Jihočeská univerzita v Českých Budějovicích
Přírodovědecká fakulta

Návrh řešení studeného startu doporučovacího systému

Bakalářská práce

Jakub Maštalíř

Školitel: Doc. Ing. Ladislav Beránek, CSc., MBA

České Budějovice 2019

Jihočeská univerzita v Českých Budějovicích
Přírodovědecká fakulta

ZADÁVACÍ PROTOKOL BAKALÁŘSKÉ PRÁCE

Student: Jakub Maštalíř
(jméno, příjmení, tituly)

Obor – zaměření studia: Aplikovaná informatika

Katedra/ústav, kde bude práce vypracovávána: Ústav aplikované informatiky

Školitel: doc. Ing. Ladislav Beránek, CSc.
(jméno, příjmení, tituly, u externího š. název a adresa pracoviště, telefon, fax, e-mail)

Garant z PřF: .....
(jméno, příjmení, tituly, katedra – jen v případě externího školitele)

Školitel – specialista, konzultant: .....
(jméno, příjmení, tituly, u externího š. název a adresa pracoviště, telefon, fax, e-mail)

Téma magisterské práce: Návrh řešení studeného startu doporučovacího systému

Cíle práce :

Cílem doporučujících systémů je poskytnout uživatelům individuální doporučení určitých produktů nebo služeb na základě jejich preferencí. Cílem této práce je vytvoření doporučujícího systém pro určitou oblast, např. pro turisty, kteří hledají informace o určité oblasti. Práce bude řešit problém studeného startu uživatele. To je problém, kdy je potřeba dát určité kvalitní doporučení pro nového uživatele, o kterém ale doporučovací systém nemá žádné nebo málo informací. To je právě například v oblasti doporučení pro turisty, kde většina uživatelů jsou právě takoví uživatelé. Práce vyjde z přehledu možných řešení a navrhne pro daný účel nejlepší. Dále práce navrhne a implementuje vlastní doporučovací systém, který bude zpracovávat dana doporučení na základě bayesovské sítě.

Základní doporučená literatura :

S. Alag. Collective intelligence in action. Manning Pubs Co Series. Manning, 2008.

M. Chen. Research on recommender technology in E-commerce recommendation system. In Education Technology and Computer (ICETC), 2010 2nd International Conference on, volume 4, page V4. IEEE, 2010.

Z. Huang, D. Zeng, and H. Chen. A comparative study of recommendation algorithms in e-commerce applications. IEEE Intelligent Systems, 22(5):68–78, 2007.

C. Li. Research on E-Commerce Recommendation Service Using Collaborative Filtering. In 2009 Second International Symposium on Knowledge Acquisition and Modeling, pages 33–36. IEEE, 2009.

X. Zhang, Z. Xiong, and J.Wang. An improved recommendation algorithm in E-commerce. In Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on, pages 2317–2320. IEEE, 2008.

Financování práce :.....

Vedoucí práce :.....podpis : .....

U externích vedoucích fakultní garant práce.....podpis : .....

Garant oboru mag. studia .....podpis : .....

Vedoucí katedry/ústavu, kde bude práce vypracovávána.....podpis : .....

Případný souhlas vedoucího ústavu AV .....podpis : .....

V Českých Budějovicích dne 3.2.2016 podpis studenta : .....

Bibliografické údaje

Maštalíř J., 2019: Návrh řešení studeného startu doporučovacího systému. [Design of recommender system with cold start problem solution. Bc. Thesis, in Czech.] 54 p., Faculty of Science, The University of South Bohemia, České Budějovice, Czech Republic.

Anotace

Hlavní téma této bakalářské práce tkví v návrhu a implementaci modulu doporučovacího systému v oblasti turistiky a cestování. V teoretické části je cílem zmapovat prostředí doporučovacích systémů, jejich typy a přístupy. Dále je zde popsán problém studeného startu a teorie Bayesovských sítí, na jejichž základě budou zpracovávána a prezentována data pro doporučení. V praktické části již programujeme a testujeme doporučovací modul, je zde uveden popis využitých technologií a jsou rozebrány jednotlivé funkcionality doporučovacího modulu zaměřeného na ubytování v Českých Budějovicích.

Summary

The main topic of the bacholor´s thesis is design and implementation of the recommender system´s module in the field of tourism and travelling. In the theoretical part the goal is mapping out an environment of the recommender system, their types and approaches. Furthermore there is described the problem of cold start and the theory of Bayes networks on the basis of which the data for the recommendation will be processed and presented. In the practical part we are programming and testing the recommender module. There is a description of the used technologies and individual functionalities of the recommender system aimed at the acommodation in České Budějovice are dismembered.

Prohlášení

Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby stejnou elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích dne 16. 4. 2019

Jakub Maštalíř

Poděkování

Rád bych poděkoval panu doc. Ing. Ladislavu Beránkovi, CSc., MBA. za odborné vedení při zpracování mé bakalářské práce, jeho cenné připomínky a rady.

Obsah

1 Úvod

V dnešní době, kdy již můžeme neomezeně cestovat na jiné kontinenty, po Evropě nebo tuzemsku, je právě cestovní ruch a turistika důležitou součástí našich životů, jelikož nás vede k poznávání nových oblastí a může být příležitostí k podnikání. Žijeme v čase, kdy drtivá většina lidí hledá informace o daném místě dopředu na internetu nebo pomocí mobilních aplikací přímo na místě. Již odzvonilo klasickým turistickým průvodcům a brožurám. Moderní technologie a aplikace s těmi nejsofistikovanějšími algoritmy jsou standardem této doby. Rozhodl jsem se proto zaměřit na toto odvětví a vypracovat doporučovací systém, který bude zacílen na turistiku ve městě České Budějovice.

Předchůdci doporučovacích systémů se objevili již v 70. letech dvacátého století. Ty dnešní mají za úkol předložit určitou položku uživateli na základě jeho preferencí. Od jejich vzniku již nějaký čas uplynul a tyto systémy se staly našim každodenním nástrojem, a to jak při vyhledávání hudby, filmů, knih, pojištění tak právě i ve sféře cestování. Může se jednat o systémy velmi specifického vyhledávání nebo webové stránky, které navštíví miliony návštěvníků denně.

Doporučovací systémy v turistice se v posledních letech dočkaly velkého rozmachu. Lidé utrácí za své výlety velké množství peněz a pro podnikatele je tedy velmi lákavé provozovat portál určený k plánování jejich dovolené. Cestovatelé mohou navštěvovat spousty webových stránek zaměřených na destinace po celém světě, ubytování a letenky. Zajímavým řešením jsou také mobilní aplikace, které pracují s naší aktuální polohou a nabízí nám nejbližší možné památky a turistické atrakce, které stojí za zhlédnutí. Člověk již nemusí vyhledávat nejbližší informační centrum ve městě, ale dostane se mu doporučení i v tom nejzapadlejším koutě jím navštíveného místa.

2 Cíle práce a metodika

2.1 Cíle práce

Cílem teoretické části mé bakalářské práce je seznámení se s problematikou doporučovacích systémů, jejich možnými přístupy, omezeními a přiblížit čtenáři konkrétní doporučovací systémy. Dále se zaměřím na problém studeného startu v tématice doporučení. Závěrem se budu zabývat Bayesovskou pravděpodobností, potažmo Bayesovskými sítěmi.

V praktické části je zvolen vhodný přístup pro daný účel, kterým je doporučení v turistice. Pomocí něj je navržen a implementován vlastní modul, který bude doporučení zpracovávat na základě Bayesovské sítě.

2.2 Metodika práce

V teoretické části práce provedu komentovanou rešerši k tématu doporučovacích systémů a jejich typů. Uvedu přehled přístupů a nástrojů, společně s uvedením konkrétních příkladů z reálného světa. Dále se budu snažit přiblížit čtenáři problém studeného startu a přejít od základů teorie pravděpodobnosti k Bayesovským sítím. Informace čerpané z literatury budou sepsány a vhodně okomentovány tak, aby byla dodržena návaznost s praktickou částí.

V části praktické se nejprve zamyslím nad konceptem celého problému a vyberu turistickou oblast, pro kterou zpracuji jednoduchý modul doporučovacího systému, který bude implementován pomocí programovacího jazyku Java. Posléze bude nutné zvolit vhodné řešení z metod uvedených v teoretické části. V závěru práce plánuji popsat jednotlivé komponenty vytvořeného modulu a ověřit jejich správnost.

3 Uživatelské preference

Uživatelská preference v doporučovacích systémech znázorňuje míru oblíbenosti položky u uživatele. Ta s nejvyšší oblíbeností je mu následně předložena. Lidé mají různé požadavky jak v běžném životě, tak při vyhledávání zboží v systémech e-commerce, proto se preference dělí na krátkodobé a dlouhodobé.

Většina doporučovacích systémů se skládá ze dvou objektů. Uživatele U a položky I doplněné o hodnocení R. To lze nazvat jako uživatelskou preferenci. Uživatel se většinou skládat z atributů jako je věk, pohlaví a povolání. Také se může jednat o sadu jeho dřívějších hodnocení. U položky platí, že je to objekt, který obdržel řadu hodnocení. Může být ale více obsáhlejší a obsahovat také různá fakta a informace.

Klíčem k úspěchu doporučovacích systémů je správně předpovědět položku uživateli. K tomu je zapotřebí vytvořit funkci, která bere cílového uživatele U a položku I jako vstup a vrací předpovídanou hodnotu R od uživatele U na položku I, která je co nejblíže reálné hodnotě. Tedy té hodnotě, kterou by uživatel položku pravděpodobně ohodnotil, pokud by mu byla známa. Tato funkce poté může být použita na zbytek skrytých položek a ta s největším předpovídaným ohodnocením bude předložena uživateli. To lze matematicky znázornit jako funkci zobrazení: [1]

$$U \times I \rightarrow R$$

3.1 Krátkodobé preference

Krátkodobou preferencí se rozumí aktuálně zvýšený zájem o určitou položku. Například je-li nutností koupě levného automobilu, drahé sportovní vozy se budou v preferencích pohybovat blízko 0, i když za jiných okolností jsou často stejným uživatelem vyhledávány. [3]

3.2 Dlouhodobé preference

Dlouhodobé preference vyjadřují uživatelův postoj, kterým se dlouhodobě řídí. Například nákup dražších, ale prověřených spotřebičů oproti těm levnějším nebo preference konkrétní značky na úkor jiných atd. [3]

3.3 Předmět preference

Předmětem preference je atribut, který uživatele u dané položky nejvíce zajímá. Každý uživatel má jiné požadavky. Atributy položky se ve většině případů nemění. Pokud má uživatel v plánu koupit úspornou lednici a pořizovací cena je pro něj druhořadá, vyšší váhu má atribut energetická třída oproti atributu cena. Možné atributy: [4]

- **Nominální:** barva, značka.
- **Numerické:** rozměry, spotřeba, otáčky, výkon.
- **Specifické:** obtížně zaznamenatelné atributy (zvuk, tvar).

3.4 Identifikace uživatele

Aby bylo možné preference využívat, je nutné uživatele v systému jednoznačně identifikovat. K tomu lze využít následující postupy:

- **Registrace uživatele:** nejspolehlivější způsob identifikace. Pod kombinací jména a hesla se skrývá právě jeden uživatel. Jednoznačná identifikace uživatele, který může používat webový prohlížeč, potažmo počítač, s více lidmi. Nevýhodou je nutnost registrace. Mnoho uživatelů tato podmínka odradí.
- **Identifikace pomocí IP adresy:** identifikace uživatele pomocí IP adresy je dalším způsobem, ovšem ne příliš spolehlivým. Tutéž IP adresu může sdílet více počítačů. Jeden člověk může přistupovat do systémů z různých zařízení, opět jiná IP adresa. Identifikace konkrétní osoby v systému je poté obtížná. Hodí se v případě, pokud systém využije informaci, z jaké destinace (státu) uživatel přistupuje do systému.
- **Identifikace pomocí COOKIES:** identifikace uživatele na základě kombinace prohlížeče a počítače. Jedná se o jednoznačnou identifikaci uživatele, jestliže používá počítač pouze sám. Opět nelze rozpoznat stejného uživatele, pokud přistupuje z jiného zařízení. Dalším problémem může být zákaz ukládání cookies ze strany uživatele, tedy nemožná identifikace tímto způsobem.

