. [41]

Výpočet pravděpodobností v tomto kroku se počítá pro určitá slova w. Pravděpodobnost P(w) udává výskyt slova v dokumentu. Další pravděpodobností v tomto kroku jsou například Pbd (boolean document), udávají, kolikrát se slovo w objeví v dokumentu děleno počtem dokumentů D. Další je Psw (sliding window), v této pravděpodobnosti se hledá výskyt slova v takzvaném oknu, které je posouváno například deset slov před slovem w a deset slov za ním, pokud se dané slovo w nevyskytne znovu v daném oknu, nezapočítá se jako společný výskyt. [41]

Existuje i implementace Wrod2Vec skrze koherentní skóre tématu. To zařadí i sémantiku slova do jeho skóre [42]

Předposledním krokem je potvrzení opatření, který je z těchto kroků nejdůležitější. [43] Používá pravděpodobnosti vypočítané v předchozím kroku a páry, které byly vytvořeny v segmentaci. Převádí vztah mezi slovy na pravděpodobnost podle korpusu a výskytu slov v dokumentu a tím ukazuje, jak silný vztah slova v páru mezi sebou mají. Používá dvě techniky výpočtu, přímou a nepřímou metodu. Nepřímá metoda může zachytit i sémantiku mezi páry slov. [41]

Posledním krokem je agregace. Zde se jedná jen o agregaci všech výsledků z předchozího kroku do jednoho výsledku a výsledkem je í skóre na škále od 0 do 1. [41]

7 Analytická platforma KNIME

V roce 2017 vydal americký časopis The Economist článek [44], ve kterém shrnul, že dnešním nejcennějším zdrojem už není ropa nebo jiná komodita, ale data. Giganti jako Google, Amazon a Microsoft během jednoho dne shromáždí neuvěřitelné množství dat od svých uživatelů. Tato data zpracují a prodají je dále. Zpracovat se ovšem musí pomocí nějakých nástrojů pro tento úkol vytvořených. Je potřebná platforma, ke které lze připojit velké množství dat z různých zdrojů, počítat nad nimi statistické operace a dále je zpracovávat a získávat z nich vzory, které ukazují chování uživatelů na daných stránkách. Pro tento účel lze využít open-source platformu KNIME. Není to jen samostatná platforma, ale slouží i jako integrační platforma, do které lze připojit další potřebné nástroje pro datovou vědu nebo business inteligence. [45]

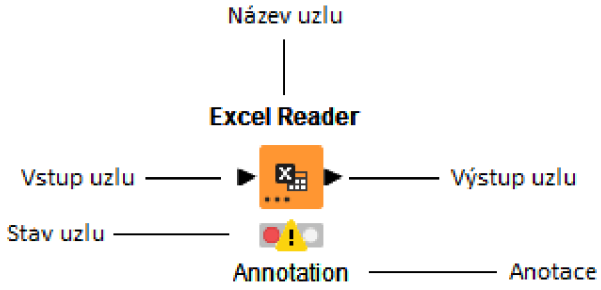
KNIME (zkratka z anglického The Konstanz Information Miner) je modulární prostředí, které skrze vizuální sestavení a interakci usnadňuje práci s daty. Umožňuje integraci nejnovějších algoritmů pro manipulaci s daty nebo jejich grafickou vizualizaci pro lepší pochopení. KNIME je navržený v programovacím jazyce Java a grafický editor je implementován pomocí plug-in Eclipse. Pomocí toho grafického prostředí je lépe viditelný workflow dat, se kterými se pracuje. Všechna data, která prochází skrze tento workflow, jsou zapouzdřena v tabulce DataTable. Tato tabulka uchovává všechna metadata o datech a jejich typech. Zároveň má každý řádek svůj unikátní identifikátor. [46]

Platforma KNIME byla navržena se třemi hlavními objektivy [46]:

- Interaktivní prostředí – tok dat lze upravit nebo rozšířit jen obyčejnými uzly, které lze vzít z nabídky a použít je ve pracovním prostředí ihned.
- Modularita – jednotlivé uzly jsou na sobě nezávislé. Data jsou zapouzdřena a typy nejsou předem definované. Nové uzly lze přidat a odebrat bez problému.
- Škálovatelnost – přidávání nových uzlů je okamžité a není třeba stahovat další podpůrné balíčky.

Základní jednotkou stavby workflow v platformě KNIME je takzvaný uzel. Uzel přijme data na vstupu a zpracuje je. Poté vysílá výsledek ze svého výstupu. Toto zpracování může být statistické modelování, manipulace s daty, vizualizace dat nebo obyčejné čtení a zápis. Na obrázku 4 lze vidět grafické zpracování uzlu a jeho popis. Jedná se

uzel, který dokáže zpracovat data z Excelu a na výstupu je uloží do tabulky DataTable. Základem je název uzlu s jeho vstupy a výstupy. Dále je zde anotace, kam lze dopsat popis uzlu a co provádí jako komentář. Poslední a důležitý prvek je stav uzlu. Tento stav udává, v jakém kroku zpracování se uzel nachází. Uzel může nabývat čtyři stavy. Červený kruh jako je na obrázku 4 značí, že uzel je neaktivní a není ještě konfigurovaný. Žlutý kruh značí, že uzel je nakonfigurovaný, ale nebyl spuštěn. Dalším je zelený kruh, který značí úspěšné provedení nebo zpracovaní akce, kterou uzel zastává. Posledním je červený kruh s křížkem uprostřed, ten značí, že se stal nějaký problém, ale zpracování bylo provedeno. [47]



Obrázek 4 Popis uzlu; zdroj: vlastní zpracování

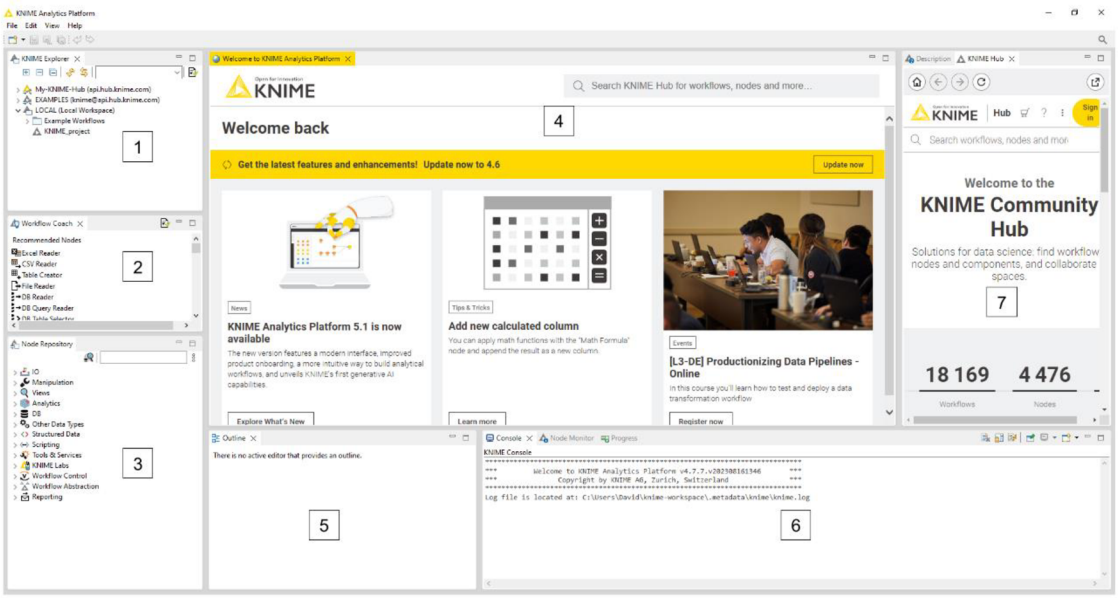
Jen uzly, které mají stejné porty, mohou být spolu spojeny. Na obrázku 4 je vidět označení portu pro datovou tabulku, která je označena černě vyplněnými trojúhelníky. Na obrázku 5 lze vidět další možné porty, které jsou dostupné v platformě KNIME. Jak již bylo uvedeno, první je zde datová tabulka. Dále jsou zobrazeny proměnné toku. To jsou takové proměnné, které jsou vyjádřeny za jednotku času. Jedná se například o množství, produkci, příjem a velikost konzumace, jsou symbolizovány červenými kruhy. [48] Model PMML, značkovací jazyk prediktivního modelu, je symbolizován modrými čtverci. Klasické připojení k databázi je symbolizováno červenými čtverci. Model souboru rozhodovacích stromů je symbolizován šedými čtverci. Poslední je port vstupu a výstupu pro obrázek a ten je symbolizován zelenými čtverci.

Obrázek 5 Různé porty uzlů; zdroj: vlastní zpracování

Použité uzly potom tvoří takzvaný workflow. Jedná se o pracovní postup, který udává, jak uzly budou zpracovávat data po sobě. Formálně se toto nazývá jako přímý acyklický graf. Acyklický graf lze popsat jako reprezentaci sérii aktivit, kde se musí dodržet jejich postup a směr. [49] Přímý v názvu udává, že směr toku dat je jen jeden a acyklický poukazuje na to, že v grafu nejsou žádné cykly. Celý tento pracovní postup je možné vidět v editoru v platformě KNIME se všemi výše popsanými stavy uzlů. [47]

Jak bylo již dříve zmíněno, všechna data jsou uložena v tabulce DataTable. K datům je přístup pomocí iterace instancí DataRow a je třeba najít unikátní identifikátor, který má daná instance dat. Vyhledávání pomocí RowID nebo indexu zde není implikováno z důvodu škálovatelnosti. Kvůli tomu neukládá platforma KNIME v paměti informace o všech řádcích a může tak pracovat s velkým obsahem dat, pokud ovšem velikost dat přeteče limit, je poté část tabulky s daty uložena na pevný disk. [47]

Dále je třeba představit grafické prostředí platformy KNIME, ve které je praktická část práce prováděna. Předem je důležité zmínit, že mnoho návodů a různých podpůrných materiálů je dostupných online právě díky tomu, že platforma KNIME je open source. Existuje fórum, kde se dá dohledat cokoliv, co se týká platformy KNIME [50]. Dále je tu Community Hub [51], což je veřejné úložiště pro workflow, příklady použití, rozšíření i uzly, které se v platformě používají. Také lze najít KNIME self-paced courses, neboť tým, který pracuje na vývoji platformy, dal k dispozici výukové materiály pro práci s platformou KNIME. [55]



Obrázek 6 Prostředí platformy KNIME; zdroj: vlastní zpracování

Obrázek 6 ukazuje uvítací grafickou obrazovku rozhraní platformy KNIME při jejím prvním spuštění. Je rozdělena do sedmi sekcí, které jsou dále popsány. Jako každý program má i tento hlavní menu, kde se nachází možné volby, jako je soubor, editace, pohled a nápověda. Hned pod hlavním menu se nachází panel nástrojů, který také není nijak výjimečný oproti ostatním a obsahuje základní nástroje jako uložit, otevřít a tlačítka zpět a dopředu. Důležité jsou již zmíněné sekce, do kterých je grafické rozhraní rozděleno.