Jako ideální řešení identifikace uživatele v systémech e-commerce se jeví kombinace identifikace pomocí cookies pro nepřihlášené uživatele a možnost registrace uživatele (následná identifikace po přihlášení). [3]

3.5 Techniky získávání uživatelských preferencí

Společným znakem převážné většiny doporučovacích systémů je snaha využít zpětné vazby uživatelů a použít ji jako zdroj informací o daném člověku k následnému doporučení. Rozlišujeme dva druhy přístupů, jak toho dosáhnout: explicitní a implicitní. [3]

3.5.1 Explicitní přístup

Společnosti jako Alza.cz, Youtube.com a jiné využívají explicitní přístup jako vlastní prostředek k ohodnocení položky, při kterém požadují, aby návštěvník jejich stránek ohodnotil obsah, který nabízejí. Děje se tak pomocí bodové stupnice nebo jednoduše přirazením tzv. palce (like/dislike), jak je tomu u již zmíněného portálu Youtube.com. U tohoto hodnocení však musí návštěvník projít a ohodnotit velké množství položek, aby se mu dostalo odpovídajícího doporučení. Tento typ získávání zpětné vazby zahrnuje několik problémů. Uživatel nemusí vždy ohodnotit obsah pravdivě a také se jeho preference mohou po určitém čase měnit. [3]

3.5.2 Implicitní přístup

U implicitního přístupu je oproti explicitnímu třeba sledovat uživatelskou aktivitu v daném systému. Implicitní přístup shromažďuje informace o uživateli a jeho chování. Konkrétně mapuje čas strávený na stránce, otevírání dalších odkazů, skrolování, zakoupení položky atd. Pomocí takto získaných znalostí o chování uživatele doporučovací systém určuje, jak velký má uživatel o danou položku zájem a přiradí jí patřičné ohodnocení. Tato metoda může být podpořena explicitním hodnocením k dosažení co nejpřesnějšího přiřazení. [3]

4 Doporučovací systémy

Doporučovací systémy jsou zaměřené na osobní doporučení pro uživatele v oblastech, jakými jsou například filmy, hudba, konkrétní webové stránky a sociální sítě. Mezi ty nejdůležitější ovšem patří moduly doporučovacích systémů v e-shopech.

První doporučovací systém nesl název Tapestry a vznikl v roce 1992. Byl to e-mailový systém vyžadující dotazy vycházející od uživatele pro následné doporučení. Tento počin si vysloužil zájem jak ze strany vědeckých kruhů, tak i podnikatelských subjektů a odstartoval rozvoj v této oblasti. [2]

Dále se doporučování zaměřilo spíše do komerčních oblastí pro zajištění zisků internetových obchodů. Mezi největší milníky využití doporučovacích systémů patří zařazení doporučení na portálu Amazon.com, který patří i aktuálně k nejvýznamnějším hráčům v této oblasti. [5]

V dnešní době internetu přijdou lidé do styku s doporučovacími systémy při každodenní návštěvě populárních stránek jako youtube.com, facebook.com a další. Hnacím motorem současné společnosti jsou ovšem zisky, proto majitelé e-shopů vkládají nemalé peníze do vývoje a zlepšování vlastních řešení doporučení. Kvalitní doporučovací systém musí dokázat uživateli nabídnout ty nejvhodnější položky z mnohdy několikatisícové nabídky. To určuje, jestli je daný systém úspěšný a přiměje návštěvníka ke koupi, či nikoliv. Doporučovací systémy a jejich zlepšování hrají v dnešním světě velkou roli a jejich téma je aktuální.

4.1 Typy doporučovacích systémů

Existuje několik způsobů, jak vytvářet doporučení pro uživatele založených na odlišných technikách. Na základě rozdílů v podkladech, vstupních datech a doporučovacích algoritmech definujeme pět různých doporučovacích technik a jejich kombinace, které nazýváme hybridní přístup, přináší další. [6]

V podkapitolách níže jsou popsány dvě základní techniky: kolaborativní filtrování a doporučení založené na obsahu. Následně je popsána kapitola hybridní přístup, která kombinuje zmíněné techniky a odstraňuje dílčí nedostatky.

4.1.1 Kolaborativní filtrování

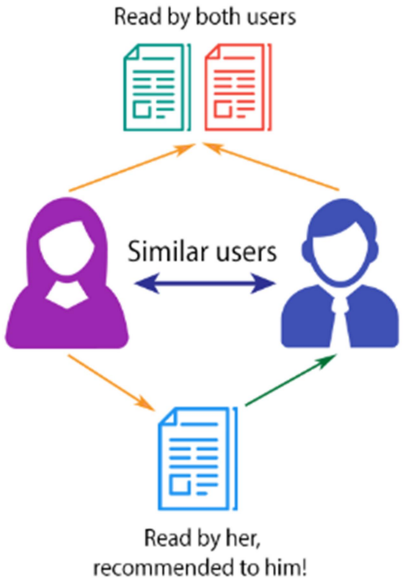
Lze říci, že kolaborativní filtrování je nejúspěšnější metodou dnešních doporučovacích systémů. Je založeno na myšlence, že doporučení položky cílovému uživateli může probíhat na základě hodnocení ostatních uživatelů v daném systému. Dělí se na dva přístupy, může být použito ve srovnání uživatel-uživatel ve filtrování založeném na podobnosti uživatelů nebo položka-položka ve filtrování založeném na podobnosti položek.

Tento přístup můžeme dále rozdělit na paměťový a modelový. V modelovém jsou data využita jako trénovací množina pro určení klasifikace na neznámých hodnotách. Tento model je poté určující pro vytvoření doporučení. Oproti tomu paměťový model pracuje s reálnou databází za běhu systému. Paměťový model má větší přesnost, protože využívá aktuální data. Problém ovšem nastává při výpočtech nad velkou databází. [1]

Technika založená na podobnosti uživatelů (User-based)

Kolaborativní filtrování, které má za cíl predikovat hodnocení položky, musí nejprve nalézt v systému podobné uživatele, následně využít jejich hodnocení a generovat doporučení cílovému subjektu. [1]

Obrázek 1: Kolaborativní filtrování



Zdroj:An Overview of Recommendation Systems

Dle [1] je u této techniky uživatel přistupující do systému součástí podmnožiny jemu podobných uživatelů. Na základě podobnosti se vypočítá hodnocení jím nenavštívených položek. Ty nejlépe ohodnocené se mu předloží. Tato technika pracuje na principu podobného vkusu mezi uživateli. Zahrnuje jak položky ohodnocené v minulosti, tak ty budoucí.

Systém kolaborativního filtrování potřebuje strukturovaná data, která obsahují uživatele, položky a jejich hodnocení. Tato struktura by měla mít formu matice (uživatel × položka). V takové matici je každý uživatel reprezentován řádkem, položka sloupcem. Hodnoty v matici jsou uživatelská hodnocení položek. Takový příklad můžeme vidět v tabulce níže. *Symbol x značí neohodnocený film uživatelem.*

Tabulka 1: Matice doporučovacího systému zaměřeného na filmy

	Pupendo	Batman	Pelíšky	Matrix
Jakub	x	5	5	4
Martin	3	2	4	3
Nela	2	4	4	3
Jan	5	x	5	4

K zjištění podobnosti dvou uživatelů je zapotřebí vypočítat váhu podobnosti mezi aktivním uživatelem systému a zbylými uživateli. To se nejčastěji provádí pomocí Pearsonova korelačního koeficientu:

$$W_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)\,(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2 \sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}}$$

kde $w_{a,u}$ je váha podobnosti, u značí uživatele a a značí aktivního uživatele. I je množina položek ohodnocena oběma uživateli, $r_{u,i}$ je hodnocení položky i uživatelem u a r_u je průměrné hodnocení uživatele u.

Dalším krokem je výběr uživatelů, kteří mají největší podobnost s aktivním uživatelem. Této množině se říká sousedství.

Výpočet předpovědi hodnocení z kombinace vybraných uživatelských hodnocení je součástí kroku tři. Tato předpověď se obvykle počítá pomocí následujícího vzorce:

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in K}(r_{u,i} - \bar{r}_u) \times W_{a,u}}{\sum_{u \in K} W_{a,u}}$$

kde $p_{a,i}$ je předpověď hodnocení aktivního uživatele a pro položku i a K je množina nejpodobnějších uživatelů – sousedství.

Toto řešení je jedno z nejstarších a nejúčinnějších. Lze ho ovšem v reálném čase realizovat pouze v menších systémech. [7] [8]

Technika založená na podobnosti položek (Item-based)

Tato technika je vhodná pro systémy, které obsahují velké množství uživatelů a položek. Oproti předchozí metodě se zde porovnává podobnost položek, ze kterých se tvoří doporučení. Při této technice je možno data vypočítat offline. Je tedy reálné zpracovat výsledky doporučení i pro velké matice složené z milionů uživatelů a položek. To by při výpočtu v reálném čase nebylo možné. [7]

Podobnosti mezi dvěma položkami i a j se dají vypočítat v režimu offline například pomocí již zmíněné Pearsonovo korelace takto:

$$W_{i,j} = \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_i)\,(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U}(r_{u,i} - \bar{r}_i)^2}\,\sqrt{\sum_{u \in U}(r_{u,j} - \bar{r}_j)^2}}$$

kde U je množina všech uživatelů, kteří hodnotili obě položky i a j, $r_{u,i}$ je hodnocení uživatele u položky i a r_i je průměrné hodnocení položky i všemi uživateli.

Potom hodnocení položky i pro uživatele a může být predikováno pomocí jednoduchého váženého průměru:

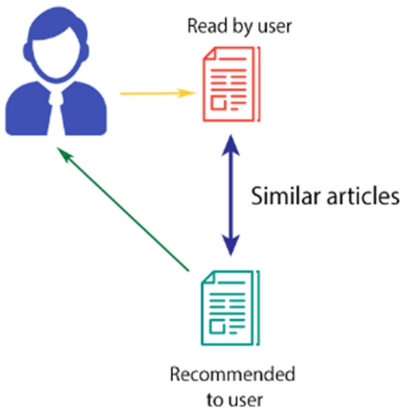
$$P_{a,i} = \frac{\sum_{j \in K} r_{a,j}\, W_{i,j}}{\sum_{j \in K} |W_{i,j}|}$$

kde K je množina sousedních položek, hodnocených uživatelem a, které jsou nejvíce podobné položce i. [7] [8]

4.1.2 Doporučení založené na obsahu

Doporučení založené na obsahu je druhým způsobem generování doporučení v doporučovacích systémech. Využívá myšlenky, že se uživateli budou líbit položky podobné těm, které dřív zakoupil nebo kladně ohodnotil.

Obrázek 2: Content-based filtering



Zdroj:An Overview of Recommendation Systems

Zatímco kolaborativní přístup doporučuje cílovému uživateli položky, které jsou kladně ohodnoceny podobnými uživateli (user-based), tento přístup se pokouší vytvořit uživateli profil, který je použit pro predikování ohodnocení neznámých položek. Aby byly předpovědi přesné, uživatelské profily musí reprezentovat jejich vkus. Tyto profily mohou být budovány implicitně sledováním zpětné vazby nebo explicitně na základě specifikace uživatelem.

Klíčem k vytvoření dobře fungujícího systému založeném na obsahu, je disponovat informacemi o položkách. Toto se provádí za pomoci atributů položek a jejich hodnot. Takový příklad je uveden v tabulce níže. Podobně jako u uživatelských profilů, také položky mají své obsahové profily, které se skládají ze stejných atributů nebo klíčových slov. [1]

Tabulka 2: Příklad hotelu s jeho atributy

Atribut	Hodnota
Název	Clarion Congres Hotel České Budějovice
Adresa	Pražská tř. 2306/14, 370 04 České Budějovice
Počet hvězdiček	4
Finanční náročnost	Vysoká
Kuchyně	Česká, italská

Uživatelský a položkový profil se obvykle definuje pomocí váhových vektorů. Pro uživatele u mějme $w_u = (w_{u1}, \ldots, w_{un})$, pro položku i $w_i = (w_{i1}, \ldots, w_{in})$, kde n je počet atributů nebo klíčových slov v systému. Váha w_u odráží uživatelské preference atributů nebo klíčových slov a váha w_i znázorňuje úroveň zařazení stejných atributů nebo klíčových slov v položkách.

Ve filtrování založeném na obsahu se pro výpočet užitečnosti položky pro uživatele často využívá kosinové podobnosti, kterou lze také použít v kolaborativním filtrování podobností mezi uživateli:

$$Užitečnost(u, i) = \cos(w_u, w_i) = \frac{w_u \,.\, w_i}{||w_u|| \times ||w_i||}$$

kde u je uživatel, i je položka, w_u je váhový vektor uživatelského profilu a w_i je váhový vektor položky. [9]

Tohoto filtrování lze využít v mnoha systémech. Nejčastěji jde ovšem o takové, kde atributy a klíčová slova položek mohou být automaticky získávány. Tak jako tomu je u doporučení textových položek, kterými jsou např. vědecké dokumenty nebo noviny. Nicméně v oblastech jako je ubytování, musí být atributy položek doplňovány manuálně. V případě, kdy takových položek nepřibývá velké množství, je tento způsob efektivní. Problém ovšem nastává v oblastech, kde se systém může rozšiřovat i o tisíce položek denně (např. multimédia). To je jednou z nevýhod oproti kolaborativnímu filtrování, které nepotřebuje znát obsah položek a je využitelné v libovolné oblasti. [1]

Dalším problémem je provázanost položek na základě identických hodnot atributů. Takové položky jsou v doporučovacím systému neoddělitelné. Přesto, že jsou v celkovém kontextu naprosto rozdílné, jsou doporučeny uživateli. [1]

Závěrečný problém v obsahově filtrovaném systému je ten, že se uživateli dostane velmi zřídka, nebo vůbec, překvapivého doporučení. Doporučení bude založeno pouze na předešlých preferencích a hodnoceních uživatele. Na portfolio doporučení se nikdy nedostane například *Science fiction* kniha, i když kvalitní, pokud uživatel hodnotil pouze *detektivní* knihy. [10]

4.1.3 Další druhy doporučení

Na základě [6] mezi další doporučovací techniky patří *demografické*, *utility-based a knowledge – based.* Tyto techniky jsou málokdy považované za samostatné, ale často doplňují právě kolaborativní filtrování nebo filtrování založené na obsahu v hybridních přístupech.

Demografický přístup je postaven na doporučeních zakládajících se na osobních atributech uživatelů. Mezi takové atributy může patřit pohlaví, věk, národnost, vzdělání nebo příjem. Na základě těchto údajů jsou uživatelé zařazováni do skupin a dostává se jim podobného doporučení. Jedná se vlastně o druh kolaborativního doporučení.

Utility-based systém se nepokouší vytvářet dlouhodobé zobecňování chovaní jejich uživatelů, ale raději bere v potaz krátkodobou uživatelskou preferenci se sadou dostupných doporučení. Tato technika vyžaduje výpočet užitečnosti každé položky pro uživatele. Výhodou zmíněného přístupu je možnost zohlednění i jiných než přímo položkových atributů, kterými mohou být například dostupnost produktu a spolehlivost dodavatele pro uživatele, již mají okamžitou potřebu nákupu.

Posledním přístupem v tomto výčtu je *knowledge-based* technika, která je založena na znalostech o uživatelích a položkách. Doporučovací systém určuje, která položka bude na základě uživatelských preferencí předložena. Pokud nelze těmto preferencím vyhovět, systém nalezne jinou možnost. Je vhodná pro použití při doporučování například nemovitostí, které nekupujeme tak často a jsou unikátní a není tudíž možné získat dostatečný počet ohodnocení. V takovém systému musí být data získávána explicitně.

4.1.4 Hybridní přístup

Hybridní přístup kombinuje dvě nebo více doporučovacích technik za účelem překonání slabých stránek jednotlivých metod a zároveň využívá jejich přednosti. Studie, kterou představil [6] definuje sedm hybridních metod:

- **Weighted:** konečná hodnota položky se vypočítává kombinací vážených výstupů z několika nezávislých doporučovacích technik.
- **Switching**: v závislosti na aktuální situaci je vybrána jedna z několika technik, která slouží ke generování doporučení.
- **Mixed:** doporučovací techniky nezávisle generují doporučení, která jsou předkládána současně. Tímto způsobem lze vyřešit problém nové položky v systému při použití kolaborativního techniky a techniky založené na obsahu (content-based).
- **Feature combination**: využívá informace kolaborativní techniky jako doplňkové vlastnosti položek v přístupu založeném na obsahu (content-based).
- **Cascade**: Tímto způsobem je myšleno vyfiltrovat množinu položek jednou technikou, která je před doporučením uživateli vyselektována druhou.
- **Feature augmentation**: jedna technika je použita pro vytvoření hodnocení, které může být využito jinou k vytvoření doporučení.
- **Meta-level**: nejprve je vytvořen model první technikou, který je použit jako vstup pro druhou, která generuje doporučení.

Ačkoli je možná kombinace většiny technik, nejrozšířenější je hybridní doporučovací systém založený na kolaborativní a obsahové (content-based) technice. Dle [1] definujeme 4 způsoby, jak tyto techniky kombinovat:

- Obě techniky vykonávat samostatně a kombinovat jejich výstupy (doporučení).
- Použít techniky založené na obsahu jako vstup kolaborativního přístupu.
- Použít techniky kolaborativního filtrování jako vstup přístupu založeného na obsahu.
- Vytvořit model, který kombinuje a sjednocuje obě techniky.

4.2 Problém studeného startu

Mezi problémy studeného startu lze zařadit neznalost nových uživatelů, položek v doporučovacím systému a také studený start samotných systémů, které jsou zaváděny. Dále v podkapitolách dle [1] [11] [12]

4.2.1 Problém studeného startu uživatele

Problémem v tomto případě je kvalitní doporučení novému uživateli doporučovacího systému. Tato situace nastává v obou hlavních technikách doporučení, jak v kolaborativní tak v té založené na obsahu. V prvně zmíněné technice filtrování jsou podobnosti mezi uživateli vypočítávány na základě stejných hodnocení. Pokud ovšem nový uživatel neposkytl žádné hodnocení, není mu možné nic doporučit. Podobný problém nastává i ve filtrování založeném na obsahu. Nový uživatel neohodnotil dostatečné množství položek a systém tedy nemá možnost sestavení kvalitního profilu pro následné doporučení. Řešením je vyplnění krátkého formuláře, ohodnocení určitého počtu položek pro zjištění uživatelských preferencí nebo registrace uživatele. Další možností může být předložení obecně nejoblíbenější položky novému uživateli.

4.2.2 Problém studeného startu položky

Tento problém vzniká u nových položek, které mají v systému málo nebo žádné hodnocení. To se týká doporučovacích systémů založených pouze na kolaborativním filtrování, kde je zapotřebí právě ohodnocení položek uživateli. Položky, které jsou v systému nové, a žádný uživatel je doposud neohodnotil, nemohou být ani doporučeny, jestliže nejsou použita žádná další opatření k vyloučení této situace. V obsahově filtrovaných systémech k tomuto problému nedochází za předpokladu, že nové položky obsahují patřičné informace hned při vstupu do systému.

4.2.3 Problém studeného startu systému

Poslední problém nastává u nových doporučovacích systémů, které mají čerstvě přidané uživatele a nedostatečně ohodnocené položky. Zmíněná situace se nejčastěji vyskytuje opět u kolaborativního filtrování. Tento problém vlastně kombinuje oba předešlé. Výsledkem je extrémně řídká matice ohodnocení.

4.3 Příklady reálných doporučovacích systémů

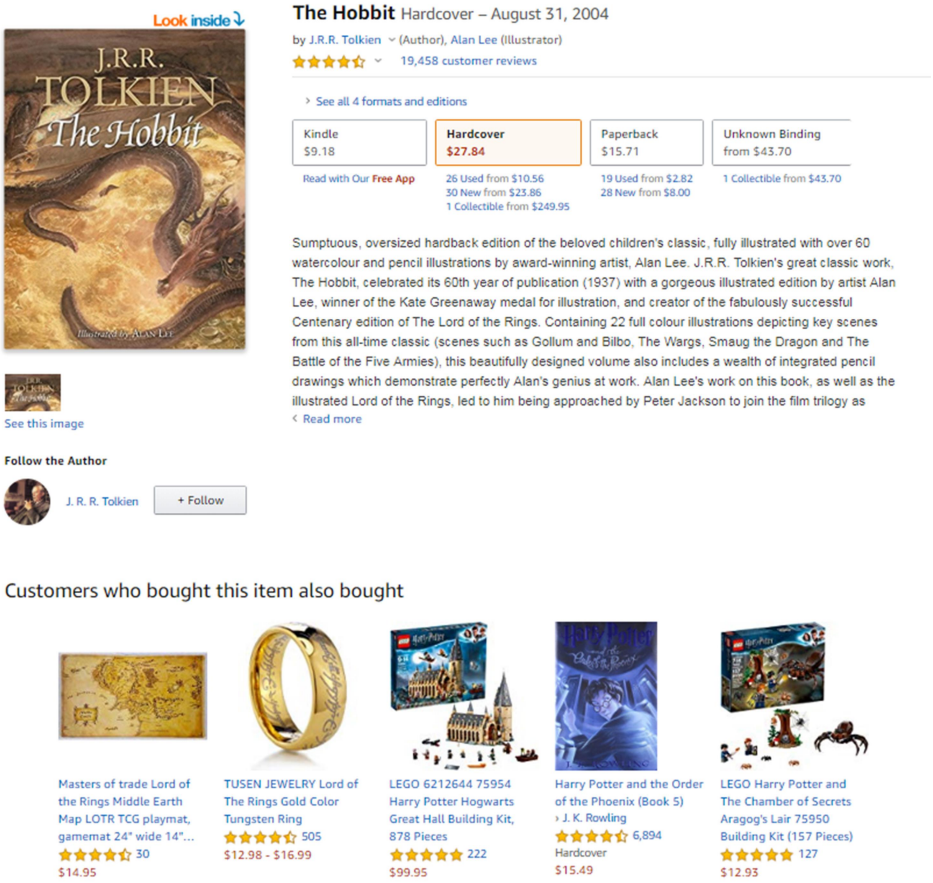
4.3.1 Amazon.com

Za jeden z prvních moderních a nejznámějších doporučovacích systémů lze považovat internetový obchod Amazon.com. Byl založen roku 1994 zakladatelem Jeffem Bezosem původně jako online knihkupectví. V dnešní době však na tomto portálu lze nalézt kromě knih také hudbu, filmy, elektroniku a prakticky vše, co člověka napadne. V kategorii knih dle [16] je možno nalézt několik typů doporučovacích systémů:

- **Customers who Bought:** Každá kniha na portálu Amazon.com obsahuje, jako na mnoha obdobných portálech, informaci o právě prohlíženém objektu, v tomto případě knize. Zde je možné nalézt dva druhy doporučení. První z nich doporučuje knihy zakoupené jinými uživateli společně s knihou, kterou si zákazník právě prohlíží. Druhý doporučuje autory, jejichž knihy jsou často nakupovány těmi zákazníky, kteří si zakoupili knihy autora právě prohlížené knihy.
- **Eyes:** Funkcí Eyes je možnost zasílání upozornění do emailové schránky zákazníka, pokud se objeví nová položka v Amazon.com katalogu knih. Požadavek může obsahovat jméno autora, název knihy, ISBN nebo informace o datu uveřejnění. Lze zadávat jednoduché požadavky nebo složitější na základě booleovské logiky (AND/OR).
- **Amazon.com Delivers:** Variace funkce Eyes. Zákazník si v seznamu vybere kategorii nebo žánr, který preferuje (knihy určitého autora, biografie, vaření). Takovýto odběratel následně dostává emailová upozornění, pokud se v daném odvětví objeví nová kniha.
- **Book Matcher:** Uživatel má možnost ohodnotit knihu, kterou přečetl, na bodové škále 1-5, přičemž 5 znamená nejvyšší známku. Po ohodnocení několika knih si uživatel může vyžádat seznam knih, které by ho mohly zajímat. Je mu tedy předloženo několik knih, které by měly odpovídat jeho vkusu, ale ještě je neohodnotil. Uživatel může přidat zpětnou vazbu tím, že tyto doporučené knihy ohodnotí.
- **Customer Comments:** Tato funkce nabízí uživateli slovní doporučení, které je zobrazené na informační stránce každé knihy společně s hodnocením v rozmezí 1 až 5

hvězdiček. Toto doporučení je vlastně slovní názor na knihu od uživatelů, kteří si ji pořídili.