Platformu KNIME tvoří [47]:

1. KNIME Explorer. Tento panel ukazuje seznam dostupných workflow ve spuštěné instanci. Při prvotním spuštění nabízí několik příkladů workflow v odvětví péče o zákazníky a maloobchodního prodeje
2. Workflow coach. Tato sekce je poněkud unikátní, neboť nabízí různé uzly, které s největší pravděpodobnostní budou navazovat na uzel, který byl aktuálně použit. Je zde vyobrazen název uzlu a pravděpodobnost jeho použití v procentech.
3. Repozitář uzlů. Zde jsou na výběr všechny možné uzly v platformě KNIME. Další uzly se dají doinstalovat pomocí nápovědy, pokud nabídka není dostatečná.
4. Workflow editor. Na obrázku 6 je zobrazena uvítací stránka při prvním spuštění, ale běžně se v tomto podokně sestavují již zmíněné workflow.

5. Outline. Zde se nachází menší verze workflow editoru, která vypomáhá, pokud se pracuje s workflow, který už svou velikostí přesahuje podokno editoru.
6. Konzole. V této oblasti se vypisují možné chyby a upozornění pro uživatele.
7. KNIME Community Hub. Zde je možné vyhledávat materiál k platformě KNIME zároveň se zde po vybrání uzlu objeví celý popis jeho funkcí.

8 Modelování extrakce témat

Po představení platformy KNIME a základních prvků je na řadě už samotná extrakce témat z nestrukturovaného textu. Praktická část je provedena ve verzi platformy 4.7.7., která byla vydána 23. 8. 2023 a v době psaní této práce byla nejaktuálnější stabilní verzí.

8.1 Selekce dat

Dataset pro extrakci témat z nestrukturovaného textu byl vybrán ze stránky Kaggle.com, která se sama propaguje jako „AirBnB pro datové vědce“. Jedná se o platformu, na kterou byla datovými nadšenci uspořádána sbírka, dále ji vyvíjí a školí nováčky i datové veterány. [53]

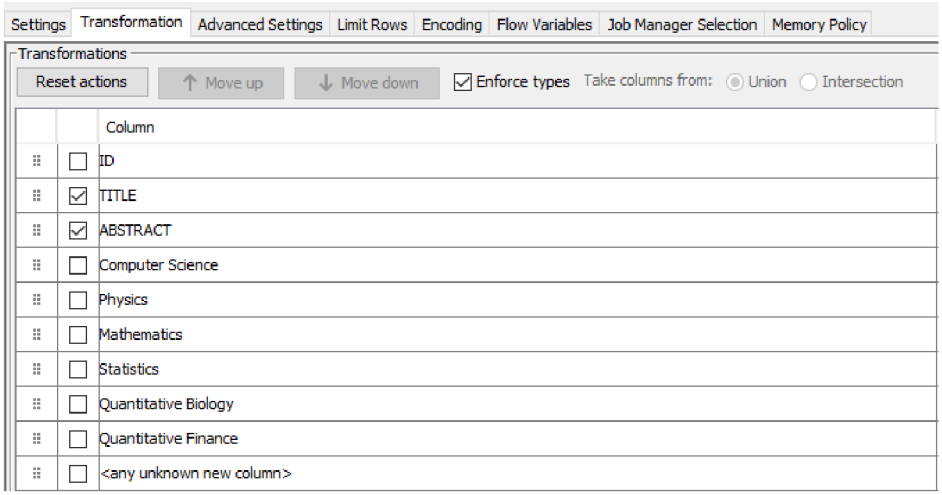
Jde o dataset s výzkumnými články. Výzkumné články nespadají pod kategorii nestrukturovaného textu, protože mají kapitoly, ale v tomto datasetu jsou vybrány jen jejich nadpisy a abstrakty. Z toho se budou extrahovat témata v platformě KNIME. Dataset disponuje 21 000 unikátními záznamy mezi daty 1. 2. 2017 a 9. 7. 2020. Tento dataset byl vybrán kvůli jeho skóre na stránce Kaggle, kde dosahuje hodnocení 10/10 v použitelnosti, ale i v kategoriích důvěryhodnosti, úplnosti a kompatibility. Bylo proto předpokládáno, že by se neměly objevit žádné problémy se selekcí dat. Dataset je také pod licencí Database contents license, takže je volně k použití. Dostupný na URL: https://www.kaggle.com/datasets/blessondensil294/topic-modeling-for-research-articles?select=train.csv. [54]

První věc, kterou bylo potřeba provést, je načtení vybraných dat do platformy KNIME. Jak bylo uvedeno výše, na to už existuje uzel, který je možné využít. Data jsou ve formě souboru CSV (Comma Separated Values), kde jsou data oddělena od sebe čárkou. Tudíž je třeba použít uzel přímo na čtení dat z CSV souborů.

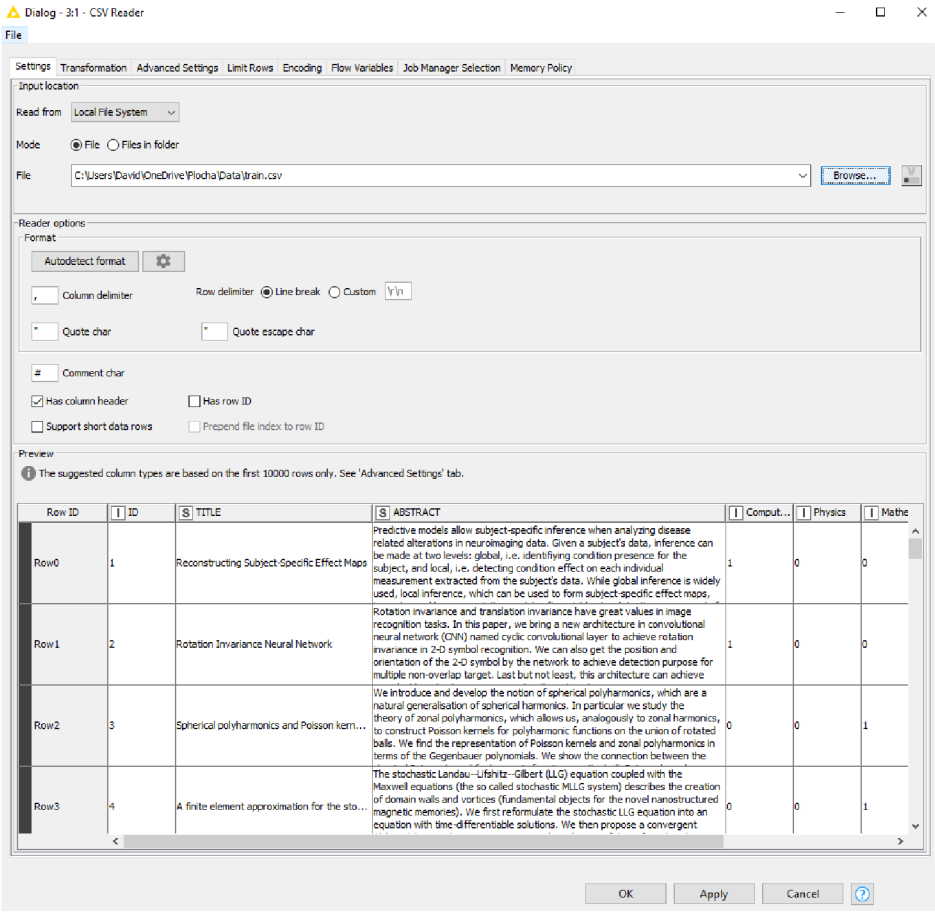
Obrázek 7 Uzel na čtení souborů CSV; zdroj: vlastní zpracování

Nyní je třeba vybrat a nakonfigurovat soubor, který bude použit pro extrakci témat. Na obrázku číslo 8 je vidět výběr souboru dat. Hned pod tímto výběrem se nachází specifikace rozdělovačů pro soubor. Je možné vybrat autodetekci a platforma KNIME vybere správný rozdělovač nebo vyplnit do kolonek vlastní znaky pro oddělovače. Pokud CSV obsahuje v prvním řádku metadata o sloupcích, je třeba zaškrtnout „has column header“, aby tato data nebyla dále použita a fungovala jen jako popisky sloupců pro orientaci. Hned pod tímto výběrem je možné vidět náhled dat, jak byla rozvržena, a zda zde není problém v rozdělovači. V náhledu se zobrazuje jen prvních tisíc řádků. Je možné si všimnout, že data, která budou použita, mají ještě dalších 6 sloupců. Tyto sloupce symbolizují, k jakému tématu daný článek patří. Dokumenty mohou patřit do počítačové vědy, fyziky, matematiky, statistiky, kvantitativní biologie nebo kvantitativních financí. S těmito sloupci se nadále nebude pracovat. Je možné přejít na záložku Transformace a zde je odebrat ze seznamu. Tyto sloupce budou ke konci použity na kontrolu a přesnost modelu.

Na obrázku 9 lze vidět záložku transformace, kde byly odebrány všechny ostatní sloupce, které nebudou použity dále v extrakci témat. Sloupec ID byl také odebrán, protože platforma KNIME používá vlastní unikátní identifikátor RowID. Doporučuje se také odebrat možnost „any unknown new column“, protože by se mohly v dokumentu objevit prázdné sloupce a hodnoty. [47]



Obrázek 8 Konfigurace CSV Reader uzlu; zdroj: vlastní zpracování

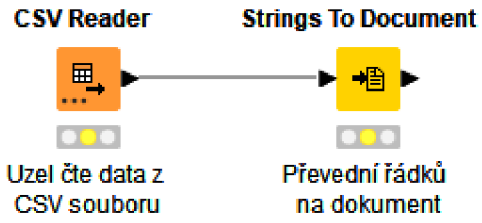


Obrázek 9 Záložka transformace; zdroj: vlastní zpracování

Na dalších záložkách, které ukazuje obrázek 8, jsou možné další akce s dokumentem CSV, jako je limitování řádků. Zde se dá také nastavit přeskočení prvního řádku, pokud by obsahoval nesouvisející data. Poté je možné nastavit i jen určitý počet řádků, se kterými se bude pracovat. Další záložka je kódování, kde je třeba nastavit kódování dokumentu. Standardem je UTF-8 pro CSV souboru. Poté následuje flow variables záložka, kde lze vidět, jak na sebe navazují předchozí záložky a jak spolu spolupracují. V záložce job manager selection lze vybrat, jak bude workflow předáván do dalšího uzlu a jak budou na sebe uzly navazovat. Zde je nejlepší možností ponechat defaultní nastavení. Poslední záložkou je memory policy, zde je možné vybrat, jestli se primárně data a tabulky budou ukládat na disk nebo budou uloženy v cache rychle přístupné paměti RAM. Standardně je zde vybrána možnost s pamětí RAM. Po projetí a vyplnění všech těchto záložek stačí jen potvrdit a stav uzlu se změní z červené na žlutý, což značí, že uzel je nakonfigurovaný, ale žádná akce nebyla provedena. [47]

Jsou zde ale jen záznamy v tabulce, pro vytvoření dokumentů je třeba z každého řádku vytvořit jeden dokument. Pro tuto situaci existuje uzel Strings to document.