Obrázek 3: Ukázka doporučení ze stránky Amazon.com



Zdroj: www.amazon.com

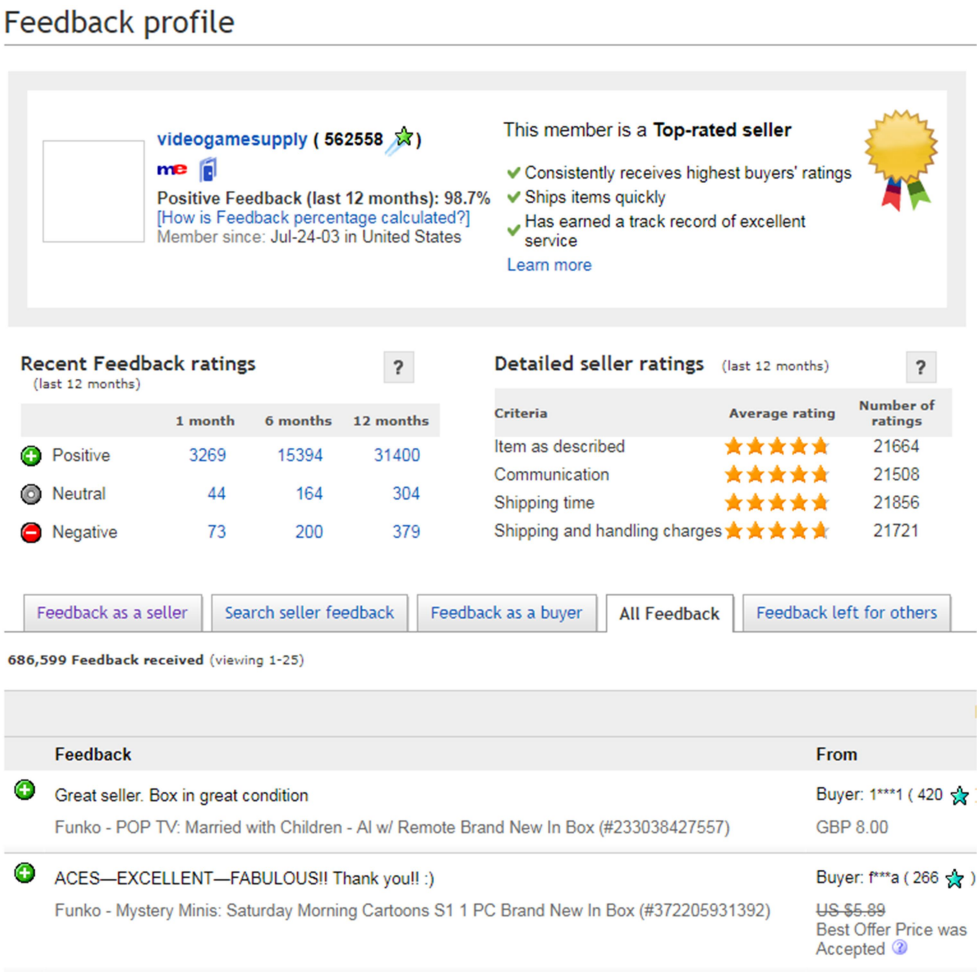
4.3.2 eBay

Společnost eBay je nejznámější internetovou aukční síní ve Spojených státech, potažmo ve světě. Tuto společnost se sídlem v San José založil Pierre Omidyar roku 1995. Velmi široká nabídka prodejců a kupujících z celého světa umožňuje nakoupit nebo prodat prakticky cokoliv. Aukce probíhají stylem klasického „přihazování“ nebo lze využít možnosti „nakup hned“. Některé typy nabídek umožňují pouze druhou variantu prodeje. [17]

Samotné doporučování probíhá pomocí modulu **Feedback Profile**. Ten umožňuje prodejcům a kupujícím do těchto profilů *zpětné vazby* přidávat zkušenost s uskutečněným obchodem. Tato zkušenost se skládá z hodnocení (pozitivní, neutrální, negativní) a komentáře. Díky takovému profilu lze snadno vypozorovat, jak je daný uživatel spolehlivý či nikoliv. Hodnocení je rozděleno do kategorií: poslední měsíc, půlrok a rok, ale také do

detailnějších hodnocení, jakými jsou popis položky, komunikace, rychlost a cena přepravy. [16]

Obrázek 4: Ukázka "Feedback profile" na ebay.com



Zdroj: www.ebay.com

5 Teorie pravděpodobnosti a Bayesovské sítě

Bayesovské sítě využívají teorii pravděpodobnosti (konkrétně Bayesova větu) a teorii grafů k modelování problémů, u kterých dochází ke zpracování nejistých informací. Aplikují pravděpodobnostní vzorce k tomu, aby vyhledávaly cíle rychleji. V následujících podkapitolách tuto teorii rozvedu

5.1 Podmíněná pravděpodobnost

Předpokládejme náhodný jev *A,* jeho pravděpodobnost *P(A)*, která leží v intervalu ⟨0;1⟩. Výsledek pravděpodobnosti $P(\mathrm{A}) = 0$ nazývejme *jevem nemožným*, výsledek $P(\mathrm{A}) = 1$ *jevem jistým*. Pravděpodobnost uskutečnění jevu A za předpokladu, že nastal jev B, se zapisuje *P(A/B)* a nazývá se *podmíněná pravděpodobnost*. Ta je rovna často používanému součinovému zápisu: [13]

$$P(AB) = P(A|B) \times P(B)$$

5.1.1 Bayesova teorie

Bayesovská teorie, potažmo Bayesova věta se pojí k anglickému matematikovi jménem Thomas Bayes. Tato teorie vysvětluje vztah mezi podmíněnou pravděpodobností a její opačnou podmíněnou pravděpodobností. Vzorec níže znázorňuje výpočet podmíněné pravděpodobnosti, který lze vyvodit ze vzorce v podkapitole 4.1. Mějme dva náhodné jevy *A* a *B* s pravděpodobnostmi *P(A)* a *P(B)*, přičemž $P(B) > 0$. Potom platí:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

kde *P(A|B)* je podmíněná pravděpodobnost jevu *A* za předpokladu, že nastal jev *B*, a naopak *P(B|A)* je pravděpodobnost jevu *B* podmíněná výskytem jevu *A*. Také lze pojmenovat jako pravděpodobnost hypotézy *H* při evidenci *E*. [14]

Hypotéz *H* bývá zpravidla více a značíme je H_n. Nás ovšem zajímá ta, pro danou evidenci *E*, s nejvyšší podmíněnou pravděpodobností H_{MAP}. Důležitá je pro nás pouze maximální pravděpodobnost, nikoliv konkrétní hodnota, čímž lze výpočet více zjednodušit zanedbáním jmenovatele. Z předešlého vzorečku, kde *A=H, B=E*. [14]

$$H_{MAP} = P(E|H_{MAP}) \times P(H_{MAP}) = max_n(P(E|H_n) \times P(H_n))$$

Podle této věty lze stanovit vliv jedné evidence na uvažovanou hypotézu. Jak ale stanovit podmíněnou pravděpodobnost, pokud máme evidencí více? Konkrétně $P(H|E_1....., E_k)$. Jedním z možných řešení je bayesovská síť, která navíc počítá se skutečností, že jevy mohou být navzájem závislé. [14]

5.2 Bayesovské sítě

Dle [14] je: „*Bayesovská síť je acyklický orientovaný graf zachycující pomocí hran pravděpodobnostní závislosti mezi náhodnými veličinami. Ke každému uzlu u (náhodné veličině) je přiřazena pravděpodobnostní distribuce tvaru* $P(u|rodiče(u))$*, kde rodiče(u) jsou uzly, ze kterých vycházejí hrany do uzlu u. Uspořádejme (a očíslujme) všechny uzly sítě tak, že rodiče jsou před svými dětmi (mají nižší pořadové číslo). Potom pro každý uzel* ui *platí, že je podmíněně nezávislý na všech uzlech s nižším pořadovým číslem s výjimkou svých rodičů podmíněno svými rodiči.*“

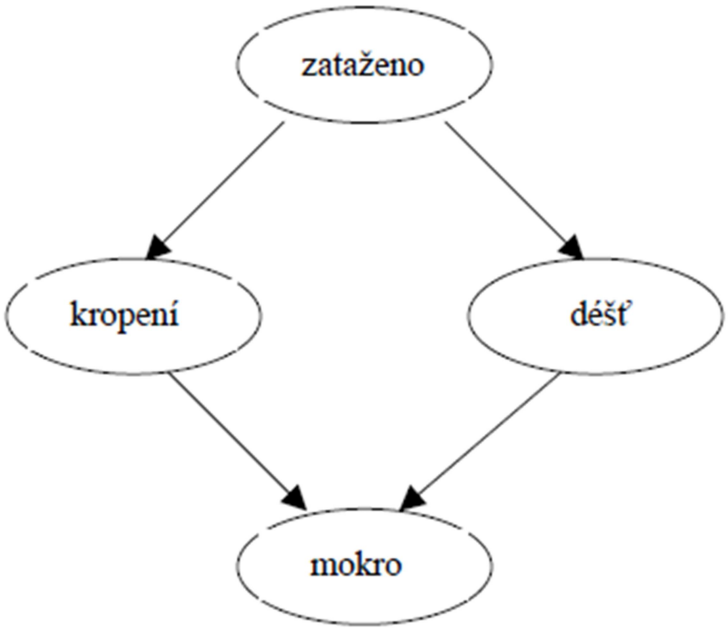
Z toho lze vyvodit:

$$P(U_i|rodiče(U_i))$$

To umožňuje vypočítat sdruženou pravděpodobnostní distribuci celé sítě takto:

$$P(u_1, \ldots, u_n) = \prod_{i=1}^{n} P(u_i \mid rodiče(u_i))$$

Obrázek 5: Model Bayesovské sítě



Zdroj: Dobývání znalostí z databází [14]

U Bayesovskě sítě této podoby by měla sdružená distribuce tvar:

$$P(Z,K,D,M) = P(Z)P(K|Z)P(D|Z)P(M|K,D)$$

Obrázek 6: Podmíněné pravděpodobnosti uzlů

Z	P(K=0)	P(K=1)
0	0.5	0.5
1	0.9	0.1

P(K|Z)

P(Z=0)	P(Z=1)
0.5	0.5

P(Z)

Z	P(D=0)	P(D=1)
0	0.8	0.2
1	0.2	0.8

P(D|Z)

K	D	P(M=0)	P(M=1)
0	0	1.0	0.0
1	0	0.1	0.9
0	1	0.1	0.9
1	1	0.01	0.99

P(M|K,D)

Zdroj: Dobývání znalostí z databází [14]

Pomocí takovéto sítě lze provádět pravděpodobnostní odvozování (inference). Ze struktury sítě (obrázek 5) a pravděpodobností přiřazených jednotlivým uzlům (obrázek 6), lze vypočítat maximální podmíněnou pravděpodobnost libovolného uzlu. Je tedy možné vypozorovat, co je příčinou toho, že je mokro, jestli kropení nebo déšť. To se nazývá *diagnostická inreference*, neboli od zdola-nahoru (od pozorování jevu k příčině). Dále můžeme vypočítat, s jakou pravděpodobností bude mokro, pokud je zataženo. V tomto případě se jedná o *kauzální inferenci*, postup shora-dolů (od příčin k důsledkům). [14]

Bayesovské sítě mají spoustu výhod. Díky uchovaným závislostem mezi proměnnými v celém modelu dokáží vyřešit situace s chybějícími daty. Dále v sobě uchovávají dva typy znalostí: Vazby mezi atributy (hrany v grafu) a pravděpodobnostní hodnoty těchto atributů (ohodnocené uzly v grafu). To nám při modelování a učení sítě umožňuje z dat odvodit strukturu sítě i pravděpodobnosti. Další z možností je vycházet ze známé struktury a z dat odvozovat pouze podmíněné pravděpodobnosti. [14]

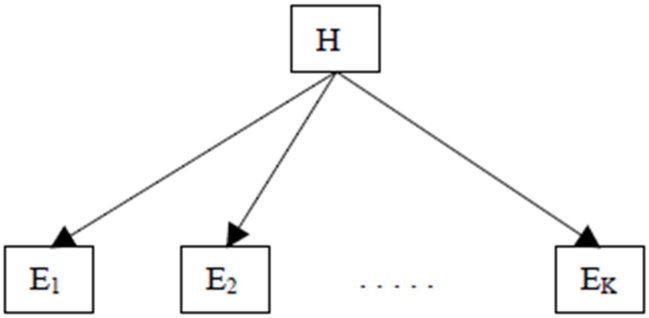
5.2.1 Bayesovský klasifikátor

Jedním ze způsobů, jak využít Bayesovské sítě, je i grafické znázornění Bayesovského klasifikátoru, který vychází z překladu, že jednotlivé evidence $E_1,\ldots..E_K$ jsou podmíněně nezávislé, při platnosti hypotézy H. Někdy nazývaný také naivním, protože předpoklad podmíněné nezávislosti je v reálných úlohách jen málokdy splněn. Tento způsob přesto vykazuje úspěšné klasifikování a je jednoduchý na implementaci. Ke klasifikaci za pomoci Bayesovského klasifikátoru se hypotéza s nejvyšší podmíněnou pravděpodobností H_{MAP} vypočítá: [14]

$$H_{MAP} = \prod_{k=1}^{K} P\left(E_k|H_{MAP}\right) \times P(H_{MAP}) = max_n \left(\prod_{k=1}^{K} P\left(E_k|H_n\right) \times P(H_n)\right)$$

V Bayesovské síti by reprezentace takovéhoto klasifikátoru vypadala následovně:

Obrázek 7: Model Bayesovské sítě pro Bayesovský klasifikátor



Zdroj: Dobývání znalostí z databází [14]

Tato síť obsahuje pouze jeden uzel, který je rodičem zbylých vstupních uzlů. Ty jsou navzájem nezávislé a nejsou tudíž spojené hranami. Sdružená distribuce vypadá následovně:

$$P(H, E_1, E_2, \ldots, E_k) = P(H)P(E_1|H)P(E_2|H) \ldots P(E_k|H)$$

6 Praktická část

Dříve se doporučování zaměřovalo spíše na odvětví jako filmy, hudba, zábava a dalších. Na přelomu tisíciletí však došlo k velkému rozšíření i v doporučovacích systémech zaměřených na turisty. V tomto odvětví je několik rozdílných služeb, které lze doporučit. Dle [15] je možné rozdělit takto:

- ubytování,
- restaurace,
- památky,
- letenky,
- základní informace o dané oblasti,
- průvodce oblastí,
- cestovatelský zážitek.

Některé systémy se zaměřují pouze na jeden z těchto sektorů, ale většinou jde o komplexnější řešení zahrnující dva a více. Systémy zabývající se určitou oblastí mohou nabízet velké zeměpisné oblasti nebo konkrétní místa. Detailnější řešení je možno naleznout v [15] , kde je také zmíněno porovnání mezi webovými a mobilními doporučovacími systémy určených pro turisty. Ty mobilní mohou využívat například aktuální pozici uživatele podle GPS.

Z těchto příkladů jsem se rozhodl vytvořit jednoduchý doporučovací systém, konkrétně zaměřený na ubytování v Českých Budějovicích. Ten bude řešen formou webového modulu a doporučení bude zpracovávat na základě Bayesovské sítě, která je bude vypočítávat z hodnocení uživatelů. Tento modul bude dále schopný navrhnout ubytování na základě konkrétních preferencí.

6.1 Zvolená metoda

Po nastudování materiálů uvedených v teoretické části této bakalářské práce a rozhodnutí vytvořit modul doporučovacího systému, který bude doporučovat ubytování v Českých Budějovicích, se mi jako nejlepší varianta zdála ta, která využívá podobnosti uživatelů a jejich hodnocení. Tedy mít v modulu vytvořen nástroj, jenž je založen na explicitním přístupu. Ten nám pomůže generovat data potřebná k tvorbě doporučení, v tomto konkrétním případě je nástrojem zákaznické hodnocení položek. Data posléze využívá

Bayeskovská síť pro klasifikaci jednotlivých uživatelů do potřebných skupin. Cílem je novému návštěvníkovi stránek doporučit takové ubytování, které by ho mohlo zajímat. Toho lze dosáhnout právě tím, že v databázi systému budou informace o uživatelích s podobnými parametry nově příchozího, jako je věk, národnost a pohlaví.

6.2 Využitá technologie

Pro tuto práci bylo zvoleno vývojové prostředí Netbeans ve verzi 8.2, která poslední nesla název NetBeans, od verze 9.0 již známé jako Apache NetBeans. NetBeans IDE je svobodné, zdarma distribuované, integrované vývojové prostředí (IDE), které vlastní firma Oracle Corporation. V současné době je NetBeans v režii Apache Software Foundation. Technologický základ tvoří platforma NetBeans a primárně je určena pro vývoj v programovacím jazyce Java, ale díky modulární softwarové architektuře umožňuje programování i v jiných programovacích jazycích.

Zdrojový kód do verze 8.2 dostupný pod licencí Common Development and Distribution License (CDDL) v1.0 a GNU General Public License (GPL) v2. Od verze 9.0 licencován jako Apache licence. [18]

6.2.1 Java Enterprise Edition

Java Enterprise Edition (neboli Java EE, od roku 2018 vyvíjena pod názvem Jakarta EE) je součást platformy Java určená pro vývoj a provoz podnikových aplikací a informačních systémů. Základem pro platformu Java EE je platforma Java SE a nad ní jsou definovány součásti tvořící Java EE.

Technologie v Java EE

Dle [19] jsou součástí platformy Java EE především specifikace pro:

- **JSP (Java Server Pages):** technologie umožňuje vkládat speciální direktivu do HTML kódu, která spustí Java kód. Na daná místa ve stránce se tak vloží data, která získala Java např. z databáze.
- **JSF (Java Server Faces):** konkurenční a modernější technologie k JSP. Celá webová stránka je reprezentována jako XML soubor. Web se skládá z již připravených komponent (formuláře, tabulky, seznamy), které lze jednoduše plnit daty z Javy.

- **JDBC (Java DataBase Connectivity):** JDBC je standardní rozhraní pro práci s různými typy databází v jejich jazyce SQL.
- **JPA (Java Persistence API):** JPA je rozhraní, umožňující objektovou práci s daty. S databází nekomunikujeme přímo v SQL, ale pomocí mezivrstvy ORM. Pracujeme tedy pouze s objekty.
- **EJB (Enterprise Java Beans):** Komponenty obchodní logiky.

6.2.2 Aplikační server GlassFish

Serverovou aplikaci zastřešující všechny knihovny, které dle specifikací Java EE platformy zajišťují požadovanou funkcionalitu, označujeme pojmem aplikační server. Tyto knihovny implementují veškerá API obsažená v Java EE. Kromě toho poskytuje aplikační server další klasické služby jako např. administrátorskou konzoli, logování atp. Ze známých implementací Java EE platformy můžeme zmínit např. JBoss od firmy Red Hat, JRun od firmy Adobe Systems. Mezi další patří implementace od Sun Microsystems, nyní spadající pod Oracle Corporation s názvem GlassFish. 0

Právě GlassFish, konkrétně verze 5.0, aplikační server vyvinutý pro platformu Java EE je použit v této práci. Řadí se mezi open source projekty podléhající licencím GPL a CDDL.

6.2.3 Databázový systém

Jako databázový systém jsem zvolil open source relační databázi Apache Derby napsanou v jazyce Java a vydávanou pod *Apache licence*. Přímo v Netbeans nese název Java DB a je postavena na SQL standardech. [21]

Derby Databáze umožňuje funkčnost ve dvou režimech:

- **Plnohodnotný databázový server:** jedná se o serverovou verzi postavenou na vlastním JVM. K připojení je nutný klient JDBC běžící na vlastních JVM. Připojování je možno jak z localhostu, tak i ze sítě. Jako vlastní databázové servery běží i řada konkurentů, např. MySQL, MS-SQL, PostGRE SQL, apod.
- **Vestavěnou databázi (Embedded):** jedná se prakticky o vložení databázového enginu do našeho projektu (programu). S danou databází pracujeme přes vytvořenou instanci. Což znamená, že různým Java programům můžeme vložit vlastní databázi a nemusíme klíčová data ukládat do externích. [22]

6.3 Návrh doporučovacího systému

V praktické části jsem se rozhodl vytvořit jednoduchý doporučovací modul, který bude mít na starost výpočty a následné doporučování. Tento modul by mohl sloužit jako doplněk ke komplexnímu ubytovacímu systému. V aplikaci je demonstrované řešení studeného startu, a to jak v problému studeného startu uživatele, tak i položky. Na pozadí všech výpočtů je zakomponovaná jednoduchá Bayesovská síť, popsaná v kapitole 5.2.1 ve spojení s hodnocením uživatelů.

Modul se zabývá ubytováním v Českých Budějovicích, přičemž v tabulce níže lze vidět rozdělení ubytování do kategorií a konkrétních ubytovacích zařízení.