Transformuje zadané řetězce do formy dokumentů. Pro každý řádek se vytvoří nový dokument a tento dokument se k němu připojí. Řetězce z předem specifikovaných sloupců slouží jako název, autor nebo obsah dokumentu. Takto jsou vytvořeny dokumenty, ze kterých je poté extrahováno téma. [55]



Obrázek 10 Workflow pro vytvoření dokumentů; zdroj: vlastní zpracování

V tomto uzlu už nastává při vytvoření dokumentu takzvaná tokenizace, kdy každé slovo stojí o samotě jako token v dokumentu pro zpracování přirozeného jazyka. Vzhledem k tomu, že budou dokumenty v angličtině, je doporučen OpenNLP English WordTokenizer. Pro jiné jazyky se doporučuje OpenNLP Whitespace Tokenizer, který bere slovo jako každý znak mezi dvěma mezerami. Bohužel nefunguje u většiny asijských jazyků jako mandarínština, korejština a japonština. [55]

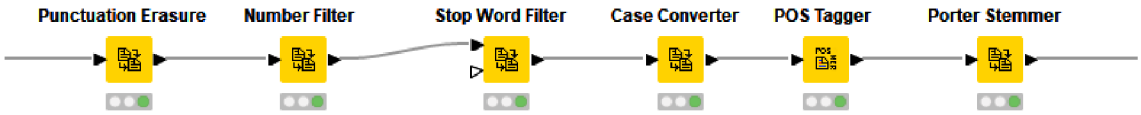
8.2 Předzpracování dat

Nyní je čas předzpracovat data, aby se z nich mohla extrahovat témata. To obnáší čtyři různé procesy, které mají za úkol převést text do podoby, se kterou už bude počítač moci pracovat.

Jedná se o tyto procesy [55]:

1. Tagování – přidávání značek k slovům za účelem obohacení textu o informace. Příkladem je zde Part Of Speech Tagging.
2. Čištění – odstranění interpunkce, stop slov a i čísel.
3. Stemming/Lammetization – převedení na kořen slova nebo transforamce na základní tvar slova.
4. Transformace – pro LDA není potřeba, ale je použita při výběru uzlu BoW dále v praktické části.

Všechny tyto procesy je možné použit v platformě KNIME. Pro přehlednost workflow je vytvořen takzvaný meta uzel. To je uzel, který v sobě obsahuje další uzly nebo komponenty. [47] Výše popsané procesy budou reprezentovány jednotlivými uzly v platformě KNIME.



Obrázek 11 Uzly pro předzpracování dat; zdroj: vlastní zpracování

První uzel Punctuation Erasure zařizuje, že bude z dat odstraněna všechna interpunkce, která by v modelování mohla dělat problémy. Dalším uzlem je Number filter, který se postará, aby v dokumentu nebyly žádné číslice. V tom jsou obsažena i všechna matematická znaménka jako + a –, desetinné čárky nebo tečky. Následuje Stop Word Filter, který odstraňuje stop slova, která jsou v jazyce běžná, ale nepřidávají žádné informace. Protože jsou vybraná data v angličtině, budou to zejména předložky typu a, an a the. Tento uzel už má v sobě zabudovaný korpus stop slov pro šest jazyků a defaultně je zde vybrána angličtina.

Jak je zřetelné na obrázku 11, uzel na odebrání stop slov má dva vstupy a je to zapříčiněno tím, že je možné přidat vlastní korpus se stop slovy, které by uživatel nechtěl zachytávat v extrakci témat. Obdobou tohoto je uzel Dictionary Filter. Pokud je třeba odebrat z dokumentu slova, o kterých je předem známo, že jsou v dokumentu, ale nejsou v běžném korpusu pro stop slova, lze použít tento uzel, který také používá uživatelem vytvořený korpus.

Následující uzel je Case Converter, který změní velká písmena na malá nebo opačně, záleží na tom, jaké nastavení si uživatel vybere. Další uzel je Part Of Speech Tagger, tento uzel přiřazuje slovní druhy ke slovům v textu. Používá knihovnu Penn Treebank Project pro tagování, které má normalizované značení slovních druhů. Předposledním uzlem je Porter Stemmer neboli stemming, jak bylo zmíněno v teoretické části. Tento uzel převede slova na jejich kořen. Pokud by bylo potřeba, jsou zde i uzle na rozpoznání pojmenovaných entit, které už jsou předem navržené, a je zde jen potřeba vybrat, jakou entitu chce uživatel sledovat v textu. Na výběr jsou lidé, lokace, organizace, datum, čas a peníze. [55]

Po proběhnutí práce meta uzlu pro předzpracování dat je možné si prohlédnout data pomocí uzlu Document Viewer. Tento uzel se používá, protože tabulka, kterou obsahuje každý uzel, není přehledná a nedá se v ní vyhledávat, je proto lepší použít tento uzel. Tento uzel nemá žádný zvláštní dopad na workflow a stačí ho jen vložit a vyvést k němu spoj z konce uzlu, kde je třeba sledovat data.

Na obrázku 12 lze vidět nezpracovaný text, který ještě člověk dokáže přečíst bez problému. Text má všechny náležitosti, které se od něj očekávají, jako interpunkce, časovaná slovesa, číslice a spojky.

3D ab initio modeling in cryo-EM by autocorrelation analysis

UNKNOWN

Single-Particle Reconstruction (SPR) in Cryo-Electron Microscopy (cryo-EM) is the task of estimating the 3D structure of a molecule from a set of noisy 2D projections, taken from unknown viewing directions. Many algorithms for SPR start from an initial reference molecule, and alternate between refining the estimated viewing angles given the molecule, and refining the molecule given the viewing angles. This scheme is called iterative refinement. Reliance on an initial, user-chosen reference introduces model bias, and poor initialization can lead to slow convergence. Furthermore, since no ground truth is available for an unsolved molecule, it is difficult to validate the obtained results. This creates the need for high quality ab initio models that can be quickly obtained from experimental data with minimal priors, and which can also be used for validation. We propose a procedure to obtain such an ab initio model directly from raw data using Kam's autocorrelation method. Kam's method has been known since 1980, but it leads to an underdetermined system, with missing orthogonal matrices. Until now, this system has been solved only for special cases, such as highly symmetric molecules or molecules for which a homologous structure was already available. In this paper, we show that knowledge of just two clean projections is sufficient to guarantee a unique solution to the system. This system is solved by an optimization-based heuristic. For the first time, we are then able to obtain a low-resolution ab initio model of an asymmetric molecule directly from raw data, without 2D class averaging and without tilting. Numerical results are presented on both synthetic and experimental data.

Obrázek 12 Nezpracovaný text; zdroj: vlastní zpracování

Po předzpracování textu na konci uzlu je výsledkem text na obrázku 13, který pro člověka nemá velkou informační hodnotu a stěží lze pochopit, o co se jedná. Zde byla odstraněna interpunkce, číslice, které stály o samotě, stop slova a změna na kořen slova. Takto předzpracovaný text lze použít v LDA pro detekci témat a poté vizualizovat výsledky.

3d ab initio model cryo-em autocorrel analysi

UNKNOWN

single-particle reconstruct spr cryo-electron microscopi cryo-em task estim 3d structur molecul set noisi 2d project unknown view direct algorithm spr start initi refer molecul altern refin estim view angl molecul refin molecul view angl scheme call iter refin relianc initi user-chosen refer introduc model bia poor initi lead slow converg furthermor ground truth avail unsolv molecul difficult valid obtain result creat qual ab initio model quickli obtain experiment data minim prior valid propos procedur obtain ab initio model directli raw data us kamautocorrel method kammethod lead underdetermin system miss orthogon matric system solv special highli symmetric molecul molecul homolog structur avail paper knowledg clean project suffici guarante uniqu solut system system solv optimization-based heurist time abl obtain low-resolution ab initio model asymmetr molecul directli raw data 2d class averag tilt numer result synthetic experiment data

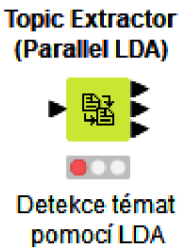
Obrázek 13 Předzpracovaný text; zdroj: vlastní zpracování

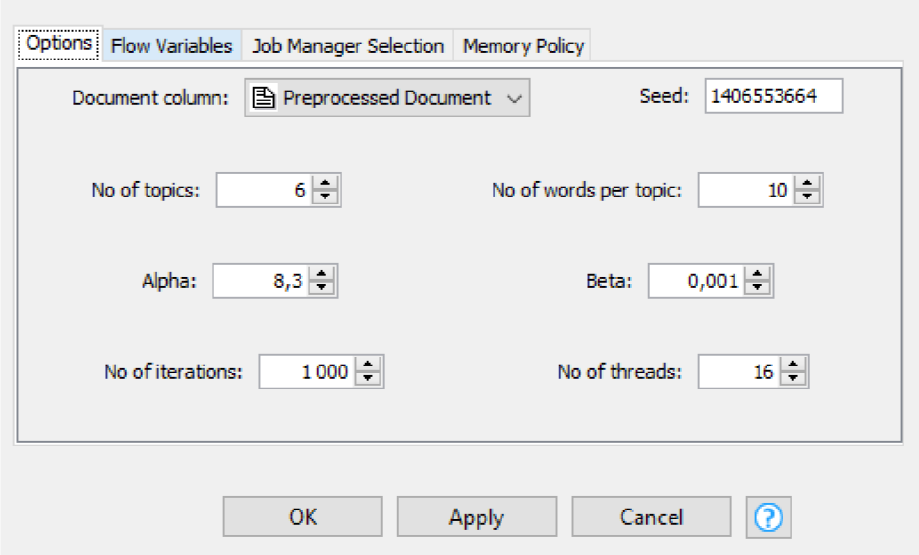
8.3 Extrakce témat

Po zpracování textu je možné přejít již k dané detekci či extrahování témat pomocí LDA. Jak bylo uvedeno v teoretické části práce, jedná se o pravděpodobnostní model. Přesněji LDA je generativní pravděpodobnostní algoritmus bez dohledu, který zjistí *k* nejvýznamnějších témat v souboru dat popsaných *N* nejrelevantnějšími klíčovými slovy. K práci s LDA musíme přijmout jisté statistické předpoklady [55, 34]:

1. Pořadí slov v dokumentu není důležité.
2. Pořadí dokumentu v souboru dokumentu není důležité.
3. Počet témat je třeba znát předem.
4. Stejné slovo může patřit do více témat.
5. Dokument *d* je chápán jako směsice témat *k.*
6. Téma *k* má multinominální rozdělení nad slovníkem slov *w.*

V platformě KNIME existuje spousta variant LDA, ale v této praktické části bude použita varianta Topic Extractor (Parallel LDA). Tato varianta má paralelní zpracování LDA, na které navazuje Sparse LDA sampling metoda, což je jiné pojmenování pro Gibbsonovo vzorkování, které bylo upraveno pro práci s LDA. [55]





Obrázek 14 Uzel a konfigurace LDA; zdroj: vlastní zpracování

Tento uzel bude navazovat na předzpracovaná data, která vycházejí z meta uzlu a už jsou připravená ke zpracování tímto uzlem. Nyní je třeba nastavit uzel LDA a tím je myšlen výběr α, β a *k*. Také lze vidět, že tento uzel má tři výstupy, z každého výstupu jsou jiné informace, shora je první tabulka dokumentu s tématy, druhý výstup ukazuje slova, která popisují téma. S tímto výstupem se poté bude pracovat dále. Posledním, třetím výstupem je výstup s iteračními statistikami. Nyní je třeba nakonfigurovat uzel LDA.

V okně na obrázku 14 je vyobrazen uzel a konfigurace LDA v platformě KNIME. Je zde vidět Document column, což udává, který dokument se má zpracovávat, a napravo od něj je seed, což je textové okno pro náhodné číslo. Pro jiný seed běží algoritmus jinak, pokud by uživatel chtěl získat stejné výsledky, je třeba použít stejný seed. Dále zde je počet témat, které už předem známe, je jich šest, proto v této části bude $k = 6$. Dalším textovým polem je možnost výběru, kolik chce uživatel slov, která budou popisovat téma. Pro tuto praktickou část to bude 10 slov. Dále jsou zde parametry Dirichletova rozdělení α a β. K těmto parametrům dosud není žádná teorie, která by pomohla vybrat ty nejoptimálnější hodnoty. Pár teorií již vzešlo, ale spíše se týkají klasifikace témat jako Elbow Method, kde se používá metoda shlukování k-means. Obvykle se uvažuje tento vztah: $\alpha = 50/k$, a proto je vybráno 8,3 a β je defaultně 0,01. Ale jelikož se jedná o vědecké články, které mají velmi podobný korpus, byl parametr β změněn na 0,001.

Pokud uživatel chce změnit tyto parametry, je důležité mít na paměti vztahy těchto parametrů [55, 32]:

- α – vyšší hodnota α znamená, že dokumenty jsi jsou více podobné
- β – vyšší hodnota β znamená, že si jsou témata více podobná mezi sebou

Posledními dvěma textovými poli je počet iterací, který je také defaultně nastaven na 1 000 opakování, zvýšení tohoto počtu má za následek, že proces LDA bude trvat déle. Další je počet vláken, které současně poběží najednou, je také defaultně nastaven na šestnáct. Po spuštění uzlu je přečteno všech dvacet jedna tisíc dokumentů a provedeno tisíc iterací generativního modelu.

Table "default" - Rows: 60 | Spec - Columns: 3 | Properties | Flow

Row ID	S Topic id	S Term	D Weight
Row0	topic_0	learn	8,648
Row1	topic_0	network	7,210
Row2	topic_0	model	6,055
Row3	topic_0	method	4,054
Row4	topic_0	data	3,958
Row5	topic_0	train	3,873
Row6	topic_0	neural	3,753
Row7	topic_0	imag	3,637
Row8	topic_0	propos	3,446
Row9	topic_0	deep	3,351

Obrázek 15 Výsledek LDA pro téma 0; zdroj: vlastní zpracování

Na obrázku 15 lze vidět výsledky pro téma 0, kde je slovo, které toto téma popisuje, a vedle něho je jeho váha. Jednotlivé váhy, které jsou přiřazeny slovům, popisují význam, který má toto slovo při generování konkrétního tématu. Výsledky lze vyexportovat pomocí uzlu excel writer do souboru xlsx a jsou uvedeny v příloze 1. této práce. Na obrázku 15 je možné vidět, že téma ještě nemá název a má pouze topic id, které ho pouze označuje pro viditelnost a rozpoznání od jiných výsledků témat. Výsledky pro další témata jsou uvedeny v následující kapitole.

8.4 Výsledky a vizualizace

Níže je prezentován plný rozsah výsledku pro $k = 6$. Je dáno šest témat se slovy, která je nejvíce vystihují. Protože není pro téma štítek, je uvedeno několik způsobů, které lze použít pro identifikaci tohoto štítku neboli názvu tématu.

Topic_0 = (learn, network, model, method, data, train, neural, propos, imag, deep)

Topic_1 = (space, function, result, graph, prove, algebra, gener, equat, set, bound)

Topic_2 = (systém, phase, magnet, field, quantum, model, effect, energi, dynamic, interact)

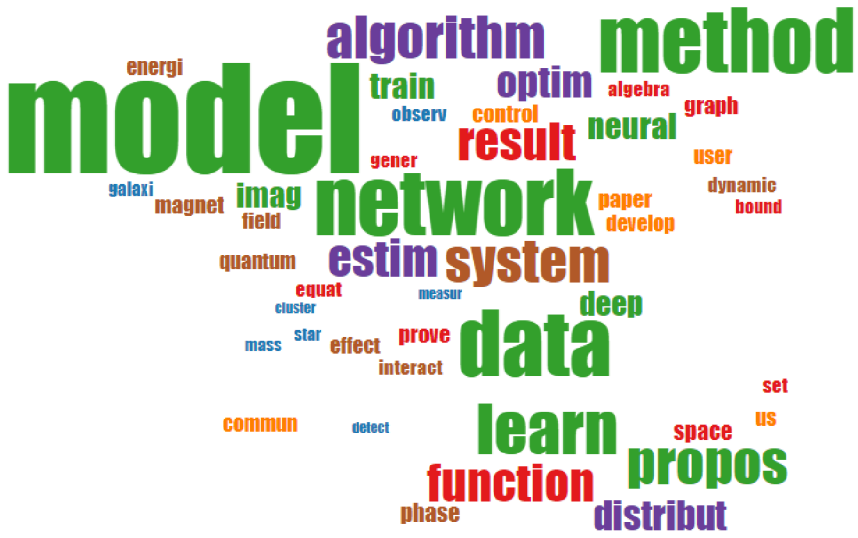
Topic_3 = (model, method, algorithm, estim, optim, distribut, data, function, propos, result)

Topic_4 = (observ, galaxi, star, mass, model, measur, detect, cluster, result, data)

Topic_5 = (systém, network, data, model, paper, control, user, us, commun, develop)

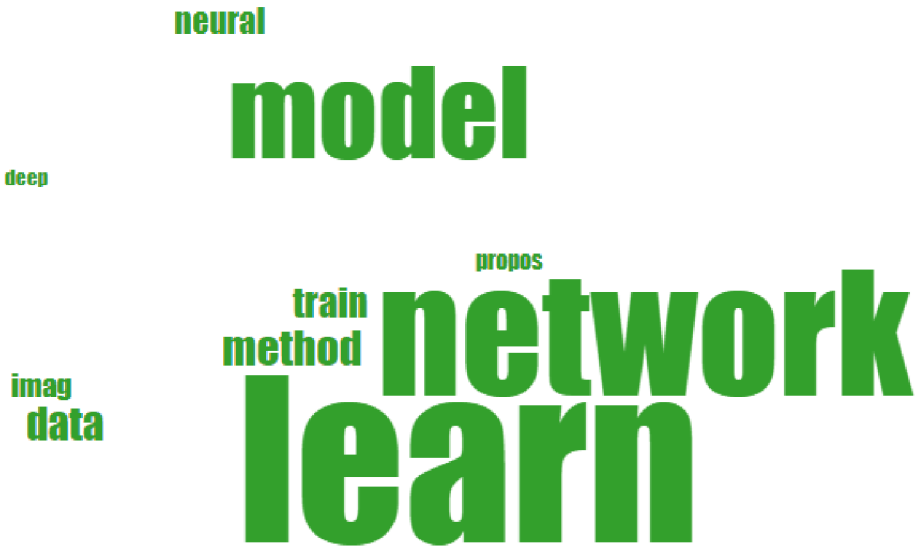
První je pomocí vizualizace, kdy je vytvořen word cloud z nejčastějších slov a touto vizualizací se ukážou důležitá slova pro téma podle jejich vah. Je to vytvořeno pomocí uzlu Color Manager, kde jsou vybrány různé barvy pro každé téma, aby je šlo lépe rozeznat, a poté pomocí uzlu Tag cloud, který už vytvoří word cloud. V uzlu Tag cloud jen třeba nastavit, jak se bude vytvářet (slova v tématu), velikost slova bude podle weight neboli váhy slova v tématu.

Na obrázku 16 lze vidět word cloud pro všechna témata, tmavě zelená slova převažují velikostí, protože mají největší váhu. Velikost je způsobena také agregovanými výsledky, protože dokument může patřit do více témat a vědecké články používají podobný korpus napříč tématy, proto jsou na obrázku 16 slova agregována. Tato vizualizace jen ukazuje důležitá slova a jejich váhu místo jen zápisu tabulce, aby měl uživatel představu o důležitosti slov pro každé téma. Nyní tyto word cloudy budou provedeny pro jednotlivá témata. V tomto případě pomocí váhy, která zde bude transformována ve velikost slova v grafu, tak bude zjištěna jeho důležitost a lze se rozhodnout pro označení tématu. Pro vybrání jen určitého tématu vizualizací je potřeba použít uzel Row Filter, který bude mít jako podmínku název tématu, jako je topic_0 nebo topic_1. Tato témata a slova pak pustí dále ke zpracování a je tak vytvořena vizualizace jen pro jedno téma místo všech.