Tabulka 3: Položky ubytování v doporučovacím systému

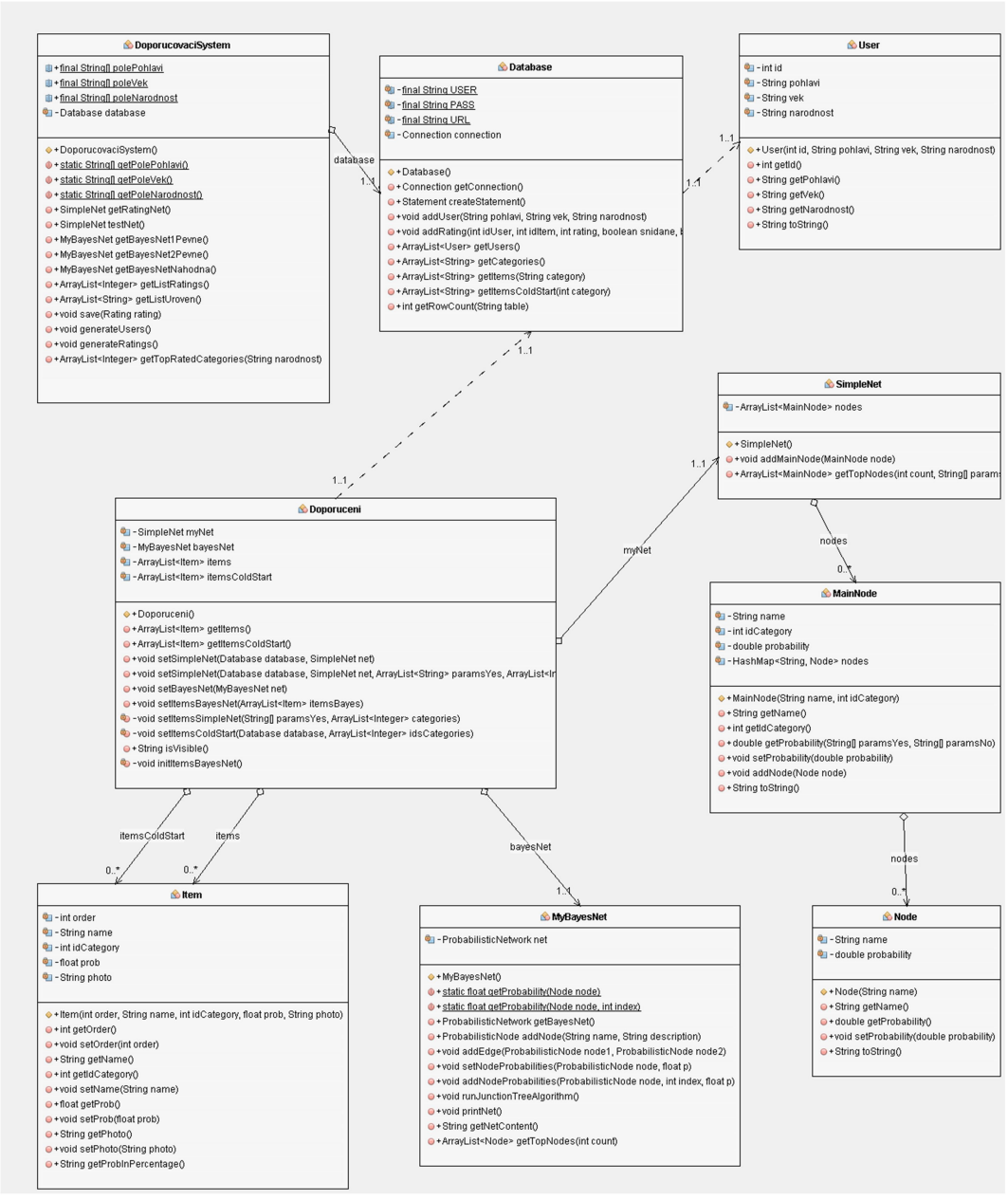
Kategorie	Ubytování
Koleje	kolej K1, kolej K2, kolej K3, kolej K4, kolej K5, kolej K6, koleje Pedagog
Ubytovny	Ubytovna Hochtief, Ubytovna Stavounion, Ubytovna U nádraží, Ubytovna Zásoby s.r.o.
Penziony	Penzion Centrum, Penzion Pegast, Penzion Smetanka, Penzion u Rudolfa
Hotely	Hotel Adler, Hotel Budweis, Clarion Congres Hotel, Grandhotel Zvon, HC Hotel, Hotel Zátkův Dům

Jedná se o modul, u kterého si klademe za cíl zpracovat a vypočítat doporučení, které bude probíhat na pozadí. Nebylo snahou ani záměrem v rozsahu této bakalářské práce zpracovávat komplexní frontend webovou aplikaci připomínající například portál booking.com.

6.3.1 Diagram tříd

V níže uvedeném diagramu jsou zahrnuty nejdůležitější třídy pro práci s modulem, včetně práce s databází a využití Bayesovských sítí. Nejsou zde zahrnuty všechny třídy pro práci s GUI.

Obrázek 8: Diagram tříd



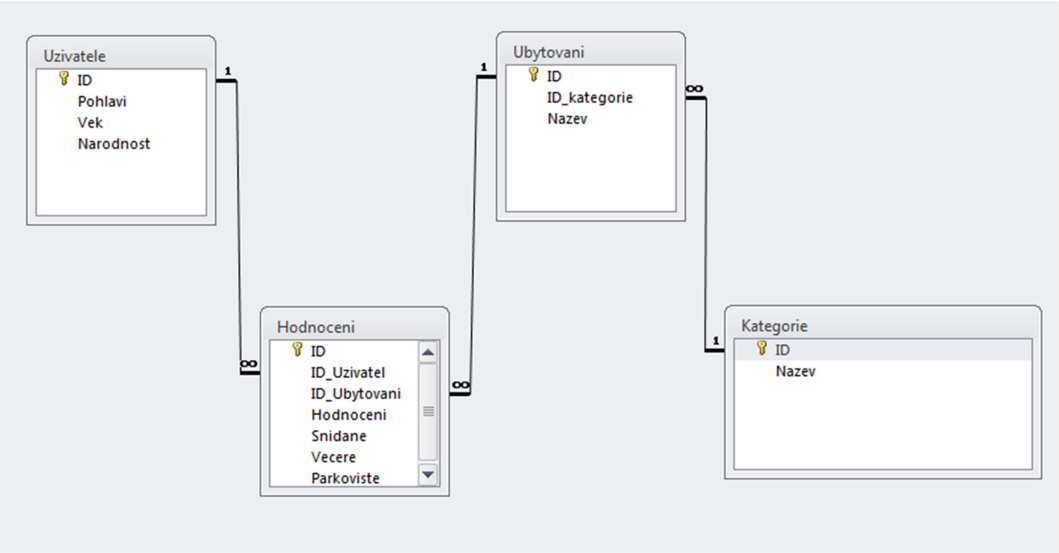
Popis jednotlivých tříd:

- **DoporucivaciSystem:** tato třída pracuje s databází a vytváří ohodnocenou síť pomocí metody getRatingNet, která je popsána v kapitole 6.3.4.
- **Database**: jedná se o třídu, která pracuje s databází. Konkrétně načítá data pro tvorbu ohodnocené sítě a ukládá hodnocení uživatelů.
- **Doporuceni:** uchovává informace o zobrazených položkách jak studeného startu, tak vyhledávání. K tomu slouží kolekce ArrayList<Item> itemsColdStart a ArrayList<Item> items.
- **Item:** představuje položku ubytování, která je využívána JSF pro její zobrazení.
- **SimpleNet:** tato třída symbolizuje jednoduchou Bayesovskou síť založenou na Bayesově klasifikátoru. Je tvořena kolekcí uzlů ArrayLst<MainNode> nodes neboli jednotlivých položek a umožňuje výpočet těch nepravděpodobnějších, tzv. TopNodes, na základě četnosti a hodnocení.
- **MainNode:** třída představuje konkrétní ubytování a kolekce ArrayList<Node> nodes uchovává parametry ubytování a umožňuje získání pravděpodobnosti při zadaných parametrech.
- **Node:** tato třída představuje jeden parametr (tj. věk, pohlaví, národnost, snídaně, večeře či parkoviště) u konkrétní položky. Obsahuje informaci o pravděpodobnosti daného parametru.
- **MyBayesNet:** využívá knihovnu unBBayes pro ukázku složitějšího typu Bayesovské sítě.
- **User:** uchovává informace o uživateli (věk, pohlaví a národnost).

6.3.2 Schéma databáze

Použitá relační databáze se skládá ze 4 tabulek: Uživatelé, Ubytování, Kategorie a Hodnocení, které je tou nejdůležitější z uvedených. Právě data z tabulky Hodnocení jsou využívána pro výpočet doporučení.

Obrázek 9: Schéma relační databáze



Jedná se o jednoduchý model databáze, který je využíván čistě pro potřebu doporučovacího modulu a v komplexnějším řešení by byl součástí většího databázového celku.

6.3.3 Hodnocení položek

Jednou z funkcionalit tohoto modulu je hodnocení jednotlivých ubytování. K tomu slouží záložka Hodnocení ve webovém prostředí.

Obrázek 10: Záložka Hodnocení v doporučovacím modulu

V rozsahu této bakalářské práce slouží tato webová stránka pouze k hodnocení ubytování a jeho následnému uložení do databáze. Hodnocení napomáhá k tvorbě doporučení novým uživatelům. V komplexní aplikaci určené k vyhledávání ubytování by tento modul mohl být rozšířen následovně: *Mějme databázi uživatelů, u kterých víme, že se přes náš systém ubytovali. Takový zákazník obdrží email s prosbou o hodnocení jeho ubytování, jak je dnes běžným zvykem. Zde provede hodnocení ubytování, které navštívil a vybere služby, které využil.* Jak je možno vidět, v databázi se informace o uživateli skládají z hodnot ID, Pohlaví, Věk a Národnost.

Obrázek 11: Tabulka uživatelé v databázi

#	ID	POHLAVI	VEK	NARODNOST
1	1	muž	50 a více	Německo
2	2	žena	50 a více	Francie
3	3	muž	20-35	Anglie
4	4	žena	0-20	Francie
5	5	muž	35-50	Anglie
6	6	žena	0-20	Česká republika
7	7	žena	50 a více	Česká republika

6.3.4 Základní doporučení na základě hodnocení položek

Nejdůležitějším bodem modulu je záložka Doporučené. Zde se nachází doporučené ubytování pro nové uživatele, jež vychází z hodnocení a četností ubytování, přičemž o samotný výpočet se stará Bayesovská síť.

Obrázek 12: Doporučení založeného na hodnocení uživatelů

Jednoduchá bayesova síť - na základě hodnocení uživatelů

1. Hotel Budweis	10.0 %	
2. Kolej K1	8.67 %	
3. Clarion Congres Hotel	6.0 %	

Výpočet doporučení

Doporučení těchto konkrétních položek je vypočteno pomocí Bayesova vzorce, který vychází z předpokladu určení pravděpodobností na základě četností. Hodnocení položek nám v systému napomáhá ke zkvalitnění doporučení, protože bez něj by systém jako první položku doporučil tu nejvíce navštěvovanou. Ubytování je možno hodnotit na bodové škále 1-5. Každý bod má váhu 0,2, aby výsledné hodnocení nabývalo hodnoty v rozsahu 0,2-1, tedy maximální hodnocení by bylo 100 %.

Obrázek 13: Tabulka hodnocení v databázi

#	ID	ID_UZIVATEL	ID_UBYTOVANI	HODNOCENI	SNIDANE	VECERE	PARKOVISTE
1	1	1	11	4	☑	☑	☐
2	2	2	13	4	☑	☑	☑
3	3	3	10	5	☐	☐	☑
4	4	4	11	5	☑	☑	☑
5	5	5	12	3	☐	☑	☐
6	6	6	9	4	☑	☐	☑
7	7	7	1	5	☐	☐	☑

Pro každou položku ubytování v tabulce Ubytování je proveden výpočet H(map) pro výpočet maximální pravděpodobnosti položky. Ten vychází z dat na obrázku 13 a vypadá následovně:

$$H(map) = P(hotel\ Budweis) \times P(hodnocení\ hotelu\ Budweis)$$

- *P(hotel Budweis)* = četnost hodnocení hotelu Budweis ve sloupci ID v celé tabulce. V tomto případě 7 záznamů z 60. P = 7 / 60 = 0,1166.
- *P(hodnocení hotelu Budweis)* = pravděpodobnost doporučení na základě hodnocení uživatelů, v tomto případě hotelu Budweis. Průměrné hodnocení tohoto konkrétního hotelu je vynásobeno váhou 0,2. Tím je získána pravděpodobnost doporučení na základě hodnocení. Průměrné hodnocení hotelu Budweis činí 4,2855. P = 4,2855 × 0,2 = 0,8571.