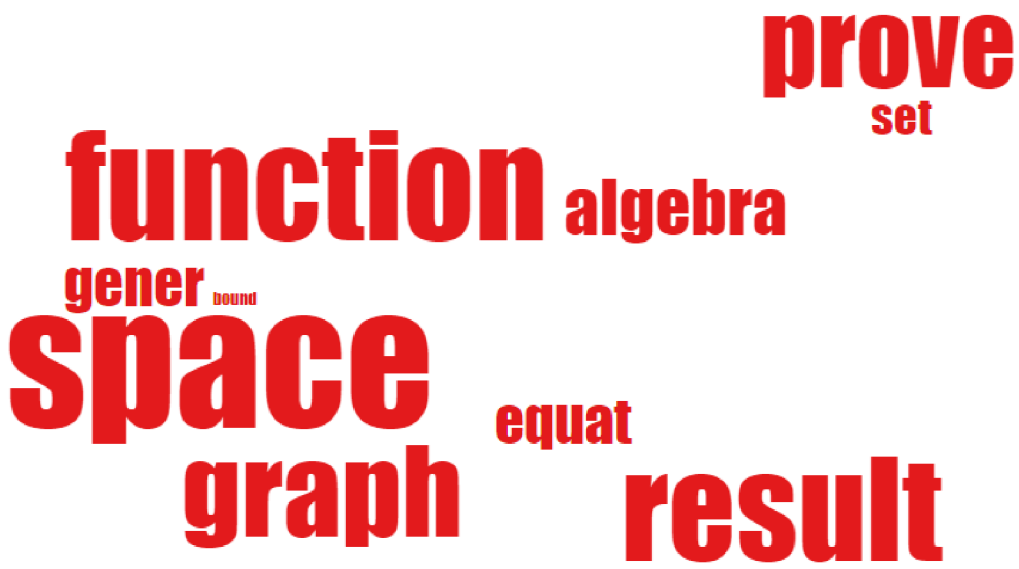


Obrázek 16 Vizualizace výsledků ve word cloudu; zdroj: vlastní zpracování

Na obrázku 17 lze vidět word cloud pro topic_0, kde jsou vizuálně zvýrazněna slova, která mají pro téma větší význam neboli váhu oproti ostatním. Ze slov, jako je učit, sítě, model, data, neural, hluboké, lze předpokládat, že se jedná a statistiku.



Obrázek 17 Vizualizace topic_0; zdroj: vlastní zpracování

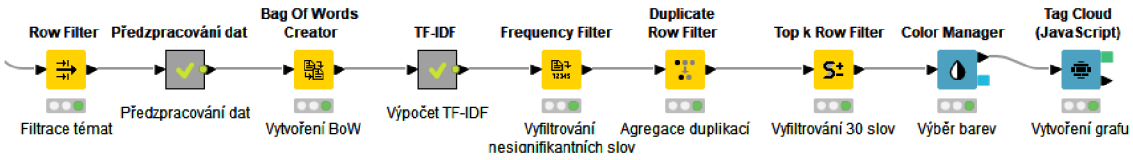


Obrázek 18 Vizualizace topic_1; zdroj: vlastní zpracování

Na obrázku 18 je vidět word cloud pro topic_1, kde jsou zvýrazněná slova prostor, funkce, graf, výsledek, důkaz a algebra. Díky těmto slovům můžeme usoudit, že se jedná o téma matematika. Tuto analýzu můžeme provést pro ostatní témata a dostaneme následující výsledky v podobě označení pro témata.

- Topic_0 = Statistika
- Topic_1 = Matematika
- Topic_2 = ….
- Topic_3 = ….
- Topic_4 = Fyzika
- Topic_5 = IT

Po analýze prvních čtyř dojde k problému, kdy důležitá slova pro téma jsou si podobná a nejde mezi nimi jasně určit, o čem jaké téma je s velkou určitostí. Toto ovšem dává smysl, protože dokument může patřit do jednoho či více témat. V této situaci přijde na řadu BoW a TF-IDF, tyto metody budou použity na filtrování nejvíce používanějších slov v dokumentech, které mají dané téma přiřazené pomocí LDA. Je třeba vytvořit další workflow, který bude vycházet z LDA a bude používat další uzly.



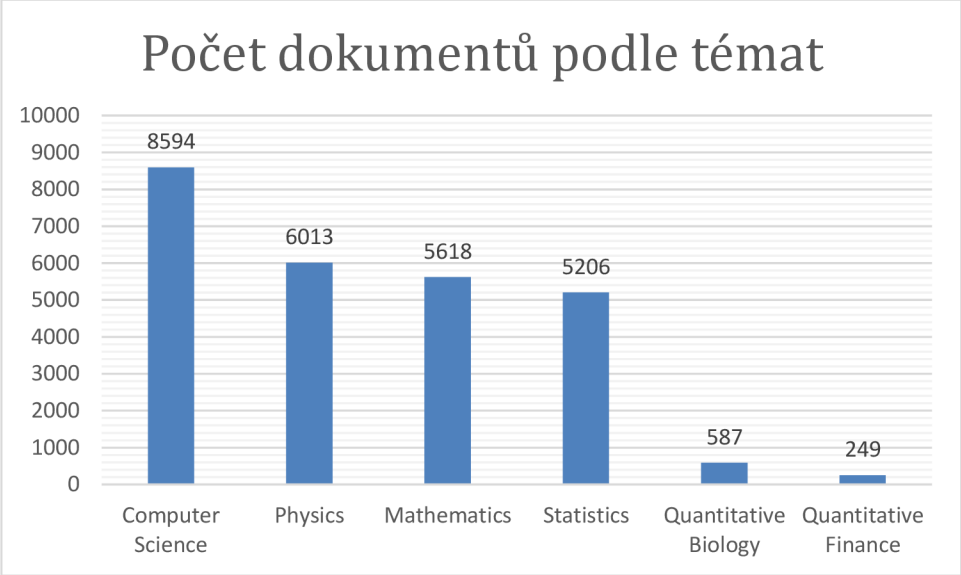
Obrázek 19 Workflow pro BoW a TD_IDF; zdroj: vlastní zpracování

Nejdříve přijde na řadu uzel Row Filter, podle kterého lze vyfiltrovat dokumenty, které mají přiřazené téma, jež je sledováno. V tomto příkladu jsou použita zbylá dvě témata topic_2 a topic_3. Vyfiltrované dokumenty jsou zpracované v meta uzlu pro předzpracování dat, kde se odstraní interpunkce, čísla, stop slova a dictionary filtr, jenž obsahuje slova, která měla témata podobná a způsobila, že výsledky nešlo odlišit od sebe. Jedná se o slova function, result, system, network, data, model, propos a method. Není zde stemming, protože už není třeba zkracovat slova na kořenový tvar, neboť výsledek bude číst člověk a počet řádků skrze filtr je velmi omezen. Přijde na řadu BoW, kde se dokumenty transformují na vektor kvůli zpracování pomocí TF-IDF. [28]

Další meta uzel je pro počítání TF-IDF. Obsahuje tři uzly pro výpočet TF, IDF a poté vynásobení mezi sebou skrze uzel Math Formula. Hodnota TF-IDF je vyhodnocena pomocí dvou metrik, jednou je metrika, kolikrát se slovo v dokumentu vyskytuje, a druhou pak inverzní frekvence dokumentu daného slova v souboru dokumentů. Inverzní frekvence dokumentu značí, jak časté nebo vzácné je slovo v celém souboru dokumentů. [12] Poté přijde na řadu frekvenční filtr, který vyfiltruje nesignifikantní slova podle kritérií z TF-IDF: čím blíže nule je výsledek, tím více je slovo signifikantní, hranice byla nastavena od 0 do 0,025. [12] Následujícím uzlem je Duplicate Row Filter, který se zbaví duplicitních výsledků slov. Dalším uzlem je top k Row Filter, který vybere prvních 30 slov seřazených podle výsledků TF-IDF sestupně. Dále jsou uzly pro vizualizaci, jako LDA je zde Color Manager, u kterého jsou vybrány stejné barvy pro téma jako u LDA a tag cloud, který vytvoří word cloud na konci toho workflow.

Po následné analýze se stále nedaří k tématům 2 a 3 přiřadit správná označení. I přes tento workflow jsou klíčová slova stále podobná a nelze přesně určit, jak tato témata označit. Po zkoumání datasetu lze z grafu 1 dojít k následujícímu závěru. Je viditelné, že témata, která byla identifikována, mají minimálně 5 000 tisíc dokumentů v datasetu, na rozdíl od témat biologie a finance, které mají 587 a 249. Vzhledem

k velikosti těchto vzorků nemůže LDA přesně učit rozdíl mezi nimi. Jedná se o výzkumné články, takže mají hodně podobný korpus, který používají, a podobná slova pak LDA používá jako klíčová slova pro téma.



Graf 1 Počet dokumentů podle témat; zdroj: vlastní zpracování

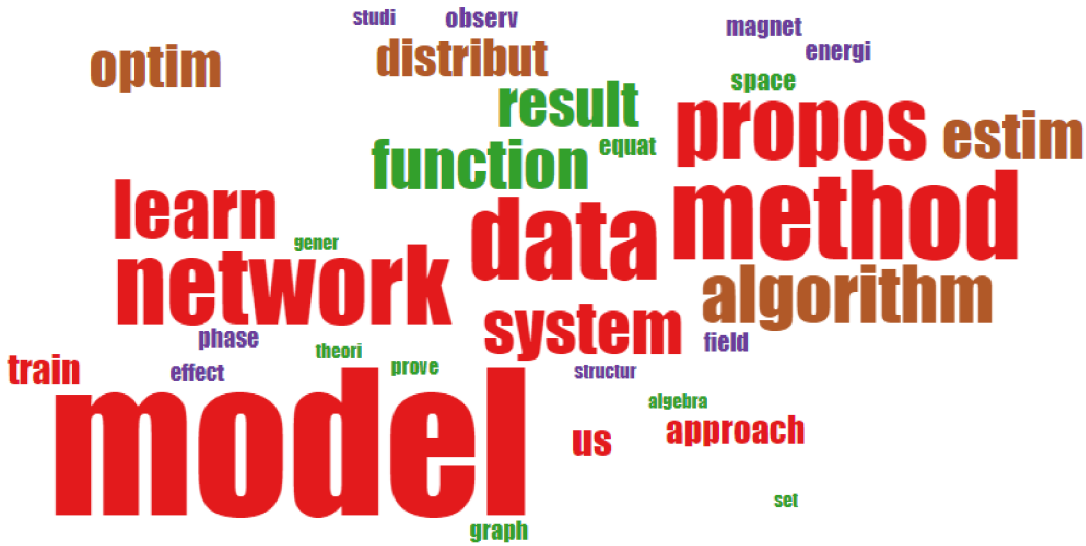
Jednou z možností je tato data odstranit a zjistit, jak se model bude chovat poté. Předtím se ale doporučuje upravit hyperparametry *k, α* a *β*. Obvykle se uvažuje tento vztah: α = 50/k, pak α po odstranění dvou témat bude 12,5. Parametr β může zůstat stejný 0,001. [55]

8.5 Úprava modelu

V datasetu byl odstraněn sloupec pro biologii a finance. V modelu byly jen upraveny parametry k = 4, α = 12,5 a parametr β zůstane stejný 0,001. Po úpravě LDA uzlu dostáváme následující výsledky pro témata:

- Topic_0 = (equat, graph, result, prove, gener, theory, algebra, set)
- Topic_1 = (network, learn, model, data, systém, us, propos, train, method, approach)
- Topic_2 = (model, method, algorithm, estim, optim, distribut, propos, data, function, result)
- Topic_3 = (observe, energi, field, system, model, magnet, phase, effect, studi, structur)

Opět jsou zde výsledky, kdy se ve třech ze čtyř témat vyskytuje slovo model. Přesto i tak slova už lépe reprezentují daná témata a dají se od sebe odlišit mnohem lépe než předchozí iterace. Pokud zde nastane problém, lze znovu použít meta uzel na dohledání klíčových slov pro téma, který vypíše top 30 slov v dokumentech, kterým bylo přidáno toto téma.



Obrázek 20 Word cloud pro k=4; zdroj: vlastní zpracování

Na obrázku 20 lze vidět word cloud, který byl vytvořen pro nynější iteraci modelu kdy $k = 4$. Topic_0 má tmavě zelenou barvu. Topic_1 je zbarveno do červena, topic 2 je zobrazeno pomocí hnědé barvy a poslední topic_3 je fialové. Podle těchto výsledků lze označit témata takto:

- Topic_0 = Matematika
- Topic_1 = IT
- Topic_2 = Statistika
- Topic_3 = Fyzika

Pro lepší pochopení lze zobrazit dalších top 30 slov, aby byla tato rozhodnutí potvrzena. Pro ilustraci je zobrazeno topic_0 a jeho top 30 nejpoužívanějších slov v dokumentech, které byly zařazené pod toto téma. Na obrázku 21 lze vidět tato slova. Jejich velikost je dána výpočtem TF-IDF namísto váhy, jako bylo v LDA. Nachází se zde slova jako matice, vysoký rozměr, časoprostor a jednoduše propojené. Na první pohled se může zdát, že tato slova by mohla být i pro téma 4 tedy fyziku, ale při

provedení workflow pro toto téma dostáváme jiné výsledky, které jasně ukazují, že témata jsou správně zařazena.

Obrázek 21 Top 30 slov pro topic_0; zdroj: vlastní zpracování

Obrázek 22 Top 30 slov pro topic_3; zdroj: vlastní zpracování

Na obrázku 21 lze pozorovat top 30 slov pro téma matematika a na obrázku 22 lze vidět slova pro téma fyzika. Při změně parametrů k a α došlo k lepšímu rozložení klíčových slov mezi témata a k přesnému určeních jednotlivých témat a tím se dosáhlo extrakce témat z nestrukturovaného textu. Bohužel nedostatek dokumentů pro téma biologie a finance zapříčinil, že témata nešlo dostatečně rozlišit, aby se dalo konstatovat, že v datasetu opravdu existují. Doporučeným způsobem pro zamezení

této chyby je buď doplnit jednotlivé články, nebo smazat dokumenty, které spadaly pod tato témata. Byl vybrán krok pro smazání těchto dokumentů. [55]

8.6 Hodnocení modelu

Hodnocení modelu je provedeno skrze koherenci témat. V platformě KNIME existuje uzel, který používá koherenci témat jen jako jednu z metrik měření modelu na extrakci témat. Tento uzel se nazývá Topic Scorer. Používá sémantickou koherenci tématu, která měří, jak jsou tato témata koherentní, a to tak, že kontroluje výskyt klíčových slov v tématu a jejich společný výskyt v dokumentu. Nezáleží zde na externím korpusu, který by byl v modelu připojen. [56]

Vznik sémantické koherence tématu zapříčinila nevyužitelnost informací o společném výskytu slov v tematickém modelu pro extrahování témat. V tomto případě se vrací k evaluaci modelu skóre v minusu a čím blíže k nule směřuje, tím více je téma sémanticky koherentní. Tato evaluace může být uplatněna i v průběhu modelu, sledování společného výskytu slov mělo za následek, že Gibbsovo vzorkování probíhalo rychleji. [57]

V tomto uzlu probíhají ještě další dva výpočty, které pomohou vyhodnotit model. Jedná se o exclusivity score a neighbor distance score. Exclusivity score je založené na metodě FREX, která počítá exkluzivní termíny v tématu a také počítá, jak vzácné je téma v dokumentu a v dokumentech se stejným tématem. Zde je skóre od 0 do 1. [58]

Skóre pro nejbližší sousední téma v modelu se zde nazývá neighbor distance metric. Témata jsou reprezentována normalizovaným vektorem klíčových slov podle témat. Vypočítává se cosinová vzdálenost tématu k nejbližšímu, protože témata si mohou být podobná a v důsledku může dokument obsahovat jedno a více témat. [56]

S Topic id	D Semantic Coherence	D Exclusivity (FREX)	D Neighbor Distance Metric (within-model)	S Nearest Topic (within-model)
topic_2	-22.409	0.449	0.602	topic_1
topic_1	-22.44	0.473	0.602	topic_2
topic_3	-29.613	0.536	0.794	topic_1
topic_0	-34.461	0.529	0.86	topic_2

Obrázek 23 Výsledky uzlu Top scorer; zdroj: vlastní zpracování

Na obrázku 23 lze vidět výsledky uzlu Topic Scorer pro model LDA při detekování témat. Nejvíce sémanticky koherentní je téma 2 statistika, následuje téma1 IT, na třetím místě je téma 4 fyzika a posledním, nejhorším skórovaným tématem je téma 0 matematika. Skóre je blízko k nule a lze tedy říci, že je velmi sémanticky koherentní.

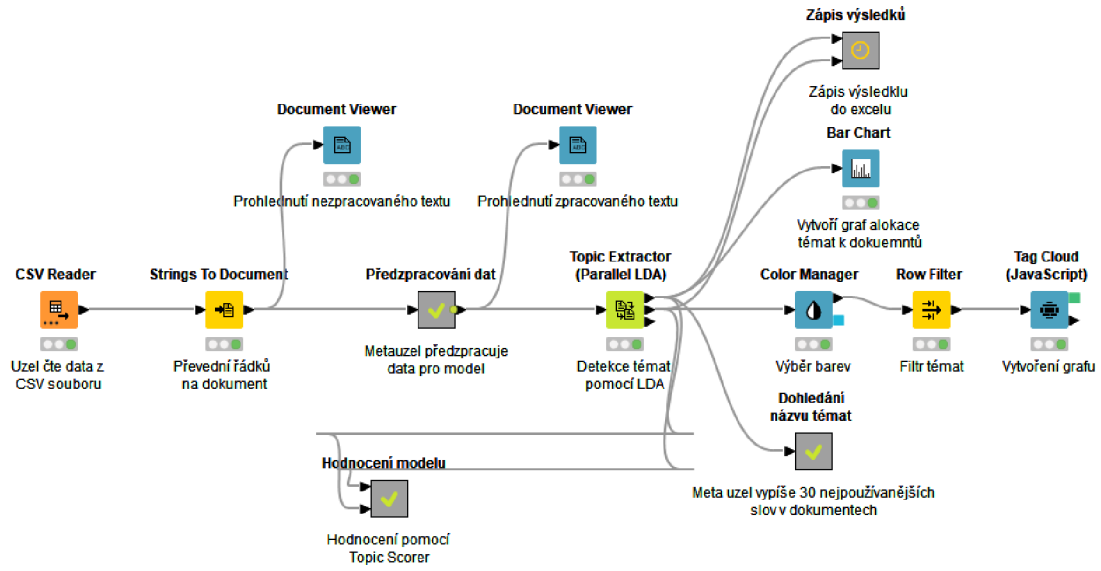
Na druhou stranu zde je skóre pro exkluzivitu, kdy jsou výsledky kolem 0,5, takže termíny jsou průměrně relevantní v daných tématech. To je zde podpořeno skutečností, že například slovo model se objevuje ve více než jednom tématu jako klíčové slovo. [58]

Posledním skóre je zde skóre pro nejbližšího souseda. Může ukazovat, kolikrát budou nejbližší témata v daném dokumentu spojená, protože dokument může obsahovat jedno až více témat.

Z těchto výsledků lze říci, že vytvořený model je velmi dobrý, co se týče sémantické koherence tématu. Latentní Dirichletova alokace po úpravě proběhla úspěšně. Slova tudíž spolu souvisí a lze je v dokumentu společně najít, ale protože se jedná o výzkumné články, jak bylo výše uvedeno, je zde zaběhnutý jistý standardní korpus. Proto není velká exkluzivita mezi slovy, která jsou v dokumentu použita.

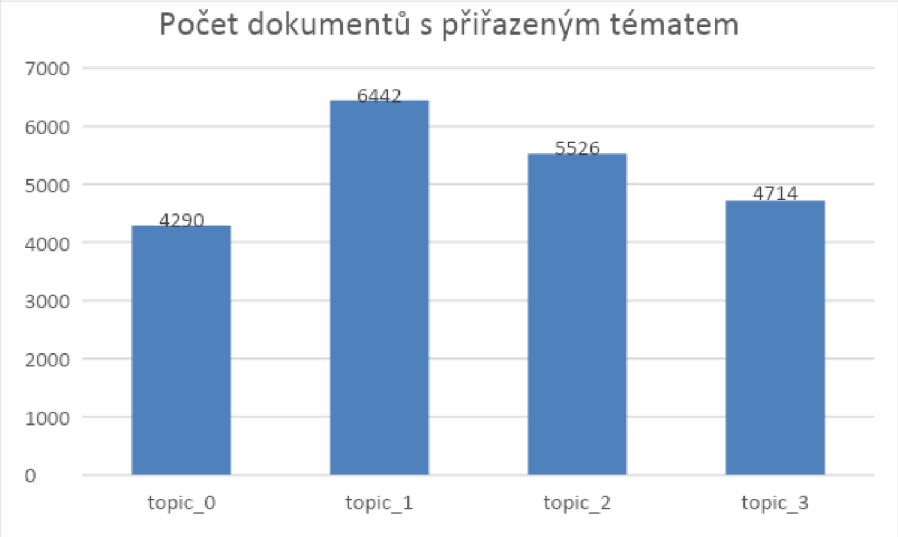
8.7 Výsledky modelu

Po zhodnocení modelu jako úspěšného lze přejít na výsledky zařazení LDA dokumentů podle témat. Na obrázku 24 lze vidět úplný model v platformě KNIME, který byl použit k detekci témat v této práci se všemi použitými uzly, jež jsou výše popsány. Celý model je anotován, aby bylo možné pochopit workflow, co každý uzel provádí, a připojena nápověda, jak daný uzel funguje.