H(map) = 0,0999 = 10,00 % jak je možno vidět na obrázku 12.

Výpočet je proveden pro všechny položky a následně jsou seřazeny sestupně 3 nejlepší výsledky. Tento vzorec je používán ve všech následujících kapitolách, kde jsou ovšem do výpočtu přidány další parametry.

V navrženém modulu je jednou z nejdůležitějších metod metoda nazvaná getRatingNet, která se stará o sestavení celé ohodnocené sítě. Je tedy využita při každém zobrazení doporučení, jak prvotním, tak při použití vyhledávání.

Obrázek 14: Metoda getRatingNet

```
public SimpleNet getRatingNet() throws SQLException {
    SimpleNet net = new SimpleNet();

    Statement statement = database.createStatement();
    ResultSet resultItems = statement.executeQuery("SELECT Id, Nazev, ID_Kategorie FROM Ubytovani");

    int count = database.getRowCount("Hodnoceni");
    // polozky
    while (resultItems.next()) {
        int idItem = resultItems.getInt("Id");
        String name = resultItems.getString("Nazev");
        int idCategory = resultItems.getInt("ID_Kategorie");

        MainNode mainNodeItem = new MainNode(name, idCategory);
        statement = database.createStatement();
        ResultSet resultRating = statement.executeQuery("SELECT Hodnoceni FROM Hodnoceni WHERE ID_Ubytovani = " + idItem);
        int absoluteFrequencyItem = 0;
        double averageRating = 0;
        while (resultRating.next()) {

            averageRating += resultRating.getInt("Hodnoceni");
            absoluteFrequencyItem++;
        }
        averageRating /= absoluteFrequencyItem;
        double p = averageRating * 0.20; // 1-5 * 0,2 => 0,2-1
        System.out.println("Pravdepodobnost doporuceni na zaklade prumerneho hodnoceni: " + p);
        double relativeFrequencyItem = (double)absoluteFrequencyItem/count;
        System.out.println("Relativni cetnost je: " + relativeFrequencyItem);
        // nastavi pravdepodobnost uzlu !!! relativni cetnost * prumerne hodnoceni (* 0.20)
        mainNodeItem.setProbability(relativeFrequencyItem * p);
```

Pomocí SQL dotazu Select jsou z databáze získány atributy ID, Název a ID_Kategorie z tabulky Ubytování, které jsou uloženy do proměnných pro další zpracování. Každá položka je následně uložena do uzlu MainNode. Z tabulky hodnocení jsou vybrány řádky s hodnocením, u kterých je proveden výpočet četnosti a průměrného

hodnocení podle jednotlivých idItem. Pomocí mainNodeItem.setProbability se každé položce přiradí pravděpodobnost doporučení.

6.4 Testování navrženého modulu na problematice studeného startu

V této kapitole uvádím testování funkčnosti celého modulu na kvalitě doporučení nově příchozích uživatelů, o kterých nemá systém žádné informace a doporučení položky, která zatím nemá žádné hodnocení.

Důležitým zdrojem dat pro toto testování je tabulka 4 uvedená níže, kde je zobrazen počet ubytovaných zákazníků podle jejich národnosti v jednotlivých kategoriích.

Tabulka 4: Četnost zákazníků rozdělených podle kategorie ubytování a národnosti

Kategorie	Národnost	Počet
Ubytovny	Velká Británie	1
	Česká republika	2
	Francie	1
	Německo	0
	Slovensko	6
Koleje	Velká Británie	6
	Česká republika	4
	Francie	3
	Německo	3
	Slovensko	6
Penziony	Velká Británie	0
	Česká republika	1
	Francie	4
	Německo	0
	Slovensko	0
Hotely	Velká Británie	4
	Česká republika	4
	Francie	3
	Německo	10
	Slovensko	2

6.4.1 Problém studeného startu na straně uživatele

První a již zmíněnou možností řešení studeného startu je zakomponování hodnocení do modulu doporučení. Uživateli se tak zobrazí jen ty nejnavštěvovanější a nejlépe hodnocené položky, tedy ty nejoblíbenější. Toto řešení lze doporučit libovolnému doporučovacímu systému, který je založen na hodnocení položek. Popsáno v kapitole 6.3.4.

Informace o uživateli a jeho preference

Při tomto řešení je zapotřebí získat od návštěvníka stránek jisté osobní informace a preference. To se provádí na stránce s vyhledáváním, viz obrázek 15. Po zvolení volitelných parametrů a osobních údajů provede modul doporučení. U vyhledávání je použita metoda doporučení na základě obsahu, konkrétně pod parametrem Úroveň. S tímto typem doporučení se nejdříve nepočítalo, ale v tomto případě jde o velmi účinné a lehké řešení na implementaci. Pokud si uživatel zvolí úroveň požadovaného ubytování, jsou mu vybrány jen příslušné kategorie. Při takovém řešení je nutné mít databázi dobře sestavenou a položky ubytování vhodně přiřazené. Ovšem při takto malém vzorku položek tento problém prakticky odpadá.

Obrázek 15: Záložka Vyhledávání v doporučovacím modulu

Po stisku tlačítka „Provést doporučení“ je vyslán signál k provedení metody setItemsSimpleNet, které jsou předány jednotlivé parametry a případně požadovaná úroveň ubytování.

Obrázek 16: Metoda setItemsSimpleNet

```
private void setItemsSimpleNet(String[] paramsYes, ArrayList<Integer> categories) {
    String[] paramsNo = new String[] {};

    items.clear();
    ArrayList<MainNode> topNodes = myNet.getTopNodes(3, paramsYes, paramsNo, categories);
    int i = 1; // pocitadlo pro 1., 2., 3. misto
    // priprava polozky pro GUI
    for (MainNode node : topNodes) {
        items.add(new Item(i, node.getName(), node.getIdCategory(),
                (float)node.getProbability(paramsYes, paramsNo), MySystem.images.get(node.getName())));
        i++;
    }
}
```

Dále metoda topNodes nalezne prvních (3+*k*) položek s nejvyšší pravděpodobností doporučení, kde *k* určuje počet položek, které mají stejnou pravděpodobnost jako položka umístěna na 3. místě. Pokud tento případ nastane, dojde k rozšíření stávající tabulky doporučení o *k* položek.

Vycházíme ze vzorce H(map) v kapitole 6.3.4, pouze je rozšířen o vybrané parametry:

$$H(map) = P(\text{četnost } U) \times P(\text{hodnocení } U) \times \prod P(parametr_{a\ldots\ldots n}\ U)$$

kde U = ubytování, $parametr_{a\ldots\ldots n}\ U$ značí zvolený parametr u zvoleného ubytování U.

Poslední metoda uvedená na obrázku 17 ukazuje jednotlivé roznásobení parametrů, kde paramsYes označuje vybrané parametry uživatelem, paramsNo označuje ty nevybrané. Následný cyklus je počítán na základě výše zmíněného vzorec.

Obrázek 17: Metoda getProbability

```
public double getProbability(String[] paramsYes, String[] paramsNo) {
    double p = probability;
    for (String parameter : paramsYes) {
        p *= nodes.get(parameter).getProbability();
    }
    for (String parameter : paramsNo) {
        p *= (1 - nodes.get(parameter).getProbability());
    }
    return p;
}
```

Získání IP adresy

Tento způsob se jeví jako ideálním řešením pro nově příchozí návštěvníky stránky. Nemusí nic vyplňovat, ale okamžitě po zjištění, z jaké oblasti přistupují, jim je zobrazeno doporučené ubytování na základě stejné národnosti nám známých, a již dříve ubytovaných, zákazníků. A to díky údajům v databázi.

Výpočet a programová část zde probíhá standardně jako ve výše zmíněné metodě. Jen je hned na začátku volána metoda setItemsSimpleNet s parametrem "Národnost".

Na obrázku 18 je zobrazené doporučení pro uživatele z Německa. Jelikož tito uživatelé byli nejčastěji ubytováni právě v hotelech, jak lze vyčíst z tabulky 4, tak i základní doporučení se skládá z hotelů a konkrétně těch, které měly nejvyšší hodnocení a návštěvnost.

Oproti doporučení, které je založen čistě na hodnocení a četnosti je toho řešení více specifické a do výběru se již nedostane kolej K1, protože taková položka není uživateli z Německa preferována.

Obrázek 18: Doporučení ubytování pro nové uživatele přistupující z Německa

Německo - Jednoduchá bayesova síť - na základě hodnocení uživatelů

1. Clarion Congres Hotel	3.27 %	
2. Hotel Budweis	2.65 %	
3. HC Hotel	1.62 %	

6.4.2 Problém studeného startu položky

Pokud do systému přidáme novou položku, nastává problém s jejím zařazením do výběru doporučených ubytování. Položka se nedostane do výběru, člověk se v takovém místě neubytuje a neohodnotí pro modul doporučení. Takový problém lze řešit různými způsoby. Jedním z nich je přidělení průměrného hodnocení položce hned při zavedení do databáze. Taková položka má šanci se ve výčtu doporučení objevit. Další možností je přidat ji automaticky do tabulky doporučení jako novou položku, která stojí za zhlédnutí. Já jsem ovšem volil možnost nabídnout novou položku takovému uživateli, kterého by mohla zajímat, a ne jí slepě zobrazovat každému.

Toto řešení využívá dvou způsobů. Prvním a tím jednodušším je doporučení položek na základě zvolené úrovně na stránce vyhledávání a doporučení i nových, neohodnocených položek spadajících do určené úrovně, potažmo kategorie. Zde je využito pouze jednoduchého filtru při zvolené úrovni ubytování.

Druhým a sofistikovanějším řešením je využití znalosti o národnosti přistupujícího uživatele. Těmto uživatelům vypočteme nejnavštěvovanější kategorii ubytování dle národnosti.

Pro testování správnosti navrženého modulu pro položku, která ještě nemá žádné ohodnocení, jsem zvolil případ návštěvníka přistupujícího ze Slovenska. Takový údaj, jak již víme, je možno zjistit z IP adresy.

Na základě tabulky 4 můžeme vidět, že uživatelé přistupující ze Slovenska si pro svůj pobyt nejčastěji volí koleje a ubytovny.

Obrázek 19: SQL dotaz z metody getTopRatedCategories

```
ResultSet resultCount = database.createStatement().executeQuery("SELECT COUNT(Hodnoceni.ID) AS total FROM Hodnoceni "
        + "INNER JOIN Uzivatele ON Hodnoceni.ID_Uzivatel = Uzivatele.ID WHERE Uzivatele.Narodnost = '" + narodnost + "'");
while (resultCount.next()) {
    count = resultCount.getInt("total");
}
```

Díky SQL dotazu na obrázku 19 v metodě getTopRatedCategories zjistíme celkový počet hodnocených položek dle zjištěné národnosti.

Obrázek 20: SQL dotaz 2 z metody getTopRatedCategories

```
ResultSet resultCategories = database.createStatement().executeQuery("SELECT ID FROM Kategorie");
HashMap<Integer, Double> relativeFrequencies = new HashMap<>();
while (resultCategories.next()) {
    int id = resultCategories.getInt("ID");

    ResultSet resultRating = database.createStatement().executeQuery("SELECT COUNT(Hodnoceni.ID) AS total FROM Hodnoceni"
            + " INNER JOIN Ubytovani ON Hodnoceni.ID_Ubytovani = Ubytovani.ID INNER JOIN Uzivatele on Hodnoceni.ID_Uzivatel"
            + "= Uzivatele.ID WHERE Ubytovani.ID_Kategorie = " + id + " AND Uzivatele.Narodnost = '" + narodnost + "'");
    while (resultRating.next()) {
        int absoluteFrequency = resultRating.getInt("total");
          System.out.println("ABS: " + absoluteFrequency);
        relativeFrequencies.put(id, (double)absoluteFrequency/count);
    }
}
```

Ve stejné metodě se nachází dotaz pro výpočet četnosti hodnocení každé kategorie s ohledem na národnost. Podle počtu hodnocení jednotlivých kategorií daným národem je rozhodnuto pro doporučení nové neohodnocené položky z kategorie s nejvyšším zastoupením. Právě díky nejvyššímu počtu hodnocených kategorií známe i nejčastější ubytování podle zvoleného národu.

Poslední část programovacího kódu metody getTopCategories se stará o seřazení podle hodnocených kategorií navštívených v tomto případě Slováky. V cyklu je zahrnuta podmínka pro doporučení položky z druhé nejvíce navštěvované kategorie, pokud rozdíl od té první není větší než 5 %. To je právě případ u doporučení nové položky pro Slovenské návštěvníky. Jak lze vidět v tabulce 4, kde právě Slováci navštívili nejvíce ubytovny a koleje, konkrétně v počtu 6 navštívení u obou kategorií.

Obrázek 21: Doporučení ubytování pro nové uživatele přistupující ze Slovenska

Slovensko - Jednoduchá bayesova síť - na základě hodnocení uživatelů

1. Kolej K2	1.78 %	
2. Kolej K4	1.56 %	
3. Ubytovna U nádraží	1.22 %	

Nové doporučené položky

Kolej K5	-	
Ubytovna Zásoby s.r.o.	-	

Na obrázku 21 z webové stránky doporučení vidíme, že pro uživatele přistupující ze Slovenska modul doporučuje právě položky z kategorie koleje a ubytovny. Z trénovacích dat víme, že právě tyto položky jsou nejoblíbenější pro Slovenské občany. Na základě toho jim je doporučena nová položka bez ohodnocení ze stejné kategorie.

7 Závěr

Cílem této bakalářské práce bylo zmapovat doporučovací systémy, jejich různá řešení a na základě takovéto rešerše zvolit vhodnou metodu a navrhnout vlastní doporučovací modul zaměřený na turistickou oblast, který nebude omezen problémem studeného startu a výsledky bude zpracovávat za pomoci Bayesovské sítě.

Na základě teoretické části jsem se rozhodl pro řešení za pomoci Bayesova klasifikátoru – jednoduché Bayesovské sítě, která vychází z předpokladu nezávislosti jevů. Toto řešení je jednoduché na implementaci a není časově tak složité v porovnání s hlubokými Bayesovskými sítěmi, u kterých může složitost dosahovat 2^n. Dále byla zvolena metoda kolaborativního filtrování založená na podobnosti uživatelů. Pokud jsou uživatelé stejné národnosti, věku nebo pohlaví, klasifikujeme je do stejných skupin. K tomu bylo ovšem zapotřebí získat od uživatele potřebné informace. V praktické části byl tento způsob demonstrován primárně na parametru národnost, ale stejně tak funguje i na parametrech věk a pohlaví. Bayesův klasifikátor by v takovém případě jen využíval jiné parametry pro výpočet doporučení.

Studený start byl řešen různými způsoby. Pro uživatele byly zvoleny možnosti: doporučení založené na hodnocení položek, doporučení na základě uživatelské interakce a po rozklíčování oblasti přístupu z IP adresy. Jako nejlepší se jeví řešení pomocí znalosti oblasti přístupu uživatele, protože není zapotřebí zisku informací od uživatele ani případné registrace. Pro studený start položky byly vytvořeny dvě možnosti řešení: zakomponování výběru úrovně na stránce vyhledávání a tudíž doporučení nových položek z preferované kategorie. Druhou možností je doporučení nových položek na základě nejoblíbenější kategorie podle oblasti přístupu. Hlavním východiskem řešení studeného startu je tedy doporučení nejoblíbenějších položek nebo takových, které jsou svázané s nově příchozím uživatelem na základě podobnosti uživatelů.

Při návrhu modulu vznikl problém, kdy již při samotném modelování Bayesovské sítě se závislými jevy přicházelo v úvahu roznásobení parametrů v řádech tisíců. Z tohoto důvodu jsem posléze volil řešení za pomoci jednodušší Bayesovské sítě, která se nazývá Bayesovský klasifikátor. Ten dokáže až překvapivě úspěšně klasifikovat a zařazovat uživatele do podobných kategorií s jemu podobnými uživateli a doporučit tak konkrétní ubytování. Toto bylo ověřeno doporučením na základě národnosti. I při dalším doporučení, jež si uživatel ovlivňuje sám, dochází k úspěšnému zařazení, které zákazník jistě ocení.

Tato práce pro mě byla velkým přínosem, jelikož jsem se seznámil s problematikou doporučovacích systému, která je zvláště v dnešní době velmi aktuální.

Při budoucím rozvoji této práce by bylo jistě zajímavé rozšířit takový modul o více turistických atrakcí a doporučovat návštěvu konkrétních zařízení v mobilní aplikaci na základě souřadnic GPS.

8 Literatura a zdroje

[1] ADOMAVICIUS, Gediminas a Alexander TUZHILIN. Towards the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*. 2005(17(6), 734–749. DOI: https://ieeexplore.ieee.org/document/1423975. Dostupné z: http://pages.stern.nyu.edu/~atuzhili/pdf/TKDE-Paper-as-Printed.pdf

[2] GOLDBERG, David, David NICHOLS, Brian M. OKI a Douglas TERRY. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*. 1992, 35(12), 61-70. DOI: https://dl.acm.org/citation.cfm?doid=138859.138867. ISSN 0001-0782.

[3] PEŠKA, Ladislav. *Uživatelské preference v prostředí prodejních webů*. 2010. Dostupné z: https://is.cuni.cz/webapps/zzp/detail/84549. Diplomová práce. Univerzita Karlova v Praze. Vedoucí práce Prof. RNDr. Peter Vojtáš, DrSC.

[4] VOJTÁŠ, Peter. *Modely uživatelských preferencí*. 2010. Dostupné z: http://www.ksi.mff.cuni.cz/~vojtas/vyuka/NDBI021PrincipyUzivatelskychPreferenci/1112_NSWI021_DotazovaniSPreferencemi/DBI021modelyUzivatele.ppt

[5] EKSTRAND, Michael D., John T. RIEDL a Joseph A. KONSTAN. Collaborative Filtering Recommender Systems. Foundations and Trends® in Human–Computer Interaction [online]. 2011(Vol. 4: no. 2), 81-173 [cit. 2019-01-09]. Dostupné z: http://files.grouplens.org/papers/FnT%20CF%20Recsys%20Survey.pdf

[6] BURKE, Robin. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* [online]. 2002, 12(4), 331-370 [cit. 2019-01-09]. ISSN 1573-1391. Dostupné z: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.8200&rep=rep1&type=pdf

[7] MELVILLE, Prem a Vikas SINDHWANI. Recommender Systems. SAMMUT, Claude a Geoff WEBB, ed. *Encyclopedia of Machine Learning*. Springer. Berlin, 2010, s. 829-838. Dostupné z: http://www.prem-melville.com/publications/recommender-systems-eml2010.pdf

[8] VALA, Martin. E-learning – doporučovací systémy [online]. Brno, 2012 [cit. 2019-01-11]. Dostupné z: https://is.muni.cz/th/359917/fi_b/bp_final_vala.pdf. Bakalářská práce. Masarykova univerzita. Vedoucí práce Mgr. Jan Géryk.

[9] WANG, Shuaiqiang, Jiankai SUN, Byron J. GAO a Jun MA. Adapting vector space model to ranking-based collaborative filtering. Proceedings of the 21st ACM international conference on Information and knowledge management - CIKM '12 [online]. New York, New York, USA: ACM Press, 2012, 2012, 1487-1491 [cit. 2019-01-28]. DOI: 10.1145/2396761.2398458. ISBN 9781450311564. Dostupné z: http://web.cse.ohio-state.edu/~sun.1306/Published_Works/CIKM_12_Adapting_Vector_Space_Model_to_Raanking-based_Collaborative_Filtering.pdf

[10] SHARDANAND, Upendra a Pattie MAES. Social information filtering. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*. New York, New York, USA: ACM Press, 1995, 1995, , 210-217. DOI: 10.1145/223904.223931. ISBN 0201847051. Dostupné z: http://jolomo.net/ringo/chi-95-paper.pdf

[11] PARK, Seung-Taek, David PENNOCK, Omid MADANI, Nathan GOOD a Dennis DECOSTE. Naïve filterbots for robust cold-start recommendations. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06*. New York, New York, USA: ACM Press, 2006, 2006, , 699-705. DOI: 10.1145/1150402.1150490. ISBN 1595933395. Dostupné z: https://www.researchgate.net/publication/221654661_Naive_filterbots_for_robust_cold-start_recommendations

[12] SCHEIN, Andrew I., Alexandrin POPESCUL, Lyle H. UNGAR, David M. PENNOCK a Dennis DECOSTE. Methods and metrics for cold-start recommendations. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '02*. New York, New York, USA: ACM Press, 2002, 2002, , 253-260. DOI: 10.1145/564376.564421. ISBN 1581135610. Dostupné z: https://repository.upenn.edu/cgi/viewcontent.cgi?article=1141&context=cis_papers

[13] BOHÁČ, Zdeněk. *Základy teorie pravděpodobnosti* [online]. In: . [cit. 2019-01-17]. Dostupné z: http://homen.vsb.cz/~boh10/1_Pravdepodobnost.pdf

[14] BERKA, Petr. Bayesovská klasifikace. *Dobývání znalostí z databází*. Praha: Academia, 2003, s. 182-196. ISBN 80-200-1062-9. Dostupné z: https://sorry.vse.cz/~berka/docs/izi456/kap_5.6.pdf

[15] KABASSI, Katerina. Personalizing recommendations for tourists. *Telematics and Informatics* [online]. 2010, **27**(1), 51-66 [cit. 2019-01-22]. DOI: 10.1016/j.tele.2009.05.003. ISSN 07365853. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S073658530900029X

[16] SCHAFER, J. Ben, Joseph KONSTAN a John RIEDI. Recommender systems in e-commerce. *Proceedings of the 1st ACM conference on Electronic commerce - EC '99*. New York, New York, USA: ACM Press, 1999, 158-166. DOI: 10.1145/336992.337035. ISBN 1581131763. Dostupné z: http://portal.acm.org/citation.cfm?doid=336992.337035

[17] KAPOUN, Jan. *Historie Ebay* [online]. [cit. 2019-04-11]. Dostupné z: https://businessworld.cz/ostatni/historie-ebay-5786

[18] *Vítejte u NetBeans a na stránkách www.netbeans.org* [online]. [cit. 2019-04-11]. Dostupné z: https://netbeans.org/index_cs.html

[19] ČÁPKA, David. *Lekce 1 - Úvod do Java Enterprise Edition (JEE)* [online]. [cit. 2019-04-11]. Dostupné z: https://www.itnetwork.cz/java/jee/java-enterprise-edition-uvod-do-jee-j2ee

[20] HANEL, David. *Vývoj webových aplikací pomocí frameworku JavaServer Faces*. 2001. Dostupné také z: https://vskp.vse.cz/20253_vyvoj_webovych_aplikaci_pomoci_frameworku_javaserver_faces. Bakalářská práce. Vysoká škola ekonomická v Praze. Vedoucí práce Ing. Luboš Pavlíček.

[21] *Apache Derby* [online]. [cit. 2019-04-07]. Dostupné z: https://db.apache.org/derby/

[22] MICHALOVIČ, Robert. Lekce 1 - Derby DB - Informace, nastavení prostředí. *www.itnetwork.cz* [online]. [cit. 2019-04-07]. Dostupné z: https://www.itnetwork.cz/java/jdbc/derbydb/derby-db-informace-nastaveni-prostredi

9 Seznam obrázků

10 Seznam tabulek

11 Přílohy

Příloha č. 1

Obsah přiloženého CD

- text bakalářské práce ve formátu PDF,
- projekt webové aplikace v prostředí NetBeans,
- složka s databází a návodem připojení k projektu.