Obrázek 24 Úplný model pro detekci témat; zdroj: vlastní zpracování

V grafu 2 lze vidět počet dokumentů s přiřazeným tématem. Opět největší počet zařazených dokumentů spadá pod téma 1, což je téma IT. Tento výsledek souhlasí i s výsledky, jak byly zařazeny dokumenty na začátku při stažení z Kaggle. Druhé téma v pořadí je statistika, třetí je fyzika a čtvrtým v počtu dokumentů je matematika. V porovnání s rozložením v datasetu jsou tu nesrovnalosti, ale to je předvídatelné, protože model nebyl 100 %. Model dosáhl úspěšnosti přiřazení správného tématu v 67 % případů při porovnání s CSV souborem, kde jsou témata předem vložena.



Graf 2 Počet dokumentů s přiřazeným tématem; zdroj: vlastní zpracování

Úspěšnost 67 % ve správné detekci témat není příliš vysoká, protože standard v tomto odvětví je 70 %, aby byl model klasifikován jako přesný. [59] Je třeba ale uvést několik okolností, které zapříčinily tento problém. LDA má problém detekovat témata, pokud platí [60]:

1. Není dostatek textů pro detekování tématu.
2. Dokumenty neprojednávají uceleně jedno téma.

Retrospektivně bod 1 a bod 2 dávají smysl, neboť abstrakt k vědeckému článku mívá velikost 200 slov. Zejména u metody LDA, která odvozuje témata na základě jednotlivých dokumentů, platí, že pokud v dokumentu není dostatek slov, neexistuje dostatek dat pro odvození spolehlivého rozdělení témat pro daný dokument. [34] To je možné vidět i v empirické studii tweetů z roku 2010, kdy velikost jednoho tweetu byla omezena na 140 znaků a přesnost byla 47 %. [61]

Bod 2 podporuje skutečnost, že témata byla detekována ve vědeckých článcích, které se nemusí prolínat stejným korpusem, ale i tématy. Například autoři vědeckých článků často používají statistiku, neboť se snaží vyvodit závěry z pravděpodobností u dat, která získali z průzkumů nebo testů. Tudíž zde může dojít k prolnutí dvou témat, jako je například medicína a statistika. Sice se to děje ve většině dokumentů, ale je důležité poukázat, že ve vědeckém článku může být toto rozdělení půl na půl.

Při zohlednění těchto poznatků je možné pohlížet na přesnost 67 % o něco příznivěji, neboť si tento model dokázal poradit i s malým počtem slov na dokument, a přesto dokázal dosáhnout přesnosti blízko standardu 70 %. [59] Potvrzuje se tak doporučení používat LDA na větším množství dat, aby přesnost detekování témat byla vyšší.

9 Závěr

Cílem diplomové práce bylo vytvořit model pro extrakci témat z nestrukturovaného textu. Byly přitom využity teoretické poznatky o analýze textu, přirozené řeči a zpracování textů. Textová analýza přinesla poznatky o tom, že přirozená řeč je plná detailů a nuancí, a proto je pro počítač obtížně zpracovatelná. Na tento poznatek bylo navázáno kapitolou o zpracování textu, která ukázala, že pomocí různých metod a technik lze přirozenou řeč převést neboli vektorizovat, tedy převést na čísla ve vektoru, se kterým je počítač schopen pracovat. Dále zde byly popsány základy NLP a LDA a jich vývoj od roku 1990. I tyto metody byly použity v praktické části.

Spojením těchto poznatků byl vytvořen model v platformě KNIME, který byl schopen extrahovat témata z nestrukturovaného textu. Extrakce témat byla provedena nad datasetem, který obsahoval odborné články. Byl využit jen jejich nadpis a abstrakt, neboť kdyby se použil celý, jednalo by se o text semi-strukturovaný. Tento dataset obsahoval šest témat, ale pomocí modelu byla definitivně určena jen čtyři. Bylo to způsobeno tím, že ke dvěma neurčeným tématům nebyl k dispozici větší počet dokumentů. Téma biologie mělo pouze 587 dokumentů a téma finance mělo 249 dokumentů. Oproti ostatním dokumentům, které měly počty v tisících, se jedná o opravdu malý vzorek. Dále v souvislosti s tím, že odborné články mají podobný korpus, nebylo možné bezpečně určit rozdíl mezi nimi ani po vyfiltrování 30 nejpoužívanějších slov v tématu. Z této zkušenosti vyplývá, že by bylo vhodné dataset doplnit o články s danými tématy, aby témata měla zhruba stejné množství článků jako ostatní témata.

Následně byl zvolen přístup upravení parametrů, protože v praxi není předem počet témat znám a standardní praxí je úprava parametrů. Doporučuje se metoda Elbow, která dokáže optimalizovat počet témat *k*, která jsou pomocí LDA hledána. [55] Přesto metoda neudává jedno číslo, ale několik možných *k*, která mají po výpočtu nejmenší rozptyl. Zde v práci nebyla potřeba, protože *k* bylo jen zmenšeno o počet neznámých témat.

V další části bylo tedy *k* upraveno na 4 a hyperparametry α a β. Parametr α byl upraven podle vztahu α/50 a parametr β zůstal stejný, neboť nebyla zatím potřeba ho měnit. Po spuštění takto upraveného modelu už bylo možné identifikovat všechna

témata, která LDA našla pomocí generativního procesu. Tato témata byla určena podle lidského myšlení a byla podpořena výběrem 30 nejpoužívanějších slov pro dané téma. Vyhodnocením modelu pomocí uzlu Topic Scorer bylo zjištěno, že jsou témata sémanticky koherentní, a to v nadprůměru, neboť se skóre blížilo nule a počet zařazených dokumentů se lišil proti datasetu o 31 %. To bylo způsobeno vyřazením dvou témat, která v datasetu jsou, ale jak již bylo uvedeno, nešlo je bezpečně od sebe odlišit natolik, aby se dalo s jistotu říci, že opravdu jsou v datasetu přítomna.

Výsledkem byl model, který již dokázal extrahovat témata, a tato témata bylo možné jednotlivě odlišit. Byly zde použity i techniky k podpoření tohoto výběru, které i nadále potvrdily správnost výběru názvu tématu a jejich koherence.

10 Seznam použité literatury

[1] Ignatow, Gabe a Rada Mihalcea. An introduction to text mining : research design, data collection, and analysis. USA: SAGE Los Angeles, 2018, 320 s. ISBN 9781506337005.

[2] Sidnell, J. Conversation Analysis. Oxford Research Encyclopedia of Linguistics. [online]. [cit. 30.01.2023]. Dostupné z: https://oxfordre.com/linguistics/view/10.1093/acrefore/9780199384655.001.0001/acrefore-9780199384655-e-40.

[3] Some Notes on Foucault on Discourse | Epoché Magazine. [online]. Dostupné z: https://epochemagazine.org/19/some-notes-on-foucault-on-discourse/

[4] Given, L. The SAGE Encyclopedia of Qualitative Research Methods. SAGE Publications, 2008, 1072 s. ISBN 9781452265896

[5] FELDMAN, Ronen a James SANGER. The text mining handbook: advanced approaches in analyzing data. New York: Cambridge University Press, 2007. ISBN 978-0-521-83657-9.

[6] Webb, L. and Wang, Y. (09 2013) 'Techniques for Sampling Online Text-Based Data Sets', in, pp. 95–114. doi: 10.4018/978-1-4666-4699-5.ch005.

[7] Tokenization of Textual Data into Words and Sentences and Definition?. Great Learning: Online Courses, PG Certificates and Degree Programs [online]. Copyright © 2013 [cit. 01.02.2023]. Dostupné z: https://www.mygreatlearning.com/blog/tokenization/

[8] Ganesan, K. All you need to know about text preprocessing for NLP and Machine Learning, KDnuggets. [cit. 06.02.2023]. Dostupné z: https://www.kdnuggets.com/2019/04/text-preprocessing-nlp-machine-learning.html

[9] Lane, H., Hapke, H., & Howard, C. (2019). Natural Language Processing in Action: Understanding, analyzing, and generating text with Python. Manning Publications. ISBN 9781617294631

[10] Horev, R. (2018) Bert explained: State of the art language model for NLP, Medium. [cit. 15.04.2023]. Dostupné z: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

[11] Schade, M. (no date) How CHATGPT and our language models are developed: Openai help center, How ChatGPT and Our Language Models Are Developed | OpenAI Help Center. [cit. 15.04.2023]. Dostupné z: https://help.openai.com/en/articles/7842364-how-chatgpt-and-our-language-models-are-developed

[12] Stecanella, B. (2019) Understanding TF-ID: A simple introduction, MonkeyLearn Blog. [cit. 16.04.2023]. Dostupné z: https://monkeylearn.com/blog/what-is-tf-idf/

[13] Yadav, S. (2023) Word classes and part-of-speech tagging in NLP, Scaler Topics. [cit. 16.04.2023]. Dostupné z: https://www.scaler.com/topics/nlp/word-classes-and-part-of-speech-tagging-in-nlp/

[14] Chiche, A. and Yitagesu, B. (2022) 'Part of speech tagging: A systematic review of deep learning and machine learning approaches', Journal of Big Data, 9(1). doi:10.1186/s40537-022-00561-y.

[15] Burns, E. and Brush, K. (2023) What is deep learning and how does it work?, Enterprise AI. [cit. 11.09.2023]. Dostupné z:https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network

[16] Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, Juan Miguel Gómez-Berbís, Named Entity Recognition: Fallacies, challenges and opportunities, Computer Standards & Interfaces, Volume 35, Issue 5, 2013, Pages 482-489, ISSN 0920-5489, https://doi.org/10.1016/j.csi.2012.09.004.

[17] Taylor, P. (2023a) Data Growth Worldwide 2010-2025, Statista. [cit. 11.09.2023]. Dostupné z: https://www.statista.com/statistics/871513/worldwide-data-created/

[18] Petasis, G. *et al.* (2000) 'Automatic adaptation of proper noun dictionaries through cooperation of machine learning and Probabilistic Methods', *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. doi:10.1145/345508.345563.

[19] Zhang, S. and Elhadad, N. (2013) 'Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts', *Journal of Biomedical Informatics*, 46(6), pp. 1088–1098. doi:10.1016/j.jbi.2013.08.004.

[20] Budi, I. and Suryono, R.R. (2023) 'Application of named entity recognition method for Indonesian datasets: A Review', Bulletin of Electrical Engineering and Informatics, 12(2), pp. 969–978. doi:10.11591/eei.v12i2.4529.

[21] Barney, N. (2023) What is named Entity recognition (ner)?: Definition from TechTarget, WhatIs.com. [cit. 22.04.2023]. Dostupné z: https://www.techtarget.com/whatis/definition/named-entity-recognition-NER

[22] Kaushik, S. (2023) "Clustering | Introduction, different methods, and applications (Updated 2023)," Analytics Vidhya [cit. 11.05.2023]. Dostupné z: https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/.

[23] Shah, N. and Mahajan, S. (2012) "Document Clustering: A Detailed Review," International Journal of Applied Information Systems, 4(5), pp. 30–38. Available at: https://doi.org/10.5120/ijais12-450691

[24] Merriam-Webster. (n.d.). Topic. In Merriam-Webster.com dictionary. [cit. 15.05.2023]. Dostupné z: https://www.merriam-webster.com/dictionary/topic (nepoužívat podtržení a barvení hypertext. odkazů; je to detail, ale zkrátka to trošku kazí grafickou podobu práce)

[25] Liu, L. et al. (2016) "An overview of topic modeling and its current applications in bioinformatics," SpringerPlus, 5(1). [cit. 15.05.2023]. Dostupné z: https://doi.org/10.1186/s40064-016-3252-8.

[26] Deerwester, S. et al. (1990) “Indexing by latent semantic analysis,” Journal of the American Society for Information Science, 41(6), pp. 391–407. [cit. 15.05.2023]. Dostupné z: https://doi.org/10.1002/(sici)1097-4571(199009)41:6.

[27] Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003) “Latent dirichlet allocation,” Journal of Machine Learning Research, 3, pp. 993–1022. [cit. 15.05.2023]. Dostupné z: https://doi.org/10.5555/944919.944937.

[28] Qader, W.A., Ameen, M.M. and Ahmed, B.I. (2019) “An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges,” 2019 International Engineering Conference (IEC), 2019. [cit. 18.05.2023]. Dostupné z: https://doi.org/10.1109/iec47844.2019.8950616.

[29] What is bag-of-words model?: AI terms explained - AI For Anyone. [cit. 18.05.2023]. Dostupné z: https://www.aiforanyone.org/glossary/bag-of-words-model.

[30] Yan, D. et al. (2020) “Network-Based Bag-of-Words model for text classification,” IEEE Access, 8, pp. 82641–82652. [cit. 19.05.2023]. Dostupné z https://doi.org/10.1109/access.2020.2991074.

[31] Pascual, F. (2019) Topic Modeling: An Introduction [cit. 19.05.2023]. Dostupné z: https://monkeylearn.com/blog/introduction-to-topic-modeling/.

[32] Churchill, R. and Singh, L. (2022) 'The evolution of topic modeling,' ACM Computing Surveys, 54(10s), pp. 1–35. https://doi.org/10.1145/3507900.

[33] Dutta, M. (2022) 'Word2Vec For Word Embeddings -A Beginner’s Guide,' Analytics Vidhya [cit. 03.06.2023]. Dostupné z: https://www.analyticsvidhya.com/blog/2021/07/word2vec-for-word-embeddings-a-beginners-guide/.

[34] Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003b) 'Latent dirichlet allocation,' Journal of Machine Learning Research, 3, pp. 993–1022. https://doi.org/10.5555/944919.944937.

[35] Kirsch, W. (2019) 'An elementary proof of de Finetti’s theorem,' Statistics & Probability Letters, 151, pp. 84–88. https://doi.org/10.1016/j.spl.2019.03.014.

[36] Liu, S. (2022) 'The Dirichlet distribution: What is it and why is it useful?,' Built In [cit. 05.06.2023]. Dostupné z: https://builtin.com/data-science/dirichlet-distribution.

[37] A Deep Dive into Latent Dirichlet Allocation (LDA) and Its Applications on Recommender Systems [cit. 06.06.2023]. Dostupné z: https://rosetta.ai/blog/a-

deep-dive-into-latent-dirichlet-allocation-lda-and-its-applications-on-recommender-systems.

[38] Ibanez, D. and Ibanez, D. (2023) 'Topic Modeling with Latent Dirichlet Allocation | Baeldung on Computer Science,' Baeldung on Computer Science [cit. 06.06.2023]. Dostupné z: https://www.baeldung.com/cs/latent-dirichlet-allocation.

[39] Serrano.Academy. (2020, March 19). Latent dirichlet allocation (Part 1 of 2) [Video]. YouTube. https://www.youtube.com/watch?v=T05t-SqKArY

[40] Gelfand, A.E. (2000) 'Gibbs sampling,' Journal of the American Statistical Association, 95(452), p. 1300. https://doi.org/10.2307/2669775.

[41] Pedro, J. (2022) 'Understanding topic coherence Measures - towards data science,' Medium, 15 January. [cit. 17.06.2023]. Dostupné z: https://towardsdatascience.com/understanding-topic-coherence-measures-4aa41339634c.

[42] Zvornicanin, E. and Zvornicanin, E. (2023) 'When coherence score is good or bad in topic modeling? | Baeldung on Computer Science,' Baeldung on Computer Science [cit. 17.06.2023]. Dostupné z: https://www.baeldung.com/cs/topic-modeling-coherence-score.

[43] Röder, M., Both, A. and Hinneburg, A. (2015) 'Exploring the space of topic coherence measures,' Proceedings of the Eighth ACM International Conference on Web Search and Data Mining https://doi.org/10.1145/2684822.2685324.

[44] The Economist (2017) 'The world's most valuable resource is no longer oil, but data,' The Economist, 11 May. [cit. 22.06.2023]. Dostupné z: https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data.

[45] Emery, J. (2023) 'What is KNIME and Tips for Getting Started,' phData . [cit. 22.06.2023]. Dostupné z: https://www.phdata.io/blog/getting-started-with-knime/.

[46] Berthold, M.R. et al. (2009) 'KNIME - the Konstanz information miner,' SIGKDD Explorations, 11(1), pp. 26–31. https://doi.org/10.1145/1656274.1656280.

[47] Hayasaka, S., Rosaria S. A Guide to KNIME to Analytics Platform for Beginners. 5th ed., KNIME Press, 2019. ISBN 978-3-033-02850-0

[48] Enotes World (2023) 'Concept of stock and flow variables,' eNotes World [cit. 22.06.2023]. Dostupné z https://enotesworld.com/concept-of-stock-and-flow-variables/.

[49] Hazelcast (2019) Directed Acyclic Graph (DAG) Overview & Use Cases | Hazelcast. [cit. 22.06.2023]. Dostupné z https://hazelcast.com/glossary/directed-acyclic-graph/.

[50] KNIME Community Forum. (n.d.). KNIME Community Forum. https://forum.knime.com/

[51] KNIME Community Hub. (n.d.). KNIME Community Hub. https://hub.knime.com/

[52] KNIME Self-Paced courses. (n.d.). KNIME. https://www.knime.com/knime-self-paced-courses

[53] What is Kaggle, Why I Participate, What is the Impact? | Kaggle. [cit. 25.08.2023]. Dostupné z: https://www.kaggle.com/discussions/getting-started/44916.

[54] Topic modeling for research articles (2020). [cit. 25.08.2023]. Dostupné z: https://www.kaggle.com/datasets/blessondensil294/topic-modeling-for-research-articles/data?select=train.csv.

[55] Tursi, V. and Silipo, R. KNIME from words to wisdom. KNIME Press, 2018 ISBN 978-3-9523926-2-1

[56] Topic Scorer (Labs) – knime [cit. 29.09.2023]. Dostupné z: https://hub.knime.com/knime/spaces/Examples/latest/00_Components/Text%20Processing/Topic%20Scorer%20(Labs)~5_W2h2g6hBY_M0Bc.

[57] Mimno, D. et al. (2011) 'Optimizing semantic coherence in topic models,' Empirical Methods in Natural Language Processing, pp. 262–272. http://dirichlet.net/pdf/mimno11optimizing.pdf.

[58] Bischof, J. and Airoldi, E.M. (2012) 'Summarizing topical content with word frequency and exclusivity,' arXiv (Cornell University) [cit. 29.09.2023]. Dostupné z:. https://arxiv.org/abs/1206.4631v1.

[59] Which is more important: model performance or model accuracy? | Fiddler AI. (n.d.). [cit. 10.12.2023]. Dostupné z:. https://www.fiddler.ai/model-accuracy-vs-model-performance/which-is-more-important-model-performance-or-model-accuracy

[60] Why latent dirichlet allocation sucks. (2018, March 6). ⁂ George Ho. [cit. 10.12.2023]. Dostupné z:. https://www.georgeho.org/lda-sucks/

[61] Liu, H., & Davison, B. D. (2010). Empirical study of topic modeling in Twitter. SOMA '10: Proceedings of the First Workshop on Social Media Analytics. https://doi.org/10.1145/1964858.1964870

Seznam obrázků

Seznam tabulek

Seznam grafů

Přílohy

1) DP_Extrakce_témat_Illner_David.knam
    a. Obsahuje úplný workflow pro detekci témat
    b. Vytvořeno v platformě KNIME
2) Data.rar
    a. Obsahuje dataset s dokumenty

UNIVERZITA HRADEC KRÁLOVÉ
Fakulta informatiky a managementu
Akademický rok: 2020/2021

Studijní program: Informační management
Forma studia: Prezenční

Podklad pro zadání DIPLOMOVÉ práce studenta

Jméno a příjmení:	**Bc. David Illner**
Osobní číslo:	**I2000076**
Adresa:	Palackého 887, Úpice, 54232 Úpice, Česká republika
Téma práce:	Extrakce témat z nestrukturovaného textu
Téma práce anglicky:	Extraction of topics from unstructured text
Jazyk práce:	Čeština
Vedoucí práce:	Ing. Martina Husáková, Ph.D. Katedra informačních technologií

Zásady pro vypracování:

Cíl práce:
Cílem práce je vytvořit model pro extrakci témat z nestrukturovaného textu v analytické platformě KNIME a zhodnotit výsledky tohoto modelu.
Osnova:

1. Úvod
2. Text Mining
3. Metody předzpracování dat
4. Analytická platforma KNIME
5. Modelování témat
6. Výsledky a hodnocení
7. Závěr a doporučení

Seznam doporučené literatury:

F. Provost, T. Fawcett Data Science for Business (2013). ISBN 9781449361327
Bakos, B. KNIME Essentials Paperback (2013). ISBN 1849699216.
KNIME platform homepage: https://www.knime.com/
UDEMY.com kurz: https://www.udemy.com/course/knime-bootcamp/
Tursi, V., Silipo, R. From Words to Wisdom – An Introduction to Text Mining with KNIME. URL: https://www.knime.com/knimepress

Podpis studenta: Datum:

Podpis vedoucího práce: Datum: