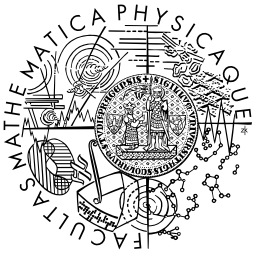


MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Bc. Patrik Machala

HCI modely pro multi-objective doporučovací systémy

Katedra softwarového inženýrství

Vedoucí diplomové práce: Mgr. Ladislav Peška, Ph.D.

Studijní program: Informatika - Softwarové a datové inženýrství

Studijní obor: ISDP

Praha 2024

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ....

Podpis autora

Na tomto místě bych chtěl poděkovat rodině za podporu po celou dobu studia. Dále bych rád poděkoval vedoucímu práce Mgr. Ladislavu Peškovi, Ph.D. za věnovaný čas a cenné rady po celou dobu práce. A také chci poděkovat RNDr. Patriku Dokoupilovi za zpřístupnění a představení implementace doporučovacího systému.

Název práce: HCI modely pro multi-objective doporučovací systémy

Autor: Bc. Patrik Machala

Katedra: Katedra softwarového inženýrství

Vedoucí diplomové práce: Mgr. Ladislav Peška, Ph.D., Katedra softwarového inženýrství

Abstrakt: Jednou z nejvíce rozvíjejících se oblastí information retrieval (vyhledávání informací) jsou doporučovací systémy. Ty se typicky snaží o doporučení několika málo nejrelevantnějších neboli nejvhodnějších položek uživateli ze všech kandidátů, jejichž počet se může pohybovat i v řádu milionů. Ukazuje se ale, že samotná relevance nestačí. Proto se tato práce zaměřuje na multi-objective doporučovací systémy využívající i tzv. beyond-relevance kritéria kvality doporučování. Cílem práce je zjištění nových poznatků o tomto specifickém typu doporučování, a to především v dosud ne příliš prozkoumaném propojení s oblastí interakcí uživatele se systémem.

Softwarovým výstupem práce je webová aplikace a upravený doporučovací systém. Tyto dvě komponenty byly použity v uživatelské studii, kde jsme mimo jiné zkoumali, zda uživatelé stojí o explicitní nastavení parametrů multi-objective doporučovacího systému pomocí přidělení vah ke každému z kritérií, porovnávali různé varianty metrik pro tato kritéria a mechanismů pro nastavení vah, rozdílnou detailnost textů a vizualizace explanations.

Z výsledků našeho experimentu plyne, že uživatelé vnímají přínos nastavení vah pro kritéria kvality doporučování k vylepšení doporučování. Zároveň se nám také podařilo zjistit, že nejvhodnějším mechanismem pro přidělení vah jsou posuvníky neboli slidery. Provedená uživatelská studie také potvrzuje, že uživatelé preferují detailnější vysvětlení doporučení.

Klíčová slova: Doporučovací systémy HCI Multi-objective doporučovací systémy

Title: Human-computer interaction model for multi-objective recommender systems

Author: Bc. Patrik Machala

Department: Department of Software Engineering

Supervisor: Mgr. Ladislav Peška, Ph.D., Department of Software Engineering

Abstract: One of the most developing research fields of information retrieval are recommender systems. They typically try to recommend a few of the most relevant or most suitable items to users from all the candidates when the number of canidates can be in the order of thousands or millions. However, it turns out that relevance alone is not enough. Therefore, this work focuses on multi-objective recommender systems using the beyond-relevance objectives. The aim of the thesis is to find out new knowledge about this specific type of recommendation, especially in the connection with the field of HCI, i.e. user and computer interaction that has not been explored much so far.

The software output of the work is a web application and a modified recommender system. These two components were used in a user study, where, among other things, we investigated whether users were willing to explicitly set the parameters for a multi-objective recommender system by assigning weights to each of the objectives, compared different variants of metrics for these objectives, mechanisms for assigning weights and different level of detail of texts and visualization of the explanations of the recommendations.

The results of our experiment show that users perceive the benefit of setting weights for objectives to improve recommendations. We also managed to find out that the most suitable mechanism for assigning weights are sliders. The conducted user study also confirms that users prefer a more detailed explanation of recommendations.

Keywords: Recommender systems HCI Multi-objective recommender systems

Obsah

Úvod

Oblast doporučovacích systémů se v posledních několika letech těší velké popularitě jak ve vědecké komunitě tak v komerční sféře. V současnosti, kdy obchodníci a poskytovatelé nabízejí obrovské množství položek (řádově od tisíců po miliony) a zároveň bojují o zákazníky či uživatele, je nutné uživatele rychle zaujmout. Právě tento fakt vedl k velkému rozvoji doporučovacích systémů, které dokážou v různých doménách vybrat z tohoto velkého počtu jen jednotky nebo menší desítky položek, kterými chtějí zaujmout, tzn. získat nebo udržet uživatele. Doporučování dnes využívají nejen obrovské portály jako je YouTube nebo Netflix, ale i malé e-shopy.

Historicky byla doporučení vybírána na základě největší odhadované relevance. Relevance se většinou počítala podle podobnosti položky s těmi položkami, které již uživatel ohodnotil / viděl / použil / koupil nebo podle chování jemu nejpodobnějších uživatelů k dané položce. Již téměř 20 let se ukazuje, že prostá maximalizace relevance doporučení nemusí být vždy nejlepší možností z pohledu uživatele. To si můžeme ilustrovat na příkladu, kdy položkami k doporučování jsou filmy a uživatel ohodnotil první 3 filmy ze série Harryho Pottera a cílem je doporučení 5 nejrelevantnějších filmů. V tomto případě by pravděpodobnými doporučeními bylo následujících 5 filmů s Harrym Potterem, což ale pro uživatele není ideální minimálně ze 2 důvodů. Zaprvé je pravděpodobné, že o těchto filmech již ví, takže doporučení pro něj není nové, nebo minimálně není takové, na které by nepřišel sám. Druhým problémem je nabídka zcela stejného typu filmu, a to proto, že uživatel může v tu chvíli hledat film jiného žánru a z doporučení si tak nemusí vybrat.

Proto se do popředí dostávají tzv. beyond-relevance kritéria kvality doporučování, která se zaměřují na jiné cíle než relevance. Mezi ty nejpoužívanější patří diverzita, která pomáhá zvyšovat různorodost seznamu doporučení, a novelty, jež zvýhodňuje položky, které by mohly být pro uživatele nové. Jedním z cílů této práce je zjistit, jaká kritéria, případně jaké jejich metriky jsou pro uživatele zajímavé a pomáhají z jejich pohledu zlepšovat doporučování.

Dalším poznatkem z odborné literatury i praxe je, že uživatelé mají větší důvěru v doporučovací systém, pokud ten dovede podat věrohodné vysvětlení důvodu doporučení jednotlivých objektů či doporučovacího algoritmu a také pokud uživatel může chování algoritmu vhodným způsobem ovlivnit. V tomto případě, kdy se zaměřujeme na multi-objective doporučovací systémy, dostali při uživatelské studii, která je součástí této práce, uživatelé možnost měnit důležitost jednotlivých kritérií pro výběr doporučení. Zároveň je skóre každé doporučené položky vzhledem k metrikám vizualizováno a krátce vysvětleno. Dalším cílem práce tedy je zjištění, zda uživatelé vůbec stojí o možnost nastavení důležitosti kritérií kvality doporučování a zda je pro ně vysvětlení doporučení vycházející z toho, jak si položka vede vzhledem k jednotlivým kritériím, přínosné.

Při návrhu webové aplikace pro provedení uživatelské studie ke splnění výše zmíněných cílů byla vzata v úvahu nedávná práce, z které plyne, že uživatelé se lépe cítí v prostředí, které více připomíná real-world portály. To znamená, že je vhodné, aby uživatelské rozhraní nebylo uzavřené jen na nutné kroky uživatelů, ale více připomínalo rozhraní typické pro portály poskytovatelů / obchodníků.

Dílčím doplňujícím cílem práce je tak zjištění, jestli uživatelé v této doméně ocení možnosti, které nabízí standartní portály, jako je textové vyhledávání, podrobnější filtrování, blokování položek a přizpůsobení samotné aplikace.

1. Doporučovací systémy

V této kapitole jsou popsány doporučovací systémy včetně jejich základních typů. Následně si představíme kritéria kvality doporučování, podle nichž lze vybírat doporučení, a varianty doporučovacích systémů, ve kterých se metriky těchto kritérií kombinují. Následná sekce bude věnována tzv. explanations, tedy vysvětlením doporučení, která mají za cíl zvyšovat uživatelovu spokojenost s těmito systémy. Poté se ještě zaměříme na HCI v oboru doporučování a na závěr kapitoly bude popsán MovieLens dataset použitý v této práci.

1.1 Obecný popis

Doporučovací systémy jsou nástroje pro interakci uživatele s velkými a komplexními informačními prostory. Poskytují personalizovaný vhled v těchto prostorech pomocí upřednostnění položek, které by mohly být pro uživatele zajímavé (Burke a kol., 2011).

Tyto systémy využívají různé typy vstupů uživatele získané z jeho historie chování. Ty mohou být explicitní a implicitní. Typickým příkladem explicitního vstupu je hodnocení položky na různých škálách od unární (líbí) přes binární (líbí nebo nelíbí) po číselnou (1 - 10). Implicitní vstup může být například kliknutí na položku a zobrazení jejího detailu. Doporučovací systémy mohou využívat i mnoho dalších akcí jako jsou zobrazení položky uživateli, jeho historie vyhledávání apod., obecné vlastnosti uživatele či položky nebo současný kontext, což může být například roční období.

Na základě těchto vstupů je cílem doporučovacího systému vybrat z velkého množství malý počet položek, které jsou zobrazeny uživateli. Formálně můžeme problém doporučování popsat následovně: Mějme množinu U všech uživatelů a I množinu všech položek k doporučení, což mohou být například knihy, filmy, restaurace atd. Velikost množiny I může být obrovská, řádově se pohybuje od stovek, přes tisíce, až po miliony položek. Zároveň počet uživatelů $|U|$ může být také velký, v některých případech řádově až v milionech. Mějme hodnotící funkci d, která měří užitečnost položky i pro uživatele u, tj. $d : U \times I \rightarrow R$, kde R je lineární uspořádání. Potom pro každého uživatele $u \in U$, chceme vybrat takovou položku $p' \in P$, která maximalizuje užitečnost položky pro uživatele (Adomavicius a Tuzhilin, 2005). Formálněji:

$$\forall u \in U, i'_u = \arg\max_{i \in I} d(u,i)$$

Dříve se odlišovaly doporučovací systémy od systémů pro vyhledávání informací (information retrieval systems) tím, že doporučovací systém je personalizovaný, tzn. že různí uživatelé dostávají různé výsledky (Burke a kol., 2011). To už dnes, kdy i část systémů pro vyhledávání informací funguje personalizovaně, neplatí. Zásadním rozdílem ale zůstává, že doporučovací systémy na rozdíl od těch vyhledávacích fungují bez explicitního uživatelova dotazu.

Způsob získání doporučení může být u kompikovanějších systémů navíc rozdělen do 4 fází: retrieval, filtrování, ohodnocení a seřazení (Higley a kol., 2022). Retrieval se používá hlavně u domén s obrovským počtem položek, kde je potřeba většinu kandidátů rychle vyloučit a k samotnému ohodnocení předat menší

počet variant řádově ve stovkách až tisících. To je důležité pro snížení vysoké časové náročnosti především fáze ohodnocení závislé většinou zhruba lineárně na počtu objektů. V další části filtrování dochází k odebrání nevhodných kandidátů, tedy takových, kteří nesplňují všechny podmínky, aby mohly být doporučeny. Takovými podmínkami můžou být specifická byznysová pravidla nebo i něco jednoduššího, jako je vyřazení vyprodaných produktů. Fáze ohodnocení probíhá na základě funkce d popsané v předchozím odstavci. Posledním krokem je seřazení, do kterého mimo výstupu předchozí fáze mohou vstupovat další faktory, jako jsou bussiness cíle apod. Do této poslední části patří i zapracování beyond-relevance kritérií, což jsou diverzita, novelty a mnohé další.

Nejčastěji se používají dva typy doporučování. První možností je kolaborativní filtrování. Tato metoda ohodnocuje položky podle toho, jak k nim přistupují uživatelé, kteří se chovají podobně jako uživatel, kterému je doporučováno. Vstupem pro algoritmy tohoto typu je často matice hodnocení, případně jiného typu interakce mezi uživateli a položkami. Druhým způsobem je content-based filtrování. Tato metoda vybírá doporučení na základě předchozích akcí (hodnocení, interakce, koupě, ...) uživatele, kdy se snaží vybrat takové objekty, které nejvíce odpovídají preferencím uživatele. Preference uživatele jsou získány z vlastností položek v jeho profilu, tzn. těch, které hodnotil, koupil apod. Na rozdíl od kolaborativního filtrování tak nevyžaduje content-based doporučování akce ostatních uživatelů. Protože oba způsoby mají své výhody i nevýhody, jsou na vzestupu hybridní doporučovací systémy, které obě tyto metody kombinují.

1.2 EASE

Jedním z algoritmů fungujích na bázi kolaborativního filtrování je EASE (Steck, 2019). Ten funguje na následujícím principu. Na vstupu má matici interakcí $X \in \mathbb{R}^{|U|\times|I|}$, kde $|U|$ je velikost množiny uživatelů a $|I|$ velikost množiny položek. Pokud mezi uživatelem a položkou proběhla interakce, je v základní verzi v matici na daném místě 1, v opačném případě 0. Interakcí může být například koupě nebo pozitivní hodnocení položky. V případě víceškálové zpětné vazby, můžou hodnoty v matici X nabírat i jiné hodnoty. Cílem algoritmu je vypočítat matici vah $B \in R^{|I|\times|I|}$ pomocí optimalizační úlohy:

$$\min_B ||X - XB||_F^2 + \lambda \times ||B||_F^2$$

$$za\ podmínky\ diag(B) = 0$$

kde $||\cdot||_F$ značí Frobeniovu normu a λ je jediný parametr L2-norm regularizace a zároveň jediný hyperparametr celého modelu. Podmínka $diag(B) = 0$ zamezuje triviálnímu řešení $B = \mathbb{I}$, kde $\mathbb{I}$ je jednotková matice.

Po vypočtení B se odhadovaná relevance položky i pro uživatele u vypočítá následovně:

$$R_{u,i} = X_{u,\cdot} \times B_{\cdot,i}$$

Díky použití kvadratické chybové funkce má EASE dokonce i analytické řešení pro nalezení matice B (viz alg. 1), což výrazně zrychluje natrénování modelu.

Výhodou tohoto modelu je fakt, že je jednoduchý a po natrénování dokáže také rychle vypočítat odhad relevance položky pro uživatele na základě vynásobení 2

Algorithm 1 Analytické řešení nalezení matice B (Steck, 2019)

Vstup: matice interakcí $X \in \mathbb{R}^{|U|\times|I|}$, parametr L2-norm regularizace λ
Výstup: matice vah mezi položkami B
$G = X^T X$
$G[diag] += \lambda$
$P = G^{-1}$
$B = P \,/\, (-diag(P))$
$B[diag] = 0$

vektorů. Zároveň je tento algoritmus konkurenceschopný i v porovnání s mnohem složitějšími modely a dokonce v některých doménách dosahuje state-of-the-art výsledků.

1.3 Kritéria kvality doporučování

V původních doporučovacích systémech byla jediným zvažovaným kritériem relevance. Ukazuje se, že jen samotná relevance nestačí (McNee a kol., 2006) a do popředí se dostávají i další kritéria, jejichž kombinace zvyšuje spokojenost uživatele s doporučováním (Herlocker a kol., 2004). V této kapitole si představíme všechna kritéria a varianty jejich metrik vyskytující se v této práci. Zdaleka ale nejde o všechna používaná v oboru doporučovacích systémů.

1.3.1 Relevance

Jak je zmíněno výše, relevance je základním kritériem, podle kterého se vybírají doporučení pro uživatele. Pomocí relevance se snaží doporučovací systémy najít takové položky, které odpovídají uživatelovým zájmům nebo potřebám.

Relevanci si lze představit jako funkci d, která udává užitečnost položky i pro uživatele u, tj. $d : U \times I \to R$, kde U je množina všech uživatelů a I množina všech objektů k doporučení a R je lineární uspořádání. Funkce d je ovšem neznámá, proto je typicky cílem doporučovacích algoritmů, které se zaměřují pouze na relevanci, co nejpřesnější odhad funkce d. Výstupem algoritmu je tedy pouze odhadovaná relevance vypočítaná na základě funkce d'. Čím lépe d' aproximuje d, tím relevantnější doporučení uživatel dostává.

Odhadovaná relevance je odvislá od každého doporučovacího modelu. Funkce d' může být v zásadě cokoli od vynásobení 2 vektorů odpovídající uživateli a položce až po výstup komplikované neuronové sítě. Relevance může být spočtena na základě chování podobných uživatelů, na podobnosti položky s položkami, s nimiž uživatel již nějakým způsobem interagoval, dále například na základě jejich vlastností nebo jen těch položek, které uživatel prohlédl v posledních několika minutách.

1.3.1.1 Metriky

Pokud měříme relevanci, měříme, jak moc odpovídá odhadovaná relevance té skutečné. Metriky měřící relevanci doporučení můžeme rozdělit na 3 základní

typy (Alhijawiová a kol., 2022), do kterých patří vždy mnoho konkrétních metrik, proto zmíníme jen ty nejzákladnější.

Tím prvním jsou metriky přesnosti predikce hodnocení. Ty jsou vhodné pouze, pokud výstupem doporučovacího modelu je predikce implicitní nebo explicitní zpětné vazby uživatele, tedy typicky hodnocení. V tomto případě nás zajímá rozdíl mezi odhadovanými hodnoceními a těmi skutečnými. Příkladem takových metrik jsou jsou MAE (mean absolute error) a RMSE (root mean squared error), tedy střední absolutní chyba a odmocnina střední kvadratické chyby.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{r}_i - r_i|$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{r}_i - r_i)^2}$$

kde $\hat{r}_i$ je predikované a r_i skutečné hodnocení položky i. Nevýhodou těchto metrik je fakt, že cílem doporučovacího systému obvykle není co nejpřesnější predikce zpětné vazby uživatele, ale seřazení položek a na základě něj doporučení těch nejvíce relevantních.

Jiným typem metrik jsou ty měřící relevanci na celém seznamu doporučení. Tyto metriky jsou závislé na velikosti seznamu doporučení, počtu relevantních položek a počtu relevantních položek v seznamu doporučení. Relevantní položku interpretujeme jako tu, na níž od uživatele dostaneme pozitivní zpětnou vazbu například v podobě pozitivního hodnocení. Typickou variantou těchto metrik jsou Precision, která měří podíl počtu relevantních položek v seznamu doporučení a velikosti tohoto seznamu, a Recall, která měří podíl počtu relevantních položek v seznamu doporučení a celkového počtu relevantních položek, které jsme mohli doporučit.

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

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

kde TP jsou relevantní doporučení, FP jsou nerelevantní doporučení, TN nedoporučené relevantní položky a FN nedoporučené nerelevantní položky. Tyto metriky již více odpovídají typickému účelu doporučovacího systému. Jejich nevýhodou ovšem je, že neberou v potaz pořadí v seznamu doporučení.

Tento problém řeší metriky relevance závislé na pořadí. Ty dávají větší důraz na měřené relevance u prvních položek v seznamu doporučení. To je rozumný přístup z toho důvodu, že uživatel většinou prochází položky postupně a k těm na konci seznamu doporučení se nemusí vůbec dostat. Některé metriky navazují na ty předchozího typu pomocí ořezání seznamu doporučení a zaměření se pouze na několik prvních položek v seznamu. Takovými metrikami jsou například Precision@k nebo Recall@k. Z té první vychází také MAP (mean average precision), která se počítá přes všechny uživatele počítá takto:

$$MAP = \frac{1}{|U|}\sum_{u\in U}\frac{\sum_{i=1}^{N} Precision@i \times 1_{I_u^+}(i_u^*)}{|I_u^+|}$$

kde U je množina uživatelů, N velikost seznamu doporučování a $1_{I_u^+}$ je charakteristická funkce, která je rovna 1, pokud i-tá položka v seznamu doporučení je relevantní pro uživatele (tzn. patřící do množiny I_u^+), a 0 jinak. Oblíbenou metrikou je DCG (discounted cumulative gain), kterou počítáme takto:

$$DCG = \frac{\sum_{i=1}^{N} r_i^u}{\log_2(i+1)}$$

kde r_i^u je skutečná relevance i-té položky v seznamu doporučení pro uživatele u. Z této metriky vychází ještě její normalizovaná varianta nDCG (normalized discounted cumulative gain), která je definována takto:

$$nDCG = \frac{DCG}{IDCG}$$

kde $IDCG$ je DCG stejného seznamu doporučení ovšem s ideálním seřazením položek sestupně dle r_i^u.

1.3.2 Diverzita

Obecně v oblasti information retrieval (vyhledávání / získávání informací) diverzita zajišťuje různorodost výsledku. Přímo pro doporučovací systémy to znamená, že s vyšší diverzitou by se měla snižovat vzájemná podobnost položek v seznamu doporučení. Podobnost položek může být závislá na jejich vlastnostech (např. žánry u filmů, hudby a knih), nebo kolaborativní, kde je vyšší, pokud je dvojice objektů podobně hodnocena velkou skupinou uživatelů. Diverzita může být počítána jako vlastnost celého seznamu doporučení, nicméně vzhledem ke způsobu použití v této práci je zvolen způsob, kdy je počítána pro každou položku zvlášť.

Nutnost diverzifikace výsledků doporučování byla poprvé popsána v roce 2001 (Bradley, 2001). Protože je pohled na to, co je to diverzita, subjektivní, vzniklo mnoho různých metrik, kde je diverzita v doporučování měřena různými způsoby (Kunaver a Porl, 2017). Představíme si pouze varianty, které byly použity pro výzkum v této práci.

1.3.2.1 Intra-list diverzita

Intra-list diverzita (diverzita uvnitř seznamu) je původní varianta, jak počítat diverzitu v oblasti doporučování (Bradley, 2001). Jedná se také o nejčastěji používánou variantu diverzity (Kunaver a Porl, 2017). Hodnota intra-list diverzity roste, pokud je položka v průměru méně podobná ostatním položkám v seznamu doporučení.

Formálněji je intra-list diverzita položky i vzhledem k seznamu doporučení vypočítána takto:

$$div(i|R) = \frac{1}{|R|} \sum_{r_j \in R} (1 - SIM(r_j, i))$$

kde $i \in I$ je položka z množiny těch k doporučení, R je množina položek, které již jsou součástí doporučení, a SIM je funkce počítající podobnost 2 položek.

Intra-list diverzitu můžeme upravit tak, abychom získali jinou metriku diverzity, kterou můžeme použít pro multi-objective doporučování. Zejména se lze zaměřit na nahrazení průměru jinou agregační funkcí.

1.3.2.2 Diverzita na základě maximální podobnosti

Beyond-relevance kritéria byla ještě před doporučovacími systémy zkoumána v oblasti information retrieval. Zde uživatel vyhledává např. na webu pomocí dotazu a algoritmus mu vrací možné výsledky, jako jsou dokumenty. Carbonell a Goldsteinová (1998) zadefinovali maximální marginální relevanci, kdy skóre každého možného výsledku vyhledávání je počítáno jako rozdíl relevance a podobnosti. Jelikož se pohybujeme v oblasti vyhledávání na webu, je skóre relevance spočteno jako podobnost dokumentu s uživatelovým dotazem. Zajímavější je to, jak autoři navrhují počítat diverzitu, kde se snaží, aby dokument byl co nejvíc odlišný od jakéhokoli jiného dokumentu, který bude součástí vráceného seznamu výsledků. To znamená, že pokud mám k dispozici hodnotu podobnosti všech dvojic dokumentů, tak by pro maximalizaci diverzity měl být vybrán ten dokument, který má nejnižší maximum podobnosti s ostatními dokumenty, které budou součástí odpovědi na uživatelův dotaz.

Formálně se maximální marginální relevance počítá takto:

$$MMR = arg \max_{D_i \in R \setminus S} [\lambda \, (Sim_1(D_i, Q) \; - \; (1 \; - \; \lambda) \max_{D_j \in S} Sim_2(D_i, D_j))]$$

kde Q je dotaz uživatele, R je seřazený seznam dokumentů získaný z information retrieval systému, $S \subset R$ je množina již vybraných dokumentů, Sim_1 je podobnostní metrika mezi dotazem a dokumentem a Sim_2 je podobnostní metrika dvojicí dokumentů.

Nyní z definice maximální marginální relevance můžeme získat metriku fungující na principu intra-list diverzity. Jak je zřejmé, pro získání metriky pro diverzitu samotnou potřebujeme oddělit 2 části argmax funkce a zachovat pouze tu druhou. Pro naše potřeby je navíc nutná úprava, aby tato metrika odpovídala oboru doporučovacích systémů. Na základě tohoto postupu získáme novou variantu diverzity, kterou nazveme diverzitou na základě maximální podobnosti.

Formálně je diverzita na základě maximální podobnosti pro položku i spočtena takto:

$$div(i|R) \; = \; 1 \; - \; \max_{r_j \in R} SIM(r_j, i)$$

kde $i \in I$ je položka z množiny všech možných k doporučení, R je množina položek, které již jsou součástí doporučení, a SIM je funkce podobnosti 2 položek. Rozdíl oproti intra-list diverzitě je tedy v tom, že se nepoužívá průměrná podobnost přes položky v seznamu doporučení, ale maximální podobnost.

1.3.2.3 Binomická diverzita

Binomická diverzita je metrika, která pracuje s žánry (Vargas a kol., 2014). Přestože není složité si představit, co by měla zajišťovat diverzita doporučení vzhledem k žánrům, definice takové metriky už není tak jednoznačná.

Autoři této varianty používají k výpočtu několik vlastností, jaké může mít seznam doporučení s ohledem na žánry. Tou první je coverage, tedy pokrytí žánry. Použití této vlastnosti zaručuje, aby co nejvíce žánrů bylo obsaženo v seznamu doporučení. Druhou vlastností vstupující do výpočtu je redundance, jejíž minimalizace přispívá k tomu, aby co nejméně docházelo k opakování žánrů. Třetím aspektem je velikost seznamu doporučení, který musí obě předchozí vlastnosti ve

výpočtu brát v potaz. Je těžší pokrýt všechny žánry seznamem obsahujícím 3 položky než seznamem obsahujícím položek 50.

Zároveň je také nutné vzít v úvahu počet položek patřících k jednotlivým žánrům, což můžeme ilustrovat na našem příkladu doporučování filmů z posledních 30 let. Je vhodné, aby bylo důležitější pokrýt žánr akčního filmu, než žánr westernu, když počet westernů je více než dvacetinásobně menší než počet akčních filmů. To souvisí i s faktem, že u dostatečně velkého seznamu nejde redundanci zabránit a více penalizována by měla být redundance méně zastoupených žánrů. Tento problém je řešen využitím binomického rozdělení, což je diskrétní rozdělení pravděpodobnosti popisující četnost výskytu úspěchu v N pokusech, které jsou nezávislé a v nichž se pravděpodobnost úspěchu nemění. Náhodná veličina $X \sim B(N, p)$ tohoto rozdělení má pravděpodobnostní funkci:

$$P(X = k) = \binom{N}{k} p^k (1 - p)^{N-k}$$

kde N je počet pokusů a p pravděpodobnost úspěchu.

Pro položku i a množinu žánrů *G(i)* patřící k položce i uvažujeme Bernoulliho experiment, zda žánr g patří do *G(i)*. Pro množinu položek I označíme počet položek patřících do daného žánru jako počet úspěchů $k_g^I = |\{i \in I : g \in G(i)\}|$. Pro seznam doporučení R velikosti N použijeme pravděpodobnost p_g pro výpočet toho, jak adekvátní je počet k_g^R položek pokrývající žánr g. K výpočtu p_g je použita personalizovaná pravděpodobnost žánrů na základě položek, které již uživatel ohodnotil, nebo s nimi jinak interagoval ($p''_g = \frac{k_g^{I_u}}{|I_u|}$). Dalším faktorem je globální pravděpodobnost mezi profily všech uživatelů ($p'_g = \frac{\sum_u k_g^{I_u}}{\sum_u |I_u|}$). Výsledná pravděpodobnost žánru u položky relevantní pro uživatele p_g je závislá na parametru α, který kombinuje personalizovanou a globální pravděpodobnost žánru:

$$p_g = (1 - \alpha)\, p'_g + + \alpha\, p''_g$$

Dále definujeme, že R je seznam doporučení, G množina všech žánrů, $G(R) \subseteq G$ množina žánrů, k němuž patří alespoň jedna položka z R, a X_g je náhodná veličina s binomickým rozdělení $Bi(|R|,p_g)$.

Nyní můžeme zadefinovat výpočet Coverage, tedy pokrytí:

$$Coverage(R) = \prod_{g \notin G(R)} P(X_g = 0)^{1/|G|}.$$

K výpočtu redundance je nutné nejdříve zadefinovat pravděpodobnost toho, že žánr je v seznamu doporučení pokryt k nebo více položkami za předpokladu, že je v seznamu doporučení pokryt aspoň 1 položkou:

$$P(\, X_g \geq k \,|\, X_g > 0\,) = 1 - \sum_{l=1}^{k-1} P(X_g = l \,|X_g > 0).$$

Non-redundancy, tedy opak redundance zadefinujeme následovně:

$$NonRed(R) = \prod_{g \in G(R)} P(X_g \geq k_g^R \,|\, X_g > 0)^{1/|G(R)|}.$$

Binomická diverzita seznamu doporučení R je tedy rovna:

$$BinDiv(R) = Coverage(R) \times NonRed(R)$$

1.3.3 Novelty

V oblasti information retrieval je novelty (novost) často chápána způsobem, jakým v oblasti doporučovacích systémů rozumíme diverzitě (Clarke a kol., 2008). Zatímco diverzita se snaží o co největší různorodost položek uvnitř seznamu doporučení, novelty měří rozdílnost položky oproti tomu, co již bylo viděno (případně ohodnoceno, koupeno,..), a to jak z pohledu uživatele, kterému doporučuji, tak z toho globálního, kdy jsou vzati v úvahu všichni uživatelé (Vargas, 2014). Stejně jako u diverzity si představíme pouze ty konkrétní varianty novelty, jež jsou použity pro výzkum v této práci.

I na novelty existuje řada různých pohledů a na ně navázaných metrik. Varianty novelty můžeme obecně rozdělit na popularity-based novelty pracující s odhadovanou globální známostí položky a distance-based novelty pracující s odlišností od uživatelova profilu.

U popularity-based variant je novelty závislá na pravděpodobnosti, že položka je pro uživatele neznámá. Fomálněji je pro položku i počítáná takto:

$$nov(i \mid \theta) \;=\; 1 \;-\; p(known \mid i, \theta)$$

kde θ je kontext uživatele, p pravděpodobnostní funkce a $known$ jev označující, že uživatel již položku zná.

Druhým typem je distance-based novelty, u které je novelty závislá na podobnosti položky s již známými položkami, což jsou většinou položky, které uživatel již viděl / hodnotil / pořídíl. Formálněji je pro položku i počítána jako:

$$nov(i \mid \theta) \;=\; f(\{d(i,j) \mid j \;\in\; \theta\})$$

kde $d(i,j) = 1 - SIM(i,j)$, kde SIM je funkce podobnosti 2 položek, θ je kontext uživatele a f může být jakákoli agregační funkce.

Nyní už představíme konkrétn9 metriky novelty. Začneme očekávaným doplňkem popularity spadající pod popularity-based varianty a následně popíšeme i dvě metriky řadící se mezi varianty distance-based novelty.

1.3.3.1 Očekávaný doplněk popularity

Jelikož je ve většině domén obvyklé, že velká část interakcí uživatelů probíhá s malým procentem nejpopulárnějších položek, je novelty také často používána v e-komerci jako prostředek k doporučení tzv. long-tail položek, což jsou ty málo pořizované (Vargas a kol., 2014).

Očekávaný doplněk popularity jako varianta popularity-based novelty používá globální kontext všech uživatelů (Vargas, 2014) a novelty položky i je počítána následovně:

$$nov(i \mid \theta) \;=\; 1 \;-\; \frac{|\{\, u \;\in\; U \;:\; K(u,i)\}|}{|U|}$$

kde U množina všech uživatelů, θ profil uživatele a K funkce, která říká, zda uživatel interagoval s položkou, což může být jakákoli akce, jako je zobrazení, nákup, hodnocení, přehrání či jiná interakce.

1.3.3.2 Distance-based novelty na základě maximální podobnosti

Tato varianta distance-based novelty vychází z definice diverzity na základě maximální podobnosti (viz kapitola 1.3.2.2). Vzhledem k tomu, že cílem je měření novelty položky, nedochází k porovnání s dalšími položkami v samotném seznamu doporučení, ale s těmi, které již uživatel ohodnotil (případně s nimi jinak interagoval).

Formálně je tedy distance-based novelty na základě maximální podobnosti položky i definována následujícím způsobem:

$$nov(i \mid K) \;=\; 1 \;-\; \max_{k_j \in K} SIM(k_j, i)$$

kde $i \in I$ je položka z množiny kandidátů k doporučení, K je množina položek, které již uživatel hodnotil (případně s nimi jinak interagoval) a *SIM* je funkce podobnosti 2 položek.

1.3.3.3 Intra-list distance-based novelty

I tato varianta distance-based novelty vychází z definice podobně nazvané varianty diverzity (viz kapitola 1.3.2.1). I zde vzhledem k tomu, že cílem je měření novelty položky, nedochází k porovnání s dalšími položkami v samotném seznamu doporučení, ale s těmi, které již uživatel ohodnotil (případně s nimi jinak interagoval).

Formálněji je intra-list distance-based novelty položky i spočtena takto:

$$nov(i \mid K) \;=\; \frac{1}{|K|} \sum_{k_j \in K} (1 \;-\; SIM(k_j, i))$$

kde $i \in I$ je položka z množiny těch k doporučení, K je množina položek, které již uživatel hodnotil (případně s nimi jinak interagoval) a *SIM* je funkce podobnosti 2 položek.

1.3.4 Popularita

Většina doporučovacích algoritmů je k popularitě pozitivně předpojatá. Populárnější položky jsou tak častěji doporučovány, proti čemuž se v některých doménách bojuje (Cañamares a Castells, 2018). Příkladem omezení popularity je právě zvažování výše zmíněné novelty. Na druhou stranu, populární položky mohou jednak uživatele přilákat a následně i zvyšovat důvěru uživatele v doporučovací systém.

1.3.4.1 Popularita dle známosti

Nejpřímější variantou popularity je známost položky. To znamená, že popularita položky je měřena podle počtu uživatelů, kteří ji znají. Jde o inverzní metriku k variantě novelty očekávánému doplňku popularity (viz kapitola 1.3.3.1). Formálněji je tato varianta pro položku i spočtena následovně:

$$pop(i \mid K) \;=\; \frac{|\{\, u \;\in\; U \;:\; K(u,i)\}|}{|U|}$$

kde U je množina všech uživatelů a K funkce, která říká, zda uživatel interagoval s položkou.

Existují i varianty zaměřující se na aktuální popularitu. To lze řešit použitím pouze těch interakcí $K(u,i)$ proběhlých v nedávné době (dnes, v tomto měsící, za poslední rok, ...). Další možností je přiřadit ke každé interakci $K(u,i)$ váhu $w_{K(u,i)}$ závislou na stáří interakce (čím novější tím větší). Místo počtu uživatelů $|\{\, u \in U : K(u,i)\}|$, je tak popularita odvislá od součtu vah $\sum_{u \in U : K(u,i)} w_{K(u,i)}$.

1.3.4.2 Popularita na základě hodnocení

Pro některé domény se však zdá být lepším ukazatelem popularity tak, jak ji vnímají uživatelé, průměrné hodnocení než celkový počet uživatelů s interakcí (hodnocení, nákup, ...) (Cañamares a Castells, 2018). Proto definujeme i druhou variantu popularity, která je použitelná pouze v doménách s explicitní zpětnou vazbou od uživatele, která má alespoň 2 škály (líbí - nelíbí), případně s alespoň více typy interakcí s různou důležitostí (rozkliknul, vložil do košíku, koupil). Ta je pro položku i počítána takto:

$$pop(i \mid R) \;=\; \frac{\sum_{\{u \in U \,:\, R(u,i)\; existuje\;\}} R(u,i)}{|\{u \;\in\; U \;:\; R(u,i)\; existuje\;\}|}$$

kde U je množina všech uživatelů a $R(u,i)$ hodnocení uživatele u položky i.

U této varianty tak, jak jsme ji zadefinovali, je nutné počítat pouze s položkami s dostatečnou zpětnou vazbou, abychom zabránili tomu, že mezi nejpopulárnějšími položkami budou ty s velmi malým počtem hodnocení. Pokud jsou v doméně položky s malým počtem hodnocení, je nutné jejich popularitu nastavit na nízkou hodnotu nebo počítat jiným způsobem, kde je vzata v úvahu penalizace za nedostatek uživatelů, kteří na položku vyjádřili názor.

1.3.5 Kalibrace

Kalibraci můžeme ilustrovat na případě z domény filmů. Pokud uživatel projevil zájem o 6 akčních a 4 romantické filmy, je časté, že klasický doporučovací systém pracující pouze s relevancí uživateli doporučí téměř pouze akční filmy a další žánry z jeho profilu nejsou v seznamu doporučení dostatečně nebo vůbec reprezentovány (Steck, 2018). Pro kalibrovaná doporučení by naopak mělo platit, že se seznam doporučených co nejvíce blíží rozdělení na 60 % akčních a 40 % romantických filmů. Hodnota kalibrace nám tedy říká, jak moc je seznam doporučení v souladu s odpozorovanými preferencemi uživatele.

Zásadní zmíněnou vlastností kalibrovaného doporučování tedy je, že napomáhá k tomu, aby méně zastoupené zájmy uživatele nebyly zcela vytěsněny těmi více zastoupenými. Tím částečně řeší problém tzv. filter bubbles, kdy je uživatel uzavřen jen v malé části domény. Neřeší ji ale zcela vzhledem k tomu, že použití kalibrace nevede k objevení nových okruhů (žánrů, témat, ...) domény, se kterými uživatel dosud neinteragoval, což může vést u uživatele s poměrně homogenní historií interakcí k opačnému efektu (Steck, 2018).

Výpočet kalibrace si formálně zadefinujeme tak, jak jej představil Steck (2018). Pro číselné vyjádření kalibrace musíme nejdřív určit dvě distribuce žánrů. První

je distribuce žánrů $p(g|u)$ v profilu uživatele u, tu počítáme následovně:

$$p(g|u) \;=\; \frac{\sum_{i \in R}\; w_{u,i} \;\times\; p(g|i)}{\sum_{i \in R}\; w_{u,i}}$$

kde R je množina položek, které uživatel hodnotil (případně s nimi jinak interagoval), $w_{u,i}$ je váha interakce (vázaná např. na její stáří) a hodnotu $p(g|i)$ předpokládáme danou, v jednoduchém případě 1, pokud položka patří do žánru, a 0, pokud ne. Jestliže chceme zařídit, aby kalibrace umožňovala i žánry chybějící v profilu uživatele, musíme $p(g|u)$ nahradit $p'(g|u)$. Tuto hodnotu vypočítáme takto:

$$p'(g|u) \;=\; \beta \;\times\; p_0(g) \;+\; (1 \;-\; \beta) \;\times\; p(g|u)$$

kde $p_0(g)$ je uniformní distribuce žánrů, nebo průměrná distribuce přes všechny uživatele a β je parametr určující vliv $p_0(g)$ na hodnotu $p'(g|u)$.

Druhou je distribuce žánrů $q(g|u)$ mezi doporučenými položkami, ta je spočtena takto:

$$q(g|u) \;=\; \frac{\sum_{i \in Rec}\; w_{rank(i)} \;\times\; (p(g|i)}{\sum_{i \in Rec}\; w_{rank(i)}}$$

kde Rec je množina doporučených položek a $w_{rank(i)}$ váha položky odvislá od pořadí v seznamu doporučení.

Pro výpočet kalibrace použijeme Kullback-Leiberovu divergenci, která se počítá následovně:

$$KL(p,q) \;=\; \sum_{g \in G(R)} p(g|u)\; \log \frac{p(g|u)}{q(g|u)}$$

kde R je seznam doporučení, a $G(R)$ množina žánrů, k němuž patří alespoň jedna položka z R.

Protože $KL(p,q)$ diverguje, pokud $q(g|u) \;=\; 0$, použijeme místo ní

$$\tilde{q}(g|u) \;=\; (1 \;-\; \alpha) \;\times\; q(g|u) \;+\; \alpha \;\times\; p(g|u).$$

Při použití malé hodnoty α platí $\tilde{q} \;\approx\; q$.

Metriku kalibrace definujeme pomocí Kullback-Leiberovi divergence takto:

$$C_{KL}(p,q|Rec) \;=\; KL(p,\tilde{q})$$

kde Rec je množina doporučených položek. Platí, že $C_{KL}(p,q) \;=\; 0$ v případě perfektní kalibrace a menší hodnoty značí lépe kalibrovaná doporučení.

1.4 Multi-objective doporučovací systémy

V mnoha aspektech běžného života jsou situace, kdy se snažíme nalézt kompromis mezi několika kritérii. Příkladem může být například koupě elektroniky, od které vyžadujeme několik vlastností, ale s omezeným rozpočtem pro nákup si nemůžeme pořídit věc, která splňuje všechna kritéria. V oblasti doporučovacích systémů multi-objective problém (problém více kritérií) nastává, pokud chceme optimalizovat více než jedno kritérium kvality doporučování. Každé takové kritérium, resp. jeho metrika, se podílí na výběru doporučení a zvyšování jedné často vede ke snižování metriky jiné, což lze ilustrovat na případu relevance a diverzity (Zheng a kol., 2023).

Problém multi-objective optimalizace (vícekriteriální optimalizaci) definujeme jako:

$$\min_x(f_1(x), f_2(x), f_3(x), ..., f_M(x))$$

za podmínek

$$g_j(x) \geq 0,\ j = 1,2,...,J$$
$$h_k(x) = 0,\ k = 1,2,...,K$$
$$x_i^L \leq x_i \leq x_i^U$$

kde $x \in R^N$ je proměnná s N složkami, $f_i(x)$ je funkce i-tého kritéria, tři následující podmínky definují možné hodnoty x. J udává počet funkcí g_j, pro které musí platit $g_j(x) \geq 0$. K udává počet funkcí h_k, pro které musí platit $h_k(x) = 0$. Poslední typ podmínky určuje minimální a maximální hodnotu každé složky vektoru x (Zheng a kol., 2023).

Multi-objective optimalizaci můžeme převést do oboru doporučovacích systémů. Proměnná x představuje seznam doporučení, případně jednu doporučovanou položku. Funkce f_i odpovídají i-té metrice (např. popularita dle známosti, intra-list diverzita). Podmínky ohraničující x umožňují nabírání hodnot všech možných seznamů doporučení, případně všech kandidátů k doporučení.

K řešení problému více kritérií můžeme použít různé přístupy. Nejčastějším způsobem je přeskládání původního pořadí určeného dle relevance na základě dalších kritérií. Jinými přístupy jsou metody na základě optimalizace s omezujícími podmínkami, přístupy založené na grafech, nejbližších sousedech, nebo vícerukých banditech, dále ty založené na konceptech Paretova optima a mnohé další (Jannach, 2022).

V různých doménách se objevují i jiné typy cílů, které musíme vyvažovat, než jsou samotné metriky kvality doporučování (Jannach, 2022). Jedním z nich jsou systémy s více zainteresovanými stranami, tzv. multi-stakeholder doporučovací systémy. Příkladem takového systému je jakákoli platforma e-komerce, kde uživatelovým zájmem je, aby mu byla doporučena co nejrelevantnější položka, zatímco obchodník může mít alespoň částečně protikladné byznysové cíle v podobě prodeje méně prodávané či dražší položky. Zainteresovaných stran může být i více než dvě, kdy typickým příkladem je srovnávač produktů, kde uživatel chce doporučit nejvýhodnější nabídku na základě jeho preferencí, obchodník chce být nabídnut co největšímu počtu lidí a samotný srovnávač chce získat co nejvíce na provizích.

Dalšími kritérii, mezi kterými musíme hledat kompromis jsou krátkodobé přilákání uživatelů, což vede k rychlému výdělku, a schopnost uživatele dlouhodobě udržet, případně opakovaně přivést. Příkladem takového rozhodování může být použití click-bait titulků v prostředí zpráv, které zvyšují šanci na rozkliknutí článku, ale podryvují dlouhodobou důvěru ve zpravodajský server.

Vyvažovat musíme taky vzájemně se negativně ovlivňující kritéria, která mají vliv na pocit uživatele z užívání systémů. Můžeme uvést například úplnost informací a přebytek informací. Obě tyto kritéria ovlivňujeme například počtem doporučení.

Různé nároky můžeme mít i na celkový systém doporučení. Je lepší použít vysoce komplikovaný model s lepšími výsledky, nebo výrazně jednodušší byť o něco zaostávající kvalitou doporučování? Je třeba uvažovat, o jak velký rozdíl kvality

doporučování jde, jak dlouho trvá výpočet doporučení nebo o kolik silnější, tím pádem dražší, výpočetní prostředky jsou nutné.

1.4.1 Předchozí práce na multi-objective doporučovací systémy

V této části představíme různé práce, které se v minulosti zabývaly vícekriteriálním doporučováním, které jsme popsaly v přechozí sekci. Nezmiňujeme zde návrhy algoritmů zabývající se jinou vícekriteriální optimalizací než přes jednotlivé metriky pro kritéria kvality doporučování. Začneme s prvními pokusy, kdy se v seznamu doporučení sestaveném na základě relevance měnilo pořadí podle diverzity. Na závěr podrobněji představíme návrh multi-objective doporučovacího algoritmu (Peška a Dokoupil, 2022) a po něm i následující analýzu zaměřenou na chování uživatelů při doporučování právě tímto algoritmem (Dokoupil a kol., 2023a). Na obě tyto práce bezprostředně navazujeme.

Je také třeba zmínit, že některé studie pro to, co jsme zadefinovali jako relevanci, používají termín přesnost doporučení a jako relevanci berou až celkové skóre za přispění všech metrik. Pro zachování konzistence tohoto textu ale i při popisu předchozích prací udržíme význam jednotlivých termínů, jak jsou zadefinovány v kapitole 1.3.

Ještě před prvními studiemi o multi-objective doporučovacích systémech, se problém více kritérií začal řešit v oblasti vyhledávání informací (infromation retrieval). Carbonell a Goldsteinová (1998) představili koncept maximální marginální relevance, který vybírá výsledky vyhledávání na základě metriky závislé na relevanci i diverzitě. Hodnota diverzity zde byla počítána jako maximální podobnost s jakýmkoli dokumentem, který již byl vybrán jako součást odpovědi na uživatelův dotaz.

I v oblasti doporučovacích systémů se první pokusy zaměřovaly především na zapracování diverzity. Ziegler a kol. (2005) pracovali se seznamem doporučení sestaveným na základě algoritmu počítající relevanci. Následně seřadily jednotlivé položky v seznamu na základě intra-list podobnosti, tedy vzájemné podobnosti položek, kde nejvýše byly ty s nejmenší podobností. Poté se skóre vypočítalo jako lineární kombinace těchto 2 složek s nastavitelným parametrem určujícím, jak velkou váhu má mít diverzita a tedy zároveň jak malou relevance.

Zhang a Hurley (2008) namodelovali problém vzájemně soupeřících kritérií, kdy jedním z nich je snaha o co nejvyšší diverzitu a druhým udržení relevance s uživatelským dotazem, jako binární optimalizační problém. Samozřejmě v případě doporučovacích systémů uživatel většinou nezadává explicitní dotazy, ty nahrazuje jeho uživatelský profil, tedy například jeho předcházející hodnocení položek.

Práce Jambora a Wanga (Jambor a Wang, 2010) se zaměřila pouze na doporučovací algoritmy fungující na principu kolaborativního filtrování, jejichž výstupem je predikce hodnocení položek. Problém definovali jako lineární optimalizaci maximalizace vah w závislých na vektoru predikovaných hodnocení r: $\max_{w} w^T r$ za podmínek $1^T w = 1$ a $w \succ 0$, kde $w \succ 0$, znamená, že všechny složky vektoru w jsou větší než 0. Následně zapracovává další kritéria, jako je doporučování tzv. long-tail položek, což je spjato s novelty, a dostupnost dané položky (např. zásoby na skladu v e-komerce). Tyto dvě kritéria vstupují do lineární optimalizace

jako další nutné podmínky, které musí hodnota w splňovat. Následně lze položky seřadit právě na základě vah w.

Oh a kol. (2011) se přímo zaměřují na omezení popularity a zvýšení novelty. Autoři zde používají PPT (osobní tendence k popularitě), kdy porovnávají distribuci popularity v uživatelově profilu (např. na základě pozitivních hodnocení položek) a distribuci popularity v možných seznamech doporučení. Pro toto porovnání používají Earth Mover's Distance (Rubner a kol., 1998). Ta v zásadě představuje cenu / náročnost transformace jedné distribuce do druhé. Při doporučování je pak z top k položek na základě relevance utvořen tzv. seed seznam, který respektuje distribuci popularity. Následně jsou pomocí greedy algoritmu porovnány položky v seed seznamu s položkami ze stejného intervalu popularity s vyšším skóre na základě relevance. V případě, že má porovnávaná položka ze seed seznamu horší skóre relevance, je nahrazena lepší položkou.

Adomavicius a Kwonová (2012) přišli s návrhem, který seřadí položky podle dvou kritérií. Prvním kritériem je relevance. U algoritmů, jako je faktorizace matic, je první seřazení vytvořeno na základě predikovaného hodnocení. Následně se zformuje druhé seřazení položek. Tvůrci jako možnosti pro druhou variantu navrhují inverzní popularitu (novelty), průměrné hodnocení položky, počet uživatelů, kterým se položka líbí, rozptyl v hodnocení položky a rozptyl v hodnocení položky mezi nejpodobnějšími uživateli. Následně se na skóre prvního kritéria určí mez T_R, kdy jsou nejprve do seznamu vybrány položky se skóre vyšším, než je T_R, a ty jsou seřazeny na základě skóre druhého kritéria. Ostatní položky jsou pak přidány do seznamu doporučení za ně, již na základě pořadí dle skóre prvního kritéria.

Ribeiro a kol. (2015) pracují již se třemi kritérii kvality doporučování, a to relevancí, novelty a diverzitou. Položky pro každého uživatele postupně seřadí do Pareto front na základě hodnot vzhledem k použitým metrikám. Z nich vytvoří seznam doporučení, pro který platí, že žádná položka s vyšším indexem nedominuje položce s nižším indexem. Pořád je ale nutné zařídit seřazení položek, které sobě vzájemně nedominují. Pro tento problém jsou navržena dvě řešení. Tím prvním je seřazení podle počtu položek, kterým dominují a druhou variantou je použití SVM-Rank algoritmu (Joachims, 2002).

V práci Jugovace, Jannacha a Lerneho (Jugovac a kol., 2017) se k problému multi-objective doporučování přistupuje na základě následujících kroků. Nejdříve je pro optimalizovanou metriku např. diverzity či popularity, vypočítána preference uživatele. Ta je spočtena na základě hodnoty metriky pro seznam N ním nejlépe hodnocených položek. Poté je získáno pořadí doporučení na základě relevance, které je rozděleno na k nejlepších položek (k odpovídá velikosti seznamu doporučení), čímž vznikne seznam T_U a několik dalších v pořadí bezprostředně následujících položek X_U. Následně je seznam T_U procházen v opačném pořadí a pro každou položku z T_U je hledána položka z X_U, jejíž prohození povede k co největší podobnosti preferencí uživatele pro hledanou metriku (varianta diverzity, popularity,...) a hodnoty této metriky pro upravený $T_U^{'}$. Pokud by prohození položek vedlo k ucházejícímu zvýšení podobnosti, jsou položky prohozeny. Po dostatečném počtu kroků je upravený seznam T_U^* předán jako seznam doporučení uživateli. V případě použití více kritérií zároveň (mimo relevanci), musí být podobnosti preferencí uživatele a hodnoty metrik pro $T_U^{'}$ vhodným způsobem agregovány.

Xie a kol. (2021) se zaměřují na relevanci, diversitu a vysvětlitelnost. Používají tzv. knowledge graph, jehož vrcholy jsou uživatelé, položky a případně nějaké vlastnosti položky (např. režisér v doméně filmů). Orientované hrany mezi užívatelem a položkou, případně mezi uživatelem a vlastností, reprezentují pozitivní explicitní hodnocení od uživatele. Navíc hrany mezi položkou a vlastností reprezentují, že k sobě patří. Následně jsou zvažovány tři optimalizační funkce, jejichž výsledky reprezentují hodnotu přesnosti, diverzity a vysvětlitelnosti každého seznamu doporučení. Pro výběr správného seznamu doporučení se využívá multi-objective evoluční algoritmus, kde jedinec je jeden seznam doporučení.

Za pomocí modelu s konvolučními filtry a grafové neuronové sítě řeší kompromis mezi relevancí a diverzitou Isufi a kol. (2021). Pro vzájemný vztah mezi položkami a mezi uživateli je využita Pearsonova korelace, pomocí níž se získají hodnoty pro matice korelací. Poté se na základě matic vytvoří dva grafy, graf nejbližšího souseda a nejvzdálenějšího souseda. Z těchto grafů se model učí společnou konvoluční reprezentaci. Při učení modelu regulátor vyvažuje informace obou konvolučních modelů.

V poslední z prací, kterou je třeba zmínit, byl navržen algoritmus EP-FuzzDA vycházející z D'Hondtové metody využívané pro přepočet mandátů na základě počtu hlasů ve volbách (Maleček a Peška, 2021). Dle této metody se vždy vybere kandidát ze strany s největší vahou (na začátku na základě počtu hlasů) a následně se váha této strany pokrátí. V multi-objective doporučovacích systémech je ale nutné počítat s tím, že jeden kandidát (= položka) se může objevit na více stranických kandidátkách (= seznamy doporučení např. na základě metriky). Proto je D'Hondtova metoda upravena tak, aby si s tímto problémem poradila pomocí maximálních zisků vzhledem k tzv. EP-rel-sum průběžného seznamu doporučení. Ta zajišťuje balanci mezi výběrem nejlepších položek napříč metrikami (v průměru) a zároveň volbou položky na základě doposud méně reprezentovaných metrik v seznamu doporučení.

Naše práce se od většiny výše popsaných prací liší tím, že je schopna pracovat s více metrikami. Žádná ze zmíněných studií se navíc detailněji nezaměřuje na vysvětlování doporučení v multi-objective doporučovacích systémech. Zároveň na rozdíl od našeho výzkumu není výrazněji řešena vizualizace těchto doporučení a ani varianty, jakými by mohl uživatel takovéto doporučování ovlivňovat.

Následující dvě podkapitoly popisují studie, na něž navazujeme. První práce navrhuje algoritmus pro multi-objective doporučování, který využíváme i v naší studii. Navazující analýzu interakcí uživatelů s tímto typem doporučovacího systému chceme výrazně rozšířit a doplnit o další výsledky, které nebyly v této práci zkoumány.

1.4.1.1 Proporcionalita na úrovni výsledků

Nyní si popíšeme práci jejíž výstupem je algoritmus pro multi-objective doporučování. Celý název práce je Towards Results-level Proportionality for Multi-objective Recommender Systems (Peška a Dokoupil, 2022).

Proporcionalita na úrovni výsledků znamená, že uživatelem zvolený poměr vah kritérií by se měl propsat do samotného seznamu doporučení. To znamená, že uživatel může navolit poměr například takto: 40 % relevance, 30 % diverzita a 30 % popularita. Přestože existují již dlouho varianty multi-objective doporučování, které berou v úvahu váhu jednotlivých kritérií kvality doporučování, není

Algorithm 2 Algoritmus RL-Prop (Peška a Dokoupil, 2022)

Vstup: kritéria $m \in M$ a jejich váhy w_m; $\sum_{m \in M} w_m = 1$
položky k doporučení $i \in I$
Výstup: seřazený seznam doporučení L
$L = []$; $TOT = 0$; $\forall m \in M : g_m = 0$
for $c \in [1,...,k]$ **do**
  for $i \in I \setminus L$ **do**
    $TOT_i = \max(TOT, TOT + \sum_{\forall m} g_{m,i})$
    $\forall m : r_m = TOT_i \times w_m - g_m$
    if $g_{m,i} \geq 0$ **then**
      $g_i = \sum_m \max(0, \min(g_{m,i}, r_m))$
    else
      $g_i = \sum_m \min(0, g_{m,i} - r_m)$
    end if
  end for
  $i_{best} = arg\,max_{\forall i}\, g_i$
  $L = L + i_{best}$
  $\forall m : g_m = g_m + g_{m,best}$
  $TOT = \sum_{\forall m} \max(0, g_m)$
end for

jejich poměr dobře promítnutý do výsledného seznamu doporučení. Takové systémy výrazně nadhodnocují kritéria s nejvyššími vahami a výrazně podhodnocují ty s nízkými.

Autoři navrhují nový algoritmus RL-Prop (viz alg. 2). Nejdříve přepokládejme, že pro každou metriku $m \in M$ a částečný seznam doporučení L dokážeme spočítat skóre tohoto seznamu vzhledem k vybrané metrice $m(L)$. Přidání položky i do seznamu doporučení, získáme nový seznam doporučení $L + i$. Následně můžeme vypočítat zisk g vzhledem k metrice m zařízený přidáním položky i:

$$g_{m,i} = m(L + i) - m(L)$$

Z toho vyplývá, že $g_{m,i}$ značí přidanou hodnotu položky i pro metriku m.

Při návrhu RL-Prop muselo být vzato v úvahu, že uživatelé procházejí doporučení od prvních pozic a ne vždy jej uvidí celý. Proto bylo cílem, aby proporcionalita metrik fungovala i na částečných seznamech (první doporučení, první 2 doporučení, první 3 doporučení, ...). Algoritmus tedy musí fungovat iterativně s postupným přidáváním po jedné položce do seznamu doporučení.

Důležitou vlastností algoritmu je také, že musí penalizovat nízké hodnoty jednotlivých metrik. Ilustrovat to lze na situaci se dvěma metrikami se stejnou vahou. Jedna položka má skóre metrik 3 a 3, druhá 71 a 97. Přestože první položka lépe zachovává proporcionalitu mezi kritérii, je očividné, že mnohem lepším doporučením je položka druhá. Mimo položky zhoršující poměr metrik je tak nutné penalizovat také položky s nízkým skóre.

Pro otestování algoritmu byly zvoleny varianty metrik pro relevanci, diverzitu a novelty. Odhadovaná relevance byla výstupem faktorizace matic. Pro diverzitu byla použita intra-list diverzita (viz kapitola 1.3.2.1) a pro novelty očekávaný

doplněk popularity (viz kapitola 1.3.3.1).

Důležité bylo také vyřešit naškálování hodnot různých metrik tak, aby měly hodnoty stejný význam. Pokud nejrelevantnější položka má hodnotu relevance 97 a položka nejvíc přispívající k diverzitě má hodnotu diverzity 17, neměl by být poměr těchto hodnot interpretovaný jako 97:17, ale jako 1:1. Tento problém se pokusili autoři řešit dvěma metodami. Tou první byla normalizace pomocí kumulativní distribuční funkce, druhou testovanou metodou byla standardizace. První varianta se ukázala v offline vyhodnocení jako vhodnější.

Kód[1] obsahující algoritmus RL-Prop a naimplementované varianty metrik byl v rámci této práce jednak rozvinutý a jednak změněný z použití pro offline vyhodnocení na použití pro real-time computing při komunikaci s naší webovou aplikací, která odesílá požadavky na seznam doporučení pro skutečné uživatele.

1.4.1.2 Propojení interakcí a sklonu uživatele k multi-objective doporučování

Na předchozí práci autoři navázali v Looks Can Be Deceiving: Linking User-Item Interactions and User's Propensity Towards Multi-Objective Recommendations (Dokoupil a kol., 2023a). V této studii porovnali interakce a spokojenost uživatelů s multi-objective doporučováním a klasickým single-objective doporučováním na základě relevance. Zároveň se zaměřili na souvislosti mezi tím, jaký důraz dává uživatel na jednotlivá kritéria kvality doporučování při nastavení jejich důležitosti, jaké položky jsou mu zobrazeny a jaké položky následně vybírá.

Uživatelská studie probíhající na datech z domény filmů byla rozdělena do několika fází, kdy v každé byly uživateli zobrazeny doporučení obou algoritmů a on mohl následně označit položky, které se mu líbí. Uživatel měl zároveň možnost měnit poměr vah jednotlivých kritérií, konkrétně relevance, diverzity a novelty, které ovlivňovaly výběr položek do seznamu doporučení pomocí RL-Prop (Alg. 2).

Jedním z důležitých výstupů práce je pozorování, že zatímco uživatelé v prvních fázích mnohem více vybírali položky doporučené pouze na základě relevance, v těch pozdějších (od 3. či od 4. fáze) už byly více vybírány položky získané z multi-objective doporučování.

Zároveň bylo zjištěno, že distribuce přidaných hodnot položek v rámci výpočtu algoritmu RL-Prop (viz kapitola 1.4.1.1) v seznamu doporučení postrádala segmenty s velmi malými a velmi vysokými hodnotami, což bylo zřejmě zapříčiněno kovariancí jednotlivých metrik, kdy například doporučování diverzních položek většinou zvyšuje i novelty. Jak relevance, tak diverzita byla v seznamu doporučení poddimenzována oproti uživatelem explicitně specifikovaným váhám. Co se týče porovnání s jeho následným chováním v podobě výběru položek, tak u prvního z kritérií byl problém poddimenzování potlačen tím, že uživatel se vyhýbal položkám s nízkou hodnotou relevance. U diverzity se problém objevuje i u vybraných položek uživatelem, což je zapříčiněno větším počtem položek v seznamu doporučení s nízkou přidanou hodnotou pro diverzitu.

Uživatelé tedy výrazně přeceňují důležitost diverzity, kdy se jejich deklarovaný sklon k tomuto kritériu kvality doporučování a následné chování při výběru položek, velmi liší. Celkově je míra nesouladu mezi metrikami kritérií a představami uživatelů o nich podstatná.

[1] Implementace RL-Prop k dispozici zde: https://github.com/pdokoupil/moo-as-voting-fast

1.5 Explanations

Jak bylo již zmíněno výše, samotná přesnost doporučení, které se snaží doporučovací systémy docílit pomocí odhadované relevance, při jejich vyhodnocování nestačí (McNee a kol., 2006). Jednou ze zásadních vlastností doporučovacích systémů, která zvyšuje šanci na dosažení business cílů (počet nákupů v e-shopu, prodloužení předplatného na portálu s filmy a seriály,...) v dané doméně, je důvěra v samotné doporučování. Tato důvěra je zvyšována vysvětlováním, proč byla ta která položka doporučena (Chen a Puová, 2005). Pro tato vysvětlení se v literatuře používá termín explanations.

Způsobů, jak vysvětlovat doporučení je mnoho. Můžeme je rozdělit na personalizované („Tento film je vysoce hodnocen uživateli, kteří hodnotí podobně jako vy“) a nepersonalizované („Zákazníci, co koupili tuto knihu, koupili také“). Zároveň jsou pro dobrou funkci explanations nutné navrhnout správně dvě věci. Tou první je způsob vizualizace doporučení, tou druhou je text vysvětlení (Gedikli a kol., 2014). Text by většinou měl být závislý na způsobu doporučování tak, aby opravdu správně podal uživateli informaci, proč je mu daná položka zobrazena. Jiná vysvětlení tak jsou očekávána pro doporučovací algoritmy na bázi kolaborativního filtrování a jiná pro content-based přístupy.

Výše zmíněná důvěra uživatele v doporučovací systém je ale jen jeden parametr, který můžeme zlepšovat či zvyšovat pomocí explanations. Důvěra je sice výrazně závislá na kvalitě doporučování, uživatelé ale dokážou lépe přijmout a odpustit nevhodná doporučení, pokud dostanou rozumná vysvětlení, proč k nim došlo (Tintarevová a Masthoffová, 2007).

Pokud vysvětlujeme uživateli věrohodně to, jak systém funguje, resp. na jakém základě vybírá doporučení, zvyšujeme jeho transparentnost. Pomocí transparentnosti může uživatel pochopit, proč nedostává doporučení, která si přeje, a případně pak správně ovlivnit doporučovací systém, aby mu nabízel lepší položky. Protože ne vždy je možné nebo rozumné odhalit uživateli přesný způsob, jak doporučovací systém funguje, můžeme rozlišovat mezi opravdovou transparentností a transparentností vnímanou uživatelem.

Dalším důležitým cílem explanations je účelnost (effectivity). Účelnost znamená, že systém díky vysvětlení svých doporučení pomohl uživateli nalézt vhodnou položku. Zlepšení účelnosti můžeme dosáhnout pomocí vysvětlení, jež umožní uživateli dělat správná rozhodnutí. Rozumnými explanations pro zvýšení účelnosti může být porovnání položek na základě uživatelových preferencí. Nejčastějším přístupem, jak přesněji vnímat účelnost, je snaha o co největší přiblížení odhadované kvality položky uživatelem a skutečností.

Souvisejícím cílem je účinnost (efficiency), která měří, jak rychle jsme schopni dosáhnout výsledku, jako je výše zmíněné nalezení vhodné položky. Účinné vysvětlení může být například zdůvodnění, v jakých parametrech je jeden produkt lepší než druhý (např. velikost obrazovky u monitoru). Účinnost můžeme měřit jednak na čas a jednak počtem kroků (kliknutí) nutných k doporučení vhodné položky.

Pokud uživateli umožníme k vysvětlením ještě jejich zpětnou vazbu, zvýšíme také parametr kontrolovatelnosti. Uživatel může systému dát vědět, co je špatně například pomocí odmítnutí nerelevantních položek nebo částečnou úpravou svého profilu(hodnocení, ...), aby dostával jiná doporučení.

Vlastností systémů, k níž pomáhají explanations, důležitou v některých doménách, zejména v oblasti e-komerce, je přesvědčitelnost. Jak již z názvu vyplývá, tato vlastnost zvyšuje schopnost systému přesvědčit uživatele například k nákupu produktu. Obecně zlepšení tohoto parametru je výhodné pro systém, ne už tak vždy pro uživatele.

Zásadním cílem je spokojenost uživatele. I ta je samozřejmě závislá na přesnosti doporučovacího systému, můžeme ji však nadále zlepšovat jednak příjemnější ovladatelností či přehledností stránky nebo aplikace, ale také právě pomocí explanations nebo podrobným popisem položky.

Je potřeba také zmínit, že většina cílů doporučení spolu souvisí. Transparentnost většinou zvyšuje důvěru a zároveň mnohem lépe funguje společně s kontrolovatelností, než pokud systém vysvětlování zlepšuje jen jeden z těchto dvou parametrů. Úzce svázané jsou i účelnost a účinnost, které jsou navíc obě silně závislé na kvalitě samotného doporučování. Zvýšení jakéhokoli z parametrů s výjimkou přesvědčitelnosti většinou povede také ke zvýšení spokojenosti uživatele se systémem. Na druhou stranu přílišná transparentnost může mít negativní vliv na účinnost systému. Většina parametrů je navíc nepříznivě ovlivňována vysokou přesvědčitelností. Rozhodnutí, k jakým parametrům by měly explanations nejvíce přispívat, je odvislé od domény, pro niž je doporučovací systém nasazen.

Je také třeba zmínit vliv úplnosti a přesnosti vysvětlení toho, jak doporučovací systém pracuje. Úplnost značí, do jaké míry je celý postup výběru doporučení vysvětlen, a přesnost znamená, jak moc každá část explanation pravdivě popisuje doporučovací systém. Ukazuje se, že úplnost vede k představě uživatele o systému, která je věrná skutečnosti. Uživatelé také preferují co nejkonkrétnější explanations vztahující se přímo k doporučení položky. Nedostatečná přesnost vysvětlení snižuje vnímanou úplnost explanations (Kulesza a kol., 2013).

Z výzkumu Gedikliho, Jannacha a Geho (Gedikli a kol., 2014) vyplývá několik poznatků. Tím prvním je, že uživatelé nejrychleji zpracují stručnější explanation s méně informacemi. Dalším výstupem je doporučení používat informace o položce, které jsou specifické pro danou doménu, což zajišťuje zvýšení efektivity (účelnosti). Je také dobré se zaměřit na to, aby uživatelé považovali vysvětlení za transparentní, což má přímý vliv na jejich spokojenost se systémem. Není vhodné zaměřovat se příliš na účinnost (= efficiency), jelikož uživatelé pro správná rozhodnutí potřebují většinou více času. Co se týče ochoty věnovat čas porozumění systému, pak podle většiny uživatelů výrazně pomáhá, pokud jsou vysvětlení stručná a pochopitelná. Někteří z nich navíc uvádějí jako další motivaci pochopení důvodu nesprávných doporučení. Několik dalších poznatků, jako je rada vizualizovat explanations tak, jak jsou na ně uživatelé zvyklí z populárních webových stránek, nejsou pro naši práci, kde chceme místo těch klasických ukazovat vysvětlení na základě skóre jednotlivých metrik, použitelné.

1.6 HCI v doporučovacích systémech

Jak je popsáno výše, nejen způsob doporučování ale i explanations rozhodují o spokojenosti uživatele s doporučovacím systémem. Dalším faktorem jsou metody, jakými spolu systém a uživatel komunikují. To zahrnuje jednak vizualizaci doporučení včetně případné vizualizace jeho vysvětlení a dále i způsob, jakým uživatel může fungování celého systému ovlivňovat. Všechny tyto prvky patří

do oblasti interakce člověka a počítače, pro niž je zaveden pojem HCI.

1.6.1 Interakce

Důležitou součástí každého systému personalizovaného doporučování je schopnost získat preference uživatele (Tintarevová a Masthoff, 2022). Pokud nyní vynecháme implicitní zpětnou vazbu, kterou získáme díky mapování uživatelových kroků v systému, a zaměříme se čistě na to, jak získat uživatelovy preference explicitně, pak nejtypičtější variantou je nechat uživatele hodnotit položky. Takové hodnocení může být jednoškálové (položka se mi líbí), dvouškálové (líbí - nelíbí), víceškálové (počet hvězdiček) nebo v podobě psaného textu (recenze).

Výběr správné škály hodnocení nemusí být tak nepodstatný, jak by se mohlo na první pohled zdát. Pokud používáme hodnotící škálu obsahující neutrální hodnotu (tříškálové, pětiškálové), pak uživatelé mají tendenci volit právě neutrální bod, aby odpověděli méně negativně, což je způsobeno společenskou vhodností, kdy uživatelé hodnotí tak, jak je to podle nich společensky žádoucí. Na druhou stranu škály bez neutrální varianty nutí skutečně nerozhodnuté uživatele volit pozitivní nebo negativní hodnocení, což pak vede ke zkreslení směrem k vyšším a nižším odpovědím (Garland, 1991). Přítomnost neutrálních bodů ve škále vede k méně extrémním odpovědím a vyšším hodnocením (Weijters a kol., 2010).

I pro stejnou škálu hodnocení záleží jednak na seřazení možností, kdy varianta zleva od nejvyššího po nejnižší hodnocení vykazuje vyšší hodnocení než ta opačná (Yanová a Keusch, 2015) a jednak na použitých číslech reprezentujících preference uživatele. Škála od -4 (nejhorší) do 4 zaznamenává výrazně vyšší (tzn. blíže hodnotě nejvyššího skóre) průměr hodnocení než škála od 1 (nejhorší) do 9, což je způsobeno tím, že varianty se zápornými čísly jsou uživatelem vnímány jako více negativní (Amoo a Friedman, 2001).

Je také vhodné mimo samotná doporučení dát uživateli alternativu v podobě vyhledávání oblíbených položek k ohodnocení. Další možností je získávat uživatelovy preference přímo. Například specifikací filmového žánru a oblíbeného filmu z tohoto žánru a případně herce. Systém pak doporučí film z vybraného žánru podobný uživatelově oblíbenému filmu, ve kterém v ideálním případě hraje jeho oblíbený herec. V moderních konverzačních přístupech může uživatel psát své preference textově a systém se může doptat na podrobnější informace.

Další variantou je možnost změnit seznam doporučení. Obecně může uživatel požádat o výměnu celého seznamu doporučení nebo jeho části. V doméně filmů či hudby může blokovat např. režiséra, herce, resp. interpreta. V jiných doménách, kde doporučování proběhne až poté, co uživatel specifikuje požadavky, je možné, že neexistuje žádná položka, co by je splňovala. Uživateli pak může být nabídnuto, co by mohl změnit, aby nějaká doporučení k dispozici byla. To můžeme ilustrovat na příkladu portálu doporučujícího hotely, kde není žádný k dispozici pro dané místo, termín a cenovou hladinu. Uživatelovi může být nabídnuta změna na jiný termín, případně zvýšení akceptovatelné ceny.

Souvisejícím prostředkem je i kritika doporučení závislá na doméně. Uživateli je umožněno na jednotlivá doporučení, případně celý seznam, předat zpětnou vazbu, která přesně specifikuje, co je na něm špatně. Příkladem je předání relativních zpráv jako „levnější“, „větší obrazovka“, „novější produkty“ v e-komerci. Systém takové zpětné vazbě většinou dobře rozumí a zvládne na ni rychle reagovat

(Jannach a kol., 2017).

V případě delšího seznamu doporučení lze uživateli nabídnout změnu těchto výsledků. To jde například v podobě filtrování výsledků, alternativou je i volba toho, podle čeho se mají doporučení seřadit.

Variantou je i ohodnocení doporučení (nikoli položky). Uživatel může dát najevo, zda mu doporučení přijde zajímavé, případně vyjádřit spokojenost s doporučením na vícehodnotové škále. Dále může uživatel dát systému najevo, že danou položku již viděl.

Více interaktivní systémy dovolují uživatelovi více specifikovat své zájmy. Například již zmíněným výběrem oblíbených žánrů (filmy, hudba, knihy,...), požadavky na položky, jako je maximální cena, minimální doba doručení atd. Případně může uživatel konkrétněji specifikovat, na základě čeho má systém vybírat doporučení, což lze provést různými způsoby odvislými od varianty doporučovacího systému.

Dalším prostředkem, pomocí kterého lze měnit chování doporučování, jsou interaktivní explanations. To znamená, že lze dát systému zpětnou vazbu, že jeho předpoklad je mylný (např. u vysvětlení „Tato kniha by se vám mohla líbit, protože jste četl...“). U explanations si jde také představit uživatelovu volbu jejich vizualizace nebo úrovně podrobnosti.

V doménách, kde máme k dispozici více různých doporučovacích systémů nebo hybridní doporučovací systém, může uživatel způsob doporučování přímo vybírat, resp. ovlivňovat specifikováním vah pro druhý případ. Každá taková změna by měla vést ke změně i ve výsledcích doporučování.

1.6.2 Personalizace

Nyní odhlédneme od toho, co vše může uživatel měnit či ovlivňovat, a zaměříme se na jeho vlastnosti. Obecně platí, že personalizovaná doporučení fungují mnohem lépe než ta nepersonalizovaná. Navíc víme, že k dalšímu vylepšení personalizovanému doporučování pomůže znalost osobnostních rysů uživatele. Příkladem je, že otevřenější uživatelé jsou více nakloněni novým věcem, což v doméně doporučovacích systémů vede k vyšší důležitosti novelty (Tkalcic a Chenová, 2015).

Osobnostní rysy lze získat explicitně. V takovém případě musí uživatel vyplnit dotazník, který se zaměřuje na jednotlivé typy povah. Nejčastěji používané dotazníky pochází z International Personality Item Pool (Goldberg a kol., 2006), ty obsahují poměrně vysoký počet otázek. Výhodou této explicitní varianty je velmi přesná představa o povaze uživatele. Přesto je použití dotazníku použitelné jen v uživatelských studiích, a ne v reálném systému, kdy uživatel většinou není ochoten systému poskytnout takové informace nebo věnovat čas vyplnění dotazníku.

Pro implicitní extrakci uživatelovy osobnosti jdou použít data z jiných systémů. Především jeho interakce na sociálních sítích, případně z jeho textů v emailech nebo třeba v blozích. Na extrakci povahy přímo z chování v doporučovacím systému se zaměřují Hu a Puová (2013). V práci se snažili zjistit, zda existuje korelace mezi osobnostními rysy z The Five Factor Modelu (McCrae a John, 1992) a chováním v systému reprezentovaném 4 proměnnými - počtem hodnocení, poměrem pozitivních hodnocení, pokrytím kategorií (žánrů) a diverzitou

zájmů uživatele. Výsledkem bylo potvrzení této korelace.

Pokud máme k dispozici osobnostní rysy z dotazníku nebo z jiné služby, můžeme si pomocí nich lépe poradit s problémem nového uživatele a doporučovat mu na základě jeho osobnosti. To se dá řešit například pomocí podobnostní matice uživatelů nebo upravenou faktorizací matic, kde je součástí vektorů s latentními faktory i vektor reprezentující povahu uživatele. Dalším možným použitím je personalizovaná diverzita (míra, případně i okruhy témat / žánrů) nebo míra novelty vycházející právě z osobnostních faktorů uživatele.

1.6.3 GUI

Obecně pro webové stránky i mobilní aplikace platí, že kriticky důležitým faktorem vedoucím ke spokojenosti uživatele je způsob prezentace, tedy grafické uživatelské rozhraní (GUI) (Garett a kol., 2016). Knijnenburg a kol. (2012) tvrdí, že dokonce způsob, jakým jsou doporučení ukázána, je ještě důležitější než samotná kvalita algoritmu. Aby totiž samotný doporučovací systém měl přidanou hodnotu, je nutné, aby si uživatel položek vůbec všiml a zároveň aby byl schopen posoudit, zda jsou pro něj doporučení zajímavá.

Cílem vizualizace doporučení je zlepšit několik důležitých vlastností systému (Murphy-Hill a Murphyová, 2014). Zásadním faktorem je srozumitelnost systému. Tu ovlivňujeme tím, zda jsou doporučení zřejmá (např. doporučení v doméně knih od stejného autora v detailu jedné z jeho knih), a také snížením kognitivního úsilí uživatele nutného k pochopení toho, o jaká doporučení jde. S tím souvisí i to, jak dobře dokáže uživatel posoudit vhodnost doporučení. I k tomuto dokáže významně přispět vizualizace například pomocí rozumného zobrazení detailnějších informací o doporučené položce. V e-komerce lze výrazně vylepšit posuzovatelnost pomocí prezentace porovnání dvou nebo více položek. Vlastností úzce navázanou na explanations je transparentnost. Nestačí ovšem mít vysoce transparentní vysvětlení doporučení, je nutné, aby je uživatel jednak zaregistroval a jednak byly zobrazeny tak, že jim uživatel bude rozumět.

První výzkum v tomto směru Swearingenové a Sinhové (Swearingenová a Sinhová, 2001) přinesl několik základních tipů, jak vytvářet rozumné GUI pro doporučení. Mezi ně patří zobrazování recenzí dalších uživatelů, pokud jsou k dispozici. Obecně se vyplatí zobrazovat dostatek, tzn. spíše více informací o doporučené položce a vhodné je i použití většího počtu různých designových prvků. Velmi důležité je i zobrazení obrázku k doporučené položce, což pomáhá uživatelům vzpomenout si, zda ji již neznají.

Ozok a kol. (2010), jež na předchozí studii částečně navazují, se zaměřují na prostředí e-komerce. I zde uživatelé oceňují přesné informace o produktu, kde jako tři nejzásadnější uvádějí jméno produktu, obrázek a cenu. Zároveň autoři potvrzují důležitost recenzí od dalších uživatelů. Co se týče základního zobrazení stránek typu e-komerce, uživatelé by rádi viděli maximálně 3 doporučení přímo obsahující stručné popisy produktů, a to ve spodní části samotné stránky. Doporučení by také neměla být zobrazena v pop-up oknech.

Obě výše zmíněné práce je ovšem potřeba brát v dnešní době již s nadhledem. GUI webových portálů je dynamicky se rozvíjející prostředí a to, co bylo dříve považováno za rozumné, už dnes zdaleka nemusí platit. To je v případě studií více než dvacet let, resp. téměř patnáct let starých nutné zohlednit.

Výstupem práce Schnabela, Bennetta a Joachimse (Schnabel a kol., 2018) je, že uživatelé preferují zobrazení dodatečných informací o položce v popover okně při najetí myší před nutností na doporučení kliknout, což také vedlo k zvýšení kvantity zpětné vazby od uživatelů při jejich studii. Autoři potvrzují, že uživatelé preferují detailnější informace o produktu před těmi stručnějšími, což je stejný závěr jako u Swearingenové a Sinhové (Swearingenová a Sinhová, 2001). Podrobnější popis také znamenal vyšší kvalitu zpětné vazby, tedy zlepšení schopnosti uživatele posoudit vhodnost doporučení.

Detailněji se můžeme podívat na porovnání různých GUI variant zobrazení doporučení na základě výstupu práce Beela a Dixonové (Beel a Dixonová, 2021). Autoři navrhli sedm různých možností, jak zobrazit doporučení. Nejjednodušší zobrazení obsahovalo pouze název položky, druhé pouze obrázek, třetí bylo kombinací dvou předchozích, čtvrté zobrazovalo navíc pořadí doporučení. Další varianty vycházely již z třetí možnosti, tzn. že obsahovaly jak název tak obrázek položky, lišily se ale zvýrazněním. Páté zobrazení zvýraznilo nejlepší doporučení (první v pořadí), šesté orámovalo to doporučení, na které uživatel najel myší. Sedmá varianta fungovala stejně jako šestá s tím rozdílem, že při najetí myší na doporučení se zaměnil původní náhled za zvýrazněný popis položky. Právě tato poslední možnost vyšla v experimentu jako nejlepší těsně následovaná tou předposlední. Překvapivé bylo, že třetí nejlepší variantou byla ta první pouze s názvem položky, což ale mohlo souviset s metodou vyhodnocování pomocí CTR (mírou prokliku), kdy uživatelé mohli v tomto případě klikat na položku jen proto, že potřebovali více informací.

1.7 Vyhodnocování

V oboru doporučovacích systémů existuje několik variant vyhodnocení. V případě doporučovacího algoritmu se používá vyhodnocování offline pomocí starších statických dat a online, kdy je nový algoritmus přímo nasazen na produkci. Nyní již ale panuje shoda na tom, že samotné měření přesnosti (případně dalších metrik) algoritmu je nedostačující. Důležitými faktory, které chceme měřit je použitelnost, užitečnost a celková spokojenost uživatele s doporučovacím systémem, což vede k hlavním cílům jako jsou zisk a udržení uživatelů a také jejich „spotřeba“ (zhlédnutí, nákup, ...) položek (Knijnenburg a Willemsen, 2015).

V offline experimentech máme k dispozici dříve získaná data, což jsou většinou hodnocení nebo výběr položek uživateli. Na těchto datech vlastně simulujeme interakce uživatelů s doporučovacím systémem, kdy předpokládáme, že se chování uživatelů v době získání dat a chování v době nasazení systému nebude příliš lišit. Výhodou této evaluace je absence práce se skutečnými uživateli, což vede k nižším nákladům, a to jak, co se týče ceny výzkumu, tak, co se týče času. Offline vyhodnocování nicméně umožňuje jen velmi úzký pohled na doporučovací systém a používá se tak zejména k otestování přesnosti predikcí algoritmu. Funguje tak jako dobrý nástroj k prvnímu vyfiltrování špatných kandidátů.

Online vyhodnocování probíhá pomocí vyčlenění nějakého procenta provozu na testovanou variantu. Pro správné měření je důležité, aby uživatelé, kteří budou pracovat s novou variantou, byli vybíráni náhodně. Je také důležité, aby vše ostatní zůstalo netknuté, tzn. v případě testování 2 algoritmů musí být uživatelské rozhraní pro obě varianty stejné a naopak v případě, kdy testuji změnu v uživa-

telském rozhraní, musí uživatelé pracovat se stejným doporučovacím algoritmem. Výhodou tohoto přístupu je interakce se skutečnými uživateli, kteří provádí skutečné úkony v systému, aniž by byli jakkoli ovlivněni tím, že by věděli, že jsou součástí studie. Zároveň lze měřit, jak testovaný kandidát přispívá k celkovým cílům systému, jako je profit nebo udržení zákazníků. Nevýhodou tohoto způsobu jsou potenciální ztráty, pokud uživatelé v rámci experimentu pracují s horší variantou, než je ta současná, což může ohrožovat celkové cíle systému. Tomuto lze částečně předcházet provedením offline vyhodnocení a uživatelské studie ještě před spuštěním online evaluace (Gunawardana a kol., 2022).

1.7.1 Uživatelská studie

Pro případy, kdy chceme otestovat více než jen kvalitu algoritmu vzhledem k vybraným metrikám a zároveň ještě nechceme nový typ systému nasazovat na online prostředí, případně žádný takový portál nemáme k dispozici, je vhodným prostředkem uživatelská studie. Účastníci dostanou přístup do kontrolovaného prostředí, v současnosti nejčastěji k webové aplikaci, s doporučovacím systémem. Následně musí v tomto prostředí nějakou dobu pracovat a v průběhu studie či na jejím konci odpovídají na otázky, které jsou součástí dotazníku. Kromě odpovědí na otázky lze získat i další informace z chování uživatelů v systému, a to například kolik doporučení v seznamu zobrazí, kolik rozkliknou atd. Studie může probíhat v místnosti pod dohledem většinou zároveň s ostatními účastníky, pak se jedná o tzv. laboratorní studii.

Výhodou oproti offline experimentům je možnost testovat chování uživatelů při interakci se systémem a zároveň vliv systému na chování uživatele. Díky dotazníku máme navíc k dispozici kvalitativní data (například informace, zda uživatel považoval doporučení za relevantní), které mohou být často nutné pro správnou interpretaci těch kvantitativních (například počet doporučení, s kterými interagoval). Na druhou stranu je potřeba si uvědomit, že výsledky studie budou stále zaujaté, jelikož uživatelé se budou chovat jinak než v reálném prostředí, pokud ví, že jsou součástí experimentu (Gunawardana a kol., 2022).

Je také nutné si zadefinovat otázky, na které by měla uživatelská studie odpovědět. Je obvyklé, že dochází k porovnání různých kandidátů, ať už to jsou dva různé doporučovací algoritmy, různé typy explanations atd. K výsledkům porovnání se dá dojít dvěma způsoby. Jedním z nich je, že různí uživatelé pracují vždy s jedním z kandidátů. Tato varianta sice tolik časově nezatěžuje uživatele, na druhou stranu jich vyžaduje větší počet. Problémem je také nejasnost porovnání, kdy například 2 uživatelé ohodnotí kandidáta maximálním hodnocením, přitom by oba dokázali rozlišit, který z nich je lepší. Druhou variantu tedy je každého uživatele nechat pracovat s více kandidáty a nechat jej je porovnat. Nevýhodou tohoto způsobu je, že uživatelé jsou většinou více vědomí experimentu, než je tomu u první varianty (Gunawardana a kol., 2022).

Pokud uživatel pracuje s více kandidáty, jsou mu prezentovány buď všechny možnosti zároveň, nebo pracuje vždy s jednou možností a ty se postupně střídají. Při obou těchto variantách je třeba brát v ohled vliv, co je kde, resp. kdy zobrazeno. U prvního způsobu záleží na tom, který kandidát je zobrazen jako první, druhý atd., kdy bereme v úvahu, že uživatel je většinou prochází shora dolů a zleva doprava. U druhého způsobu zase zvažujeme pořadí, v jakém kandi-

dát s variantami pracoval. Důvodem, proč je nutné toto zohlednit, jsou například pozorovaná lepší hodnocení varianty, pokud následuje bezprostředně po špatném protikandidátovi (Gunawardana a kol., 2022). Navíc, pokud jsou uživateli zobrazeni všichni kandidáti zároveň, je lepší volbou sloupcové rozložení. To znamená, že jednotliví kandidáti jsou zobrazeni vedle sebe, ne pod sebou (Dokoupil a kol., 2023b).

Co se týče dotazníku, je nutné snažit se zajistit, aby uživatel neměl pocit, že by například měl odpovídat na hodnotící otázky více pozitivně, než jak to opravdu cítí. K tomu je důležité pokládat otázky tak, aby nenabádaly uživatele k jakékoli odpovědi. Jak by měly konkrétně vypadat otázky v dotazníku je zmíněno v ResQue frameworku (Puová a kol., 2011), který se snaží sjednotit způsob evaluace doporučovacích systému právě pomocí uživatelských studií.

Častým problémem uživatelských studií nicméně je, že jsou příliš zaměřeny na svou úlohu, tedy doporučovací systém, a mnohem méně připomínají skutečné prostředí webových portálů, ve kterých se dnes doporučování využívá. Uživatelé jsou tak tlačeni do jediné možnosti, jak procházet položky, kdežto obecně by ocenili více způsobů, které spolupracují (Kleemann a kol., 2022). Příkladem je vyhledávání na základě filtru a následné doporučování pouze těch položek, které nastavení filtru odpovídají. Proto je pro lepší vyhodnocení nutné navrhnout uživatelskou studii z širší perspektivy tak, aby nebyla zcela vázaná na jedinou úlohu doporučování a obsahovala více prvků. Taková studie nabídne více odpovídající pohled na to, jak může doporučovací systém uživateli pomoci v dané doméně a jak může spolupráce s dalšími rozhodovacími prvky zlepšit celkové posouzení systému (Loepp, 2022).

I pokud se zaměříme na pouhou část doporučování, měli bychom systém vyhodnocovat jako celek. To znamená měřit například, jak uživatelé vnímají použitelnost prvků, kterými systému dodávají zpětnou vazbu nebo způsob zobrazení doporučení (Knijnenburg a Willemsen, 2015).

1.8 MovieLens dataset

MovieLens datasety jsou nejpoužívanějšími datasety pro offline vyhodnocování doporučovacích systémů. První MovieLens dataset byl vyvíjen na Minnesotské Univerzitě a zveřejněn již na konci minulého století v roce 1998. Dataset obsahuje hodnocení filmů uživateli. Do roku 2003 bylo k dispozici pět úrovní hodnocení filmu od 1 hvězdy po 5 hvězd, poté se škála hodnocení zdvojnásobila rozčleněním na půl hvězdy, čímž uživatel dostal možnost rozlišení na deset různých úrovní od 1/2 hvězdy po 5 hvězd. Jedno takové hodnocení obsahuje ID uživatele, ID filmu, skóre hodnocení (0,5 až 5) a časové razítko. Všechna tato hodnocení byla získaná od uživatelů stránky MovieLens.

Vzrůstající popularita datasetu je zapříčiněna jednak rozvojem výzkumu v oblasti doporučovacích systémů a také proto, že filmy jsou doménou s několika vhodnými vlastnostmi pro vyhodnocení doporučovacího systému. Jednak je doména dobře známá drtivé většině uživatelů, navíc hodnocení filmů je vysoce subjektivní záležitostí, což je vhodné právě kvůli personalizaci doporučení, o které se systémy snaží (Harper a Konstan, 2015).

MovieLens Dataset byl vydán v několika verzích. Ta první vydaná v roce 1998 obsahovala 100 000 hodnocení. Následně byl počet hodnocení s každou verzí na-

výšen, v roce 2003 se dataset skládal z 1 milionu hodnocení, o šest let později jich bylo již 10 milionů. V roce 2015 již byla vydána verze s 20 miliony hodnocení a tou poslední je MovieLens 25M Dataset z roku 2019, který obsahuje 25 milionů hodnocení. Mimo klasických verzí MovieLens Datasetů je k dispozici ještě MovieLens Tag Genome Dataset vydaný nejdříve v roce 2014 a poté 2021, který obsahuje různé tagy a napojení tagů na filmy s hodnotou, jak moc je tag relevantní pro daný film. Tagy jsou k dispozici i jako součást datasetů MovieLens 20M Dataset a MovieLens 25M Dataset, ty na rozdíl od předchozích verzí ale neobsahují demografická data o uživatelích (GroupLens, 2023).

Námi používaný MovieLens 25M Dataset obsahuje navíc dataset s odkazy na ID v databázích IMDB a TMDB. Využitím TMDB API můžeme obohatit dostupné informace o filmech, což jsou pouze ID, jméno, přiřazené žánry a tagy o ty konkrétnější, jako jsou režisér filmu, vystupující herci, odkaz na trailer a další (TMDB, 2023).

2. Řešený problém

Zde bychom chtěli detailněji představit řešený problém. Ten spočíval v úpravě a rozšíření existující implementace multi-objective doporučovacího systému (viz kapitola 1.4.1.1), ve vytvoření webové aplikace obsahující některé prvky běžné na většině skutečných webových portálů a v provedení uživatelské studie, jejíž cílem bylo zjistit, jak uživatelé hodnotí multi-objective doporučovací systém a práci s ním v naší webové aplikaci.

2.1 Výzkumné otázky

Co konkrétně chceme zjistit, představíme podrobněji v této sekci pomocí definování výzkumných otázek. Odpovědi na každou z výzkumných otázek by pak měly představovat výstup naší práce.

2.1.1 RQ1: Stojí uživatelé o nastavení svých preferencí k jednotlivým kritériím kvality doporučování?

Zaměřujeme se na multi-objective doporučování, kde uživateli dáváme možnost nastavit své preference k jednotlivým kritériím kvality doporučování pomocí vah. Na základě poměru vah těchto kritérií jsou následně voleny jednotlivé položky do seznamu doporučení. Uživateli tedy dáváme možnost výrazným způsobem ovlivnit, jakým způsobem jsou mu vybírána doporučení.

Není ovšem jasné, zda uživatel vůbec rozumí tomu, jakým způsobem může modifikovat doporučování, a zda skutečně pozoruje, že se zvýšenou vahou kritéria kvality doporučování dostávají přednost položky přispívající k tomuto kritériu a naopak. Oba zmíněné problémy jsou vlastně podotázkami této výzkumné otázky a jejich splnění zároveň nutnou podmínkou pro kladnou odpověď.

Důležitým aspektem ale také je snaha o snížení úsilí, které uživatelé musí vynaložit, aby dosáhli svého cíle, což je v tomto případě nalezení vhodné položky, případně položek. Koneckonců i proto jsou doporučovací systémy tak populární. Zde ovšem vyžadujeme od uživatele, aby se zamyslel a následně specifikoval, jak by měla být doporučení pro něj vybrána.

Hypotézy:

- H1.1: Uživatelé pozorují zlepšení doporučování díky možnosti nastavení vah ke kritériím.
- H1.2: Uživatelé dávají přednost automatickému doporučování bez nutnosti nastavení vah ke kritériím.

2.1.2 RQ2: Jaký mechanismus pro nastavení vah jednotlivým kritériím kvality doporučování považují uživatelé za nejvhodnější?

Navazujícím problémem je volba vhodného mechanismu, pomocí kterého uživatelé mohou specifikovat své preference. Jediným požadavkem na takový me-

chanismus je to, aby umožňoval přímé nastavení vah jednotlivým kritériím, nebo seřazení kritérií od toho nejpreferovanějšího k těm méně preferovaným.

Dokoupil a kol. (2023a) ve své studii umožnili uživatelům použít pro specifikaci svých preferencí posuvníky (sliders). Již Shneiderman (1997) popisuje posuvníky jako přirozenou volbu pro filtry obsahující více polí s rozsahy číselných hodnot.

My k nim ale chceme navrhnout jiné varianty mechanismů a zjistit od uživatelů, které považují za nejvhodnější.

Hypotézy:

- H2.1: Uživatelé příliš nerozlišují mezi jednotlivými variantami mechanismů pro nastavení vah kritériím.

2.1.3 RQ3: Jaká kritéria kvality doporučování jsou pro uživatele přínosná?

Jedním z problému je také výběr používaných kritérií kvality doporučování. Od těchto kritérií vyžadujeme, aby jim uživatelé rozuměli a také aby byli schopni zpozorovat jejich vliv na výběr doporučování a vyhodnotit jejich přínos.

Dokoupil a kol. (2023a) volí tři zřejmě nejzákladnější, což jsou relevance, diverzita a novelty. K těmto třem kritériím navrhujeme ještě další alternativy v podobě popularity a kalibrace. Otázkou tedy je, zda uživatelé těmto vybraným kritériím rozumí a jsou schopni pomocí nastavení svých preferencí pozorovat doporučování více či méně přispívající k daným kritériím.

Hypotézy:

- H3.1: Uživatelé jsou schopni pochopit, k čemu přispívají jednotlivá kritéria kvality doporučování.
- H3.2: Uživatelé vnímají zvýšení či snížení vlivu jednotlivých kritérií kvality doporučování na seznam doporučení se změnou váhy kritéria.
- H3.3: Uživatelé dávají přednost relevantním doporučením.

2.1.4 RQ4: Jaké varianty metrik se pro jednotlivá kritéria kvality chovají nejblíže tomu, co od nich uživatelé očekávají?

Algoritmus RL-Prop (viz alg. 2) vybírá položky na základě hodnot odpovídajících přínosu položky k danému kritériu. Jak zmiňujeme v kapitole 1.3, existuje pro každé kritérium více metrik, na základě kterých tyto hodnoty můžeme vypočítat. I z toho vyplývá, že definice jednotlivých kritérií přesně neurčuje, jakým způsobem se dá přínos položky či seznamu doporučení vzhledem k danému kritériu měřit, z čehož plyne, že interpretace těchto kritérií kvality doporučování se může různit.

Proto navrhujeme ke všem používaným kritériím mimo kalibraci dvě až tři varianty, jak jednotlivé kritérium vnímat, což přeneseně znamená, na základě jaké metriky ho měřit. Jedním z výstupů této práce by tak měla být odpověď na otázky, jak uživatelé vnímají jednotlivá kritéria a jaká varianta metriky nejlépe splňovala to, co uživatelé od daného kritéria očekávají.

Hypotézy:

- H4.1: Uživatelé jsou schopni porovnat varianty metrik popularity a budou preferovat popularitu na základě hodnocení.
- H4.2: Nezanedbatelná část uživatelů není schopna rozlišit rozdíly v chování ostatních kritériích kvality doporučování při změně používané metriky.

2.1.5 RQ5: Jaká vysvětlení doporučení v multi-objective doporučování jsou vnímána jako nejpřínosnější?

Jak zmiňujeme v kapitole 1.5, explanations, tedy vysvětlení doporučení mají výrazný vliv na celkovou spokojenost uživatele s doporučovacím systémem. Pokud využijeme osvědčených postupů, mělo by vysvětlení skutečně odpovídat tomu, proč byla jednotlivá položka doporučena. A zároveň by vysvětlení mělo mít takový charakter, aby byl uživatel schopen na základě něj pochopit, co má změnit, aby doporučení více odpovídala jeho představám.

Vzhledem k tomu, že v našem specifickém případě je hlavním způsobem, jak ovlivnit doporučování, nastavení vah jednotlivým kritériím, měla by explanations obsahovat také přínos položky k danému kritériu. Proto bychom měli do vysvětlení doporučení zahrnout i vyjádření skóre metrik daných kritérií.

Otázkou je, jak přesně by měla být tato vysvětlení vizualizována. Konkrétně, jak by měl být vizualizován přínos metrik už z náhledu položky a jak detailně by mělo být doporučení vysvětleno v podrobnější explanation.

Hypotézy:

- H5.1: Uživatelé preferují detailnější vysvětlení doporučení.

2.1.6 RQ6: Jak uživatelé oceňují prvky webové aplikace?

Dalším faktorem, který ovlivňuje uživatelův názor na doporučovací systém, je prezentace jeho výsledků, tedy seznamu doporučení. Na základě posledních zjištění je navíc vhodné, aby prostředí uživatelské studie více odpovídalo prostředí reálných portálů, kde jsou doporučovací systémy využívány (viz kapitola 1.6.3). Proto zahrnujeme ve webové aplikaci i další prvky, které umožňují například jiný způsob procházení položek nebo vlastní nastavení.

Dílčím výstupem z naší práce by mělo být také, zda námi navržená prezentace náhledu položky v seznamu doporučení a její detail obsahují dostatek informací k tomu, aby uživatel dokázal posoudit, zda je pro něj položka vhodná.

Zároveň chceme od uživatelů zjistit, jestli oceňují další prvky již nesouvisející přímo s doporučovacím systémem, jako jsou textové vyhledávání, podrobnější filtr, blokace položek ať už přímo nebo na základě některé z jejich vlastností, možnost vlastního nastavení atd.

Hypotézy:

- H6.1: Uživatelé jsou schopni na základě poskytnutých informací v náhledu a detailu filmu posoudit, zda se jim doporučení líbí.
- H6.2: Uživatelé vnímají tradiční prvky objevující se na webových portálech (textové vyhledávání, podrobnější filtr, nastavení atd.) jako užitečné.
- H6.3: Uživatelé oceňují možnost blokování filmů přímo i na základě některé z vlastností (žánr, herec, režisér).

2.2 Požadavky

V této sekci zmíníme všechny základní požadavky, jaké máme na softwarové řešení a grafickou prezentaci. Jsou to konkrétně jednak požadavky na doporučovací systém, jeho natrénování a následné predikce, dále na webovou aplikaci, a to na prvky, které by měla obsahovat, a způsob jejich zobrazení. Na závěr se ještě zaměříme na to, jaké požadavky máme na data a jak by měla probíhat samotná uživatelská studie.

2.2.1 Doporučovací systém

Nejdříve představíme požadavky, které máme na doporučovací systém, a to od jeho základní specifikace přes způsob nasazení až po implementaci metrik kritérií a interpretaci jejich výstupů.

2.2.1.1 Základní specifikace doporučovacího systému

Vzhledem k tomu, že chceme používat více kritérií kvality doporučování, potřebujeme multi-objective doporučovací algoritmus. Od tohoto algoritmu vyžadujeme, aby bral v úvahu přiřazené váhy kritériím kvality doporučování a zároveň byl schopen změřená skóre pro jednotlivá kritéria vrátit.

2.2.1.2 Způsob nasazení

S ohledem na způsob nasazení, je nutné používat takový doporučovací algoritmus s takovými daty pro výpočet metrik jednotlivých kritérií kvality doporučování, aby mohl celý systém fungovat dynamicky se skutečnými uživateli, kteří jednak průběžně vznikají a jednak v průběhu specifikují své preference.

K tomu také potřebujeme, aby byl požadavek na predikci, tedy seznam doporučení, pro uživatele vystaven jako webové API, aby mohla probíhat komunikace mezi webovou aplikací a doporučovacím systémem. Zároveň je nutné zadefinovat způsob komunikace mezi nimi. Je třeba specifikovat data, která budou posílána z webové aplikace v rámci požadavku, a také data, která bude doporučovací systém vracet jako výsledek v odpovědi na požadavek.

2.2.1.3 Volba kritérií kvality doporučování a způsobu jejich měření

Následně je také nutné určit, jaká kritéria kvality doporučování a jaký způsob či způsoby, kterými je můžeme měřit, budeme používat. Je také nutné učinit rozhodnutí, na základě jakých dat se tyto metriky budou počítat. Například zda budeme brát v úvahu pouze pozitivní hodnocení od uživatele, nebo i ta negativní. Zda použijeme podobnost jednotlivých položek vypočítanou na základě kolaborativních dat nebo vlastností položky, jako mohou být žánry, do kterých patří atd.

2.2.1.4 Korektní interpretace hodnot jednotlivých metrik

Dalším problémem, který se musí řešit, je správná interpretace skóre metrik jednotlivých kritérií kvality doporučení. Pokud položka má skóre 35 pro nějakou

metriku relevance a skóre metriky diverzity 0,7, nemůžeme tyto dvě číselné hodnoty srovnávat, což je nutné pro typ algoritmu, který chceme použít. Musíme tedy hodnoty převést jednak na stejnou škálu, což lze poměrně jednoduše, ale pořád to neřeší další problém a tou je rozdílná distribuce hodnot každé metriky, což opět může činit hodnoty metrik dvou různých kritérií neporovnatelné. Zároveň je také potřeba vzít v úvahu, že hodnoty některých metrik můžou mít velmi rozdílnou distribuci s ohledem na pořadí v seznamu doporučení nebo na počet položek v profilu uživatele. Pro příklad vezměme v úvahu diverzitu na základě maximální podobnosti (viz kapitola 1.3.2.2), která bude obecně nabírat vyšších hodnot, když vybíráme položku na začátek seznamu, tedy když je seznam doporučení malý, a naopak nižší skóre diverzity na základě maximální podobnosti budou mít položky vybírané na konec seznamu. Podobný problém nastává u distance-based novelty na základě maximální podobnosti (viz kapitola 1.3.3.2) pouze s tím rozdílem, že ta není závislá na pořadí položky v seznamu ale na počtu položek v uživatelově profilu.

2.2.2 Webová aplikace

Několik požadavků je nutné specifikovat i na webovou aplikaci, a to jednak na její GUI a jednak na její implementaci.

2.2.2.1 Vizualizace seznamu doporučení

Před samotným návrhem aplikace je nutné si ujasnit, co vše by měla obsahovat z pohledu uživatele. Základem každého GUI aplikace spolupracující s doporučovacím systémem je samozřejmě zobrazení seznamu doporučení. Tento seznam by měl obsahovat náhledy jednotlivých položek se základními informacemi, které musí být dostačující k tomu, aby byl uživatel schopen posoudit, zda je pro něj položka vhodná. Musíme tedy zvolit konkrétní grafickou vizualizaci jak seznamu doporučení jako celku, tak jednotlivých náhledů položek.

2.2.2.2 Zpětná vazba

Zároveň uživatel musí mít možnost položky hodnotit, aby měl doporučovací systém k dispozici data, na základě kterých může vybírat doporučení. Na zvážení je také, zda neumožnit uživatelům ještě jiný typ zpětné vazby.

2.2.2.3 Mechanismus ke specifikaci preferencí k jednotlivým kritériím

Protože navíc potřebujeme, aby měl uživatel možnost měnit váhy kritérií kvality doporučování, je také nutné nabídnout uživateli nějaký mechanismus, pomocí něhož může důležitost jednotlivých kritérií nastavovat.

2.2.2.4 Výběr variant

Jelikož uživateli také chceme umožnit u některých z prvků vybírat z různých variant zobrazení, musíme navrhnout způsob, jakým bude uživatel jednotlivé možnosti volit a měnit. Výběr z více variant by měl být k dispozici minimálně u mechanismu filtru kritérií kvality doporučování, explanations a metriky, na základě

které počítá doporučovací systém skóre položky pro kritérium kvality doporučování.

2.2.2.5 Explanations

Protože nejzásadnějším způsobem, jakým uživatel může ovlivnit doporučování společně se svými hodnoceními je nastavení vah pro jednotlivá kritéria kvality doporučování, měla by vysvětlení vyjadřovat, jak moc doporučená položka přispívá k používané metrice kritéria.

V klasickém případě single-objective doporučování je obvyklé, že jsou doporučovány nejlepší položky s ohledem na jediné kritérium, takže je dostačující mít pouze variantu vysvětlení pozitivního přínosu k metrice kritéria. V našem případě multi-objective doporučování je očekávané, že některé doporučené položky mohou mít i nízké skóre metriky alespoň jednoho kritéria. Proto bychom měli pro každé kritérium navrhnout separátní textové vysvětlení odpovídající vysokému a nízkému skóre.

Zásadní je také způsob, jakým budou tyto explanations vizualizovány. Je potřeba navrhnout, jak detailní vysvětlení budou patrná již z náhledu položek a jak detailní explanations se objeví při fokusu na dané doporučení.

2.2.2.6 Rozšíření webové aplikace o další prvky

Je vhodné využít osvědčených postupů zmíněných v kapitole 1.6.3. Jelikož uživatelé oceňují, pokud jsou k dispozici detailnější informace o položce, měli bychom uživateli umožnit i zobrazení těchto informací. Zároveň je vhodné, aby aplikace, přestože bude využita na uživatelskou studii, více odpovídala typickým webovým portálům, v nichž jsou doporučovací systémy využívány. Tomu lze napomoci začleněním dalších prvků, jako je použití textového vyhledávání, případně podrobnějšího filtru na vlastnosti položek odvislých od domény, ve které pracujeme. U všech těchto prvků je nutné navrhnout způsob, jakým budou fungovat a jakým způsobem budou začleněny do GUI.

2.2.2.7 Nezávislost na doméně

Samotná implementace webové aplikace by měla být co nejméně navázána na doménu, ve které se pohybuje. To znamená, že pokud možno co největší část by měla být odstíněna od toho, zda jsou doporučované položky filmy, hudba, knihy, nebo zboží na e-shopu. Byť se závislosti na doméně v částech implementace nelze vyhnout.

2.2.2.8 Nezávislost na variantě multi-objective doporučovacího systému

Zároveň požadujeme, aby návrh webové aplikace byl odstíněn od varianty multi-objective doporučovacího algoritmu a jeho implementace. Je ovšem očekávano, že uživatel ovlivňuje doporučování pomocí přidělení vah kritériím kvality doporučení a algoritmus pro jednotlivé položky v seznamu doporučení vrací i skóre přes jednotlivá kritéria. Proto omezující podmínky na doporučovací systémy, které lze použít vychází z popisu očekávané komunikace mezi ním a webovou aplikací tak, jak jsou popsány v kapitole 3.2.2.

2.2.3 Data

Při uživatelské studii bude doporučovací systém doporučovat položky účastníkům studie. Pro to je ale nutné doporučovací algoritmus nejdříve natrénovat a k tomu potřebujeme data. A i na data máme několik požadavků.

2.2.3.1 Existence zpětné vazby

Abychom vůbec data mohli použít k natrénování doporučovacího systému, potřebujeme, aby obsahovala zpětnou vazbu. Každý záznam o zpětné vazbě musí obsahovat identifikaci uživatele, identifikaci položky, a pokud rozlišujeme více typů zpětné vazby (např. víceškálové hodnocení nebo různé typy interakcí), tak i její typ.

2.2.3.2 Dostatečné informace o položkách

Protože provedeme studii, kde budeme položky doporučovat skutečným uživatelům, musí data obsahovat také základní informace k položce, aby ty mohly být účastníkovi studie prezentovány v náhledu položky v seznamu doporučení. Pokud chceme, aby byl uživatel skutečně schopen správně rozpoznat položku a posoudit její užitečnost pro něj, potřebujeme o každé položce i další detailnější informace včetně obrázku položky.

2.2.3.3 Stáří dat

Pro uživatelskou studii je také nutné, aby obsahovala dostatečně nová data. Pokud bychom doporučovali mobilní telefony, pak by i použití pár let starých dat bez nových modelů výrazně snížilo schopnost doporučit vhodné položky pro uživatele. V jiných doménách, jako jsou knihy, nejsou omezení na stáří tak přísná, ale i zde platí čím novější data tím vhodnější.

2.2.4 Uživatelská studie

Výsledky našeho výzkumu budou získány pomocí uživatelské studie. Jelikož uživatelská studie může probíhat mnoha způsoby, zmíníme zde požadavky právě na tento typ vyhodnocení.

2.2.4.1 Přístup

Základním požadavkem na uživatelskou studii je to, aby byla pro účastníky přístupná. Vzhledem k tomu, že studie bude probíhat prací uživatele ve webové aplikaci, měla by tato aplikace dostupná přes internet. To nemusí platit, pokud bychom chtěli uživatelskou studii provádět jako laboratorní studii, kde by stačil přístup pouze přes zařízení, která by účastníci používali.

2.2.4.2 Identifikace účastníka

Zároveň potřebujeme zařídit, abychom po sběru výsledku byli schopni rozlišit akce daného uživatele pro získání kvalitativních i kvantitativních dat. Pro to musíme zvolit odpovídající způsob, podle kterého budeme uživatele identifikovat.

Bylo by také vhodné získat alespoň nějaký údaj od účastníka, na základě kterého můžeme v případě pochybností odhalit, zda se jednalo o skutečného uživatele, případně zda nějaký účastník nevyplnil studii vícekrát.

2.2.4.3 Způsob prezentace kandidátů

Z našich výzkumných otázek (viz kapitola 2.1) vyplývá, že ve více než jednom případě porovnáváme různé kandidáty, ať už jde o varianty explanations doporučení, metriky kritéria kvality doporučování nebo typ mechanismu pro nastavení vah. Proto si musíme pro všechny případy rozmyslet, zda každý uživatel bude muset pro jednotlivé zkoumané oblasti vyzkoušet všechny kandidáty, jejich část, nebo pouze jednoho z nich. V případě více kandidátů, musíme navrhnout, jestli se tito kandidáti budou zobrazovat účastníkovi studie zároveň nebo postupně. U první možnosti pak musíme navrhnout rozřazení kandidátů při vizualizaci, při druhém způsob změny kandidáta. Zároveň, pokud dochází ke zvýhodnění některých kandidátů (např. zobrazen jako první, zobrazen nahoře), je třeba zařídit, aby se každý kandidát na výhodnější pozici objevoval u stejného nebo alespoň podobného počtu účastníků.

2.2.4.4 Sběr dat

Před spuštěním uživatelské studie, je důležité rozhodnout, jaká kvantitativní (případně i kvalitativní) data budeme sbírat z uživatelových akcí. Potřebujeme také tyto události uspořádat na časové ose. Takto získaná data nám mohou pomoci s lepší interpretací kvalitativních dat získaných z dotazníku a tedy s přesnější odpovědí na výzkumné otázky (viz kapitola 2.1).

2.2.4.5 Dotazník

Zásadní odpovědi pro náš výzkum získáme z dotazníku v rámci uživatelské studie. Je nutné navrhnout všechny otázky a možné odpovědi tak, abychom z nich byli schopni odpovědět na námi definované výzkumné otázky (viz kapitola 2.1). Zároveň je nutné pokládat otázky tak, aby uživatelé necítili, že jsou do nějaké z odpovědí tlačeni.

3. Řešení

V této kapitole popíšeme, jaká data jsme zvolili, komunikaci webové aplikace a doporučovacího systému, jaké prvky obsahuje uživatelské rozhraní webové aplikace a co jsme změnili či přidali do původní implementace (viz kapitola 1.4.1.1) doporučovacího systému.

3.1 Data

Pro potřeby natrénování doporučovacího systému potřebujeme vhodný dataset. Protože necílíme v uživatelské studii na určitý typ uživatelů, je vhodné zvolit doménu, ve které se lidé obecně vyznají. Zároveň, vzhledem k definicím některých metrik závisejících přímo na preferencích uživatele, je vhodné použít doménu s vysoce subjektivní zpětnou vazbou.

Proto jsme volili zřejmě nejpoužívanější dataset pro výzkum doporučovacích systémů MovieLens 25M Dataset, který podrobněji představujeme v kapitole 1.8. Abychom mohli uživateli zobrazit dostatečný počet informací pro to, aby dokázal posoudit, zda je pro něj doporučený film zajímavý, obohatili jsme data z MovieLens o informace z The Movie Database (TMDB, 2023).

3.1.1 MovieLens 25M Dataset

MovieLens 25M Dataset byl vydán na konci roku 2019. Tento dataset, jak název napovídá, obsahuje data z domény filmů a skládá se z několika souborů. Ty zahrnují následující data, která používáme:

- Hodnocení
  - ID uživatele
  - ID filmu
  - Skóre hodnocení
  - Časová značka
- Filmy
  - ID filmu
  - Název
  - Seznam žánrů
- Odkazy
  - ID filmu
  - ID filmu v IMDB
  - ID filmu v TMDB

Mimo tato data obsahuje dataset ještě soubor s tagy a další soubor, kde jsou tyto tagy napojeny na filmy. Tagy ovšem v našem případě nepoužíváme.

3.1.1.1 Filtrace

Vzhledem k velkému počtu dat s cca 25 miliony hodnocení od 163 tisíc uživatelů pro 62 tisíc filmů, bylo nutné zmenšit velikost datasetu. A to jednak z časové i paměťové náročnosti samotného natrénování celého algoritmu, ale především k následné schopnosti doporučovat v uživatelsky přijatelném čase. V následujících řádcích si popíšeme všechny použité filtrace. Konkrétní hodnoty byly zvoleny podle toho, abychom se dostali na počet 1500 až 2000 filmů, který se ukázal jako vhodný pro dostatečně rychlé doporučování.

3.1.1.1.1 Stáří filmů

Nejdříve jsme se zbavili všech filmů starších než je rok 1990. Tímto prvním krokem jsme chtěli zajistit, aby se uživatelům zobrazovaly filmy, které by pro ně mohli být relevantní i v dnešní době, což pro většinu starších snímků mimo ty nejpopulárnější už nemusí platit.

3.1.1.1.2 Stáří hodnocení

Následně jsme odstranili i všechna hodnocení starší než z roku 2010. Zde jsme vycházeli z toho, že globální preference se v čase mohou měnit a novější hodnocení více odpovídají preferencím uživatelů v současnosti.

3.1.1.1.3 Počet hodnocení filmů

Zároveň jsme zanechali pouze ty filmy, které v průměru od roku vydání byly ohodnoceny alespoň 75 uživateli za rok. To vyřadilo minimálně známé filmy, jako jsou snímky známé spíše v jednotlivých zemích vzniku než mezinárodně.

3.1.1.1.4 Počet hodnocení od uživatelů

Také byli z datasetu odebráni všichni uživatelé, kteří ohodnotili méně než 100 filmů. Tato filtrace zanechala v datasetu jen ty uživatele s dostatečným počtem hodnocení, na nichž jsme mohli dobře natrénovat doporučovací algoritmus.

3.1.1.1.5 Finální filtrace

Na závěr jsme již jen odstranili hodnocení filmů vyřazených z datasetu. Stejně tak jsme postupovali u hodnocení od uživatelů odebraných z datasetu. Ve výsledném vyfiltrovaném datasetu zůstalo 5 187 037 hodnocení, 20 962 uživatelů a 1 728 filmů.

3.1.2 The Movie Database

Díky odkazům v MovieLens datasetu jsme získali napojení MovieLens ID filmu na jeho ID v The Movie Database. Data jsme se ještě rozhodli obohatit o tyto informace z The Movie Database pomocí jejího webového API (TMDB, 2023):

- Obrázek
- Režisér
- Herci
- Datum vydání

- Odkaz na trailer na platformě YouTube

Cílem tohoto obohacení bylo zobrazení dostatečných informací pro uživatele, aby mohl posoudit, zda je pro něj doporučený film vhodný.

3.2 Komunikace webové aplikace a doporučovacího systému

Implementaci doporučovacího systému jsme museli upravit tak, aby mohla být použita v dynamickém prostředí, kdy bude vracet seznam doporučení na základě požadavků skutečných uživatelů. Zároveň jsme chtěli vyzkoušet i jiná kritéria a různé varianty metrik u těch již použitých. Nejdříve jsme určili, jaké informace o doporučovacím systému by měla mít webová aplikace k dispozici, co by mělo být součástí požadavků na doporučovací systém a co by měl naopak vracet.

3.2.1 Parametry doporučovacího systému

Ještě před nastavením komunikace je nutné záznam o doporučovacím systému a některých jeho parametrech uložit do databáze webové aplikace. Základními informacemi je jméno doporučovacího systému a jeho URI.

Jelikož je celý projekt implementován pro práci s multi-objective doporučovacími systémy je také nutné specifikovat kritéria, se kterými pracuje, a pro ně také čtyři škály textového vysvětlení odpovídající od velmi pozitivního skóre vzhledem k metrice kritéria po velmi negativní. Případně lze vložit ke každému kritériu ještě jeho popis a příklad použití pro lepší pochopení kritéria, resp. jeho metriky, uživatelem. Řešení explanations podrobněji probíráme později v kapitole 3.4.2.

Pokud doporučovací systém umožňuje skóre některých kritérií počítat různým způsobem, je navíc nutné specifikovat varianty metrik. Každá varianta by navíc měla mít unikátní kód, který musí být správně interpretován jak webovou aplikací tak doporučovacím systémem. I zde by měly být specifikovány 4 možné explanations pro každou variantu, které jsou pak zobrazovány místo textů specifikovaných u samotného kritéria.

3.2.2 Komunikace

Předávané proměnné při komunikaci webové aplikace a doporučovacího systému, které jsou vypsané v tabulce 3.1, popíšeme podrobněji.

3.2.2.1 Struktura požadavku na doporučení

Nejdříve se podíváme na to, co je předáno doporučovacímu systému při požadavku na seznam doporučení. Nutností pro personalizovaná doporučení je identifikace uživatele, který požadavek posílá. Dále na základě možností, jaké má uživatel ve webové aplikaci, se předávají i tři seznamy položek. Jde o povolené položky, což jsou ty, které splňují uživatelovo textové vyhledávání, popřípadě ním zadané filtrování. V případě, že uživatel nic nespecifikoval, je tato proměnná prázdná a povolené jsou všechny objekty. Druhým seznamem jsou zakázané položky, což jsou jednak ty vypadávající na základě explicitních blokujících pravidel, ale také

Požadavek	Odpověď
ID uživatele	Doporučené položky
Povolené položky	Skóre každé položky pro každé kritérium
Zakázané položky	
Položky již přítomné v seznamu	
Počet doporučení	
Váhy kritérií kvality doporučování	
Varianty metrik	

Tabulka 3.1: Data posílaná při komunikaci mezi webovou aplikací a doporučovacím systémem

mezi ně patří již ohodnocené nebo v nedávné době viděné položky. Tím posledním jsou položky, které již jsou zobrazeny, což je použito, pokud uživatel požádá o načtení dalších položek k již dopoučenému seznamu, tzn. úkon „Doporučit další“, ne základní úkon „Doporučit“.

Zároveň je součástí požadavku informace, kolik položek má doporučovací systém vrátit. Jelikož jedním z našich hlavních cílů práce je zjistit, zda uživatelé stojí o nastavení svých preferencí přes jednotlivá kritéria kvality doporučování tak, aby mohli ovlivnit výstup algoritmu RL-Prop, předávají se v rámci požadavku i uživatelem specifikované váhy kritérií. Poslední složkou jsou varianty jednotlivých metrik. Na základě nich je vybrán způsob, podle něhož se počítá skóre položky pro jednotlivá kritéria.

3.2.2.2 Struktura odpovědi doporučovacího systému

Hlavním výstupem doporučovacího systému je seznam doporučených položek, který zobrazíme uživateli ve webové aplikaci. Jelikož se zaměřujeme i na vysvětlení doporučení na základě jejich skóre jednotlivých metrik, vrací doporučovací systém i tato skóre pro každou položku přes všechna používaná kritéria.

3.3 Doporučovací systém

K dispozici jsme měli naimplementovaný doporučovací algoritmus RL-Prop v jazyce python tak, jak byl použit pro práci Dokoupila, Pešky a Boratta (Dokoupil a kol., 2023a). Jádro tohoto funkčního kódu jsme zachovali neměnné, bylo ale nutné změnit způsob spuštění aplikace. Původní implementace se spouštěla skriptem, který provedl natrénování algoritmu, predikci doporučení pro všechny uživatele a změření výsledků.

Vzhledem k tomu, že potřebujeme, aby byl systém volán jako webová služba, použili jsme webový framework Flask. V zásadě jedinou dostupnou metodu, kterou od doporučovacího systému vyžadujeme, je výpočet seznamu doporučení pro konkrétního uživatele. Pro to, aby algoritmus RL-Prop mohl uživateli vybrat vhodné položky k doporučení na základě více metrik, je nutné nejdřív celý algoritmus natrénovat. Natrénování spouštíme před zpracováním prvního požadavku na seznam doporučení a poté pravidelně jednou denně.

3.3.1 Seznam metrik

Nyní vypíšeme používaná kritéria kvality doporučování a jejich varianty metrik (viz kapitola 1.3). Tučně jsou označena kritéria a metriky, které nebyly součástí původní implementace RL-Prop (viz kapitola 1.4.1.1). Průběh výpočtu metrik jednotlivých kritérií popisujeme v následujících kapitolách 3.3.2 - 3.3.6.

- Relevance
  - $\boldsymbol{EASE_{POS}}$
  - $\boldsymbol{EASE_{NEG}}$
- Diverzita
  - Intra-list diverzita
  - **Diverzita na základě maximální podobnosti**
  - **Binomická diverzita**
- Novelty
  - očekávaný doplněk popularity
  - **Distance-based novelty na základě maximální podobnosti**
  - **Intra-list distance-based novelty**
- **Popularita**
  - **Popularita dle známosti**
  - **Popularita na základě hodnocení**
- **Kalibrace**
  - **Kalibrace dle Stecka (Steck, 2018)**

3.3.2 Výpočet relevance

Hlavní logika jak natrénování tak následného výběru doporučení je realizována pomocí úpravy hlavního skriptu původní implementace. Nejdříve dochází k výpočtu matice predikcí. Způsob výpočtu probíhal pomocí faktorizace matic, implementovanou variantu tohoto algoritmu ale nebylo možné použít v dynamickém prostředí (noví uživatelé, nová hodnocení), proto byla prvním návrhem volba jiné implementace faktorizace matic, která by si s těmito změnami poradila. Přestože se s takovou variantou pracovalo v současně běžící studii autorů algoritmu, změnili jsme algoritmus pro výpočet matice predikcí na EASE (viz kapitola 1.2). Důvodem pro toto rozhodnutí byly jednak velmi dobré výsledky tohoto algoritmu výrazně překonávající faktorizaci matic a v některých doménách dosahující téměř state-of-the-art doporučování. Tím druhým byla výrazně menší početní náročnost při výpočtu predikcí na základě uživatelových hodnoceních při dotazu na jeho seznam doporučení oproti jiným kvalitním doporučovacím, avšak výrazně složitějším algoritmům. Právě predikce skóre na základě EASE slouží jako výpočet skóre pro relevanci.

Používáme dvě různé varianty EASE. Ta první pracuje pouze s pozitivními hodnoceními uživatele, ta druhá používá i negativní hodnocení. Jediný rozdíl v obou implementacích jsou hodnoty, které jsou ve vstupní matici X. Variantu, kdy uvažujeme pouze pozitivní hodnocení označíme jako $EASE_{POS}$ a variantu, která bere v úvahu i ty negativní, označíme jako $EASE_{NEG}$. Maximální hodnocení označíme jako R_{max}. Hodnota $X_{u,i}$ je určena na základě skutečného hodnocení uživatele u položky i $R_{u,i}$ takto:

$$X_{u,i} = \begin{cases} \frac{R_{u,i}}{R_{max}} & \text{if } (EASE_{POS} \ \& \ R_{u,i} > (\frac{R_{max}}{2})) \\ 0 & \text{if } (EASE_{POS} \ \& \ R_{u,i} \leq (\frac{R_{max}}{2}) \\ ((R_{u,i} \ - \ (\frac{R_{max}}{2})) * 2) \ / \ R_{max}) & \text{if } (EASE_{NEG} \ \& \ R_{u,i} \neq (\frac{R_{max}}{2}) \\ 0.01 \ / \ R_{max} & \text{if } (EASE_{NEG} \ \& \ R_{u,i} = (\frac{R_{max}}{2})) \\ 0 & \text{if } R_{u,i} \text{ neexistuje} \end{cases}$$

Jak můžeme vidět, každou hodnotu dělíme R_{max}, což zajišťuje naškálování hodnot v matici na interval $[0; 1]$, resp. $[-1; 1]$. V případě $EASE_{POS}$ varianty zachováváme jen vyšší hodnoty, než je poloviční hodnocení. V případě $EASE_{NEG}$ odečítáme od $R_{u,i}$ poloviční hodnocení, abychom převedli nižší hodnocení na negativní hodnoty. Následné vynásobení 2 zvětšuje rozdíl mezi pozitivními a negativními hodnoceními. Pro hodnocení rovné (R_{max} / 2) zapisujeme velmi malou hodnotu, abychom byli schopni na základě matice X oddělit neutrální hodnocení od těch neexistujících, což je využíváno pro výpočet metrik jiných kritérií.

3.3.3 Výpočet diverzity

Již naimplementovaná byla intra-list diverzita (viz kapitola 1.3.2.1). Ta pro svůj výpočet potřebuje matici podobnosti mezi položkami SIM, resp. matici jejich vzájemné vzdálenosti $DIST(= (1 \ - \ SIM))$. Matici SIM získávají autoři implementace na základě kosinové podobnosti mezi sloupcovými vektory matice predikcí vypočtené pro relevanci. V implementaci je také obsažena varianta výpočtu $DIST$ na základě metadat, konkrétně na základě žánrů filmů. Stejně jako autoři využíváme ve studii pouze první variantu, tedy kolaborativní vzdálenost. Je také potřeba zmínit, že máme dvě varianty matice predikcí na základě $EASE_{POS}$ a $EASE_{NEG}$, a tím pádem i dvě varianty kolaborativní $DIST$. Jednou z možností je, používat matici vzdálenosti na základě zvolené varianty relevance, což by ale vedlo k nežádoucímu efektu, kdy výběr varianty relevance má vliv na výpočet diverzity. Proto jsme stejně jako autoři implementace použili matici predikcí vypočtenou na základě pouze pozitivních hodnocení.

I pro diverzitu jsme chtěli přidat další její varianty. Implementovali jsme diverzitu na základě maximální podobnosti (viz kapitola 1.3.2.2) počítající na základě stejné matice vzdáleností $DIST$, jako je tomu u intra-list diverzity. Skóre diverzity na základě maximální podobnosti u první položky v seznamu doporučení bylo vypočítáno na základě průměrné rozdílnosti od všech dalších položek, abychom vyřešili problém, kdy první položku není s čím porovnávat. Jde o stejný způsob, jako je původně v implementaci u intra-list diverzity.

Výrazně rozdílnější metrikou je binomická diverzita (viz kapitola 1.3.2.3). K výpočtu této metriky získáváme z metadat seznam všech žánrů a napojení žánrů na jednotlivé filmy. Následně počítáme pravděpodobnosti toho, že se u pozitivně

hodnoceného filmu vyskytuje žánr na základě pozitivních hodnocení všech uživatelů. Personalizované pravděpodobnosti žánrů jsou spočteny také pro každého uživatele. Výpočet těchto pravděpodobností odpovídá definici p'_g a p''_g (viz 1.3.2.3). Následně je implementován celý výpočet binomické diverzity tak, jak jsme jej zadefinovali v kapitole 1.3.2.3. Binomická diverzita je ovšem zadefinována pouze pro celý seznam doporučení, zatímco my potřebujeme hodnotu binomické diverzity pro všechny kandidáty, z kterých vybíráme další položku v seznamu. Tuto hledanou hodnotu jsme zadefinovali jako rozdíl binomické diverzity seznamu doporučení s kandidátem i a současného seznamu doporučení R bez něj, pak tedy:

$$div(i|R) = BinDiv(R \cup \{i\}) \; - \; BinDiv(R)$$

Navíc musíme opět řešit problém první položky v seznamu. Zde místo binomické diverzity současného seznamu doporučení, který je prázdný, odečítáme nejnižší hodnotu napříč všemi kandidáty:

$$div(i|\emptyset) = BinDiv(\{i\}) \; - \; \min_{j \in I} BinDiv(\{j\})$$

kde I je množina všech kandidátů k doporučení. Vždy tak počítáme přímo přidanou hodnotu kandidáta pro binomickou diverzitu celého seznamu doporučení.

3.3.4 Výpočet novelty

Již implementovanou metrikou novelty byl očekávaný doplněk popularity (viz kapitola 1.3.3.1). Počet hodnocení každého filmu jsme získali na základě nenulových hodnot ve sloupcích matice X z $EASE_{NEG}$ varianty algoritmu EASE, protože ta na rozdíl od X z $EASE_{POS}$ obsahuje všechna uživatelova hodnocení. Můžeme tak získat všechny položky, o kterých víme, že je zná. Počet hodnocení byl následně vydělen počtem uživatelů a poté ještě odečten od hodnoty 1, čímž jsme pro každou položku získali skóre novelty na základě očekávaného doplňku popularity.

Protože jsme chtěli přidat i nějakou personalizovanou variantu novelty, implementovali jsme ještě dvě metriky vycházející z obdobných metrik u diverzity. Tou první je intra-list distance-based novelty (viz kapitola 1.3.3.3). I zde využíváme stejně napočítanou matici vzdáleností mezi položkami $DIST$ jako u variant diverzity. Rozdíl oproti intra-list diverzitě spočívá pouze v tom, že místo průměrné vzdálenosti kandidátů od položek v nedokončeném seznamu doporučení, počítáme průměrnou vzdálenost kandidáta k doporučení od seznamu položek, které ohodnotil. Tento seznam, který nazýváme profilem uživatele, získáme z jeho vektoru v matici X z $EASE_{NEG}$ varianty algoritmu EASE.

Druhou přidanou variantou je distance-based novelty na základě maximální podobnosti (viz kapitola 1.3.3.2). Stejným způsobem jako u intra-list distance-based novelty získáme matici vzdáleností mezi položkami $DIST$ a profil uživatele, tedy seznam ním hodnocených položek. Jediným rozdílem oproti předchozí metrice je, že místo měření průměrné vzdálenosti počítáme minimální vzdálenost kandidáta k doporučení od položek v uživatelově profilu.

U obou posledních variant jsme řešili problém prázdného profilu uživatele přiřazením hodnoty 1 pro všechny položky. V reálném použití je ale stejně nutné pro funkčnost algoritmu, aby uživatel alespoň nějaké položky ohodnotil.

3.3.5 Výpočet popularity

Nově přidaným kritériem kvality doporučování, které nebylo součástí původní implementace je popularita. Pro toto kritérium jsme implementovali dvě různé metriky.

První metrikou je popularita dle známosti (viz kapitola 1.3.4.1). Jelikož jde o opačnou metriku k metrice novelty očekávanému doplňku popularity, postupuje výpočet stejně jako u ní. I zde je počet hodnocení každého filmu získán z počtu nenulových hodnot ve sloupcích matice X z $EASE_{NEG}$ varianty algoritmu EASE. Volba varianty EASE má stejný důvod jako u novelty, kterým je fakt, že X z $EASE_{NEG}$ obsahuje všechna hodnocení uživatelů, tzn. všechny položky, které uživatel zná. Hodnota této metriky pro položku vypočtena vydělením počtu hodnocení počtem uživatelů.

Druhou variantou popularity je popularita na základě hodnocení (viz kapitola 1.3.4.2). Průměrné hodnocení každé položky je získáno přímo dotazem do databáze. A právě průměrné hodnocení je hodnotou této metriky pro jednotlivé kandidáty k doporučení. Problém malého počtu hodnocení jsme neřešili, protože v databázi jsou pouze filmy s dostatečně vysokým počtem hodnocení (75 a více).

3.3.6 Výpočet kalibrace

Posledním přidaným kritériem kvality doporučování je kalibrace (viz kapitola 1.3.5). Implementace vychází z definice kalibrace od Stecka (Steck, 2018). K výpočtu této metriky nejdříve získáváme z metadat seznam všech žánrů a napojení žánrů na jednotlivé filmy. Následně počítáme pro každého uživatele u personalizované pravděpodobnosti $p(g|u)$ toho, že se u pozitivně hodnoceného filmu uživatelem vyskytuje daný žánr g na základě jeho předchozích pozitivních hodnocení. Pouze pozitivní hodnocení používáme proto, jelikož lépe reprezentují uživatelovy zájmy, resp. jeho pozitivní vztah k žánrům, na jehož základě chceme kalibraci počítat. Pro výpočet $p(g|u)$ používáme vzoreček zmíněný v kapitole 1.3.5, tedy: $p(g|u) = \frac{\sum_{i \in R} w_{u,i} * (p(g|i)}{\sum_{i \in R} w_{u,i}}$, kde R je množina položek, které uživatel pozitivně hodnotil, $w_{u,i}$ je váha hodnocení a hodnota $p(g|i)$ je rovna 1, pokud film i patří do žánru a 0 jinak. Co se týče vah hodnocení $w_{u,i}$, použili jsme nejjednodušší řešení, kdy má každé pozitivní hodnocení stejnou váhu, a to hodnotu 1.

Jak zmiňujeme v kapitole 1.3.5, pouze na základě $p(g|u)$ jsou při doporučování na základě kalibrace vyřazeni všichni kandidáti, kteří patří jen k žánrům, které se v uživatelově profilu neobjevily. Toto uzavření uživatele v bublině je problém při doporučování, kde je jediným kritériem kalibrace, případně je použita pouze v kombinaci s relevancí. Vzhledem k tomu, že náš multi-objective doporučovací systém obsahuje i kritéria diverzitu a novelty, které nepřímo zařizují doporučení kandidátů patřících do žánrů mimo profil uživatele, nebylo nutné se na něj zaměřit přímo v implementaci kalibrace.

Pro výpočet kalibrace potřebujeme také rozdělení pravděpodobnosti žánrů mezi položkami v seznamu doporučení $q(g|u) = \frac{\sum_{i \in I} w_{rank(i)} * (p(g|i)}{\sum_{i \in Rec} w_{rank(i)}}$, kde Rec je množina doporučených položek a $w_{rank(i)}$ váha položky odvislá od pořadí v seznamu doporučení. I zde volíme nejjednodušší řešení, kde $w_{rank(i)}$ je vždy roven 1, a to z důvodu, že pořadí doporučení nebereme v úvahu ani při výpočtu metrik ostatních kritérií kvality doporučování. Pro výpočet $q'(g|u)$ používáme stejně jako

autor metriky $\alpha = 0.01$.

Přestože by šlo použít více variant kalibrace, případně upravit tu používanou komplikovanějšími váhami $w_{u,i}$ a $w_{rank(i)}$, implementujeme pouze popsaný způsob výpočtu metriky pro kalibraci. A to z toho důvodu, že kalibrace je poměrně složitý koncept, který nemusí být pro uživatele zcela pochopitelný.

Stejně jako u implementace binomické diverzity musíme vyřešit to, že námi používaná metrika pro kalibraci se počítá pro celý seznam doporučení, zatímco algoritmus RL-Prop (viz alg. 2) potřebuje ohodnocovat jednotlivé kandidáty k doporučení při každém kroku přidání položky do seznamu. I zde tento problém řešíme tím, že počítáme de facto přidanou hodnotu položky pro tuto metriku. Zároveň potřebujeme vyřešit to, že všechny zmíněné metriky se snažíme maximalizovat, zatímco tato metrika oceňuje nejlepší možnou kalibraci hodnotou 0 a s horší kalibrací hodnota roste. Proto definujeme kalibraci položky takto:

$$C(i|Rec) \;=\; (-\,C_{KL}(p,q|Rec \cup \{i\})) \;-\; (-\,C_{KL}(p,q|Rec))$$

kde Rec je množina doporučených položek. Stejným způsobem jako u binomické diverzity řešíme i kalibraci při výběru první položky do seznamu doporučení:

$$C(i|\emptyset) \;=\; (-\,C_{KL}(p,q|\{i\})) \;-\; \min_{j \in I}(-\,C_{KL}(p,q|\{j\}))$$

kde I je množina všech kandidátů k doporučení.

3.3.7 Normalizace

Jak jsme již zmiňovali výše, zásadním problémem je interpretace hodnot různých metrik různých kritérií kvality doporučování. Algoritmus RL-Prop (viz kapitola 1.4.1.1) totiž tato skóre různých kritérií porovnává a podle toho vybírá doporučenou položku. Proto potřebujeme měřit kvalitu jednotlivých kritérií na stejné škále a ideálně se stejnou distribucí hodnot.

Tento problém je řešen již v původní implementaci, kde je použita normalizace pomocí empirické distribuční funkce. Tento typ normalizace zajišťuje, že k-tá nejlepší položka vzhledem k metrice m_1 má stejné skóre m_1, jako je skóre m_2 k-té nejlepší položky vzhledem k metrice m_2.

Při offline vyhodnocování v původní studii Pešky a Dokoupila (Peška a Dokoupil, 2022) šlo natrénovat namapování na základě empirické distribuční funkce přímo na hodnoty metrik kritérií kvality doporučování, které budou v průběhu experimentu položky nabývat. V našem dynamickém experimentu pomocí uživatelské studie je tato přesnost nemožná, pokud netrénujeme novou normalizaci pro každého nového uživatele a také při každé změně hodnocení uživatelem, což by bylo vzhledem k času natrénování normalizace nevýhodné a i tak stále nepoužitelné pro některé z nově přidaných metrik. Normalizaci tak musíme natrénovat pouze na odhadovaných hodnotách. Rozdílem také je, že zatímco pro offline vyhodnocování byla normalizace připravena na doporučování všem uživatelům zároveň, v našem případě se doporučení počítá vždy jen pro jednoho uživatele. Proto bylo nutné způsob trénování normalizace upravit.

3.3.7.1 První návrh natrénování normalizace

Nejdříve jsme oddělili metriky, jejichž skóre nezávisí na konkrétních uživatelích, kterým je doporučováno. To jsou očekávaný doplněk popularity pro no-

velty a obě varianty popularity - popularita dle známosti a popularita na základě hodnocení. Normalizaci možných hodnot těchto metrik jsme mohli natrénovat na skutečných hodnotách, které budou nabývat při doporučování. Samozřejmě nová, případně změněná hodnoceními mají vliv na přesné hodnoty, na druhou stranu při počtu nových uživatelů v jednotkách až malých desítkách neočekáváme, že by tyto změny byly zásadní. Nové hodnoty těchto metrik pro jednotlivé položky a jejich normalizaci tak stačí natrénovat pouze jednou za určité období na základě poměru původních a nových hodnocení, která ještě nebyla součástí posledního natrénovaní.

Následně jsme pro všechny zbylé metriky kritérií kvality doporučování vytvořili odhadované hodnoty na základě následujícího postupu. Vybrali jsme náhodně vzorek 100 uživatelů, pro ty jsme simulovali doporučování. Tzn. u diverzity a kalibrace postupně vkládáme položky do seznamu doporučení a vždy počítáme skóre kandidátů vzhledem k metrikám, u novelty a relevance počítáme hodnoty metrik pouze jednou, jelikož se skóre metriky nemění vzhledem k průběžnému seznamu doporučení. Průběžný seznam doporučení jsme naplňovali na základě pseudodoporučování, kdy s určitou pravděpodobností vkládáme položku s nejvyšší hodnotou metriky a náhodnou položku jinak. Pro kalibraci a diverzitu jsme si tedy ukládali hodnoty všech kandidátů při výběru položky na každé místo v seznamu doporučení. A následně jsme náhodně vybrali vzorek z těchto hodnot, jehož velikost odpovídala počtu položek, tedy počtu kandidátů, a na těchto hodnotách

Metrika	**Závislost na pořadí v seznamu doporučení**	**Závislost na počtu již hodnocených položek**
$EASE_{POS}$	NE	**ANO**
$EASE_{NEG}$	NE	**ANO**
Intra-list diverzita	**ANO**	NE
Diverzita na základě maximální podobnosti	**ANO**	NE
Binomická diverzita	**ANO**	NE
očekávaný doplněk popularity	NE	NE
Intra-list distance-based novelty	NE	**ANO**
Distance-based novelty na základě maximální podobnosti	NE	**ANO**
Popularita dle známosti	NE	NE
Popularita na základě hodnocení	NE	NE
Kalibrace	**ANO**	ANO

Tabulka 3.2: Metriky a jejich závislost na pořadí v seznamu doporučení a na počtu již hodnocených položek uživatelem. Tučně jsou označeny problémové závislosti vzhledem k návrhu úpravy normalizace $NORM_1$.

Algorithm 3 Pseudokód trénování normalizace $NORM_1$ pro metriku diverzity

Vstup: množina uživatelů U, seznam položek k doporučení *candidates*, počet hodnot použitých pro natrénování normalizace *len_train*, velikost seznamu doporučení k
Výstup: natrénovaná normalizace *norm_diversity*

```
/*Náhodný výběr vzorku uživatelů*/
users = random_sample(U, 100)
diversity_data_points = []
for u ∈ users do
    recommendation_list = []
    candidates_for_user = candidates
    for i ∈ [1,...,k] do
        diversity_values_of_candidates = diversity_metric(
            recommendation_list, candidates_for_user)
        diversity_data_points.add(diversity_values_of_candidates)
        /*Simulace výběru doporučení*/
        recommendation_list[i] = pseudo_recommend()
        candidates_for_user.remove(recommendation_list[i])
    end for
end for
/*Náhodný výběr vzorku o velikosti len_train z diversity_data_points */
data_points = random_sample(diversity_data_points, len_train)
norm_diversity = normalization.train(data_points)
return norm_diversity
```

jsme natrénovali normalizaci pomocí empirické distribuční funkce. Tento typ natrénovaní normalizace označíme jako $NORM_1$ a jeho postup u metrik diverzity je algoritmicky vyjádřen v alg. 3.

Tento první návrh úpravy normalizace byl sice vhodný pro náš problém ve smyslu proveditelnosti, ale normalizované hodnoty skóre jednotlivých metrik nabývaly v některých případech příliš nízké nebo příliš vysoké hodnoty. To bylo způsobeno závislostí některých metrik na pořadí vybírané položky do seznamu doporučení nebo na počtu již hodnocených položek uživatelem.

V tabulce 3.2 můžeme vidět vztah metrik s oběma možnými problémovými závislostmi. Obě hodnoty relevance predikované na základě EASE jsou závislé na počtu ohodnocených filmů uživatele. Obecně tím, že čím vyšší součet má uživatelův vektor v matici X, který odpovídá hodnocením uživatelem, tím vyšší hodnoty budou mít i očekávané predikce. Pro naše použití navíc nedoporučujeme již ohodnocené filmy, proto můžeme očekávat i to, že s vyšším počtem hodnocení se zároveň snižuje počet relevantních filmů, které uživatel dosud nehodnotil, což vede k nižším predikovaným hodnotám za předpokladu, že bereme v úvahu pouze dosud nehodnocené filmy.

Co se týče variant diverzity, všechny jsou závislé na pořadí právě vybírané položky do seznamu doporučení. Nejmenší závislost lze očekávat u intra-list diverzity, která se počítá z průměru rozdílnosti s předchozími položkami v seznamu. Přesto se dá očekávat, že distribuce očekávaných hodnot bude o něco jiná při výběru například druhé položky a desáté. Největší závislost očekáváme u diverzity na základě maximální podobnosti, kdy s narůstající velikostí seznamu doporučení očekávané hodnoty rozdílnosti od nejpodobnější předchozí položky v seznamu klesají. Závislost na pořadí pozorujeme i u binomické diverzity. Vzhledem k tomu, že je výpočet této metriky závislý na pokrytí žánry a jejich neredundanci, dá se očekávat, že položky s vyšším pořadím už budou k binomické diverzitě přispívat o něco méně, přestože je tento problém částečně řešen již definicí této metriky.

U intra-list distance-based novelty a distance-based novelty na základě maximální podobnosti platí stejná odůvodnění závislosti jako u intra-list diverzity a diverzity na základě maximální podobnosti. Pouze s tím rozdílem, že tyto metriky nejsou závislé na pořadí vybírané položky v seznamu doporučení, ale na počtu položek v profilu uživatele.

Kalibrace tak, jak ji počítáme, měří, jak moc se distribuce žánrů v profilu uživatele liší od distribuce žánrů v průběžném seznamu doporučení. Z toho můžeme vypozorovat, že distribuce žánrů v profilu uživatele je závislá na počtu ohodnocených položek a distribuce žánrů v průběžném seznamu doporučení je závislá na velikosti průběžného seznamu doporučení, tedy na pořadí vybírané položky.

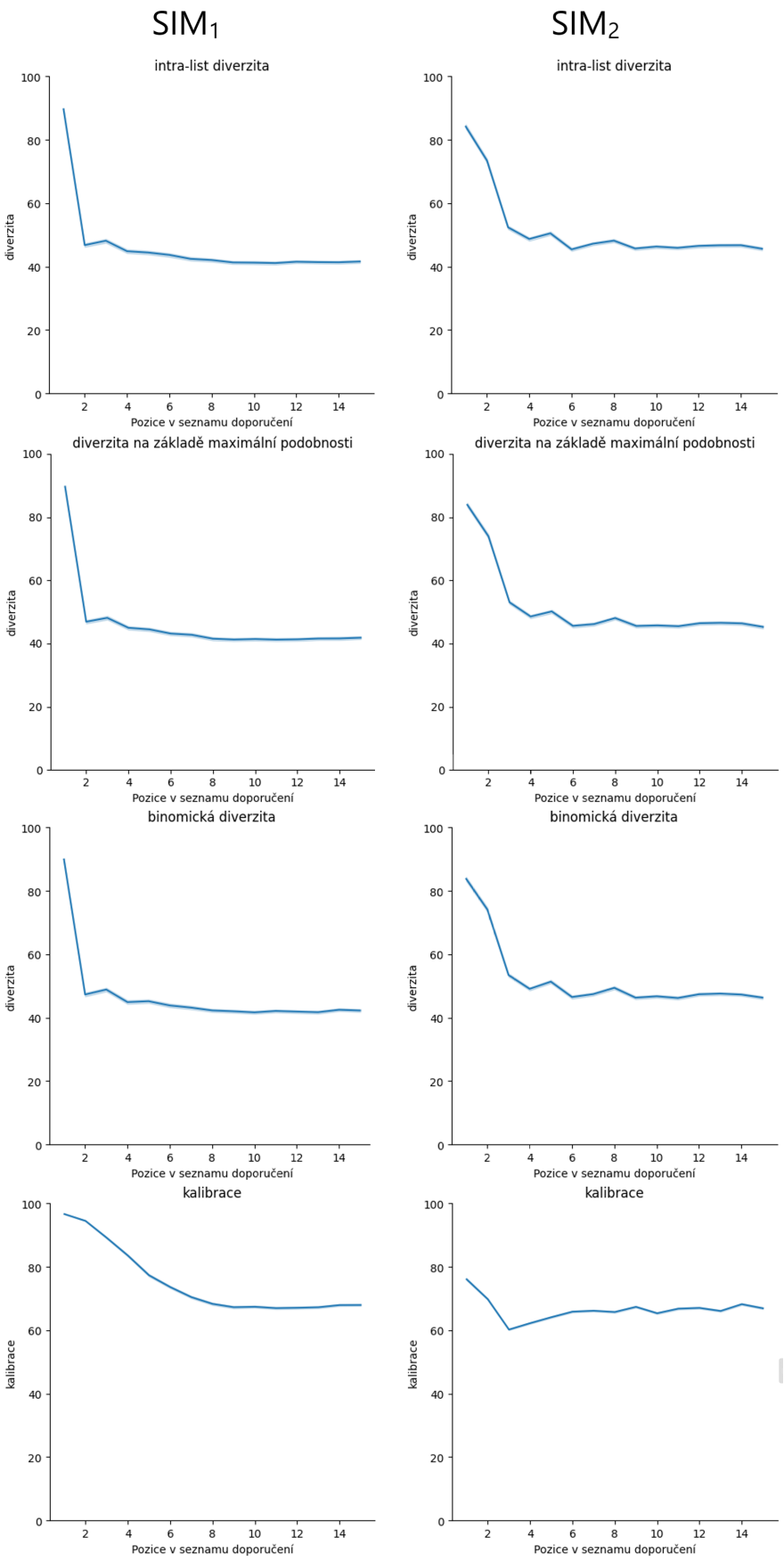
3.3.7.2 Simulace doporučování při použití normalizace $\mathbf{NORM_1}$

Všechny výše zmíněné závislosti vychází z definice jednotlivých metrik, nevíme ale, zda je skutečně nutné je všechny řešit. Proto jsme se rozhodli simulovat doporučování pro již nahrané uživatele z MovieLens a následně tuto závislost vizualizovat na grafech, na kterých budeme moc zpozorovat případnou problémovou závislost.

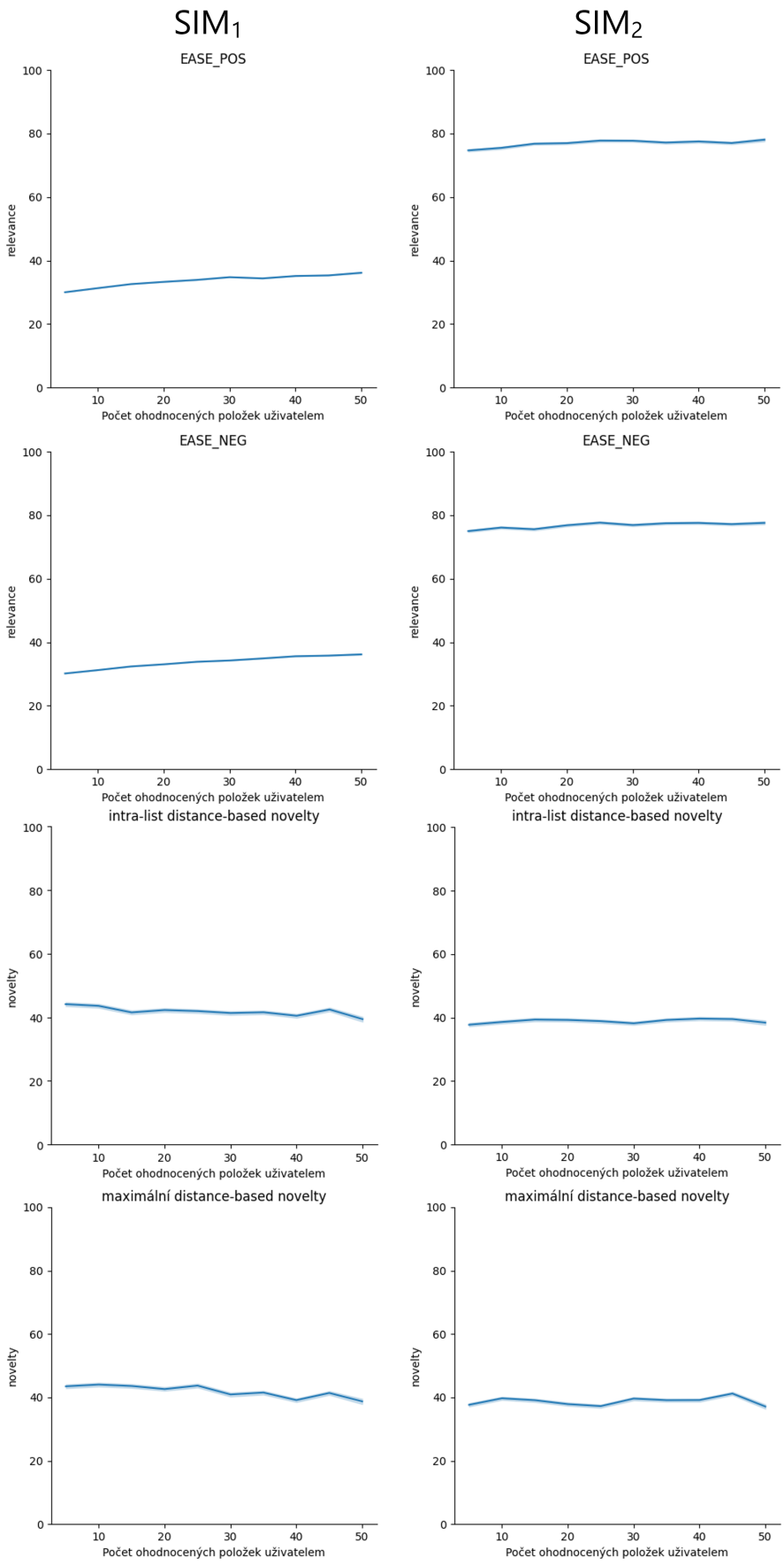
Tuto simulaci podrobně popíšeme. Z MovieLens dat jsme vybrali náhodně 1000 uživatelů. Protože tito uživatelé, které máme v databázi, mají vždy minimálně 100 hodnocení, volili jsme také náhodně počet nejstarších hodnocení,

které budeme brát v úvahu. Tento náhodný počet se pohyboval mezi 3 a 50, abychom se zaměřili na velikosti uživatelských profilů, které očekáváme během studie. Následně jsme simulovali doporučování postupně přes všechny varianty pro každého z vybraných uživatelů s náhodně vybranými váhami kritérií. Tuto simulaci označíme jako SIM_1. Poté jsme si ze skóre doporučených položek pro všechny metriky vytvořili graf, kde na ose y je normalizované skóre na základě prvního návrhu úpravy normalizace $NORM_1$ a na ose x buď pořadí v seznamu doporučení a nebo počet hodnocených položek uživatelem.

Na základě pozorování výsledků jsme vyhodnotili závislosti, které je třeba řešit. Grafy metrik s problémovou závislostí při SIM_1 jsou vidět vždy vlevo na obr. 3.1 a obr. 3.2. Závislost na pořadí v seznamu doporučení pozorujeme u všech variant metrik diverzity a u kalibrace. Problémová závislost na počtu ohodnocených položek uživatelem je zřejmá u obou variant metrik relevance, u distance-based novelty na základě maximální podobnosti a u intra-list distance-based novelty.



Obrázek 3.1: Normalizované hodnoty metrik a jejich závislost na pořadí v seznamu doporučení na základě simulací doporučování SIM_1 a SIM_2.



Obrázek 3.2: Normalizované hodnoty metrik a jejich závislost na počtu hodnocených položek uživatelem na základě simulací doporučování SIM_1 a SIM_2.

3.3.7.3 Natrénování normalizace řešící problémové závislosti

Snažili jsme se původní $NORM_1$ upravit tak, aby se lépe vypořádala se závislostmi na pořadí položky v seznamu doporučení i na počtu ohodnocených položek. Proto navrhujeme nové řešení spočívající v rozdělení natrénovaných normalizací na více normalizací odpovídající jednotlivým pozicím v seznamu doporučení, potažmo intervalu počtu ohodnocených položek, které zde podrobně popíšeme.

Algorithm 4 Pseudokód trénování normalizace $NORM_2$ pro metriku diverzity

Vstup: množina uživatelů U, seznam položek k doporučení *candidates*, počet hodnot použitých pro natrénování jedné normalizace *len_train*, velikost seznamu doporučení k
Výstup: seznam natrénovaných normalizací pro jednotlivé pozice v seznamu doporučení *normalizations_diversity*

```
/*Náhodný výběr vzorku uživatelů z množiny U*/
users = random_sample(U, 100)
normalizations_diversity = []
for u ∈ users do
    candidates_for_users[u] = candidates
    recommendation_lists[u] = []
end for
for i ∈ [1,...,k] do
    diversity_data_points = []
    for u ∈ users do
        diversity_values_of_candidates = diversity_metric(
            recommendation_lists[u], candidates_for_users[u])
        diversity_data_points.add(diversity_values_of_candidates)
        /*Simulace výběru doporučení*/
        recommendation_lists[u][i] = pseudo_recommend()
        candidates_for_users[u].remove(recommendation_lists[u][i])
    end for
    /*Náhodný výběr vzorku o velikosti len_train z diversity_data_points */
    data_points = random_sample(diversity_data_points, len_train)
    norm_diversity = normalization.train(data_points)
    normalizations_diversity[i] = norm_diversity
end for
return normalizations_diversity
```

Normalizaci metrik s problémovou závislostí na pořadí vybírané položky v doporučovacím seznamu, což jsou varianty diverzity a kalibrace, řešíme rozdělením jedné natrénované normalizace pro tyto metriky na normalizace dle jednotlivých pozic v seznamu doporučení. Upravený postup natrénování normalizace probíhá opět tak, že vybíráme náhodně vzorek 100 uživatelů, pro které simulujeme doporučování. Průběžný seznam doporučení opět postupně naplňujeme na základě pseudodoporučování, kdy s určitou pravděpodobností vkládáme položku s nejvyšší hodnotou metriky a náhodnou položku jinak. Ukládáme si hodnoty skóre metriky všech kandidátů při výběru položky na dané místo v seznamu doporučení. A následně náhodně vybíráme vzorek z těchto hodnot jen pro danou pozici (případně interval pozic), jehož velikost odpovídá počtu položek, tedy počtu

kandidátů, a na těchto hodnotách trénujeme normalizaci odpovídající pouze dané pozici (případně intervalu pozic) pomocí empirické distribuční funkce. Konkrétně je trénována normalizace pro každou z prvních 15 pozic, což vychází z očekávané velikosti doporučovacího seznamu. Od 16. do 60. pozice, což je používaná maximální velikost seznamu doporučení, je trénována vždy jedna normalizace pro interval o velikosti 5. Toto řešení je první částí druhé verze trénování normalizace, kterou označíme jako $NORM_2$, a její natrénování u metrik diverzity je algoritmicky vyjádřeno v alg. 4.

Obdobným způsobem jsme postupovali i u normalizací metrik s problémovou závislosti na počtu hodnocení uživatele, což jsou varianty relevance a novelty. Změnou je, že vybíráme náhodný vzorek maximálně 100 uživatelů vždy pro různé intervaly počtu hodnocení uživatele. To znamená pokud hledáme normalizaci pro uživatele s počtem hodnocení od 20 do 29, vybíráme vzorek jen z těch uživatelů, kteří ohodnotili alespoň 20 položek, a pokud uživatelé ze vzorku mají 30 a více záznamů hodnocení, pak z jejich hodnocení použijeme jen náhodný vzorek o náhodné velikosti z intervalu $[20; 30)$. Poté si ukládáme hodnoty kritérií a následně trénujeme normalizaci na vybraném vzorku těchto hodnot, jehož velikost odpovídá počtu položek. Konkrétně používáme intervaly $[0; 2)$, $[2; 5)$, $[5; 8)$, $[8; 10)$, $[10; 15)$, $[15; 20)$, $[20; 30)$, $[30; 40)$, $[40; 50)$, $[50; 60)$, $[60; 75)$, $[75; 90)$, $[90; 110)$, $[110; 140)$ a $[140,\infty)$. Tyto intervaly byly voleny dle očekávaného průběžného počtu hodnocení uživateli při plnění uživatelské studie. Zvyšující se velikost intervalů u vyšších hodnot vychází z odhadované menší pravděpodobnosti, že uživatel takového počtu hodnocení dosáhne, a také z očekávané méně viditelné závislosti na základě podobných velikostí uživatelova profilu. Je také nutné zmínit, že u relevance jsme oproti $NORM1$ nebrali v úvahu odhadovanou relevanci vybraných položek z náhodného vzorku hodnocení uživatele, a to proto, že již hodnocené položky nejsou uživateli doporučovány. Toto řešení je druhou částí druhé verze trénování normalizace $NORM_2$.

3.3.7.4 Simulace doporučování při použití normalizace $NORM_2$

Po implementaci tohoto druhého návrhu trénování normalizace ($NORM_2$) jsme opět simulovali doporučování se zcela stejnými parametry jako tomu bylo při simulaci doporučování s první verzí trénování normalizace ($NORM_1$). Použili jsme zcela stejné uživatele a pro každé doporučování i stejný počet jeho prvních hodnoceních, jako tomu bylo u SIM_1. Tuto simulaci označíme jako SIM_2. Na obr. 3.1, resp. obr. 3.2, můžeme vidět grafy, kde na ose y je normalizované skóre a na ose x pořadí v seznamu doporučení, resp. počet hodnocených položek uživatelem. Grafy s výsledky simulace, pro kterou byla použita druhá verze natrénování normalizace $NORM_2$, jsou viditelné na obr. 3.1 a obr. 3.2 vždy hned vpravo od vizualizace výsledků simulace při použití první varianty natrénování normalizace $NORM_1$.

Pro lepší přehlendost jsme na grafech z obr. 3.2 provedli uhlazení (smoothing) tak, že hodnoty počtu ohodnocených položek byly rozděleny na intervaly o velikosti 5. Jak je patrné z porovnání na obr. 3.1 a obr. 3.2, závislost normalizovaných hodnot jak na pořadí, tak na počtu hodnocených filmů, se nám podařilo alespoň částečně zmírnit.

Pokud se podíváme nejdříve na závislost na pořadí v seznamu doporučení zobrazenou na obr. 3.1, můžeme vidět především vždy o něco vyrovnanější hodnoty

od 4. pozice dál. Větší hodnoty u prvních pozic připisujeme definici algoritmu RL-Prop (viz alg. 2), který zejména u prvních pozic maximalizuje všechny metriky nebo alespoň většinu z nich, pokud má takovou položku k dispozici. Definicí algoritmu si zdůvodňujeme propad u druhé pozice u kalibrace, a to ve spojitosti se způsobem výpočtu této metriky (viz kapitola 3.3.6), která je navázána na kalibraci průběžného seznamu doporučení bez ní. Ze způsobu výpočtu vyplývá, že dobře kalibrované seznamy doporučení o velikosti $n-1$, snižují možný příspěvek položky na n-té pozici. Ke stejnému, byť ne na základě grafu z obr. 3.1 tolik markantnímu problému jako u kalibrace dochází i u binomické diverzity.

Jak můžeme pozorovat na obr. 3.2, závislost s ohledem na počet ohodnocených položek uživatelem, se pomocí $NORM_2$ podařilo vyřešit. Nezvažováním odhadované relevance vybraných položek z náhodného vzorku hodnocení uživatele v $NORM_2$, jsme se také dokázali vypořádat s příliš nízkými hodnotami odhadované relevance položek jak u $EASE_{POS}$ tak u $EASE_{NEG}$.

3.3.8 Výběr doporučení pro uživatele

Při požadavku na seznam doporučení pracuje doporučovací systém následovně. Nejdříve si zpracuje předané parametry, které podrobněji popisujeme v kapitole 3.2. Následně z databáze získá uživatelova hodnocení a na základě nich přepočítá hodnotu proměnných, které jsou používány při výpočtu skóre metrik, případně výběru správné normalizace. Pro varianty relevance je nutné napočítat nový uživatelův vektor predikcí z vynásobení vytvořeného vektoru na základě momentálních hodnocení uživatele odpovídající pravidlům pro vytvoření matice X (viz kapitola 3.3.2) a natrénované matice B. Na základě momentálních hodnocení upravíme i profil uživatele, což je nutné pro výpočet intra-list distance-based novelty a distance-based novelty na základě maximální podobnosti a velikost tohoto profilu je důležitá pro výběr správné normalizace. Navíc musíme přepočítat distribuci žánrů v uživatelově profilu, což je zase využiváno při výpočtu binomické diverzity a kalibrace.

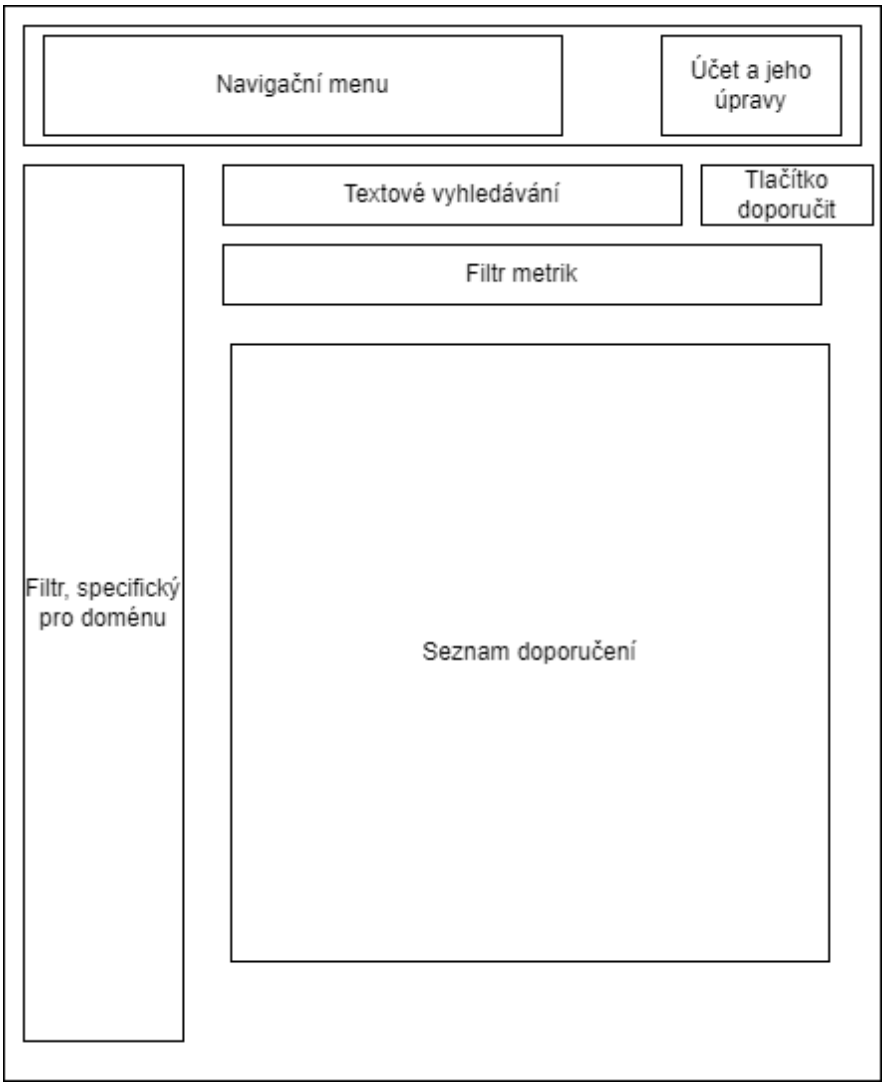
Následně už postupně vybíráme položky do seznamu doporučení. Nejdříve vypočítáme skóre každé položky na základě vybraných metrik pro každé kritérium kvality doporučování. Tato skóre následně normalizujeme. Poté algortimus RL-Prop vybere položku a ta je přidána do seznamu doporučení. Pokud seznam doporučení není dokončený, cyklus opakujeme. V opačném případě vracíme seznam doporučení a normalizované hodnoty skóre každé položky v seznamu pro jednotlivá kritéria.

3.4 Webová aplikace

Jak je zmíněno výše, cílem bylo vytvoření webové aplikace, kterou bude používat uživatel a která bude komunikovat s multi-objective doporučovacím systémem.

Právě s webovou aplikací budou pracovat účastníci uživatelské studie. Protože jsme se nechtěli nutně omezit na česky hovořící účastníky, je celé uživatelské rozhraní aplikace v angličtině.

3.4.1 Návrh GUI



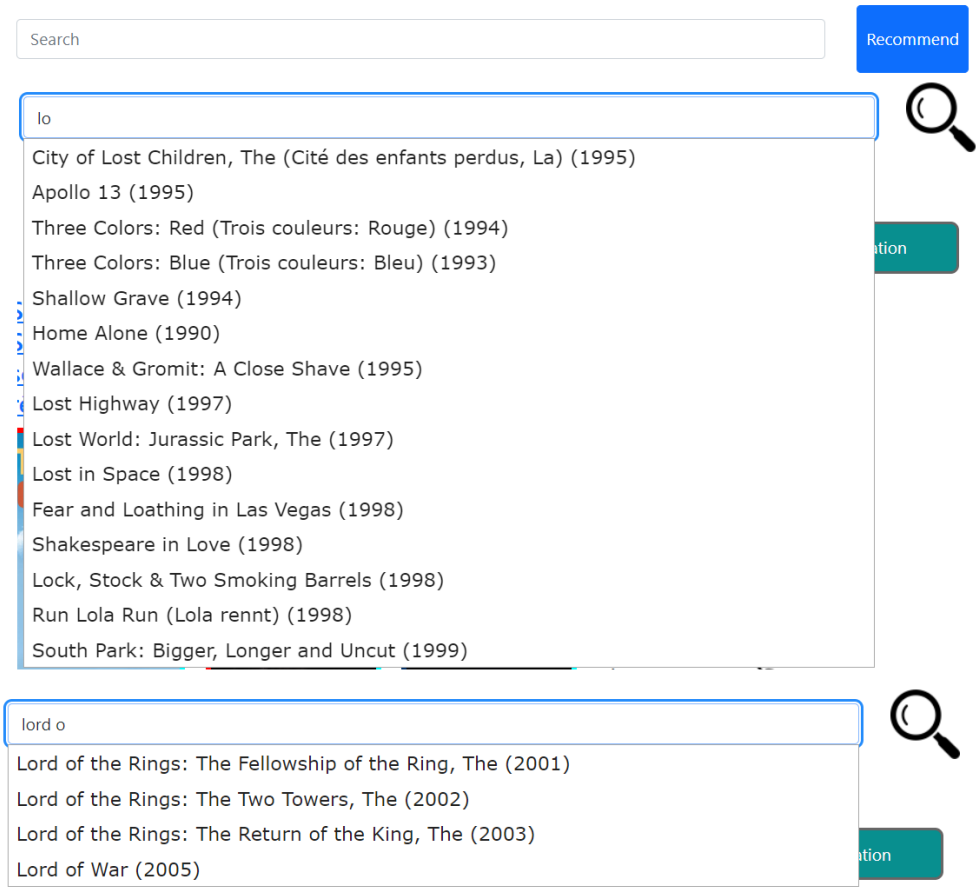
Obrázek 3.3: Navrhované rozložení hlavní stránky webové aplikace před začátkem implementace

Po ujasnění všech základních prvků, které by měla webová aplikace obsahovat jsme ještě před začátkem samotné implementace navrhli základní rozložení hlavní stránky (viz obr. 3.3).

Nyní jednotlivé prvky, jejich grafickou podobu a funkčnost popíšeme podrobněji včetně přiložených snímků obrazovky. Je také potřeba zmínit, že aplikace byla spouštěna a testována s daty z domény filmů a některé prvky byly uzpůsobeny právě pro tuto doménu.

3.4.1.1 Textové vyhledávání

Textové vyhledávání se v zásadě nijak neliší od těch, jaké můžeme najít na většině webových portálů. Vyhledávání funguje pouze na název položky, a to ze dvou důvodů. Tím prvním je, že není nutné prvek jakkoli měnit, pokud by měla aplikace sloužit pro jinou doménu, a druhým je přítomnost podrobnějšího filtru, kde uživatel může vyhledávat na základě vlastností položek.



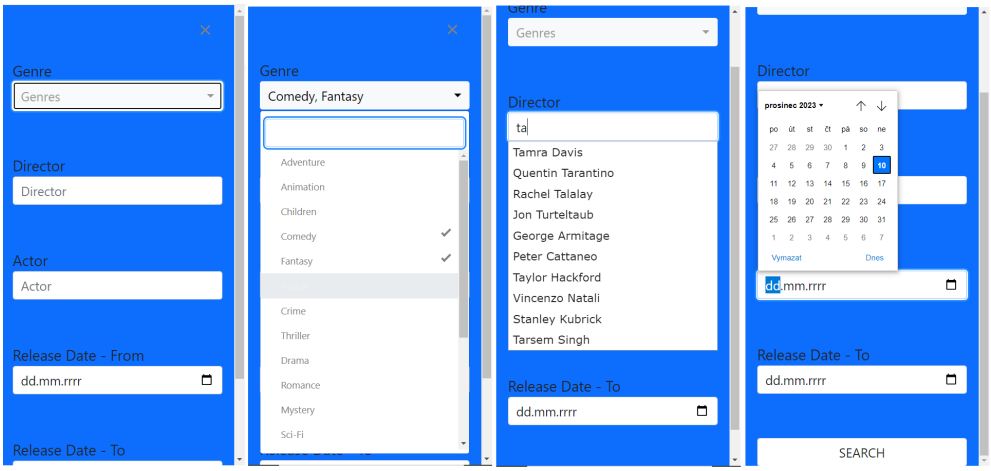
Obrázek 3.4: Kroky textového vyhledávání

Způsob vyhledávání je zobrazen na obr. 3.4. Po vyplnění jakýmkoli textem se tlačítko hned vpravo od tohoto prvku změní svůj vzhled na ikonu hledání. Součástí textového vyhledávání je také našeptávač, který navrhne položky, jejichž název obsahuje vyhledávaný text.

3.4.1.2 Podrobnější filtr

Vzhledem k nízké expresivnosti textového vyhledávání na základě jména položky, kdy uživatel musí přesně vědět, co hledá, je k dispozici i jiná forma vyhledávání. Tou je podrobnější filtr přes různé vlastnosti položky, který musí být samozřejmě specifický pro danou doménu.

V našem případě byl implementován filtr pro doménu filmů. Vyhledávat se dalo na základě žánru, režiséra, herce a data vydání. Uživatel může zvolit jakoukoli podmnožinu žánrů (výběr žádné možnosti je interpretován stejně jako výběr všech žánrů) z rozbalovacího seznamu (viz část druhá zleva obr. 3.5). Režiséra i herce může uživatel specifikovat textově a vyfiltrování se řídí stejnou podmínkou, jako je tomu u textového vyhledávání na základě názvu. To znamená, že film splňuje specifikovanou podmínku, pokud film má režiséra, nebo v něm hrál herec, jehož jméno obsahuje vyhledávaný text. I zde funguje našeptávač (viz část druhá zprava obr. 3.5). Důvodem použití výběru z rozbalovacího seznamu u žánrů a textového vyhledávání pro následující dvě položky je rozdílný počet možných hodnot u těchto vlastností. Zatímco žánrů je 19, režiséři a herci jsou v našich



Obrázek 3.5: Podrobný filtr a pole, která obsahuje

datech zastoupeni v řádu stovek až tisíců. Poslední možností, jak může uživatel filtrovat, je nastavení minimálního a maximálního data, kdy mohl být film vydán. Tyto dvě pole lze vybírat přímo z kalendáře (viz část zcela vpravo obr. 3.5).

3.4.1.3 Náhled položky

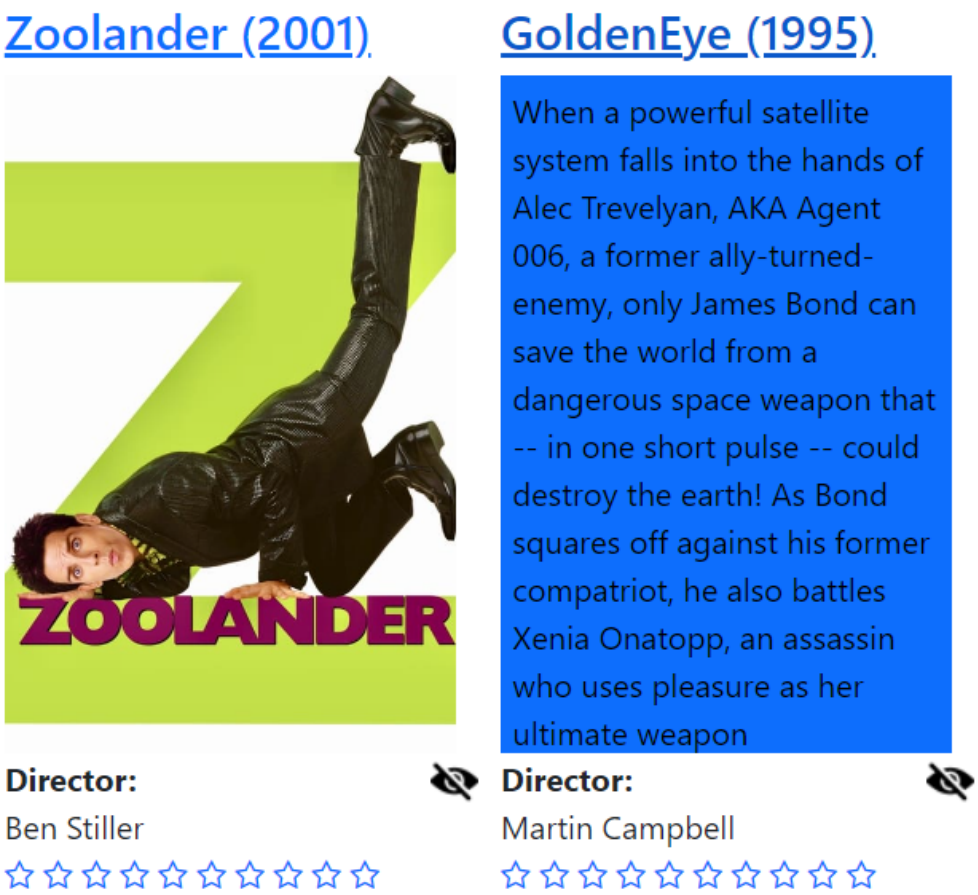
Další věcí k řešení je způsob, jak vizualizovat položky v rámci doporučovacího seznamu. Cílem by mělo být vměstnání dostatku informací o položce, aby uživatel dokázal posoudit, zda jej tento objekt zajímá. Zároveň je potřeba tento zájem vybalancovat s tím, že místo na obrazovce je omezené a uživateli chceme zobrazit více položek zároveň.

Při návrhu vzhledu položky v seznamu doporučení jsme využili výsledků ze studie Beela a Dixonové (Beel a Dixonová, 2021), která se zaměřovala na vizualizaci doporučování v doméně blogů. Jejich nejlepší varianta obsahovala název, obrázek a při najetí kurzorem myši se položka zvýraznila a obsah náhledu byl nahrazen popisem položky. Jak je zřejmé z obr. 3.6, všechny tyto vlastnosti a prvky jsme použili. Jelikož doporučovací systém funguje na základě desetiškálového hodnocení, je součástí náhledu i nástroj, pomocí kterého uživatelé mohou položky hodnotit. Vzhledem k námi zvolené doméně filmů je náhled navíc rozšířen také o režiséra filmu.

3.4.1.4 Detail položky

Jak je zmíněno výše, ke mnoha pozitivním charakteristikám doporučovacího systému pomáhá dobrá posuzovatelnost toho, zda je položka pro uživatele vhodným kandidátem. A právě tu můžeme zvýšit větším počtem informací o jednotlivých objektech, které se již z prostorových důvodů nevlezly do náhledu položky. Proto lze po kliknutí právě na náhled položky zobrazit i její detail.

Na stránce detailu položky (viz obr. 3.7) lze najít všechny informace zobrazené na jejím náhledu, tzn. název, hodnocení, popis, režiséra filmu a obrázek. Navíc jsou zde vypsány všechny další údaje o položce, které máme k dispozici, což jsou žánry, do kterých se film řadí a herecké obsazení. Součástí detailu je také trailer nahraný na platformě YouTube. Vpravo nahoře je tlačítko, pomocí kterého se uživatel může vrátit zpět na seznam doporučení.



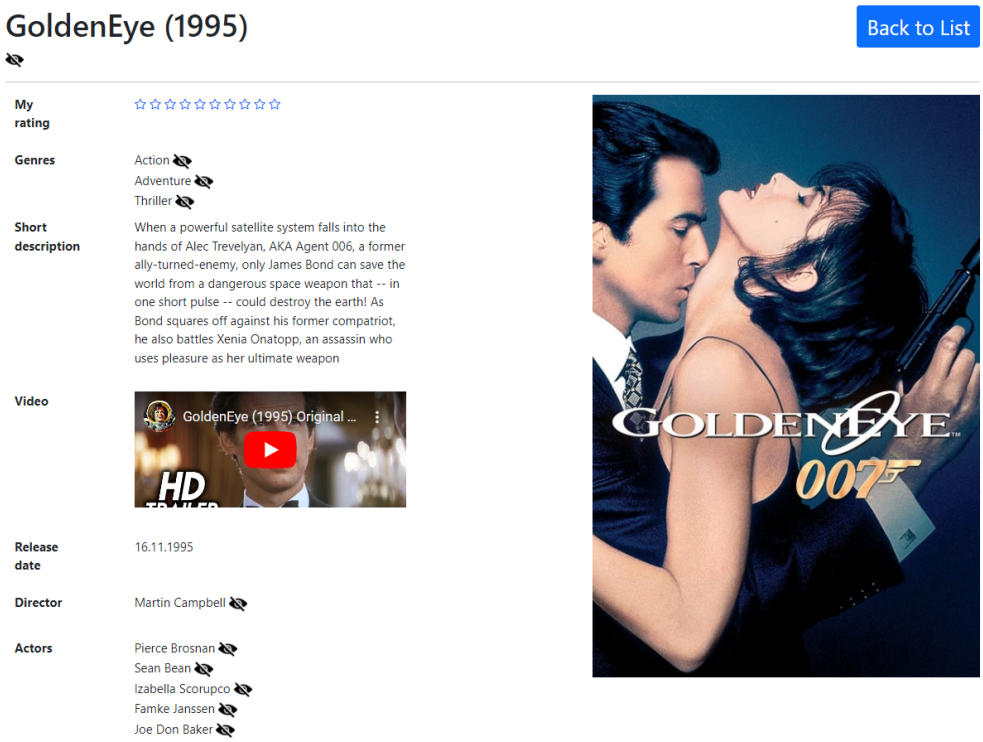
Obrázek 3.6: Náhled položky v seznamu doporučení; Náhled položky v seznamu doporučení při najetí kurzorem myši (Zdroj plakátu Zoolander: The Movie Database (2023b))

3.4.1.5 Blokace

Mimo hodnocení a nastavení filtru kritérií kvality doporučování je umožněna ještě další akce, pomocí které lze ovlivnit doporučení. Uživatel může blokovat přímo samotné položky, a to pomocí kliknutí na ikonu přeškrtnutého oka ve spodní části náhledu položky (viz obr. 3.6) a nahoře přímo pod názvem v detailu položky (viz obr. 3.7).

Přímá blokace filmu není jediný způsob, jakým lze zakázat doporučení některých objektů. Jde odmítnout filmy na základě jejich vlastností, konkrétně uživatel může blokovat některé žánry, režiséry a herce, a to pomocí kliknutí na ikonu přeškrtnutého oka u názvu žánrů resp. jmen režisérů a herců v detailu položky (viz obr. 3.7).

Pro správu blokujících pravidel je vyhrazena v aplikaci samostatná stránka. Uživatelé zde můžou kliknutím na ikonu oka zrušit své blokující pravidlo (viz obr. 3.8 vlevo). Zároveň mohou využít 2 typy formulářů pro vytváření pravidel nových, a to buď variantu po jednom pravidlu (na obr. 3.8 uprostřed), nebo druhou možnost, která dovoluje zadefinovat více bloků zároveň (na obr. 3.8 vpravo).



Obrázek 3.7: Stránka s detailem položky (Zdroj plakátu GoldenEye: The Movie Database (2023a))

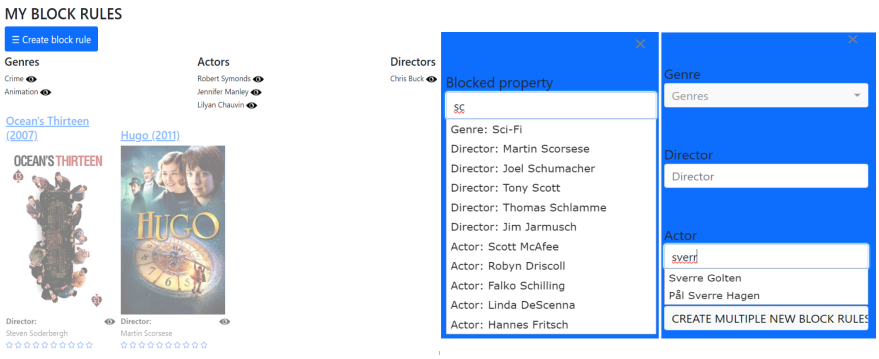
3.4.1.6 Filtr kritérií kvality doporučování

Jedním z hlavních výstupů naší práce by mělo být, zda uživatelé stojí o modifikaci důležitosti jednotlivých kritérií a jaký navrhovaný mechanismus jim pro tento úkon přijde nejvhodnější. Proto bylo nutné předložit jako možnost několik variant filtrů kritérií kvality doporučování, pomocí nichž může uživatel měnit váhy právě jednotlivých kritérií. Navrhované mechanismy by měly být pro uživatele pochopitelné, jednoduché k používání a vzájemně na sebe reagující vzhledem k faktu, že mezi kritéria lze rozdělit jen danou celkovou váhu. To znamená, že při překročení této hodnoty se musí váhy ostatních kritérií snížit.

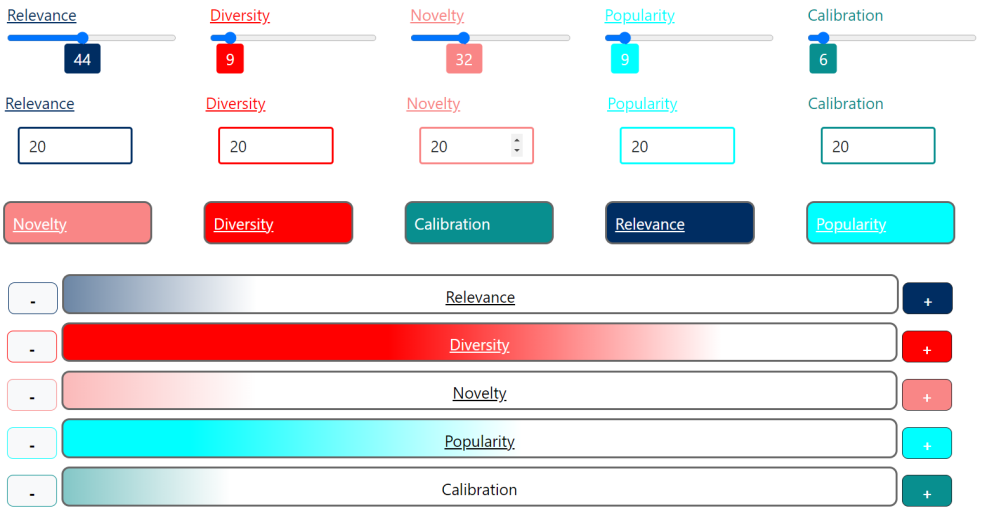
První zkoumaný nástroj jsou tzv. sliders, tedy posuvníky (na obr. 3.9 nahoře), které jsou použity ve studii Dokoupila, Pešky a Boratta (Dokoupil a kol., 2023a). Uživatel nastaví váhu pro kritérium vždy změnou ovládacího bodu odpovídajícího slideru. Pod ovládacím bodem posuvníku je zobrazena v současnosti nastavená číselná hodnota. Očekávanou předností této varianty by měla být pochopitelnost a fakt, že je již odzkoušená v rámci jiné uživatelské studie.

Druhým mechanismem jsou textová pole (na obr. 3.9 druhý shora). Uživatel může do pole přímo napsat hodnotu váhy, jakou by chtěl přidělit danému kritériu kvality doporučování. Očekávanou předností této varianty je, že jde o pohodlný způsob ke specifikaci přesných hodnot.

Třetí, o něco více odlišnou variantou, kterou můžeme vidět na obr. 3.9 jako druhou zespodu, je použití metody drag and drop („táhni a pusť“). Jak název napovídá, uživatel může měnit pořadí kritérií kvality doporučování pomocí stisknutí levého tlačítka myši nad boxem jednoho z kritérií, následným přetažením nad box jiného kritéria a uvolněním tlačítka myši. Po této akci si přetahované kritérium



Obrázek 3.8: Stránka pro správu blokujících pravidel (zdroj plakátů: TMDB (2023)); Formulář pro vytvoření jednoho blokujícího pravidla; Formulář umožňující vytvoření více blokujících pravidel zároveň

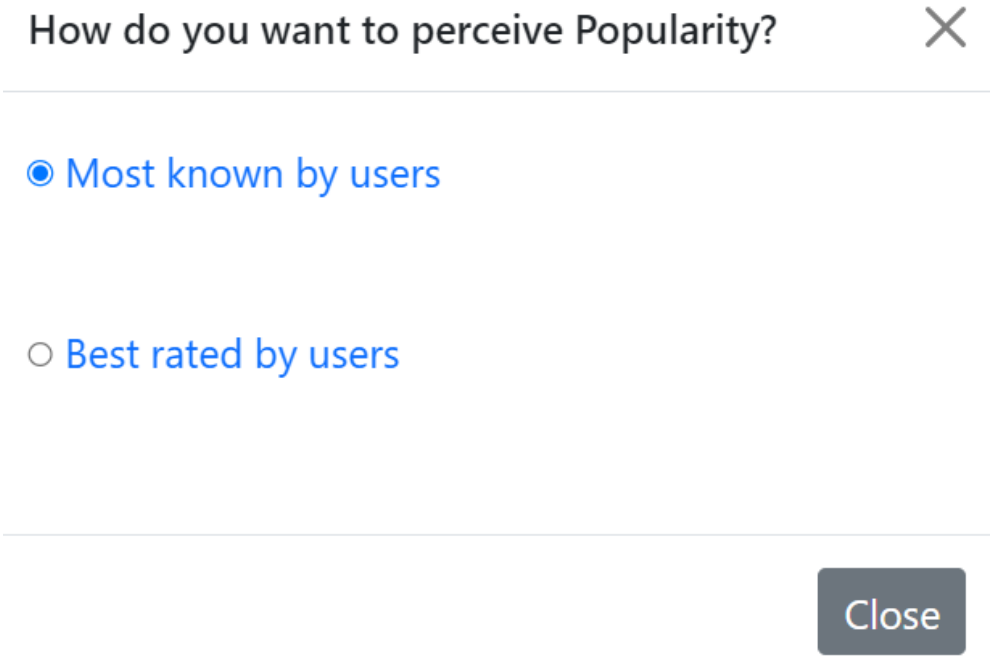


Obrázek 3.9: Varianty filtru kritérií kvality doporučování: Slidery; Textová pole; Drag and drop; Tlačítka + a -

vymění místo s druhým, nad kterým bylo upuštěno. Platí, že kritéria více vlevo mají vyšší váhu, než ty více vpravo a to v takovém poměru, kdy kritérium zcela vpravo má váhu (v rámci poměru) 1 a každé kritérium má váhu o 1 vyšší než to hned vedle něj vpravo. To při použití 5 kritérií kvality doporučování znamená, že poměr vah je 5:4:3:2:1. Očekávanou výhodou tohoto nástroje je snížení úsilí uživatele při rozhodování o přesné hodnotě váhy a také fakt, že by použitím tohoto mechanismu mělo docházet k více vyváženému doporučování, byť to se může pro některé uživatele ukázat i jako nevýhoda.

Posledním návrhem pro filtr kritérií kvality doporučování je použití tlačítek + a -, které zvyšují a snižují váhu kritéria. Součástí tohoto nástroje je i progress bar („ukazatel průběhu“), který se více zbarvuje do bíla při snižování váhy a naopak do barvy kritéria při zvyšování váhy (viz obr. 3.9 dole). Zvyšování a snižování mění váhu o 10 jednotek (maximum je 100). Očekávanou výhodou této varianty je snadná pochopitelnost a grafické znázornění váhy oproti číselnému v prvních 2 variantách, které může být pro některé uživatele příjemnější pro práci se systémem.

3.4.1.7 Volba varianty metriky

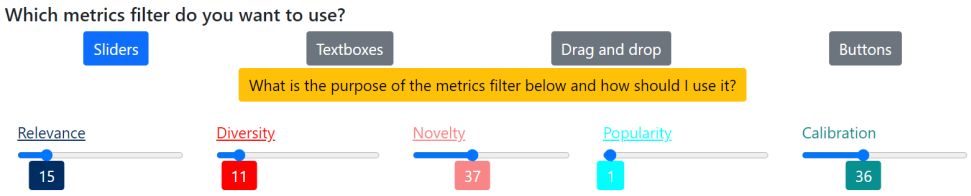


Obrázek 3.10: Volba varianty metriky popularita

Upravený doporučovací systém umožňuje většinu kritérií měřit různými způsoby. Konkrétně to platí u relevance, diverzity, novelty a popularity, pro výpočet kalibrace používáme pouze jeden způsob (viz kapitola 1.3). Protože jedním z našich cílů je porovnání různých metrik z pohledu uživatele, musíme mu v rámci webové aplikace umožnit zvolit variantu, kterou doporučovací systém používá. Jde tak o další způsob, jakým může uživatel ovlivnit výstup algoritmu, tedy seznam doporučení.

Uživatel může měnit používanou metriku kliknutím na název kritéria ve filtru kritérií kvality doporučování. Názvy kritérií s více způsoby výpočtu jsou podtrženy (viz obr. 3.9). Následně se uživateli zobrazí výběr jednotlivých metrik, který umožňuje volbu varianty pomocí přepínače, jak je vidět na obr. 3.10.

3.4.1.8 Uživatelské nastavení



Obrázek 3.11: Výběr z variant filtru kritérií kvality doporučování na stránce uživatelského nastavení aplikace

V rámci aplikace umožňujeme uživateli také měnit některé části aplikace.

Pro tato nastavení je vyhrazena samostatná stránka, kde lze nastavení spravovat a volit různé varianty prvků.

Jak zmiňujeme výše, uživatel má na výběr z více možností filtru kritérií kvality doporučování (viz obr. 3.11) a formuláře pro nové blokující pravidlo. Navíc lze měnit způsob, jakým jsou doporučení vysvětlena a to v několika vrstvách. Konkrétně jsou k dispozici různé alternativy pro náhled vysvětlení, způsob vizualizace skóre a počet kritérií s vysvětlením. Doplňujícím nastavením je přiřazení barev ke kritériím.

3.4.2 Explanations

Co se týče vysvětlení doporučení, zaměřili jsme se na dva problémy. Tím prvním je návrh textu vysvětlení, tím druhým je vizualizace explanations.

3.4.2.1 Text

Při návrhu explanations jsme začali nejdříve s textem vysvětlení doporučení. Tyto texty jsem se rozhodli rozdělit pro jednotlivá kritéria tak, aby byla pro uživatele pochopitelnější, než jeden text kombinující všechna tato vysvětlení. Jak zmiňujeme v požadavcích (viz kapitola 2.2.2.5), je nutné vzít v úvahu, že v seznamu doporučení z multi-objective doporučovacího systému se mohou objevovat položky s nižším až velmi nízkým skóre vzhledem k některým metrikám kritérií kvality doporučování. Proto navrhujeme pro každou metriku čtyři různé texty vysvětlení - velmi pozitivní, spíše pozitivní, spíše negativní, velmi negativní. Tyto texty odpovídají následujícím intervalům skóre položky pro danou metriku - $[75; 100]$, $[50; 75)$, $[25; 50)$, $[0; 25)$.

V následujícím seznamu popíšeme všechny tyto texty vysvětlení k používaným metrikám. Protože uživatelská studie probíhala v angličtině, budou texty vypsány anglicky. Jejich překlad do češtiny je k dispozici v příloze A.5.

- $EASE_{POS}$
  - Users that like similar movies as you really like this movie.
  - Users that like similar movies as you usually like this movie.
  - Users that like similar movies as you usually don't like or don't know this movie.
  - Users that like similar movies as you really don't like or don't know this movie.
- $EASE_{NEG}$
  - Users that like and dislike similar movies as you really like this movie.
  - Users that like and dislike similar movies as you usually like this movie.
  - Users that like and dislike similar movies as you usually don't like this movie.
  - Users that like and dislike similar movies as you really don't like this movie.

- Intra-list diverzita
  - This movie is different from all previous movies in the list.
  - This movie is quite different from all previous movies in the list.
  - This movie is quite similar to previous movies.
  - This movie is very similar to previous movies.
- Diverzita na základě maximální podobnosti
  - No previous movie in the list is similar to this movie.
  - No previous movie in the list is very similar to this movie.
  - Some previous movies in the list are quite similar to this movie.
  - Some previous movies in the list are very similar to this movie.
- Binomická diverzita
  - This movie is different on genres from most of the previous movies in the list.
  - This movie is quite different on genres from most of the previous movies in the list.
  - This movie is not much different on movie genres from most of the previous movies.
  - This movie is not different at all on movie genres from most of the previous movies.
- Očekávaný doplněk popularity
  - This movie is not known by most of the users.
  - This movie isn't amongst the most known by users.
  - This movie is quite known amongst users.
  - This movie is known by most of the users.
- Distance-based novelty na základě maximální podobnosti
  - This movie is not similar to any movie you rated.
  - This movie is not very similar to any movie you rated.
  - This movie is quite similar to some of the movies you rated.
  - This movie is very similar to some of the movies you rated.
- Intra-list distance-based novelty
  - This movie is not similar to movies you rated on average.
  - This movie is not very similar to movies you rated on average.
  - This movie is quite similar to movies you rated on average.
  - This movie is very similar to movies you rated on average.

- Popularita dle známosti
  - This movie is known by most of the users.
  - This movie is known amongst the users.
  - This movie is quite unknown amongst users.
  - This movie is not known by users.
- Popularita na základě hodnocení
  - This movie recieved very good ratings from users.
  - This movie recieved good ratings from users.
  - This movie is not amongst the best rated.
  - This movie isn't well rated.
- Kalibrace
  - This movie helps to make ratio of genres among movies in the list to this item more corresponding to ratio of genres among movies you have postitively rated.
  - This movie quite helps to make ratio of genres among movies in the list to this item more corresponding to ratio of genres among movies you have postitively rated.
  - This movie doesn't help much to make ratio of genres among movies in the list to this item more corresponding to ratio of genres among movies you have postitively rated.
  - This movie doesn't help at all to make ratio of genres among movies in the list to this item more corresponding to ratio of genres among movies you have postitively rated.

U každého textu vysvětlení jsme se zaměřili na transparentnost, aby uživatelé skutečně dokázali z vysvětlení odvodit, co přispívá k dané metrice, a následně pomocí hodnocení a filtru kritérií kvality doporučování mohli upravit své preference. Zároveň jsme se snažili každé vysvětlení podat co nejpochopitelnějším způsobem tak, aby mu uživatel dobře rozuměl.

3.4.2.2 Vizualizace

Pro doporučování používáme multi-objective doporučovací systém, proto nelze zvolit odpovídající explanation na celý seznam doporučení. Každá položka má různé skóre pro každé kritérium kvality doporučování a každé kritérium navíc textově vysvětlujeme jiným způsobem. Vzhledem k většímu počtu informací také není vhodné zobrazit celé odůvodnění doporučení zároveň pro každou položku ze seznamu doporučení. Proto jsme řešili nejprve způsob, jak uživateli zobrazit poměr skóre už z náhledu položky. Detailnější vysvětlení doporučení dané položky jsme zobrazili až při akci najetí myší na náhled položky.

3.4.2.2.1 Poměr skóre z náhledu položky

Ke každému kritériu přiřadili barvu, kterou používáme už ve filtru kritérií kvality doporučování (viz obr. 3.9). Navrhli jsme nejdříve dvě varianty ohraničení náhledu položky - kompletní ohraničení barvami kritérií na základě poměru skóre, ohraničení barvami kritérií pouze vlevo na základě poměru skóre. U obou těchto možností odpovídá poměr barev v ohraničení poměru skóre odpovídajících kritérií. Třetí variantou je název položky v barvě kritéria s nejvyšším skóre. Všechny varianty jsou zobrazeny na obr. 3.12

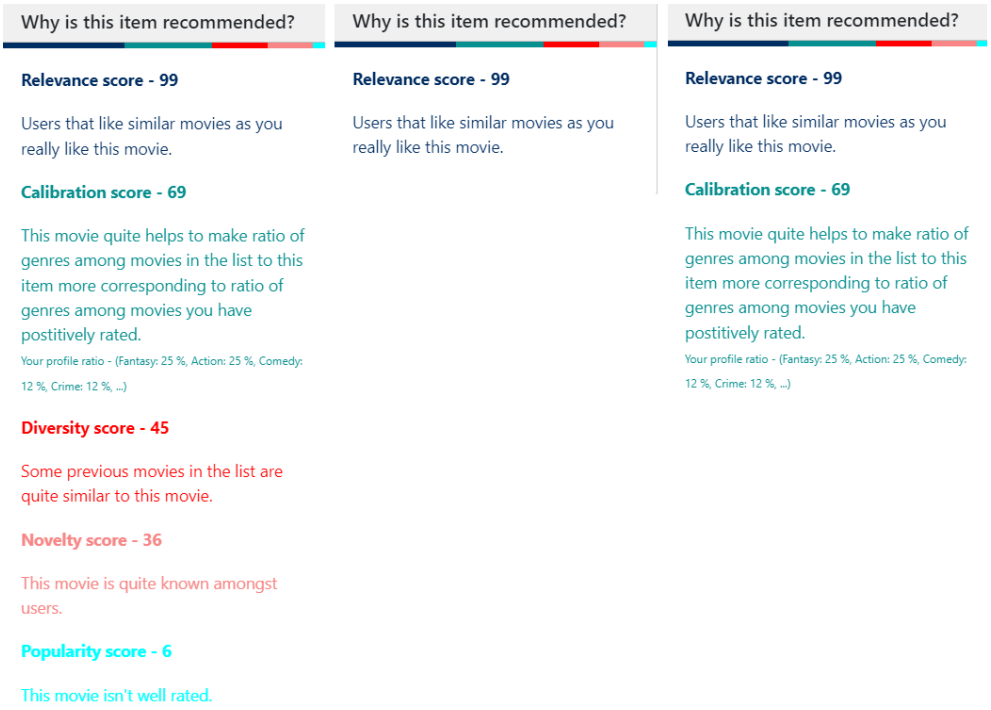
Obrázek 3.12: Varianty náhledu vysvětlení při zobrazení seznamu doporučených položek, zdroj plakátu GoldenEye: The Movie Database (2023a). Zleva jde o kompletní ohraničení barvami kritérií na základě poměru skóre, ohraničení barvami kritérií pouze vlevo na základě poměru skóre a název položky v barvě kritéria s nejvyšším skóre. Zobrazované doporučení má následující skóre jednotlivých kritérií: Relevance (tmavě modrá) 83 , diverzita (červená) 55, novelty (růžová) 55, popularita (světle modrá) 55 a kalibrace (tmavě zelená) 27.

3.4.2.2.2 Počet kritérií kvality doporučování

V původním návrhu se uživateli při najetí myší na náhled položky zobrazilo popover okno s detailem vysvětlení (skóre a text) ke všem kritériím kvality doporučování. Protože by vysvětlení všech kritérií u každé položky mohlo vést k přetížení uživatele informacemi, navrhli jsme druhou variantu, kde je zobrazováno vysvětlení jen ke kritériu s nejvyšším skóre. Protože v tomto návrhu zase naopak výrazná část vysvětlení doporučení uživateli není zobrazena, byla přidána ještě třetí varianta obsahující vysvětlení pouze těch kritérií s nadprůměrným skóre, což vzhledem k normalizaci pomocí empirické distribuční funkce a přeškálování na interval $[0; 100]$ byla odhadem pouze kritéria se skóre 50 a více. Vizualizace těchto možností je viditelná na obr. 3.13.

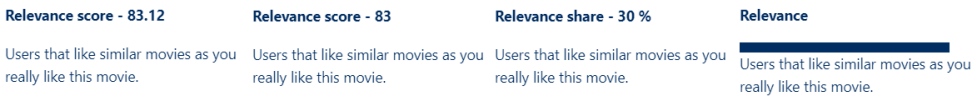
3.4.2.2.3 Vizualizace skóre

Poslední části explanations, u které navrhujeme více variant je vizualizace skóre.



Obrázek 3.13: Varianty počtu kritérií kvality doporučení v rámci explanation. Vlevo je vysvětlení všech kritérií, uprostřed vysvětlení jen ke kritériu s nejvyšším skóre, vpravo vysvětlení pouze kritérií s nadprůměrným skóre.

První variantou je zobrazovat skóre kritérií kvality doporučení, která nabývají hodnot z intervalu $[0; 100]$, taková, jaká byla předána doporučovacím systémem, se zaokrouhlením na dvě desetinná místa. Tento typ budeme nazývat hrubým skóre. Druhou obdobnou možností je zaokrouhlení na celá čísla, kterou budeme nazývat zaokrouhleným skóre. Třetí způsob je navázán na vizualizaci vysvětlení v náhledu položky (viz obr. 3.12), kdy nezobrazujeme skóre kritéria, ale procentuální podíl kritéria na součtu skóre všech kritérií, tzn. zobrazujeme hodnotu $\forall m \in M : share_m = \frac{score_m}{\sum_{n \in M} score_n}$, kde M je množina všech současně používaných metrik, z nichž každá počítá skóre pro dané kritérium. Poslední variantou je sloupcový graf (byť proložený textem explanation), kdy skóre předaná doporučovacím systémem nezobrazujeme číselně, ale graficky. Všechny způsoby vizualizace skóre jsou zobrazeny na obr. 3.14.

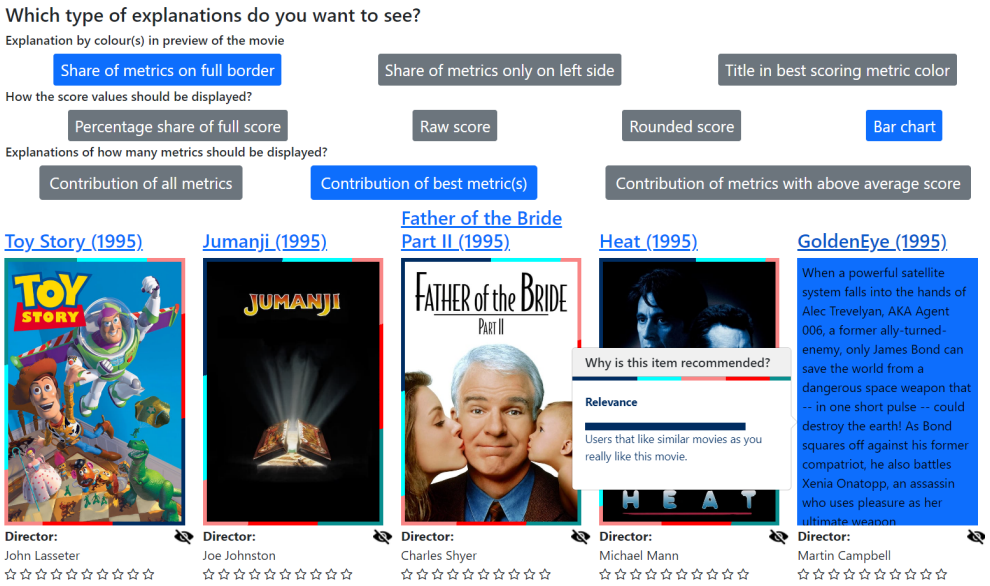


Obrázek 3.14: Varianty vizualizace skóre v explanation. Zleva jsou to hrubé skóre, zaokrouhlené skóre, procentuální podíl kritéria na součtu skóre všech kritérií a grafické vyjádření skóre na způsob sloupcového grafu.

3.4.2.2.4 Nastavení explanations

Uživatel může měnit varianty vizualizace poměru metrik v náhledu položky, počet kritérií kvality doporučování vyskytujících se ve vysvětlení a způsob prezentace

skóre v uživatelském nastavení (viz kapitola 3.4.1.8). Způsob nastavení variant vysvětlení je představen na obr. 3.15.



Obrázek 3.15: Volba variant explanations v uživatelském nastavení (Zdroj plakátů filmů: TMDB (2023))

4. Implementace

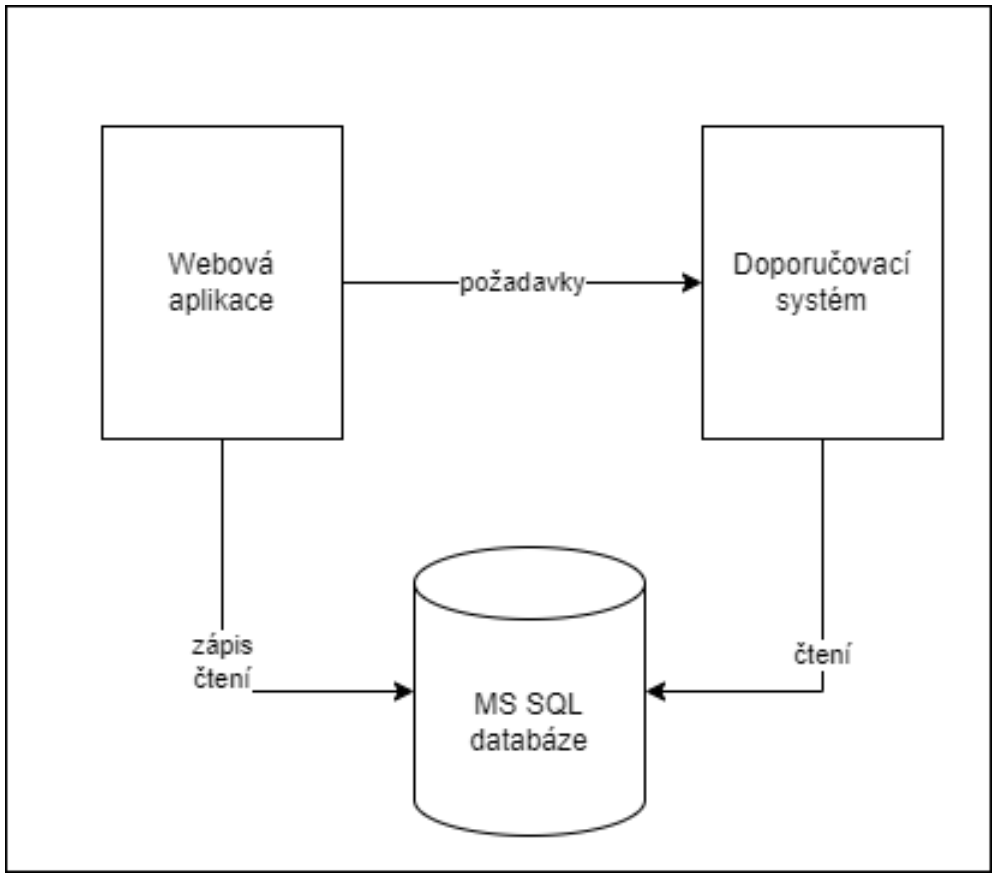
Jak je zmíněno výše, pro uživatelskou studii musela být vytvořena webová aplikace, která umožní jednak komunikaci s multi-objective doporučovacím systémem a především umožní uživatelovi upravit parametry pro tento doporučovací systém. Její implementaci zevrubně popíšeme.

Konkrétní prvky webové aplikace jsou popsány v kapitole 3.4 a její komunikace s doporučovacím systémem v kapitole 3.2.2.

Zároveň je součástí celého softwarového díla i doporučovací systém. Jeho původní implementace ale byla k dispozici a naší prací bylo ji jen upravit a rozšířit pro naše potřeby. Tyto změny jsou popsány v kapitole 3.3.

Struktura odevzdaného řešení je velmi zevrubně popsána v příloze B.1. Podrobnější popis lze nalézt v dokumentačních souborech uvnitř samotného řešení.

4.1 Architektura



Obrázek 4.1: Architektura

Celé softwarové dílo se skládá ze tří komponent, webové aplikace, doporučovacího systému a databáze. V databázi jsou uložena všechna potřebná data o uživatelích, položkách, hodnoceních, interakcích atd. Doporučovací systém je po natrénování na datech z databáze schopen doporučovat uživateli položky na základě jeho preferencí. V uživatelském rozhraní webové aplikaci může uživatel procházet

položky z databáze, nastavit své preference pro doporučování a následně posílat požadavky na doporučovací systém a zobrazovat výsledný seznam doporučení.

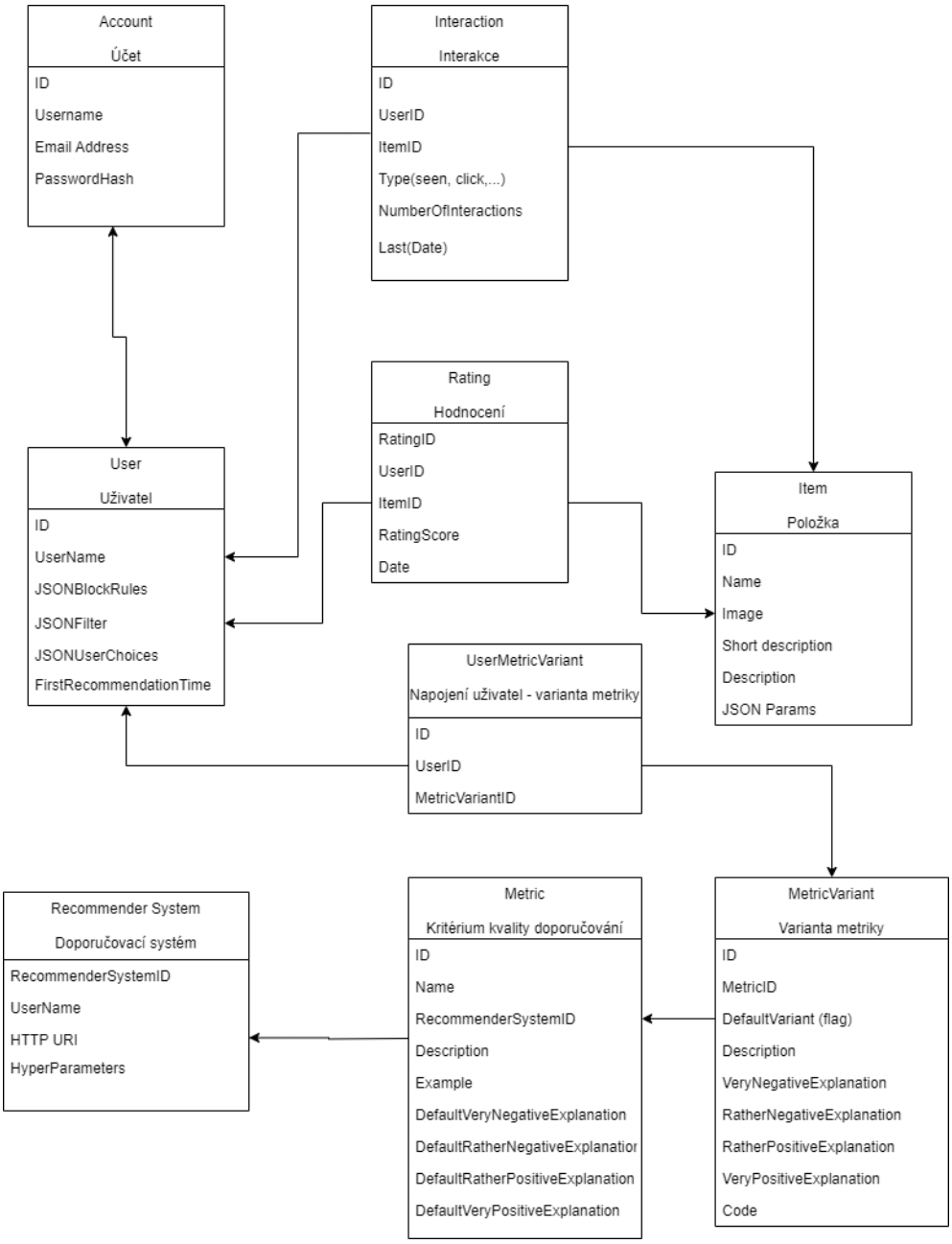
Jak je patrné z architektury na obr. 4.1, webová aplikace i doporučovací systém posílají dotazy do databáze, webová aplikace navíc posílá požadavky na doporučovací systém.

Webová aplikace jednak z databáze čte data, aby uživateli byly prezentovány současné informace, a jednak do databáze data zapisuje, případně je aktualizuje, a to zejména po akcích uživatele (vytvoření uživatele, nová interakce, nové či změněné hodnocení atd.).

Doporučovací systém data z databáze pouze čte. Tato data používá pro natrénování algoritmu a při zpracování požadavku na doporučení pro uživatele z databáze získává jeho aktuální hodnocení, na základě kterých vybírá doporučení. Žádná data doporučovací systém do databáze nezapisuje.

Komunikace webové aplikace a doporučovacího systému probíhá na základě požadavku vyvolaného uživatelem ve webové aplikaci na jeho seznam doporučení. Doporučovací systém vypočítá tento seznam a vrátí ho webové aplikace, aby ta mohla doporučení prezentovat uživateli. Pravidla této komunikace jsou podrobně popsána v kapitole 3.2.2.

4.2 Základní datový model



Obrázek 4.2: Základní datový model

Před začátkem implementace webové aplikace a vytvořením databáze, byl navržen datový model, který je detailně vykreslen na diagramu na obr. 4.2. Základními entitami vycházející z používaného MovieLens25M datasetu, ale použitelné téměř pro jakoukoli doménu, kde se pracuje s doporučovacím systémem, jsou uživatel, položka a hodnocení.

4.2.1 Účet

Uživatele jsme oddělili od základní entity reprezentující účet, protože chceme používat i uživatele bez účtu, což jsou ti již nahraní z datasetu. Účet musí obsa-

hovat informace, které popisujeme v kapitole 2.2.4.2.

4.2.2 Uživatel

Uživatel je napojen na svůj případný účet pomocí uživatelského jména. Dále obsahuje informaci o čase prvního požadavku na doporučovací systém, což později využíváme při uživatelské studii (viz kapitola 5.1.8.2). Další vlastnosti uživatele, konkrétně jeho blokovací pravidla, poslední hodnoty filtru kritérií kvality doporučování a jeho nastavení, jsou uloženy v JSONu proto, aby se nemusel měnit model při případných změnách v aplikaci nebo změně celé domény.

4.2.3 Položka

Položka obsahuje informace, které by měly být zobrazeny uživateli a zároveň existují napříč doménami. Jde o jméno, obrázek, krátký popis a podrobnější popis. Poslední vlastnost JSONParams slouží pro uložení vlastností položky specifických pro doménu. V případě filmů jde např. o režiséra, žánry či herce.

4.2.4 Hodnocení

Vlastnosti hodnocení vychází z MovieLens25M datasetu, kdy opět očekáváme, že takový návrh entity hodnocení je použitelný i v jiných doménách. Konkrétně musí být hodnocení napojeno na uživatele a položku, obsahovat skóre a čas, kdy bylo uděleno.

4.2.5 Interakce

Hodnocení není jediným způsobem, jakým může dojít k interakci mezi uživatelem a položkou. Proto definujeme i obecnou entitu interakce, která opět musí být napojena na uživatele a položku. Dále musíme specifikovat typ interakce, jako je zobrazení položky uživatelem, nebo kliknutí na ni, počet takových proběhlých interakcí a čas poslední takové interakce. Potenciálně by mohl být pro každou interakci vytvořen nový záznam, ale bylo dostačující mít maximálně jeden záznam pro kombinaci uživatel - položka - typ interakce.

4.2.6 Doporučovací systém

Od doporučovacího systému vyžadujeme jeho jméno, dle kterého ho můžeme identifikovat a URI, na němž můžeme volat jeho API při požadavku na seznam doporučení. Poslední vlastnost s hyperparametry může být použita pro nastavení speciálních vlastností multi-objective doporučovacího systému mimo ty základní popsané v kapitole 3.2.

4.2.7 Kritérium kvality doporučování

Každé kritérium kvality doporučování musí být napojeno na doporučovací systém, ve kterém se používá. Může obsahovat také krátký popis kritéria a příklad jeho užití, což používáme v nápovědě pro uživatele. Součástí jsou taky výchozí vysvětlení doporučení na čtyřech úrovních dle skóre vráceného z doporučovacího

systému (podrobněji v kapitole 3.4.2.1). Tato vysvětlení jsou používaná, pokud kritérium nemá více variant metrik.

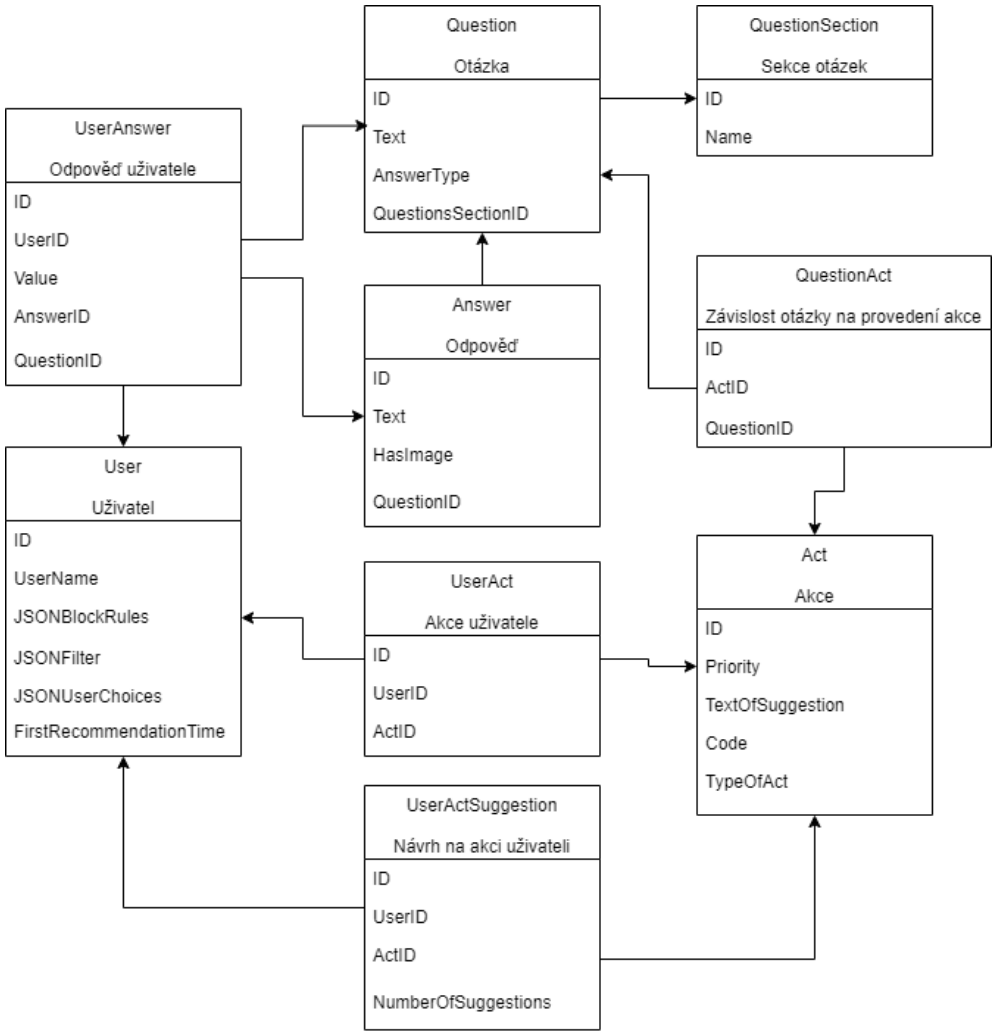
4.2.8 Varianta metriky

Tato entita reprezentuje variantu metriku, podle které se počítá skóre pro kritérium. Musí tedy obsahovat napojení na odpovídající kritérium, popis, který je opět použit v nápovědě. Je také specifikováno, zda je tato varianta metriky výchozí pro kritérium kvality doporučování, a i zde jsou zahrnuta vysvětlení pro všechny čtyři úrovně dle vypočteného skóre položky na základě této metriky. Zásadní vlastností varianty metriky je její kód, který je specifikován v požadavku na seznam doporučení tak, aby doporučovací systém věděl, kterou variantu metriky použít (viz kapitola 3.2).

4.2.9 Napojení uživatel - varianta metriky

Tato entita reprezentuje zvolenou variantu metriky uživatelem. Pro jedno kritérium kvality doporučování a jednoho uživatele musí existovat maximálně jeden takový záznam. Pokud neexistuje žádný, je používána výchozí varianta metriky.

4.3 Datový model pro uživatelskou studii



Obrázek 4.3: Datový model pro uživatelskou studii

Základní datový model byl dostatečný pro spuštění webové aplikace, které umožňovalo její funkčnost s procházením položek pomocí vyhledávání, nastavováním svých preferencí, doporučováním položek a dalšími akce mi podrobněji popsanými v kapitole 3.4.

Pro možnost provedení uživatelské studie jsme ale základní datový model museli rozšířit o další entity. Všechny přidané entity a jejich vzájemné napojení je viditelné na diagramu na obr. 4.3.

4.3.1 Otázka

Jelikož chceme uživatelskou studii zakončit dotazníkem, je nutné, aby model obsahoval entitu otázka. Otázka obsahuje text, typ odpovědi na ni (textově, Likertova škála, odpověď z databáze) a odkaz na sekci, do které patří.

4.3.2 Odpověď

Pokud se na otázku neodpovídá textově či pomocí Likertovy škály, musí být možné odpovědi na otázky specifikovány jako entity v datovém modelu. Odpověď obsahuje text, informaci, zda má být v dotazníku k odpovědi přidán obrázek pro jednoznačnou interpretaci uživatelem, a napojení na otázku, na niž může být tato odpověď použita.

4.3.3 Odpověď uživatele

Abychom byli schopni získat výsledky ze studie, musíme ukládat i odpovědi uživatele. Entita uživatelovy odpovědi musí být napojena samozřejmě na uživatele a otázku. Následně je vyplněna většinou jedna z následujících třech vlastností. Value, tedy hodnota, je použita pro otázky s odpovědí pomocí Likertovy škály, text pro otázky s textovou odpovědí a jinak je vyplněno ID odpovědi.

4.3.4 Sekce otázek

Otázky jsou sdruženy do sekcí. Tyto sekce jsou použity při průchodu uživatele závěrečným dotazníkem, kdy jsou v danou chvíli zobrazeny pouze otázky z jedné sekce. Jedinou vlastností sekce otázek je název.

4.3.5 Akce

Jelikož přístup k dotazníku podmiňujeme provedením několika akcí v aplikaci, musíme také definovat entitu akcí. O akci si uchováváme její prioritu, unikátní kód a typ akce, na základě něhož akce rozřazujeme do skupin. Podrobněji akce popisujeme v kapitole 5.1.8. Navíc ještě přidáváme text návrhu akce, který se uživateli zobrazí, pokud mu provedení této akce navrhujeme (viz 5.1.8.1).

4.3.6 Akce uživatele

Jelikož potřebujeme uložit, zda uživatel provedl akci, přidáváme i jednoduchou entitu s napojením na akci a uživatele.

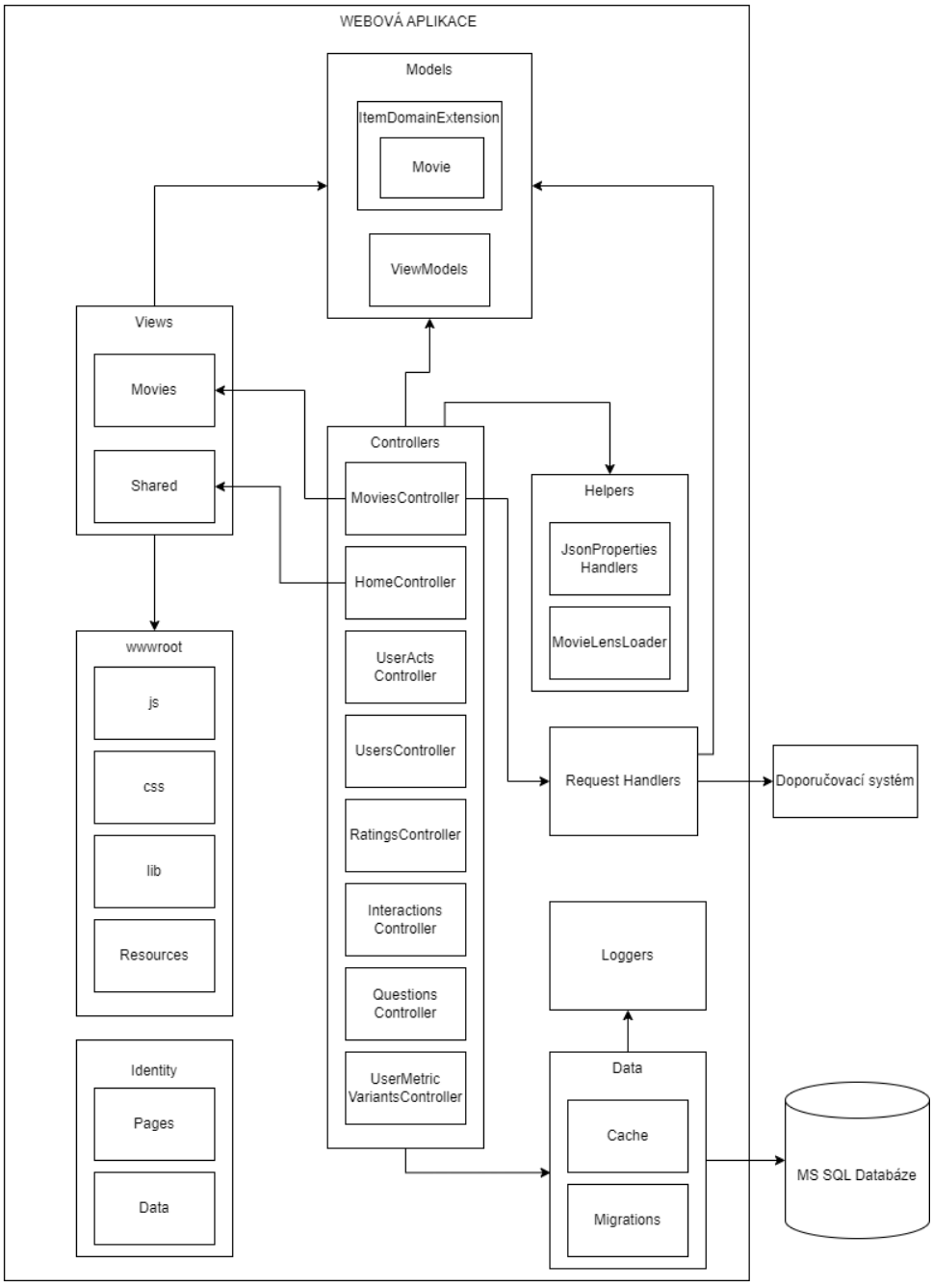
4.3.7 Návrh na akci uživateli

Vzhledem k tomu, že, jak zmiňujeme výše, od uživatele vyžadujeme provedení dostatečného počtu akcí, tak mu tyto akce potřebujeme přímo navrhovat a pamatovat si počet těchto návrhů, abychom častěji nabízeli dosud méně navrhované akce (podrobněji viz 5.1.8.1). Kromě napojení na akci a uživatele si tedy v této entitě ukládáme i počet návrhů této akce uživateli.

4.3.8 Závislost otázky na provedení akce

Zobrazení některých otázek z dotazníku uživateli je podmíněno provedením jedné či více akcí (viz kapitola 5.1.8), proto potřebujeme ukládat i tuto závislost v databázi. Tato entita obsahuje pouze napojení na akci a otázku.

4.4 Implementace webové aplikace



Obrázek 4.4: Rozdělení implementace webové aplikace

Pro implementaci byl použit framework ASP.NET Core MVC pro vytváření webových aplikací s využitím návrhového vzoru Model-View-Controller. Jak můžeme vidět na obr. 4.4, implementace webové aplikace je rozdělena do několika částí, které zároveň odpovídají adresářové struktuře. Nyní jednotlivé komponenty popíšeme.

4.4.1 Model

Model definuje datovou strukturu. Obsahuje všechny třídy z kapitol 4.2 a 4.3. Navíc obsahuje statickou třídu Movie, pomocí které může být položka (Item) interpretována jako film, což slouží k jednoduššímu přístupu k vlastnostem této entity specifických pro doménu filmů. Jelikož většina stránek, resp. jejich částí, vyžaduje více entit modelu, definujeme i ViewModels, které obsahují všechna potřebná data pro zobrazení jednotlivých stránek, jako je hlavní stránka, náhled položky atd.

4.4.2 Views - uživatelské rozhraní

Uživatelské rozhraní je tvořeno jednotlivými stránkami, resp. částmi stránek. Shared obsahuje ty (částečné) stránky , které nejsou závislé na doméně, což jsou např. filtr kritérií kvality doporučování, textové vyhledávání dle názvu atd. Movies obsahuje stránky specifické pro doménu filmů, jako jsou podrobnější filtr, detail položky atd.

4.4.3 Controllers - Kontrolery

Kontrollery obsahují řídící logiku aplikace. Zpracovávají požadavky uživatele, na základě nichž mění model a vrací uživateli stránku, resp. její část. Kontrollery jsou rozděleny podle části modelu dat, kterou mění nebo zobrazují uživateli. HomeController zpracovává požadavky uživatele na celé stránky (domovská stránka, stránka pro správu blokovacích pravidel, stránka s uživatelským nastavením a stránka s dotazníkem), případně statické části stránek (nápovědy atd.). MoviesController zpracovává úkony, které jsou závislé na doméně filmů.

4.4.4 wwwroot - Javascript, CSS a statické soubory

V používaném frameworku slouží tato část aplikace pro uložení Javascriptu, CSS a statických souborů, jako jsou obrázky. Konkrétně tedy obsahuje naši implementaci Javascriptu a CSS používanou na frontendu, uložené obrázky, které chceme zobrazovat na webu a javascriptové a CSS knihovny.

4.4.5 Identity

Identity je vygenerovaná komponenta starající se o autorizaci a autentizaci uživatele. Část pages obsahuje stránky s registrací a přihlášením a jejich logikou, Data obsahují pouze model pro entitu účtu.

4.4.6 RequestHandlers - Obsluhy žádostí

Tato část implementace se stará o zpracování žádosti na doporučení položek, případného odeslání požadavku na doporučovací systém, zpracování jeho odpovědi a vrácení doporučených položek.

4.4.7 Helpers

Helpers obsahuje třídy, které tvoří fasádu pro zpracování JSON vlastností z datového modelu položek a uživatelů. Součástí je také načítání dat z MovieLens datasetu do databáze včetně obohacení o data z The Movie Database a filtrování (viz 3.1). Pro potřeby uživatelské studie je zde obsažena i třída UserActHelper starající se o zpracování akcí uživatele a jejich navrhování.

4.4.8 Data

Tato část aplikace se stará výlučně o práci s daty. Jednak vytváří instanci relace připojení k databázi, která následně obsluhuje dotazy do databáze. Dále obsahuje ještě Cache, která zařizuje uložení některých dat, ke kterým se často přistupuje, přímo v mezipaměti aplikace. Součástí je také složka Migrations obsahující vygenerované soubory reprezentující aktualizace databáze.

4.4.9 Loggers

Součástí aplikace jsou Loggers obsahující třídu pro logování událostí aplikace a uživatele do souboru.

4.4.10 Settings - Nastavení

Komponenta Settings není na diagramu na obr. 4.4 zmíněna, protože s ní komunikují téměř všechny části kódu. Obsahuje totiž jednak definici enum tříd, které reprezentují jednotlivé varianty nastavení uživatele (mechanismus pro filtr kritérií kvality doporučování, typ vizualizace skóre v rámci vysvětlení doporučení atd.), dále výběr výchozích nastavení pro uživatele, což se využívá při registraci nového účastníka studie, a seznam systémových parametrů, které platí napříč celou aplikací. Takovými parametry jsou např. počet doporučených položek, který chceme vrátit doporučovacím systémem, název doménového kontroleru nastavený na základě používané domény atd.

5. Uživatelská studie

V této kapitole popisujeme, jakým způsobem probíhal experiment v podobě uživatelské studie, a poté představíme z něj vyplývající výsledky.

5.1 Provedení

V popisu způsobu provedení studie zmíníme, jak byla uživatelská studie účastníkům k dispozici, jaké údaje byly vyžadovány, jak byli prezentováni kandidáti, která data jsme sbírali, na co jsme se ptali v dotazníku a jak celá studie probíhala z pohledu účastníka.

5.1.1 Přístup

Webová aplikace byla nasazena na veřejně přístupnou adresu tak, aby k ní měli účastníci jednoduchý přístup přes webový prohlížeč a mohli provést a vyplnit uživatelskou studii např. z domova. Laboratorní studie, kde bychom chování účastníků při práci sledovali, nebyla pro náš výzkum nutná.

5.1.2 Identifikace účastníka

Vzhledem k tomu, že webová aplikace je koncipována tak, aby více připomínala skutečné portály využívající doporučovací systémy, vyžadovali jsme nejdříve po účastníkovi studie registraci. Při ní jsme vyžadovali e-mail a přihlašovací údaje v podobě uživatelského jména a hesla.

E-mail byl zvolen jako údaj, na základě kterého můžeme zevrubně zkontrolovat, že jde o skutečného účastníka, a v případě pochybností jej kontaktovat. Získání dalších údajů by pro případné ověření uživatele bylo vhodné, ale mohlo by jej od účasti v uživatelské studii odradit, proto jsme od něj upustili.

5.1.3 Způsob prezentace kandidátů

Porovnáváme různé varianty následujících prvků aplikace, konkrétně jde o varianty filtru kritérií kvality doporučování (viz kapitola 3.4.1.6), náhledu explanations (viz kapitola 3.4.2.2.1), počtu vysvětlených kritérií (viz kapitola 3.4.2.2.2), vizualizace skóre v rámci explanations (viz kapitola 3.4.2.2.3) a metrik pro relevanci, diverzitu, novelty a popularitu (viz kapitola 3.3.1).

Vzhledem k tomu, že se zaměřujeme hned na několik porovnání různých kandidátů, není vhodné tyto kandidáty uživateli prezentovat současně. Zároveň s ohledem na počet kandidátů a očekávaný počet účastníků, nepřipadá v úvahu, pokud chceme získat rozumné výsledky, abychom účastníkům prezentovali jen jednoho kandidáta pro každé porovnání. Proto by měl každý uživatel mít možnost vyzkoušet všechny kandidáty, přestože to pro něj může být náročné.

Kromě volby metrik pro každé kritérium, mohou uživatelé měnit varianty ostatních prvků v rámci uživatelského nastavení (viz kapitola 3.4.1.8 a 3.4.2.2.4). Výběr metriky probíhá po kliknutí na název kritéria ve filtru kritérií kvality doporučování (viz kapitola 3.4.1.7).

Stále je ovšem třeba vyřešit, jak přiřazovat první variantu z každého porovnání. Nejspravedlivější rozdělení mezi účastníky by proběhlo na základě použití všech kombinací jednotlivých kandidátů. Tato varianta je nepoužitelná, protože takových rozřazení prvních variant bychom měli v řádu tisíců, zatímco počet účastníků očekáváme v řádu desítek. Proto používáme jen různé kombinace metrik kritérií kvality doporučování, kde očekáváme potenciálně větší vliv první používané metriky na rozhodování uživatele při porovnání všech metrik kritéria. První použité varianty pro další porovnání jsou vybírány náhodně, jelikož zde očekáváme menší závislost na prvním používaném kandidátovi a větší schopnost uživatele rychle vyhodnotit, kterou z variant považuje za vhodnější, než je tomu u metrik kritérií kvality doporučování.

5.1.4 Sběr dat

O uživateli si do databáze ukládáme všechna jeho hodnocení, interakce (zobrazení a kliknutí), provedené akce, návrhy na akce a odpovědi na otázky v závěrečném dotazníku. Zároveň jsou uloženy jeho právě používané varianty filtru kritérií kvality doporučování, vizualizace explanations (náhled, skóre, počet kritérií), také jeho pravidla pro blokování a používané metriky pro každé kritérium. S výjimkou hodnocení, které uživatel ovšem může odebrat či změnit, a interakce, kde ale ukládáme pouze čas poslední takové interakce, si neukládáme čas, kdy záznam vznikl.

Proto navíc logujeme všechny zásadní činnosti uživatele i s časem, kdy jej provedl, abychom mohli rekonstruovat, jak probíhala jeho práce v rámci uživatelské studie. Konkrétně zapisujeme do logovacích souborů všechny následující záznamy o jeho působení v aplikaci:

- Přidání blokovacího pravidla
- Odebrání blokovacího pravidla
- Přidělení hodnocení
- Proběhlá interakce
- Zvolená odpověď na otázku v dotazníku
- Provedená akce
- Návrh na možnou akci
- Požadavek na doporučovací systém včetně používané metriky každého kritéria, používaného filtru kritérií kvality doporučování a přidělených vah každému kritériu

5.1.5 Dotazník

Na závěr uživatelské studie účastník vyplňuje dotazník. Otázky v dotazníku můžeme rozdělit na ty, na které účastník studie odpovídá mírou souhlasu s tvrzením pomocí tzv. Likertovy škály. Námi používaná Likertova škála se skládá z následujících odpovědí:

- Strongly agree (Rozhodně souhlasím)
- Agree (Souhlasím)
- Neutral / Don't know (Neutrální / Nevím)
- Disagree (Nesouhlasím)
- Strongly disagree (Rozhodně nesouhlasím)

Dotazník také obsahuje kontroly pozornosti, abychom dokázali vyloučit odpovědi účastníka, o kterém se můžeme domnívat, že se dostatečně nesoustředil při vyplňování dotazníku. U některých otázek navíc zobrazujeme kromě textu možné odpovědi i obrázek této varianty, aby uživatel správně chápal možné odpovědi i v případě nespojení si varianty s jejím názvem.

Otázky dělíme do sekcí a uživateli jsou vždy naráz zobrazeny jen otázky z dané sekce. Všechny otázky jsou vypsány v příloze A.1, a to včetně možných odpovědí na ně. Protože uživatelské rozhraní studie je lokalizováno do angličtiny, lze najít v příloze A.2 seznam otázek a možných odpovědí také v angličtině.

5.1.6 Průběh

Nyní popíšeme průběh studie z pohledu účastníka. Nejdříve je nutná registrace, bezprostředně po úspěšné registraci je účastníkovi studie zobrazena nápověda. Od uživatele pak vyžadujeme tři pozitivní hodnocení a poté už funguje doporučování multi-objective doporučovacího systému. Od účastníka studie je vyžadováno splnění několika akcí, ty jsou mu navíc navrhovány v okně nahoře na hlavní stránce. Jakmile jich provede dostatečný počet, je mu zpřístupněn dotazník. Po vyplnění dotazníku je studie z pohledu účastníka hotová. Nyní popíšeme jednotlivé fáze průběhu studie o něco podrobněji.

5.1.6.1 Registrace

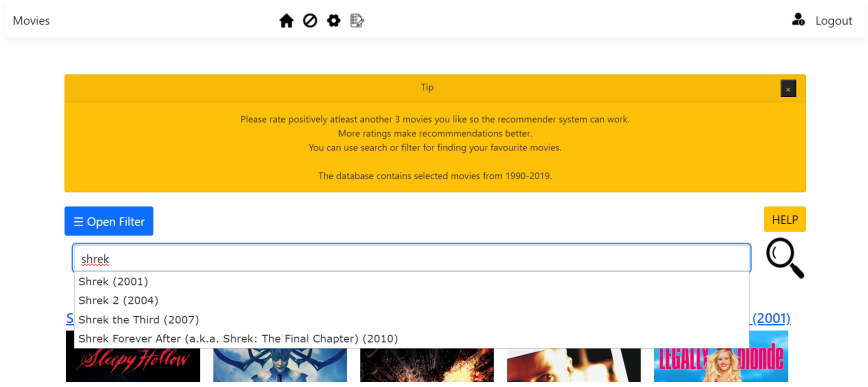
Každý nepřihlášený uživatel je po načtení webu přesměrován na stránku přihlášení. Účastník se následně pomocí odkazu pro uživatele bez vytvořeného účtu překlikne na stránku registrace. Poté vyplní následující údaje: e-mail, uživatelské jméno a heslo.

5.1.6.2 Nápověda

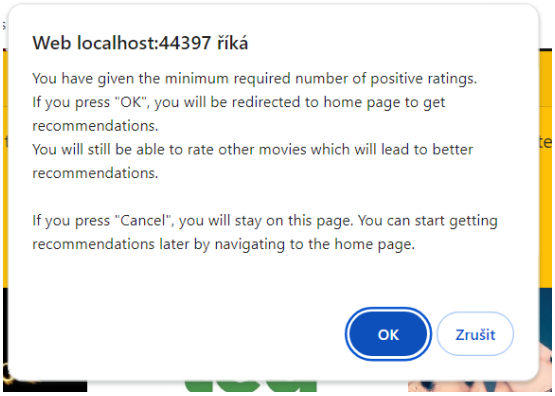
Ihned po registraci a úspěšném vytvoření nového uživatele je účastník studie přesměrován na nápovědu. Zde je popsáno, co lze kde na stránce najít, a také jsou v tomto manuálu vysvětleny kroky pro splnění uživatelské studie. Vše je doplněno snímky obrazovky pro urychlení pochopení uživatelem.

5.1.6.3 První hodnocení

Vzhledem ke způsobu návrhu multi-objective doporučování je doporučovací systém schopen zpracovávat požadavky jen od uživatele, který již nějaké položky ohodnotil. Proto nejdříve vyžadujeme od účastníka studie 3 pozitivní hodnocení,



Obrázek 5.1: Hlavní stránka po prvním načtení uživatelem, který ještě nehodnotil žádný film.



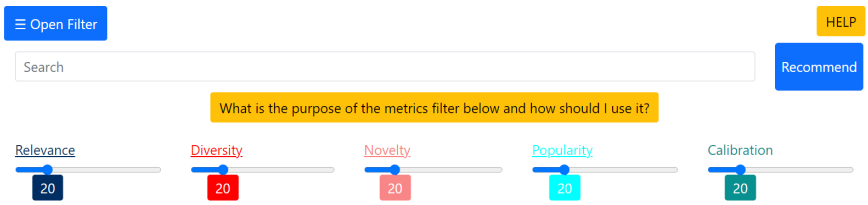
Obrázek 5.2: Dialogové okno v české lokalizaci prohlížeče, které se zobrazí bezprostředně po třetím pozitivním hodnocení uživatele.

což jsme na základě otestování zvolili jako minimální kvótu pro rozumné doporučování. Uživateli jsou nabídnuty náhodně filmy z 500 nejčastěji hodnocených, nicméně může již využít ať už textové vyhledávání dle názvu filmu tak podrobnější filtr. Jak vypadá stránka ihned po prvním načtení uživatelem, je viditelné na obr. 5.1.

Jakmile uživatel rozdá vyžadovaná pozitivní hodnocení, vyskočí mu dialogové okno (viz obr. 5.2), ve kterém je účastníkovi studie oznámeno, že už doporučování může fungovat, a je dotázán, zda ho chce začít používat. Pokud potvrdí, je hlavní stránka přenačtena a načtené filmy jsou už výstupem doporučovacího systému. V opačném případě, může pokračovat ještě v procházení a ohodnocení dalších filmů a v dialogovém okně je mu oznámeno, že doporučování spustí po přenačtení hlavní stránky.

5.1.7 Doporučování

Účastník studie nyní může měnit pomocí filtru kritérií kvality doporučování váhy pro tato kritéria a měnit tak seznam doporučení. Uživatel má taky bezprostředně nad filtrem kritérií kvality doporučování tlačítko s nápovědou k tomuto filtru včetně vysvětlení kritérií a způsobu, jakým může přenastavovat důležitost těchto kritérií. Po nastavení vah může uživatel získat doporučení kliknutím na tla-



Obrázek 5.3: Základní rozhraní hlavní stránky pod hlavním menu a nad doporučenými filmy. Obsahuje 2 typy vyhledávání, filtr kritérií kvality doporučování, nápovědu k němu, tlačítko „RECOMMEND“ a tlačítko „HELP“

čítko „RECOMMEND“. Uživatel také může otevřít opět manuál pomocí kliknutí na tlačítko „HELP“. Výše popsané základní rozhraní je zobrazeno na obr. 5.3.

5.1.7.1 Nápověda ke kritériím kvality doporučování

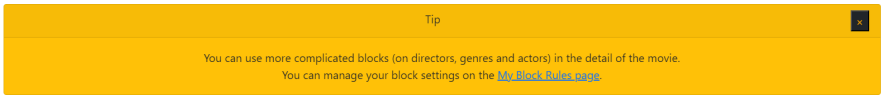
Zároveň uživatel může zobrazit nápovědu po kliknutí na tlačítko bezprostředně nad filtrem kritérií kvality doporučování viditelné na obr. 5.3, kde jsou popsána právě tato kritéria. Konkrétní texty této nápovědy jsou vypsány v příloze A.4.

5.1.8 Akce

Následně od uživatele vyžadujeme provedení několika akcí pro to, aby mu byl povolen přístup do závěrečného dotazníku. Účastník studie nemusí provést zcela všechny akce, ale pak mu nejsou zobrazeny některé otázky. Důvodem pro omezení nutnosti provedení všech akcí je už tak náročný průběh studie pro průměrného uživatele a vysoký počet informací, které si účastník musí zapamatovat, aby byl schopen odpovědět na většinu otázek.

Akce dělíme do skupin a platí, že každá akce ve skupině má stejnou prioritu. Zároveň je většinou každá skupina navázána na zobrazení alespoň jedné otázky z dotazníku. V příloze A.3 popisujeme a vyjmenováváme akce pro každou ze skupin včetně otázek, jejichž zobrazení je na provedení všech akcí ze skupiny závislé.

5.1.8.1 Návrhy na akce



Obrázek 5.4: Tip na akci, kterou by měl uživatel provést zobrazující se nahoře na hlavní stránce.

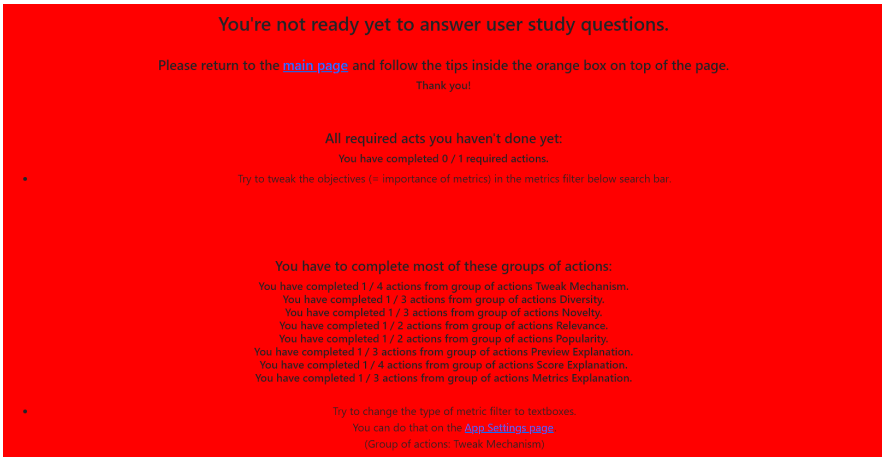
Tyto akce jsou uživateli nabízeny vždy nahoře na hlavní stránce (viz obr. 5.4). Výběr navrhované akce probíhá následovně. Nejdříve pro každou akci vypočítáme váhu nesplněné akce w_a takto:

$$w_a = \frac{\max_{a' \in A} priority(a') - priority(a) + 1}{2^{suggestions(a)}}$$

kde A je seznam všech nesplněných akcí, *priority* je funkce určující prioritu akce a *suggestions* je funkce, která vrací počet navržení akce, tzn. kolikrát byla tato akce uživateli zobrazena v rámci oranžového okna nahoře na hlavní stránce.

Navrhovaná akce je následně vybrána na základě vážené náhodné volby, kdy akce a má $\frac{w_a}{w_{a'}}$-krát větší šanci být zvolena než a'.

5.1.8.2 Připuštění k dotazníku



Obrázek 5.5: Stránka s dotazníkem v době, kdy uživatel nesplnil dostatečný počet akcí, aby mohl být k dotazníku připuštěn.

Jak již zmiňujeme výše, nevyžadujeme od uživatele splnění všech akcí, aby byl připuštěn k dotazníku. Uživatel je informován o tom, jaké akce má plnit jednak výše zmíněným návrhem akce na hlavní stránce, ale také na stránce samotného dotazníku (viz obr. 5.5).

Účastník studie ovšem musí splnit každou akci s prioritou 1. V této studii byla pouze jedna taková akce, a to použití filtru kritérií kvality doporučování. Akce s prioritou vyšší jak 3 jsou naopak nepovinné.

Otázkou tedy zůstává, jaký počet akcí vyžadovat z těch s prioritou 2 a 3. Nejdříve jsme určili, že míra splnění bude záviset na 3 proměnných:

- Počet splněných skupin akcí s prioritou 2 (P_2)
- Počet splněných skupin akcí s prioritou 3 (P_3)
- Čas strávený v uživatelské studii v minutách ($Time$)

Pro dvě úvodní proměnné využíváme toho, že akce v jedné skupině mají stejnou prioritu. Proměnnou navíc nerozmělňujeme na počet jednotlivých akcí, protože až splnění celé skupiny akcí vede k zobrazení otázky závislé na těchto akcích, což je to, co nás skutečně zajímá. Čas, na základě kterého chceme zjistit, že uživatel skutečně se systémem dostatečně dlouho pracoval, měříme od prvního doporučení doporučovacím systémem, což je doba, kdy uživatel pracuje se systémem a jeho prvky, které nás zajímají.

V návrhu pravidel pro připuštění k dotazníku na základě hodnot výše zmíněných 3 proměnných jsme vycházeli z Łukasiewiczovy t-normy definované následovně:

$$T_{LUK}(a,b) = max(0, a + b - 1)$$

Protože t-norma je funkce pracující na intervalu $[0; 1]$, museli jsme všechny proměnné nejdříve přeškálovat na tento interval, což provádíme takto:

$$P_2' = P_2 \,/\, |G_{P_2}|$$

$$P_3' = P_3 \,/\, |G_{P_3}|$$

$$Time' = \min(30, Time) \,/\, 30$$

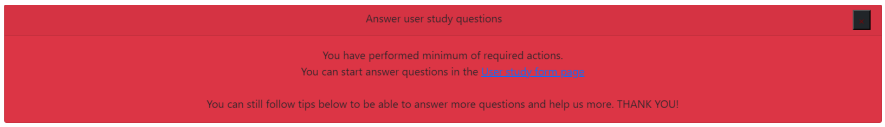
kde G_{P_2} je množina skupin akcí s prioritou 2 a G_{P_3} množina skupin akcí s prioritou 3. 30 minut volíme jako dobu, kdy očekáváme, že je účastník studie schopen splnit téměř všechny nutné akce.

Łukasiewiczovu t-normu nicméně tak, jak je definována, nemůžeme použít ze dvou důvodů. Tím prvním je počet proměnných, druhým je nemožnost přidělení větší váhy k provedení skupin akcí s prioritou 2 než k provedení těch s nižší prioritou 3. Proto vytvoříme metriku míry splnění T, která z Łukasiewiczovu t-normy pouze vychází a sama už t-normou není. Metriku míry splnění T definujeme následovně:

$$T = T_{LUK}(0.8 \times P_3', T_{LUK}(Time', 1.2 \times P_2'))$$

Uživatel je následně připuštěn k dotazníku, pokud $T > 0$. Koeficienty jsme zvolili na základě porovnání různých variant těchto koeficientů, kdy s hodnotami 0.8 a 1.2 metrika míry splnění T dosahovala kladných čísel po provedení dostatečného počtu skupin akcí.

Účastník studie je následně informován o povoleném přístupu k dotazníku v červeném okně nahoře na hlavní stránce (viz obr. 5.6).

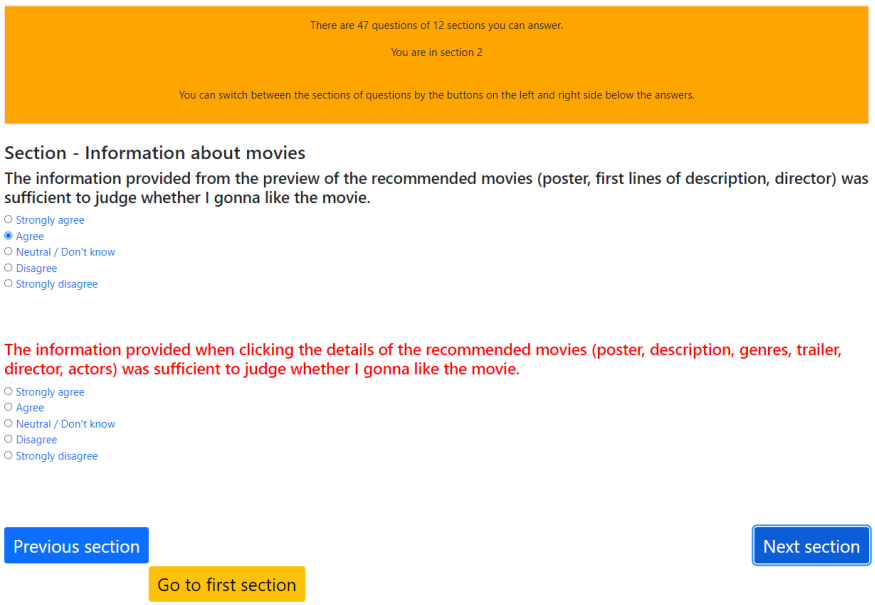


Obrázek 5.6: Informace o připuštění k dotazníku zobrazující se nahoře na hlavní stránce.

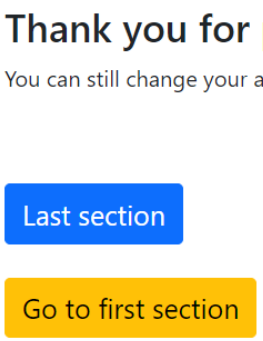
5.1.8.3 Vyplnění dotazníku

Uživatel následně prochází všechny otázky postupně přes jednotlivé sekce (viz kapitola 5.1.5). Účastník studie je nahoře na stránce informován o počtu sekcí a otázek a o sekci, ve které se právě nachází. Odpověď na všechny otázky právě zobrazené sekce je podmínkou připuštění k sekci následující. V případě, že uživatel na některou z otázek neodpoví, je označena červeně (viz obr. 5.7).

Jakmile, uživatel projde všechny sekce a odpoví na všechny otázky, je mu tato informace oznámena (viz obr. 5.8). Účastník studie se následně ještě může vrátit do jednotlivých sekcí a své odpovědi změnit. Pokud po vyplnění dotazníku, provede uživatel další akce v systému, které umožňují odpovědět na dříve nezobrazené otázky, jsou tyto otázky zobrazené v odpovídající sekci a také na závěrečné stránce tohoto dotazníku s informací o splnění studie.



Obrázek 5.7: Vyplňovaní dotazníku. Uživatel chtěl přejít do následující sekce, aniž by zodpověděl všechny otázky zobrazené sekce, proto je nezodpovězená otázka označena červeně.

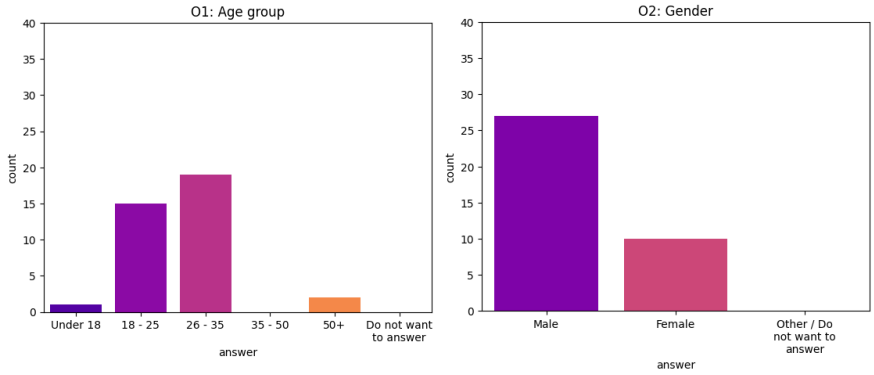


Obrázek 5.8: Informace o splnění uživatelské studie po zodpovězení na všechny otázky.

5.2 Výsledky

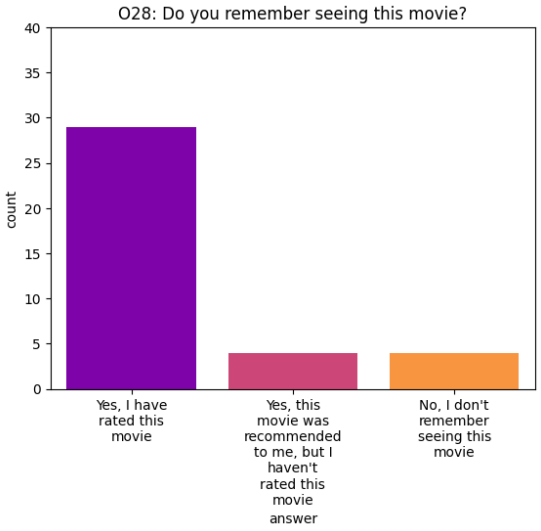
Uživatelské studie se zúčastnilo 52 lidí. Studii dokončilo 45 účastníků. Z důvodu chyby při vyloučení vícekrát zobrazených položek z doporučování pracovali uživatelé s variabilním počtem filmů. Výrazný problém s malým počtem položek k doporučení nastal až u posledního účastníka, jehož jsme museli vyřadit. Ostatní uživatelé pracovali s minimálně 350 filmy. Všech 44 zbylých účastníků správně odpovědělo na druhou kontrolu pozornosti (otázka O35), ale pouze 37 správně odpovědělo na první kontrolu pozornosti (otázka O15). Odpovědi 7 účastníků, kteří tímto ověřením neprošli, jsme z výsledků vyřadili. Nezanedbatelný počet vyřazených účastníků značí, že uživatelská studie byla pro některé z nich příliš náročná či dlouhá a stálo by za zvážení rozdělení uživatelské studie do více experimentů.

Věkové složení účastníků, kteří splnili všechny podmínky, a jejich rozdělení dle



Obrázek 5.9: Rozdělení účastníků studie dle věku a pohlaví.

pohlaví je viditelné na obr. 5.9. Zároveň jsme ještě všechny účastníky v otázce O28 zkoušeli, zda si pamatují jeden z filmů, který hodnotili. A to proto, abychom získali představu, v jaké míře si při práci ve webové aplikaci byli schopni své úkony zapamatovat. Výsledky tohoto testu lze pozorovat na obr. 5.10.



Obrázek 5.10: Kontrola pozornosti účastníka studie na to, zda si pamatuje film, který hodnotil (otázka O28).

Ve výsledcích závislých na dotazníku, bereme v úvahu pouze odpovědi těch uživatelů, které splnili kontroly pozornosti v rámci otázek O15 a O35. Data o chování účastníků studie v průběhu práce se systémem zvažujeme u všech účastníků, kteří se dostali až k dotazníku.

Různé počty odpovědí na jednotlivé otázky jsou zapříčiněny faktem, že zobrazení některých z nich je závislé na provedení skupiny akcí, což podrobněji popisujeme v kapitole 5.1.8.2.

Je také potřeba zmínit, že u otázek, na které lze odpovědět Likertovou škálu převádíme odpovědi na číselné hodnocení takto:

- Strongly agree : 1
- Agree : 0,5

- Neutral / Don't know : 0
- Disagree : -0,5
- Strongly disagree : -1

Tento převod nám umožňuje vyjádřit v grafech odpovědi účastníků studie pomocí vizualizace průměru a směrodatné odchylky číselných hodnot odpovědí, jak můžeme vidět např. na obr. 5.11.

5.2.1 Data o průběhu práce

	Průměr	**Směrodatná odchylka**
Čas od prvního požadavku na doporučení po první odpověď v dotazníku v minutách	25,71	14,81
Čas od prvního požadavku na doporučení po poslední odpověď v dotazníku v minutách	35,67	16,10
Počet odeslaných požadavků na doporučovací systém	39,39	29,32
Poměr splněných skupin akcí o prioritě 1	1,00	0,00
Poměr splněných skupin akcí o prioritě 2	0,78	0,16
Poměr splněných skupin akcí o prioritě 3	0,74	0,28
Poměr splněných skupin akcí o prioritě 4	0,68	0,28
Poměr splněných skupin akcí o prioritě 5	0,75	0,44

Tabulka 5.1: Základní statistiky práce účastníka studie se systémem

Předtím, než se dostaneme k samotným výsledkům studie, představíme některé základní statistiky, které nastíní, jak účastníci studie pracovali se systémem. Všechny popisované statistiky jsou k nalezení v tabulce 5.1.

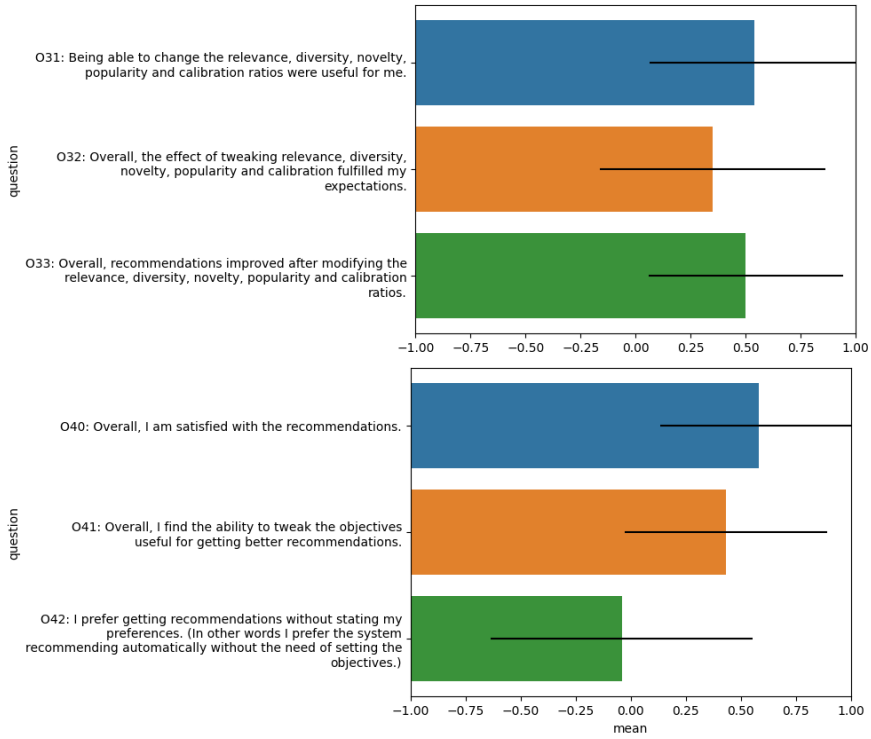
Nejdříve se podíváme na to, jak dlouho uživatelé ve studii trávili, kde jsme brali v úvahu jen ty, kteří si práci ve studii nerozdělili, což jsme filtrovali na základě maximální doby provedení 2 hodiny. Jak můžeme vidět v tabulce 5.1, uživatelé v průměru strávili prací se systémem 26 minut a zhruba dalších 10 minut jim trvalo vyplnění dotazníku. Námi prezentovaný odhadovaný čas studie jejím účastníkům 30 minut byl tedy lehce nižší, než ukázala skutečnost, ale i tak poměrně přesný.

Další statistikou, která nás zajímá, je počet odeslaných požadavků na doporučovací systém, které byly provedeny při načtení hlavní stránky nebo po stisknutí tlačítka „RECOMMEND“ na hlavní stránce. Zde vidíme, že průměrný účastník odeslal 39 požadavků, což odpovídá celkovým 585 doporučením vrácených doporučovacím systémem. Je si ale potřeba všimnout vysoké směrodatné odchylky, která značí, že počty požadavků na doporučovací systém se mezi účastníky lišily poměrně výrazně.

Od každého účastníka jsme vyžadovali splnění několika skupin akcí. Každá skupina akcí má svou prioritu od 1 (nejvyšší) po 5 (nejnižší), podrobněji akce popisujeme v kapitole 5.1.8. Všichni účastníci, kteří se dostali k vyplňování dotazníku, logicky provedli jedinou skupinu akcí s prioritou 1, protože to bylo nutnou

podmínkou připuštění k dotazníku. Zajímavější jsou hodnoty u dalších priorit, kdy poměr splněných skupin akcí s prioritou 2 a 3 rozhodoval společně se stráveným časem o připuštění k dotazníku (viz kapitola 5.1.8.2). U priority 2 pozorujeme průměrné splnění asi 4 z 5 skupin akcí, u priority 3 více než 2 ze 3 skupin akcí. Zbývající skupiny akcí již neměly vliv na připuštění účastníka studie k závěrečnému dotazníku. U priority 4 v průměru pozorujeme provedení více než 3 z 5 skupin akcí a vzhledem k tomu, že máme jedinou skupinu akcí s prioritou 5, tak hodnota průměrů udává, že ji 75 % účastníků studie provedlo.

5.2.2 RQ1: Stojí uživatelé o nastavení svých preferencí k jednotlivým kritériím kvality doporučování?



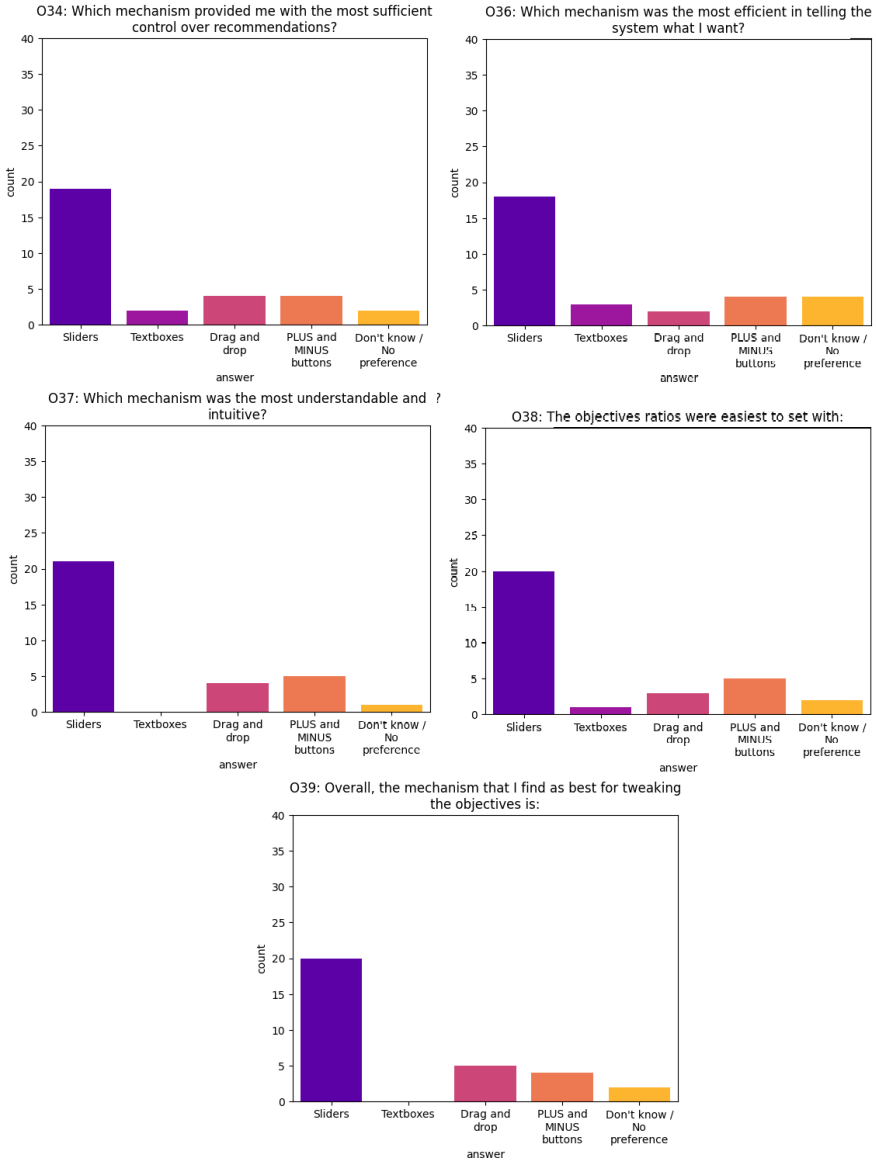
Obrázek 5.11: Odpovědi účastníků na otázky O31 - O33 a O40 - O41 s celkovým hodnocením.

Nejdříve se podíváme na odpovědi účastníků studie na otázky O31 - O33 a O40 - O41. Obecně panovala mezi uživateli spokojenost s doporučeními, jak můžeme pozorovat na obr. 5.11. Je také patrné, že účastníci v drtivé většině udávali, že možnost nastavení vah ke kritériím pro ně byla užitečná, o něco méně z nich vidí její užitečnost pro vylepšení doporučování. Většina uživatelů také pozoruje s nastavením vah k jednotlivým kritériím zlepšení doporučování (viz obr. 5.11). Tyto výsledky potvrzují hypotézu H1.1.

Ne již tak zcela jednoznačné rozložení odpovědí jsme získali na otázku O42, zda uživatelé preferují, když systém vybírá doporučení automaticky, aniž by museli specifikovat své preference pomocí nastavení vah k jednotlivým kritériím, což můžeme také pozorovat na obr. 5.11. S tímto tvrzením více účastníků nesouhlasí, než souhlasí, velký počet z nich si ovšem není jistý. Čistě na základě odpovědí v dotazníku můžeme označit hypotézu hypotézu H1.2 jako mylnou.

Je ale potřeba vzít v úvahu také jiné podmínky při uživatelské studii a nasazením takového systému na produkci, kdy u účastníků studie očekáváme větší trpělivost a větší tendenci vyvinout větší úsilí než poté u uživatelů v reálném prostředí. Proto odpověď na to, co je skutečně preferováno, není jednoznačná a spíše bychom doporučili možnost nastavení svých preferencí ke kritériím kvality doporučování zahrnout tak, aby si ji uživatelé, co o ni stojí, mohli aktivovat, ovšem nenastavit jako povinnou nebo výchozí variantu.

5.2.3 RQ2: Jaký mechanismus pro nastavení vah jednotlivým kritériím kvality doporučování považují uživatelé za nejvhodnější?



Obrázek 5.12: Odpovědi účastníků na porovnání mechanismů filtru kritérií kvality doporučování (otázkyO34 a O36 - O39).

Pro zjištění preferencí uživatele k variantám mechanismů pro filtr kritérií kva-

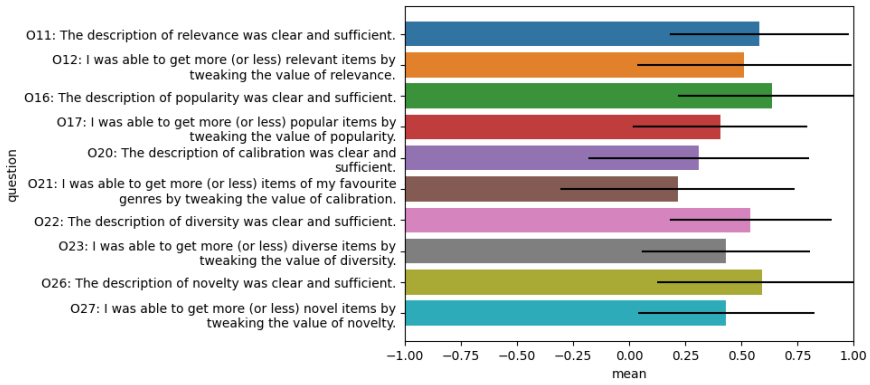
lity doporučování, jsme se ptali na otázky O34 a O36 - O39.

Z obr. 5.12 je patrné, že jako nejlepší mechanismus pro nastavení vah ke kritériím kvality doporučení účastníci studie hodnotí posuvníky (sliders), a to ve všech aspektech, na které jsme se dotazovali. V celkovém porovnání označuje posuvníky jako nejlepší 20 účastníků studie, 5 se přiklání k variantě drag and drop, 4 volí tlačítka + a - a 2 z nich nemají preferenci k žádné z možností.

Hypotézu H2.1 tak můžeme označit jako zcela mylnou a pro další výzkum v této oblasti doporučit používání posuvníků, tzn. sliderů.

5.2.4 RQ3: Jaká kritéria kvality doporučování jsou pro uživatele přínosná?

Nejprve jsme se zaměřili na to, zda uživatelé vůbec rozumí použitým kritériím kvality doporučování a zda pozorují jejich vliv na seznam doporučení. Uživatelé mohli kritériu rozumět ještě před účastí ve studii, nebo jej pochopit na základě textových explanations (viz kapitola 3.4.2.1) či nápovědy (viz kapitola 5.1.7.1). Vliv kritérií mohli účastníci studie posoudit na základě změny jejich vah ve filtru kritérií kvality doporučování a následným pohledem na seznam doporučených filmů.



Obrázek 5.13: Odpovědi účastníků ke kritériím kvality doporučování, konkrétně otázky k relevanci O11 - O12, popularitě O16 - O17, kalibraci O11 - O12, diverzitě O22 - O23, a novelty O26 - O27.

Relevanci účastníci povětšinou rozuměli, což můžeme pozorovat na obr. 5.13 (otázka O11), kdy drtivá většina účastníků studie udává, že tomuto kritériu rozumí. Zároveň i velká část z nich pozoruje vliv změny váhy relevance na relevanci doporučení (otázka O12).

Velmi podobné výsledky jsme získali i u diverzity, a to především u porozumění kritériu kvality doporučování, na které jsme se ptali v otázce O22 (viz obr. 5.13). Co se týče schopnosti systému reagovat na změnu váhy diverzity, účastnici ji opět z většiny pozorují, byť oproti relevanci stoupá počet uživatelů, kteří si nejsou jistí, a naopak klesl počet těch, kteří rozhodně souhlasí, že dokázali na základě nastavení váhy získat více či méně diverzní seznamy doporučení (otázka O23).

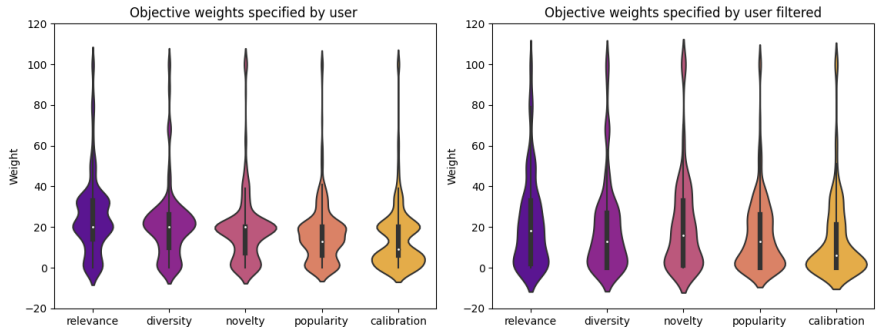
U novelty můžeme na obr. 5.13 vidět větší počet účastníků, kteří si nejsou jistí, zda byl pro ně popis tohoto kritéria jednoznačný a dostatečný (otázka O26). Schopnost ovlivnit seznam doporučení váhou přidělenou k novelty opět většina

uživatelů pozoruje (viz obr. 5.13 s odpověďmi na otázku O27), byť počet těch, co rozhodně souhlasí, je i zde menší než u relevance. Na druhou stranu počet účastníků, kteří nesouhlasí, nebo si nejsou jistí, je nižší, než je tomu u diverzity.

Popularita je známější pojem, než jsou názvy ostatních kritérií, což potvrzují i naše výsledky u otázky O16, kdy téměř všichni účastníci rozumí tomu, co popularita představuje (viz obr. 5.13). Na stejném obrázku je také vidět, že u otázky O17 opět většina účastníků udává, že dokáže nastavením váhy popularity ovlivnit doporučení více či méně populárních filmů.

Kalibrace je dle očekávání nejméně chápané kritérium i mezi účastníky studie, což je opět zřetelné na obr. 5.13. I tak ovšem většina z nich udává, že popis tohoto kritéria byl jednoznačný a dostatečný k pochopení. U tohoto kritéria téměř polovina uživatelů nedokáže říci, zda se jim pomocí změn vah kalibrace dařilo získávat méně nebo více kalibrované seznamy doporučení.

Pokud tedy shrneme odpovědi na dvě zmiňované otázky přes jednotlivá kritéria, můžeme říct, že uživatelé jsou i na základě stručného popisu schopni dobře porozumět všem pěti kritériím. Účastníci studie také pozorovali, že jsou schopni ovlivnit výsledný seznam doporučení pomocí nastavení vah relevanci, diverzitě, novelty a popularitě. Jen u kalibrace není uživatelům jasné, zda váha opravdu přispívá, resp. nepřispívá ke kalibrovanějšímu seznamu doporučení. Zůstává otázkou ovšem, zda je to způsobeno jen složitějším konceptem oproti ostatním kritériím nebo nevhodně zvoleným výpočtem pro kalibraci. Obecně ale naše výsledky potvrzují hypotézy H3.1 a H3.2.



Obrázek 5.14: Váhy přidělené účastníky studie ke kritériím při požadavku na doporučovací systém. Vlevo brány v úvahu všechny požadavky na doporučovací systém, vpravo jen ty s nevýchozím nastavením vah.

Pokud se v levé části obr. 5.14 podíváme, jaké váhy účastníci přidělovali jednotlivým kritériím při požadavku na doporučovací systém, je zřetelné, že žádné z kritérií výrazně nevybočovalo. Nejvyšší medián najdeme u relevance, a to těsně nad hranicí hodnoty 20. Přesto můžeme pozorovat, že nejvyšší váhy účastníci přidělovali především relevanci ale taky diverzitě a novelty. Nižší váhy oproti zmíněným třem kritériím byly nastavovány pro popularitu a ještě menší pro kalibraci.

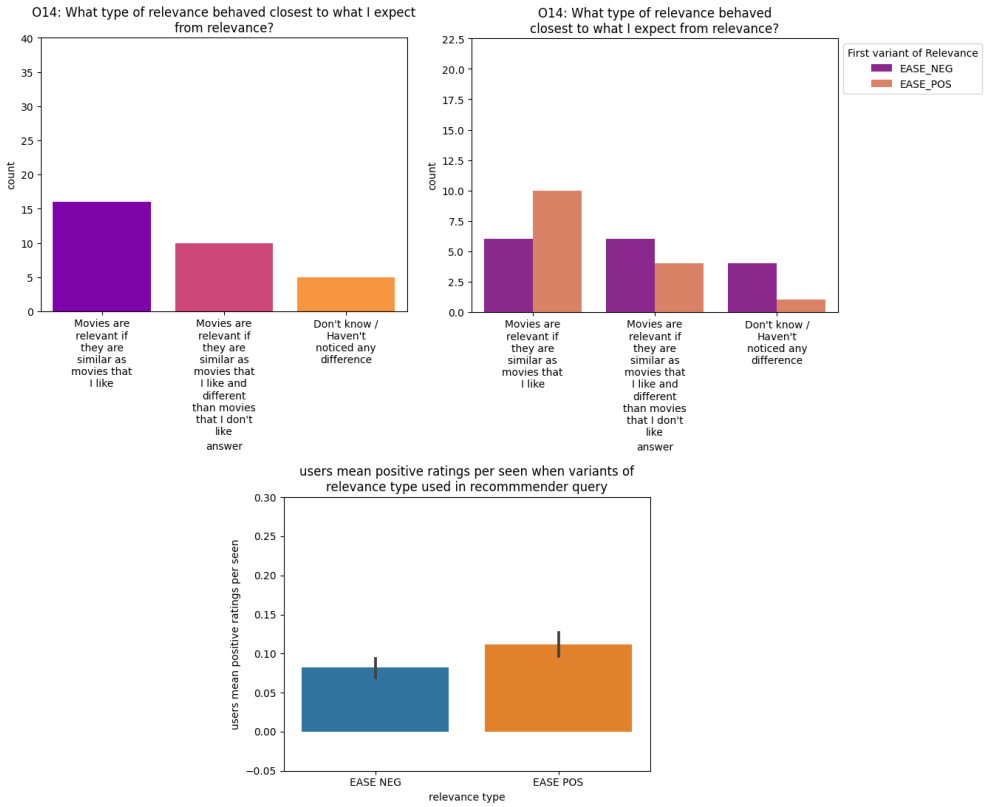
Ještě se zaměříme pouze na požadavky na doporučení, kde nebyly nastaveny výchozí váhy, tzn. pro všechna kritéria váha 20, případně u mechanismu drag and drop váha 100 rozdělena mezi kritéria poměrem 5:4:3:2:1 (v pořadí relevance, diverzita, novelty, popularita, kalibrace). V pravé části obr. 5.14 můžeme vidět, že explicitně nastavené vyšší váhy byly přiřazovány k novelty než k diverzitě, a to

až na úrovni relevance. Podobně vysoké váhy jako u diverzity pozorujeme i u popularity. I zde ale platí, že nejníže uživatelé nastavují důležitost ke kalibraci, což může být zapříčiněno nejasností kritéria nebo nepozorováním jeho přínosu, což popisujeme výše v této kapitole.

Relevance je skutečně kritériem, kterému uživatelé přidělují nejvyšší váhy, což potvrzuje hypotézu H3.3. Je ale potřeba zmínit, že při nevýchozím nastavení vah, dávali účastníci studie stejný důraz i na novelty.

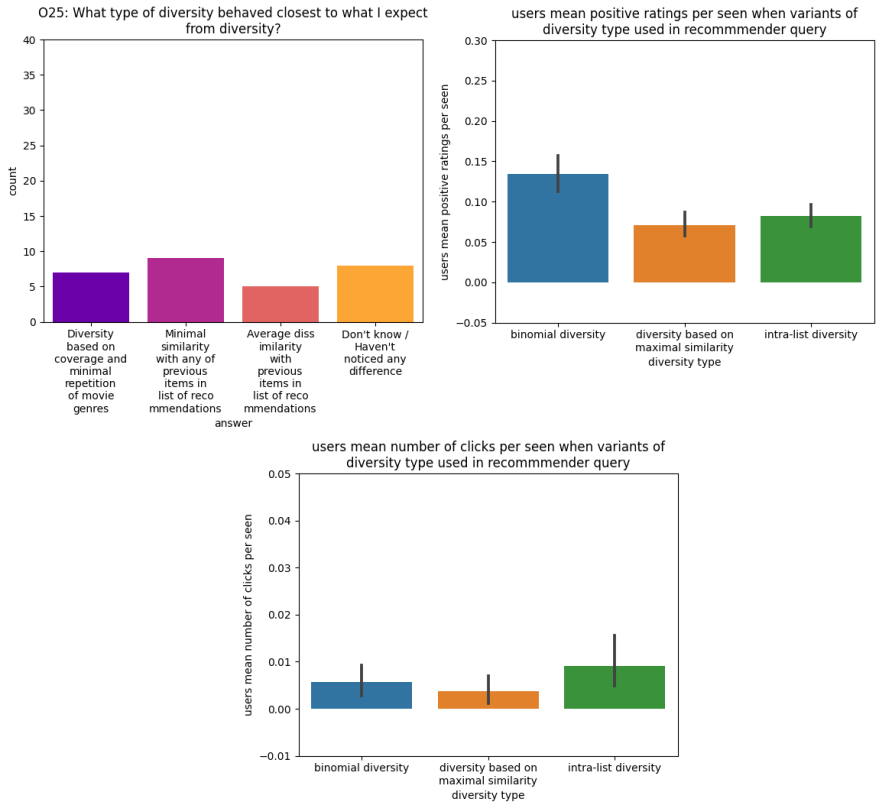
5.2.5 RQ4: Jaké varianty metrik se pro jednotlivá kritéria kvality chovají nejblíže tomu, co od nich uživatelé očekávají?

Kritéria kvality doporučování, pro která umožňujeme uživateli volit z více variant metrik, jsou relevance, diverzita, novelty a popularita (viz kapitoly 1.3 a 3.3.2 - 3.3.6). Podíváme se na to, jak účastníci chápali kritéria před účastí v uživatelské studii, na jejich explicitní zpětnou vazbu v podobě odpovědi v dotazníku, ale i na implicitní, kdy se zaměříme na uživatelův průměrný poměr počtu pozitivních hodnocení, případně kliknutí na položku vzhledem k množství zobrazených položek. Počet pozitivních hodnocení můžeme interpretovat jako přesnost doporučení, kdežto četnost kliknutí jako počet položek, které uživatele zaujaly. Je ale třeba zmínit, že počty kliknutí jsou poměrně nízké na to, abychom z nich mohli dělat jasné závěry.



Obrázek 5.15: Vlevo odpovědi na otázku O14, vpravo závislost těchto odpovědí na výchozí variantě metriky relevance, dole pozitivní hodnocení účastníků při použití jednotlivých metrik.

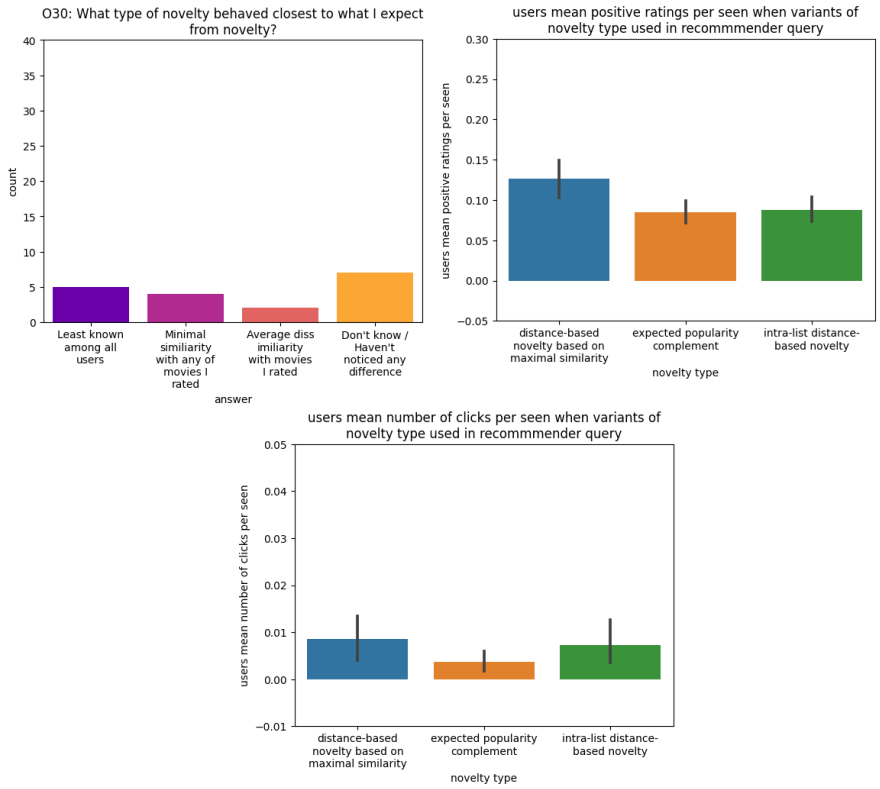
Začneme u relevance. Na obr. 5.15 vlevo nahoře můžeme pozorovat, že účastníci studie v dotazníku v odpovědi na otázku O14 hodnotí lépe variantu pracující pouze s pozitivními hodnoceními $EASE_{POS}$. To potvrzují i implicitní data, kdy se zaměřujeme na to, kolik ze zobrazených položek uživatelé pozitivně hodnotili při použití jednotlivých metrik. I na základě tohoto porovnání, které můžeme vidět na obr. 5.15 dole, vyplývá, že $EASE_{POS}$ se ukazuje jako lepší varianta než $EASE_{NEG}$. Je ale potřeba zmínit, že u obou variant odhadu relevance většina účastníků hodnotila lépe variantu, která jí byla přidělena jako první, což lze pozorovat na obr. 5.15 vpravo nahoře. To mohlo být zapříčiněno tím, že nejrelevantnější filmy byly účastníkovi studie zobrazeny již při prvních doporučeních a vzhledem k pravidlům omezující opětovná doporučení hodnocených či nedávno zobrazených filmů se tato doporučení neopakovala v pozdější fázi studie, kdy byla používaná druhá varianta relevance.



Obrázek 5.16: Vlevo odpovědi na otázku O25, vpravo pozitivní hodnocení účastníků a dole kliknutí (= rozbalení detailu) při použití jednotlivých metrik.

Jak je zřejmé z obr. 5.16 na grafu v levé horní části, účastníci studie v odpovědi na otázku O25 nejlépe hodnotí diverzitu na základě maximální podobnosti a lépe než intra-list diverzitu vnímají i použití binomické diverzity pro výpočet skóre tohoto kritéria. Poměrně vysoký počet účastníků také udává, že nebyli schopni rozlišit, která metrika nejvíce přispívala k výsledku, který očekávají od diverzity. Co se týče chování uživatelů při doporučování s těmito metrikami, můžeme na obr. 5.16 vidět, že nejvíce pozitivních hodnocení pozorujeme u binomické diverzity a nejméně u diverzity na základě maximální podobnosti. U počtu kliknutí na doporučení, které by mělo lépe ilustrovat splnění cílů diverzity, nejlépe funguje intra-list diverzita a nejhůře opět nejlépe hodnocená diverzita na základě maxi-

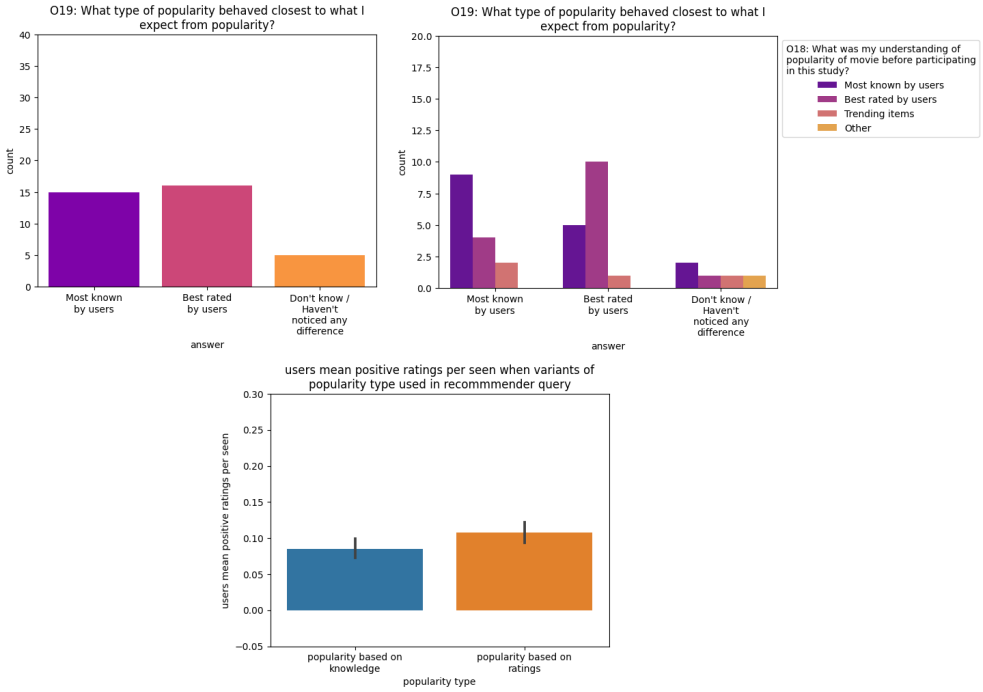
mální podobnosti, což nepotvrzuje explicitní zpětnou vazbu uživatelů z dotazníku u otázky O25.



Obrázek 5.17: Vlevo odpovědi na otázku O30, vpravo pozitivní hodnocení účastníků a dole kliknutí (= rozbalení detailu) při použití jednotlivých metrik.

U novelty můžeme z odpovědí účastníků na otázku O30, které jsou vizualizovány na obr. 5.17 v horním levém grafu, vypozorovat dva problémy. Jednak menší počet účastníků, kteří v průběhu studie nastavili všechny varianty metrik novelty a mohli tak odpovídat na tuto otázku, a jednak vysoký počet uživatelů, kteří nebyli schopni mezi přínosem jednotlivých metrik rozlišit. Nejlépe hodnocenou metrikou nicméně mezi účastníky studie je očekávaný doplněk popularity. U této nejlépe hodnocené metriky ale pozorujeme nejnižší počet kliknutí na doporučení, nejvyšší četnost kliknutí na zobrazené doporučení naopak můžeme pozorovat při použití distance-based novelty na základě maximální podobnosti. Stejné pořadí metrik určuje i průměrný počet pozitivních hodnocení na 1 zobrazení doporučené položky, což lze vidět na obr. 5.17.

U odpovědí na otázku O19 nepozorujeme na obr. 5.18 vlevo nahoře téměř žádný rozdíl mezi popularitou dle známosti a popularitou na základě hodnocení. Většina z účastníků je nicméně si schopna vybrat, kterou z variant vnímali jako lepší, což potvrzuje první část hypotézy H4.1. Zajímavé ale je, že účastníci studie se většinou drží konceptu, jakým chápali popularitu ještě před provedením studie (viz porovnání s odpověďmi na otázku O18 na obr.5.18 na pravém horním grafu). Vzhledem k tomu, že popularitu považujeme za kritérium, jehož cílem je exploatace, můžeme na spodním grafu na obr. 5.18 porovnat metriky vzhledem k průměrnému počtu pozitivních hodnocení na zobrazení. Z výsledků vyplývá, že k většímu počtu pozitivní explicitní zpětné vazby k doporučením přispívala popularita na základě hodnocení.



Obrázek 5.18: Vlevo odpovědi na otázku O19, vpravo závislost odpovědí na tuto otázku na chápání popularity před provedením studie (otázka O18) a dole pozitivní hodnocení účastníků při použití jednotlivých metrik.

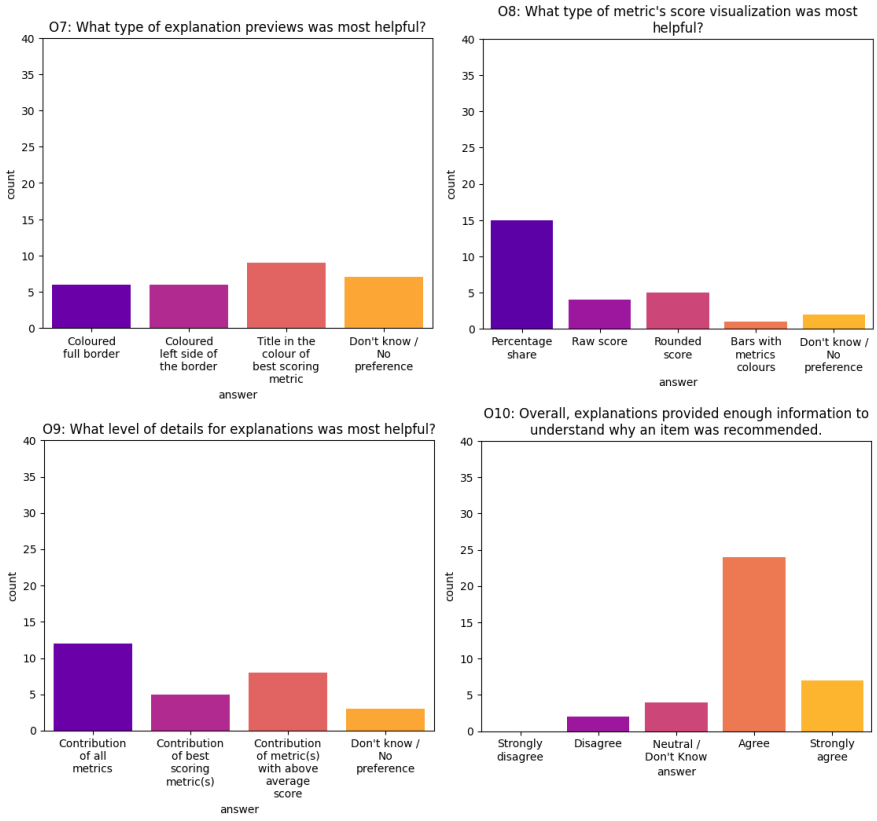
Pokud tedy shrneme výsledky porovnání metrik pro všechna zmíněná kritéria, můžeme pro odhad relevance spíše doporučit variantu pouze s pozitivními hodnoceními $EASE_{POS}$ před variantou $EASE_{NEG}$. U diverzity jsme tak jednoznačné výsledky od účastníků studie nezískali. Nejlépe sice vnímali diverzitu na základě maximální podobnosti, což ale nebylo v souladu s jejich chováním v průběhu práce se systémem. Nejméně vypovídající porovnání pozorujeme u novelty, kdy chování uživatelů opět neodpovídá jejich odpovědím v dotazníku a navíc jen malá část z nich byla schopna rozlišit přínos novelty při použití různých metrik. Varianty popularity opět hodnotí jako nejlepší podobný počet účastníků, na základě jejich chování v podobě udělování pozitivních hodnocení ale jako lepší možnost působí popularita na základě hodnocení, což je v souladu s dřívějšími poznatky (Cañamares a Castells, 2018) i druhou částí naší hypotézy H4.1.

Hypotéza H4.2 se nám potvrzuje především u novelty a v menší míře u diverzity, nicméně očekávané výraznější rozdíly mezi schopností rozlišit změny při používání různých metrik popularity a stejnou schopností u různých metrik ostatních kritérií kvality doporučování nepozorujeme.

5.2.6 RQ5: Jaká vysvětlení doporučení v multi-objective doporučování jsou vnímána jako nejpřínosnější?

U explanations doporučení multi-objective doporučovacího systému jsme porovnávali různé varianty na třech úrovních, a to u náhledu vysvětlení odpovídajícímu skóre používaných metrik jednotlivých kritérií kvality doporučování, u vizualizace hodnoty skóre a u počtu kritérií, jejichž vysvětlení zobrazujeme.

Na základě odpovědí účastníků studie na otázku O7 (viz obr. 5.19 vlevo na-



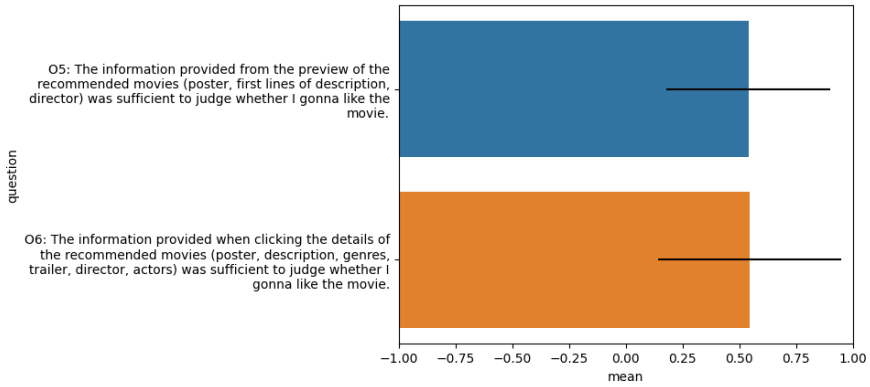
Obrázek 5.19: Odpovědi na otázky O7 - O10 k hodnocení explanations.

hoře), nedostáváme jasnou odpověď na to, jaký typ náhledu vysvětlení je nejvhodnější. Nejvíce hlasů sice získal název položky v barvě kritéria s nejvyšším skóre, ovšem další dvě možnosti jsou si velmi podobné a v součtu je více účastníků studie označilo jako nejlepší variantu. Poměrně vysoký počet z nich navíc neudává preferenci k žádné z variant.

Více vypovídající výsledky už pozorujeme na pravém horním grafu na obr. 5.19 u otázky O8, kde porovnáváme různé varianty vizualizace skóre. Účastníci studie preferují procentuální podíl na celkovém skóre (tzn. suma přes všechna kritéria) položky. Pokud sečteme hlasy pro velmi podobné návrhy hrubého skóre a zaokrouhleného skóre, tak i ty nezanedbatelný počet účastníků považuje jako nejlepší variantu. Nejhůře účastníci hodnotí grafické vyjádření skóre.

Co se týče detailu vysvětlení, na který se ptáme v otázce O9, tak z grafu v levé dolní části obr. 5.19 vyplývá, že účastníci studie preferují nejdetailnější vysvětlení se všemi kritérii. Nejhůře hodnocené bylo naopak nejméně detailní vysvětlení pouze s kritériem, pro které má položka nejvyšší skóre. Sklon k preferenci detailnějších explanations je v souladu se současnými poznatky o detailnosti informací, které podrobněji popisujeme v kapitole 1.6.3, a také s hypotézou H5.1.

Na závěr jsme se ještě účastníků studie dotazovali v otázce O10, zda pro ně byla používaná explanations dostačující k tomu, aby chápali, proč byl daný film doporučen. Z odpovědí (viz obr. 5.19 vpravo dole) plyne, že vysvětlení poskytla dostatek informací k pochopení toho, proč byla položka doporučena.

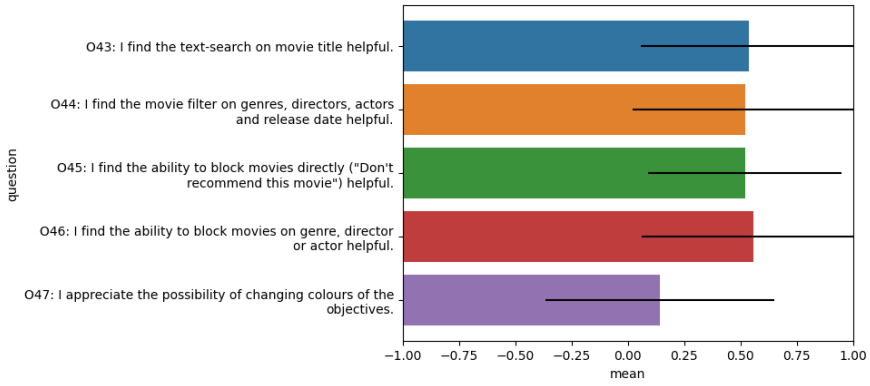


Obrázek 5.20: Odpovědi na otázky O5 a O6, jestli jsou poskytnuté informace o filmech dostatečné pro posouzení toho, zda se účastníkům studie film líbí. Nahoře informace v náhledu doporučení, dole informace v detailu filmu.

5.2.7 RQ6: Jak uživatelé oceňují prvky webové aplikace?

Nejdříve se podíváme na to, zda informace o filmech zobrazené v náhledu a detailu doporučení byly pro účastníky studie dostatečné pro posouzení toho, zda se jim film líbí, resp. bude líbit. Jak je vidět na obr. 5.20, kde jsou zobrazeny odpovědi na otázky O5 a O6, účastníci studie hodnotili poskytnuté informace jako dostatečné pro posouzení doporučeného filmu, což potvrzuje hypotézu H6.1. Poměrně překvapivé je, že nepozorujeme téměř žádný rozdíl mezi úrovní posuzovatelnosti u náhledu filmu a jeho detailu, kde je informací o něco více.

K vyhodnocení toho, jak účastníci vnímali další prvky ve webové aplikaci, vezmeme v úvahu jednak počet využití těchto prvků na základě statistik z tabulky 5.2 a odpovědí účastníků v dotazníku na otázky O43 - O47 k vyhodnocení užitečnosti zařazení toho daného prvku ve webové aplikaci.



Obrázek 5.21: Odpovědi na otázky O43 - O47 ohledně užitečnosti doplňujících prvků aplikace. Shora textové vyhledávání, podrobnější filtr, přímé blokování filmů, blokování na základě vlastnosti a změna barev kritérií kvality doporučování.

Účastníci studie udávají v dotazníku, jak můžeme vidět na obr. 5.21, že považují za užitečnou možnost textového vyhledávání i podrobnějšího filtrování, což potvrzuje i naši další hypotézu H6.2. Dále účastnící oceňují možnost blokování filmu přímo i blokování na základě některé z jeho vlastností (žánr, režisér, herec).

Akce	**Průměrný počet provedení akce**	**Směrodatná odchylka počtu provedených akcí**	**Poměr účastníků studie, kteří akci provedli alespoň jednou**
Nastavení bloku přímo na film	5,71	14,06	0,66
Nastavení blokovacího pravidla na žánr, režiséra nebo herce	1,63	3,31	0,48
Nastavení blokovacího pravidla na herce	0,74	2,19	0,19
Nastavení blokovacího pravidla na režiséra	0,09	0,30	0,10
Nastavení blokovacího pravidla na žánr	0,80	2,24	0,19
Kliknutí na náhled položky a zobrazení jejího detailu	2.75	2,94	0,89
Použití textového vyhledávání na základě názvu filmu	1,67	2,86	0,64
Použití podrobnějšího filtru k vyhledávání filmu dle žánrů, režiséra, herce, nebo data vydání	1,67	2,86	0,64
Nastavení barev pro kritéria kvality doporučování	1,45	1,46	0,75

Tabulka 5.2: Počet akcí různých typů, které provedl účastník v průběhu studie

Tyto výsledky odpovídají dřívějším studiím (např. Kleemann a kol. (2022) a Loepp (2022)). Již menší užitečnost ale účastníci vidí v možnosti vlastního nastavení barev ke kritériím.

Pokud se podíváme na to, jak účastníci studie v průběhu práce využívali tyto prvky, což lze vidět v tabulce 5.2, tak si můžeme všimnout vysokých hodnot směrodatné odchylky, a to dokonce vyšších, než jsou hodnoty průměru, což značí velkou odlišnost v počtu provedených akcí mezi uživateli. I vzhledem k tomu, že jednotlivé akce jsme účastníkům studie navrhovali (blokování herce, režiséra a žánru patří pod jednu akci blokování na základě vlastnosti), je u všech těchto akcí poměr účastníků, kteří akci provedli poměrně vysoký, kdy kromě akce nastavení blokovacího pravidla na žánr, režiséra nebo herce akci provedla alespoň polovina uživatelů. Pokud si akci blokování na základě vlastnosti, rozdělíme na jednotlivé vlastnosti, vidíme už nízké počty uživatelů, kteří je použili, což může být způsobeno tím, že narozdíl od blokování filmu přímo, je nutné se nejdříve prokliknout

na detail filmu nebo na stránku správy blokovacích pravidel. Tato nižší používání bloků ovšem nejsou v souladu s hypotézou H6.3.

Závěr

Úvodním požadavkem zadání diplomové práce bylo podrobné seznámení se s doporučovacími systémy a jejich propojení s oblastí HCI, do které patří explanations doporučení, interakce systému a uživatele a vizualizace. Konkrétněji jsme se v oblasti doporučovacích systémů zaměřovali na kritéria kvality doporučování a na multi-objective doporučovací systémy pracující s více takovými kritérii. Výstupem seznámení se se zkoumanou oblastí je kapitola 1, kde představujeme hlavní a zásadní zjištěné poznatky, které tvoří teoretickou část práce.

Dalším cílem práce byla implementace webové aplikace, která bude sloužit jako vhodné prostředí pro provedení uživatelské studie zaměřené na interakce uživatele s multi-objective doporučovacím systémem. Řešení tedy musí umožnit komunikaci s tímto multi-objective doporučovacím systémem a prezentování jeho výstupů uživateli, kdy zásadním prvkem této aplikace je možnost explicitního upravení parametrů doporučovacího systému uživatelem. Webová aplikace byla navržena a implementována tak, aby uživateli umožnila s tímto specifickým typem doporučovacího systému pracovat s co nejmenší závislostí na doméně doporučování a konkrétní implementaci samotného multi-objective doporučovacího systému. Zároveň byla do výsledného řešení promítnuta snaha, aby celý systém více připomínal prostředí reálného portálu s procházením položek umožňující i jiné způsoby průchodu než pouze pomocí doporučovacího systému. Uživatel může v aplikaci prohlížet vrácené doporučené položky, nastavovat váhy pro jednotlivá kritéria kvality doporučování, blokovat či hodnotit položky, měnit některá nastavení aplikace či používané varianty, vyhledávat textově či pomocí filtru atd. Webová aplikace musela navíc také obsahovat prvky umožňující ji použít jako prostředí pro provedení uživatelské studie, jako je závěrečný dotazník, ukládání dalších akcí uživatele a jejich navrhování.

V rámci řešení došlo i k úpravě a rozšíření původní implementace doporučovacího systému RL-Prop (Peška a Dokoupil, 2022). To zahrnovalo převedení softwarového řešení do formy webové služby zpracovávající požadavky na doporučení jednotlivým uživatelům, přidání nových kritérií kvality doporučování a jejich variant metrik a další úpravy implementace doporučovacího systému.

Závěrečnou částí práce bylo provedení experimentu v podobě uživatelské studie a zpracování jeho výsledků. V uživatelské studii, do které se registrovalo 50 účastníků a z toho ji zhruba čtyři desítky provedli celou, byla zkoumána tendence uživatelů k používání jednotlivých kritérií kvality doporučování a možnosti explicitního nastavení vah pro tato kritéria. Dále byly porovnávány různé varianty metrik pro většinu z kritérií, vysvětlení doporučení, mechanismu pro nastavení vah. Na závěr jsme se ještě zaměřili na to, jak účastníci vnímali užitečnost dalších prvků, které přímo nesouvisely s doporučovacím systémem, jako je vyhledávání, blokování položek atd.

Z provedené studie vyplynulo několik výsledků. Ukázalo se, že účastníci vnímali přidanou hodnotu práce s kritérii kvality doporučování a část z nich dokonce tuto možnost preferovala před automatickým doporučováním bez nastavení vah jednotlivým kritériím. Zjistili jsme, že jednoznačně nejlépe hodnoceným mechanismem pro nastavování vah jsou posuvníky, tzn. slidery. Účastníci studie obecně kritériím kvality doporučování rozuměli a vnímali přínos u relevance, diverzity,

novelty, popularity a v o něco menší míře u kalibrace. Co se týče porovnání různých metrik pro většinu kritérií, nedokážeme i vzhledem k menšímu počtu účastníků jednoznačně určit, jaká z variant je nejvhodnější. S ne zcela vypovídajícími výsledky jsme nuceni pracovat i u variant explanations, kdy se nám ale na základě odpovědí účastníků studie potvrdilo, že uživatelé preferují detailnější vysvětlení doporučení. Posledním výstupem experimentu je fakt, že účastníci považovali za užitečné téměř všechny prvky webové aplikace, které nebyly přímo navázány na práci s doporučovacím systémem.

Co se týče další práce, která by na tuto mohla navazovat, stojí vzhledem k počtu účastníků, kteří studii nedokončili, případně museli být vyřazení, za zvážení, zda některé zkoumané oblasti nezkoumat v rámci samostatných experimentů. Pro vyšší statistickou významnost by také bylo vhodné provést studii s větším počtem účastníků. Je také potřeba zmínit, že v této práci jsme se zaměřovali na jedinou doménu, což byly filmy, a pracovali s jediným multi-objective doporučovacím systémem, takže další možnou cestou je potvrdit naše výsledky v jiné doméně, případně při práci s jinou variantou multi-objective doporučovacího systému.

Seznam použité literatury

Adomavicius, G. a Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, **17**(6), 734–749. doi: 10.1109/TKDE.2005.99.

Adomavicius, G. a Kwonová, Y. (2012). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, **24**(5), 896–911. doi: 10.1109/TKDE.2011.15.

Alhijawiová, B., Awajan, A. a Fraihat, S. (2022). Survey on the Objectives of Recommender Systems: Measures, Solutions, Evaluation Methodology, and New Perspectives. *ACM Comput. Surv.*, **55**(5). ISSN 0360-0300. doi: 10.1145/3527449. URL `https://doi.org/10.1145/3527449`.

Amoo, T. a Friedman, H. (2001). Do Numeric Values Influence Subjects' Responses to Rating Scales? *Journal of International Marketing and Marketing Research (European Marketing Association)*, **26**, 41–46.

Beel, J. a Dixonová, H. (2021). The 'Unreasonable' Effectiveness of Graphical User Interfaces for Recommender Systems. In *Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '21, page 22–28, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383677. doi: 10.1145/3450614.3461682. URL `https://doi.org/10.1145/3450614.3461682`.

Bradley, K. (2001). Improving Recommendation Diversity. *Proc. AICS '01.*

Burke, R., Felfernig, A. a Göker, M. (2011). Recommender Systems: An Overview. *Ai Magazine*, **32**, 13–18. doi: 10.1609/aimag.v32i3.2361.

Cañamares, R. a Castells, P. (2018). Should I Follow the Crowd? A Probabilistic Analysis of the Effectiveness of Popularity in Recommender Systems. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 415–424, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3210014. URL `https://doi.org/10.1145/3209978.3210014`.

Carbonell, J. a Goldsteinová, J. (1998). The use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 335–336, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 1581130155. doi: 10.1145/290941.291025. URL `https://doi.org/10.1145/290941.291025`.

Chen, L. a Puová, P. (2005). Trust building in recommender agents. In *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*, pages 135–145.

Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomovová, O., Ashkan, A., Büttcher, S. a MacKinnon, I. (2008). Novelty and Diversity in Information Retrieval Evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, page 659–666, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581644. doi: 10.1145/1390334.1390446. URL https://doi.org/10.1145/1390334.1390446.

Dokoupil, P., Peška, L. a Boratto, L. (2023a). Looks Can Be Deceiving: Linking User-Item Interactions and User's Propensity Towards Multi-Objective Recommendations. In *Proceedings of the 17th ACM Conference on Recommender Systems*, RecSys '23, page 912–918, New York, NY, USA, 2023a. Association for Computing Machinery. ISBN 9798400702419. doi: 10.1145/3604915.3608848. URL https://doi.org/10.1145/3604915.3608848.

Dokoupil, P., Peška, L. a Boratto, L. (2023b). Rows or columns? minimizing presentation bias when comparing multiple recommender systems. In Chen, H., Duh, W. E., Huang, H., Kato, M. P., Mothe, J. a Poblete, B., editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 2354–2358. ACM. doi: 10.1145/3539618.3592056. URL https://doi.org/10.1145/3539618.3592056.

Garett, R., Chiu, J., Zhang, L. a Young, S. (2016). A Literature Review: Website Design and User Engagement. *Online journal of communication and media technologies*, **6**, 1–14. doi: 10.29333/ojcmt/2556.

Garland, R. (1991). The mid-point on a rating scale: Is it desirable? URL https://api.semanticscholar.org/CorpusID:146702037.

Gedikli, F., Jannach, D. a Ge, M. (2014). How should I explain? A comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, **72**(4), 367–382. ISSN 1071-5819. doi: https://doi.org/10.1016/j.ijhcs.2013.12.007. URL https://www.sciencedirect.com/science/article/pii/S1071581913002024.

Goldberg, L. R., Johnson, J. A., Eber, H. W., Hogan, R., Ashton, M. C., Cloninger, C. R. a Gough, H. G. (2006). The international personality item pool and the future of public-domain personality measures. *Journal of Research in Personality*, **40**(1), 84–96. ISSN 0092-6566. doi: https://doi.org/10.1016/j.jrp.2005.08.007. URL https://www.sciencedirect.com/science/article/pii/S0092656605000553. Proceedings of the 2005 Meeting of the Association of Research in Personality.

GroupLens (2023). MovieLens. URL https://grouplens.org/datasets/movielens/. Accessed: 2023-11-22.

Gunawardana, A., Shani, G. a Yogev, S. (2022). *Evaluating Recommendation Systems*, pages 547–601. In Ricci a kol. (2022). ISBN 978-1-4899-7637-6. doi: 10.1007/978-1-4899-7637-6_21. URL https://doi.org/10.1007/978-1-4899-7637-6_21.

Harper, F. M. a Konstan, J. A. (2015). The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.*, **5**(4). ISSN 2160-6455. doi: 10.1145/2827872. URL https://doi.org/10.1145/2827872.

Herlocker, J. L., Konstan, J. A., Terveen, L. G. a Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.*, **22**(1), 5–53. ISSN 1046-8188. doi: 10.1145/963770.963772. URL https://doi.org/10.1145/963770.963772.

Higley, K., Oldridge, E., Ak, R., Rabhi, S. a de Souza Pereira Moreira, G. (2022). Building and Deploying a Multi-Stage Recommender System with Merlin. In *Proceedings of the 16th ACM Conference on Recommender Systems*, RecSys '22, page 632–635, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392785. doi: 10.1145/3523227.3551468. URL https://doi.org/10.1145/3523227.3551468.

Hu, R. a Puová, P. (2013). Exploring relations between personality and user rating behaviors. *CEUR Workshop Proceedings*, **997**.

Isufi, E., Pocchiari, M. a Hanjalic, A. (2021). Accuracy-diversity trade-off in recommender systems via graph convolutions. *Information Processing & Management*, **58**(2), 102459. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2020.102459. URL https://www.sciencedirect.com/science/article/pii/S0306457320309511.

Jambor, T. a Wang, J. (2010). Optimizing Multiple Objectives in Collaborative Filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, page 55–62, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605589060. doi: 10.1145/1864708.1864723. URL https://doi.org/10.1145/1864708.1864723.

Jannach, D. (2022). Multi-Objective Recommender Systems: Survey and Challenges.

Jannach, D., Naveedová, S. a Jugovac, M. (2017). User Control in Recommender Systems: Overview and Interaction Challenges. In *International Conference on Electronic Commerce and Web Technologies*, volume 278, pages 21–33. ISBN 978-3-319-53675-0. doi: 10.1007/978-3-319-53676-7_2.

Joachims, T. (2002). Optimizing Search Engines Using Clickthrough Data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, page 133–142, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775067. URL https://doi.org/10.1145/775047.775067.

Jugovac, M., Jannach, D. a Lerche, L. (2017). Efficient optimization of multiple recommendation quality factors according to individual user tendencies. *Expert Systems with Applications*, **81**, 321–331. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2017.03.055. URL https://www.sciencedirect.com/science/article/pii/S0957417417302075.

Kleemann, T., Loepp, B. a Ziegler, J. (2022). Towards Multi-Method Support for Product Search and Recommending. In *Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '22 Adjunct, page 74–79, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392327. doi: 10.1145/3511047.3536408. URL https://doi.org/10.1145/3511047.3536408.

Knijnenburg, B. P. a Willemsen, M. C. (2015). *Evaluating Recommender Systems with User Experiments*, pages 309–352. Springer US, Boston, MA. ISBN 978-1-4899-7637-6. doi: 10.1007/978-1-4899-7637-6_9. URL https://doi.org/10.1007/978-1-4899-7637-6_9.

Knijnenburg, B. P., Willemsen, M. C., Gantner, Z., Soncu, H. a Newell, C. (2012). Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction*, **22**, 441–504. URL https://api.semanticscholar.org/CorpusID:2944283.

Kulesza, T., Stumpfová, S., Burnettová, M., Yangová, S., Kwan, I. a Wong, W.-K. (2013). Too much, too little, or just right? ways explanations impact end users' mental models. In *2013 IEEE Symposium on Visual Languages and Human Centric Computing*, pages 3–10. doi: 10.1109/VLHCC.2013.6645235.

Kunaver, M. a Porl, T. (2017). Diversity in Recommender Systems A Survey. *Know.-Based Syst.*, **123**(C), 154–162. ISSN 0950-7051. doi: 10.1016/j.knosys.2017.02.009. URL https://doi.org/10.1016/j.knosys.2017.02.009.

Loepp, B. (2022). Recommender Systems Alone Are Not Everything: Towards a Broader Perspective in the Evaluation of Recommender Systems. In Zangerle, E., Bauer, C. a Said, A., editors, *Proceedings of the Perspectives on the Evaluation of Recommender Systems Workshop 2022 co-located with the 16th ACM Conference on Recommender Systems (RecSys 2022), Seattle, WA, USA, September 22, 2022*, volume 3228 of *CEUR Workshop Proceedings*. CEUR-WS.org. URL https://ceur-ws.org/Vol-3228/paper5.pdf.

Maleček, L. a Peška, L. (2021). Fairness-Preserving Group Recommendations With User Weighting. In *Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '21, page 4–9, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383677. doi: 10.1145/3450614.3461679. URL https://doi.org/10.1145/3450614.3461679.

McCrae, R. R. a John, O. P. (1992). An Introduction to the Five-Factor Model and Its Applications. *Journal of Personality*, **60**(2), 175–215. doi: https://doi.org/10.1111/j.1467-6494.1992.tb00970.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6494.1992.tb00970.x.

McNee, S. M., Riedl, J. a Konstan, J. A. (2006). Being Accurate is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, page 1097–1101, New York, NY, USA, 2006. Association for Computing

Machinery. ISBN 1595932984. doi: 10.1145/1125451.1125659. URL https://doi.org/10.1145/1125451.1125659.

Murphy-Hill, E. a Murphyová, G. C. (2014). *Recommendation Delivery*, pages 223–242. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-642-45135-5. doi: 10.1007/978-3-642-45135-5_9. URL https://doi.org/10.1007/978-3-642-45135-5_9.

Oh, J., Park, S., Yu, H., Song, M. a Park, S.-T. (2011). Novel recommendation based on personal popularity tendency. In *2011 IEEE 11th International Conference on Data Mining*, pages 507–516. doi: 10.1109/ICDM.2011.110.

Ozok, A., Fan, Q. a Norcio, A. (2010). Design guidelines for effective recommender system interfaces based on a usability criteria conceptual model: Results from a college student population. *Behaviour & IT*, **29**, 57–83. doi: 10.1080/01449290903004012.

Peška, L. a Dokoupil, P. (2022). Towards Results-Level Proportionality for Multi-Objective Recommender Systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 1963–1968, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531787. URL https://doi.org/10.1145/3477495.3531787.

Puová, P., Chenová, L. a Huová, R. (2011). A User-Centric Evaluation Framework for Recommender Systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys '11, page 157–164, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450306836. doi: 10.1145/2043932.2043962. URL https://doi.org/10.1145/2043932.2043962.

Ribeiro, M. T., Ziviani, N., Moura, E. S. D., Hata, I., Lacerda, A. a Veloso, A. (2015). Multiobjective Pareto-Efficient Approaches for Recommender Systems. *ACM Trans. Intell. Syst. Technol.*, **5**(4). ISSN 2157-6904. doi: 10.1145/2629350. URL https://doi.org/10.1145/2629350.

Ricci, F., Rokach, L. a Shapirová, B., editors (2022). *Recommender Systems Handbook*. Springer US, New York, NY. ISBN 978-1-4899-7637-6. doi: 10.1007/978-1-4899-7637-6_21. URL https://doi.org/10.1007/978-1-4899-7637-6_21.

Rubner, Y., Tomasi, C. a Guibas, L. (1998). A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 59–66. doi: 10.1109/ICCV.1998.710701.

Schnabel, T., Bennett, P. N. a Joachims, T. (2018). Improving recommender systems beyond the algorithm. *ArXiv*, **abs/1802.07578**. URL https://api.semanticscholar.org/CorpusID:3412768.

Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Longman Publishing Co., Inc., USA, 3rd edition. ISBN 0201694972.

Steck, H. (2018). Calibrated Recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, page 154–162, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450359016. doi: 10.1145/3240323.3240372. URL https://doi.org/10.1145/3240323.3240372.

Steck, H. (2019). Embarrassingly Shallow Autoencoders for Sparse Data. In *The World Wide Web Conference*, WWW '19, page 3251–3257, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366748. doi: 10.1145/3308558.3313710. URL https://doi.org/10.1145/3308558.3313710.

Swearingenová, K. a Sinhová, R. (2001). Beyond algorithms: An hci perspective on recommender systems. In *ACM SIGIR 2001 workshop on recommender systems*, volume 13, numbers 5-6, pages 1–11.

The Movie Database (2023a). Golden Eye poster. URL https://image.tmdb.org/t/p/w500//z0ljRnNxIO7CRBhLEO0DvLgAFPR.jpg. [Online; accessed December 29, 2023].

The Movie Database (2023b). Zoolander poster. URL https://image.tmdb.org/t/p/w500//qdrbSneHZjJG2Dj0hhBxzzAo4HB.jpg. [Online; accessed December 29, 2023].

Tintarevová, N. a Masthoff, J. (2022). *Beyond Explaining Single Item Recommendations*, pages 711–756. In Ricci a kol. (2022). ISBN 978-1-4899-7637-6. doi: 10.1007/978-1-4899-7637-6_21. URL https://doi.org/10.1007/978-1-4899-7637-6_21.

Tintarevová, N. a Masthoffová, J. (2007). A Survey of Explanations in Recommender Systems. In *2007 IEEE 23rd international conference on data engineering workshop*, pages 801–810. IEEE. ISBN 978-1-4244-0832-0. doi: 10.1109/ICDEW.2007.4401070.

Tkalcic, M. a Chenová, L. (2015). *Personality and Recommender Systems*, pages 715–739. In Ricci a kol. (2022). ISBN 978-1-4899-7637-6. doi: 10.1007/978-1-4899-7637-6_21. URL https://doi.org/10.1007/978-1-4899-7637-6_21.

TMDB (2023). The Movie Database (TMDB) API. URL https://developer.themoviedb.org/reference. Accessed: 2023-11-22.

Vargas, S. (2014). Novelty and Diversity Enhancement and Evaluation in Recommender Systems and Information Retrieval. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, page 1281, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450322577. doi: 10.1145/2600428.2610382. URL https://doi.org/10.1145/2600428.2610382.

Vargas, S., Baltrunas, L., Karatzoglou, A. a Castells, P. (2014). Coverage, Redundancy and Size-Awareness in Genre Diversity for Recommender Systems. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, page 209–216, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450326681. doi: 10.1145/2645710.2645743. URL https://doi.org/10.1145/2645710.2645743.

Weijters, B., Cabooter, E. a Schillewaert, N. (2010). The effect of rating scale format on response styles: The number of response categories and response category labels. *International Journal of Research in Marketing*, **27**(3), 236–247. ISSN 0167-8116. doi: https://doi.org/10.1016/j.ijresmar.2010.02.004. URL https://www.sciencedirect.com/science/article/pii/S0167811610000303.

Xie, L., Hu, Z., Cai, X., Zhang, W. a Chen, J. (2021). Explainable recommendation based on knowledge graph and multi-objective optimization. *Complex & Intelligent Systems*, **7**, 1–12. doi: 10.1007/s40747-021-00315-y.

Yanová, T. a Keusch, F. (2015). The Effects of the Direction of Rating Scales on Survey Responses in a Telephone Survey. *Public Opinion Quarterly*, **79**(1), 145–165. ISSN 0033-362X. doi: 10.1093/poq/nfu062. URL https://doi.org/10.1093/poq/nfu062.

Zhang, M. a Hurley, N. (2008). Avoiding Monotony: Improving the Diversity of Recommendation Lists. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, RecSys '08, page 123–130, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605580937. doi: 10.1145/1454008.1454030. URL https://doi.org/10.1145/1454008.1454030.

Zheng, Y., David a Wang (2023). Multi-Objective Recommendations: A Tutorial.

Ziegler, C.-N., McNee, S. M., Konstan, J. A. a Lausen, G. (2005). Improving Recommendation Lists through Topic Diversification. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, page 22–32, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595930469. doi: 10.1145/1060745.1060754. URL https://doi.org/10.1145/1060745.1060754.

Seznam obrázků

Seznam tabulek

Seznam použitých zkratek

- **HCI** - Human-computer interaction, interakce člověka a počítače
- **GUI** - Graphical User Interface, grafické uživatelské rozhraní

A. Přílohy k textu

A.1 Seznam otázek z dotazníku v češtině

Ty otázky, na které se odpovídá pomocí Likertovy škály označíme *.
Kontroly pozornosti označíme ^.
Odpovědi, pro které se kromě textu zobrazuje i jejich grafické vyjádření označíme ~.

A.1.1 Demografické údaje

- O1: Věková skupina
  - Méně než 18 let
  - 18 - 25
  - 26 - 35
  - 35 - 50
  - 50+
  - Nepřeji si odpovídat
- O2: Pohlaví
  - Muž
  - Žena
  - Jiné / Nepřeji si odpovídat
- O3: Vím, co je to doporučovací systém.*
- O4: Jsem obeznámen se základy doporučovacích systémů a strojového učení. *

A.1.2 Informace o filmech

- O5: Informace poskytnuté z náhledu doporučených filmů (plakát, první řádky popisu, režisér) byly dostatečné k tomu, abych mohl posoudit, zda se mi film bude líbit.*
- O6: Informace poskytnuté při kliknutí na detail doporučeného filmu (plakát, popis, žánry, trailer, režisér, herci) byly dostatečné k tomu, abych mohl posoudit, zda se mi film bude líbit.*

A.1.3 Explanations

- O7: Jaký typ náhledů vysvětlení byl nejužitečnější?
  - Kompletní ohraničení barvami kritérií~
  - Ohraničení barvami kritérií pouze vlevo~
  - Název položky v barvě kritéria s nejvyšším skóre~
  - Nevím / Žádná preference

- O8: Jaký typ vizualizace skóre kritéria byl nejužitečnější?
  - Procentuální podíl˜
  - Hrubé skóre˜
  - Zaokrouhlené skóre˜
  - Sloupcový graf˜
  - Nevím / Žádná preference
- O9: Jaká úroveň podrobnosti vysvětlení byla nejužitečnější?
  - Přínos všech metrik˜
  - Příspěvek metriky (metrik) s nejlepším skóre˜
  - Příspěvek metriky (metrik) s nadprůměrným skóre˜
  - Don't know / No preference
- O10: Celkově vysvětlení poskytla dostatek informací k pochopení toho, proč byla položka doporučena.*

A.1.4 Relevance

- O11: Popis relevance byl jasný a dostatečný.*
- O12: Podařilo se mi získat více (nebo méně) relevantních filmů úpravou váhy relevance.*
- O13: Jak jsem chápal relevanci filmu před účastí v této studii?
  - Filmy jsou relevantní, pokud jsou podobné filmům, které se mi líbí.
  - Filmy jsou relevantní, pokud jsou podobné filmům, které se mi líbí, a liší se od filmů, které se mi nelíbí.
  - Filmy jsou relevantní, pokud se zdá, že se mi budou líbit.
  - Jinak
    - Textové pole pro specifikaci
- O14: Jaký typ relevance se choval nejblíže tomu, co očekávám od relevance?
  - Filmy jsou relevantní, pokud jsou podobné filmům, které se mi líbí.
    - $=> EASE_{POS}$
  - Filmy jsou relevantní, pokud jsou podobné filmům, které se mi líbí, a liší se od filmů, které se mi nelíbí.
    - $=> EASE_{NEG}$
  - Nevím / Nevšiml(a) jsem si žádného rozdílu
- O15: Třetím použitým kritériem byla relevance. Jedná se o kontrolu pozornosti a musíte odpovědět „Rozhodně nesouhlasím".^*

A.1.5 Popularita

- O16: Popis popularity byl jasný a dostatečný.*
- O17: Podařilo se mi získat více (nebo méně) populární filmy tím, že jsem upravil váhu popularity.*
- O18: Jak jsem chápal popularitu filmu před účastí v této studii?
  - Nejznámější mezi uživateli
  - Nejlépe hodnocené uživateli
  - Trendy položky
  - Jinak
    - Textové pole pro specifikaci
- O19: Jaký typ popularity se choval nejblíže tomu, co od popularity očekávám?
  - Nejznámější mezi uživateli
    - => Popularita dle známosti
  - Nejlépe hodnocené uživateli
    - => Popularita na základě hodnocení
  - Nevím / Nevšiml(a) jsem si žádného rozdílu

A.1.6 Kalibrace

- O20: Popis kalibrace byl jasný a dostatečný.*
- O21: Podařilo se mi získat více (nebo méně) filmů z mých oblíbených žánrů úpravou váhy kalibrace.*

A.1.7 Diverzita

- O22: Popis diverzity byl jasný a dostatečný.*
- O23: Podařilo se mi získat více (či méně) diverzních (rozmanitých) filmů tím, že jsem upravil váhu diverzity.*
- O24: Jak jsem chápal diverzitu filmů před účastí v této studii?
  - Filmy s různými filmovými žánry jsou diverzní
  - Filmy, které jsou hodnoceny různými uživateli, jsou diverzní
  - Filmy, které se liší od filmů, které jsem hodnotil, jsou diverzní
  - Jinak
    - Textové pole pro specifikaci
- O25: Jaký typ diverzity se choval nejblíže tomu, co od diverzity očekávám?
  - Diverzita založená na pokrytí a minimálním opakování filmových žánrů

        * => Binomická diverzita
    - Minimální podobnost s některou z předchozích položek v seznamu doporučení
        * => Diverzita na základě maximální podobnosti
    - Průměrná odlišnost od předchozích položek v seznamu doporučení
        * => Intra-list diverzita
    - Nevím / Nevšiml(a) jsem si žádného rozdílu

A.1.8 Novelty

- O26: Popis novelty byl jasný a dostatečný.*
- O27: Podařilo se mi získat více (nebo méně) nových / neznámých / neotřelých předmětů tím, že jsem upravil váhu novelty.*
- O28: Pamatujete si, že jste narazili na tento film?^ [1]
    - Ano, hodnotil jsem tento film
    - Ano, tento film mi byl doporučen, ale nehodnotil jsem ho
    - Ne, nepamatuji si, že bych tento film viděl
- O29: Jak jsem chápal novelty u filmu před účastí v této studii?
    - Filmy, které se liší od filmů, které jsem hodnotil, jsou nové / neznámé / neotřelé
    - Filmy, které byly vydány v poslední době, jsou nové / neznámé / neotřelé
    - Filmy, které uživatelé nejméně znají, jsou nové / neznámé / neotřelé
    - Filmy, o kterých jsem dříve nevěděl, jsou nové / neznámé / neotřelé
    - Jinak
        * Textové pole pro specifikaci
- O30: Jaký typ novelty se choval nejblíže tomu, co od novelty očekávám?
    - Nejméně známý mezi všemi uživateli
        * => Očekávaný doplněk popularity
    - Minimální podobnost s kterýmkoli z filmů, které jsem hodnotil
        * => Distance-based novelty na základě maximální podobnosti
    - Průměrná odlišnost od filmů, které jsem hodnotil
        * => Intra-list distance-based novelty
    - Nevím / Nevšiml(a) jsem si žádného rozdílu

[1] Účastníkovi studie byl vždy zobrazen film, který ohodnotil.

A.1.9 Kritéria celkově

- O31: Možnost měnit relevanci, diverzitu, novelty, popularitu a kalibrační poměry pro mě byly užitečné.*
- O32: Celkově efekt vyladění relevance, diverzity, novelty, popularity a kalibrace splnil má očekávání.*
- O33: Celkově se doporučení zlepšila po úpravě vah relevance, diverzity, novelty, popularity a kalibrace.*

A.1.10 Filtr kritérií kvality doporučování

- O34: Který mechanismus mi poskytl nejdostatečnější kontrolu nad doporučeními?
  - Posuvníky (sliders)˜
  - Textová pole˜
  - Drag and drop˜
  - Tlačítka + a -˜
  - Nevím / Žádná preference
- O35: Nastavil jsem hodnotu relevance na 49. Jedná se o kontrolu pozornosti a musíte odpovědět „Souhlasím“.^ *
- O36: Který mechanismus byl nejúčinnější pro nastavení toho, co chci?
  - Posuvníky (sliders)˜
  - Textová pole˜
  - Drag and drop˜
  - Tlačítka + a -˜
  - Nevím / Žádná preference
- O37: Který mechanismus byl nejsrozumitelnější a nejintuitivnější?
  - Posuvníky (sliders)˜
  - Textová pole˜
  - Drag and drop˜
  - Tlačítka + a -˜
  - Nevím / Žádná preference
- O38: Váhy kritérií se nejjednodušeji nastavovali pomocí:
  - Posuvníků (sliders)˜
  - Textových polí˜
  - Drag and drop˜
  - Tlačítek + a -˜

  - Nevím / Žádná preference

- O39: Celkově lze říci, že mechanismem, který považuji za nejlepší pro specifikaci svých preferencí ke kritériím kvality doporučování, jsou:
  - Posuvníky (sliders)˜
  - Textová pole˜
  - Drag and drop˜
  - Tlačítka + a -˜
  - Nevím / Žádná preference

A.1.11 Celkově

- O40: Celkově jsem s doporučeními spokojen.*
- O41: Celkově považuji možnost upravit váhy kritérií za užitečnou pro získání lepších doporučení.*
- O42: Dávám přednost doporučením, aniž bych uváděl své preference. (Jinými slovy dávám přednost automatickému systému doporučování bez nutnosti specifikace vah ke kritériím.)*

A.1.12 Doplňující

- O43: Textové vyhledávání dle názvu filmu považuji za užitečné.*
- O44: Podrobnější filtr umožňující vyhledávání podle žánrů, režisérů, herců a data vydání považuji za užitečný.*
- O45: Možnost přímého blokování filmů považuji za užitečnou.*
- O46: Možnost blokovat filmy podle žánru, režiséra nebo herce považuji za užitečnou.*
- O47: Oceňuji možnost měnit barvy kritérií kvality doporučování.*

A.2 Seznam otázek z dotazníku v angličtině

Podrobnější informace k otázkám lze najít v příloze A.1. Zde popisujeme otázky v angličtině tak, jak na ně byli tázáni účastníci studie.

Ty otázky, na které se odpovídá pomocí Likertovy škály označíme *.

Kontroly pozornosti označíme ^.

Odpovědi, pro které se kromě textu zobrazuje i jejich grafické vyjádření označíme ˜.

A.2.1 Demographics

- O1: Age group
  - Under 18
  - 18 - 25
  - 26 - 35
  - 35 - 50
  - 50+
  - Do not want to answer
- O2: Gender
  - Male
  - Female
  - Other / Do not want to answer
- O3: I know what is a recommender system.*
- O4: I am familiar with the basics of recommender systems and machine learning.*

A.2.2 Information about movies

- O5: The information provided from the preview of the recommended movies (poster, first lines of description, director) was sufficient to judge whether I gonna like the movie.*
- O6: The information provided when clicking the details of the recommended movies (poster, description, genres, trailer, director, actors) was sufficient to judge whether I gonna like the movie.*

A.2.3 Explanation

- O7: What type of explanation previews was most helpful?
  - Coloured full border˜
  - Coloured left side of the border˜
  - Title in the colour of best scoring metric˜
  - Don't know / No preference
- O8: What type of metric's score visualization was most helpful?
  - Percentage share˜
  - Raw score˜
  - Rounded score˜
  - Bars with metrics colours˜

  - Don't know / No preference

- O9: What level of details for explanations was most helpful?
  - Contribution of all metrics˜
  - Contribution of best scoring metric(s)˜
  - Contribution of metric(s) with above average score˜
  - Don't know / No preference

- O10: Overall, explanations provided enough information to understand why an item was recommended.*

A.2.4 Relevance

- O11: The description of relevance was clear and sufficient.*
- O12: I was able to get more (or less) relevant items by tweaking the value of relevance.*
- O13: What was my understanding of relevance of a movie before participating in this study?
  - Movies are relevant if they are similar as movies that I like
  - Movies are relevant if they are similar as movies that I like and different than movies that I don't like
  - Movies are relevant if it seems I gonna like them.
  - Other

- O14: What type of relevance behaved closest to what I expect from relevance?
  - Movies are relevant if they are similar as movies that I like
  - Movies are relevant if they are similar as movies that I like and different than movies that I don't like
  - Don't know / Haven't noticed any difference

- O15: Relevance was the third used objective. This is an attention check and you have to answer "Strongly disagree".*ˆ

A.2.5 Popularity

- O16: The description of popularity was clear and sufficient.*
- O17: I was able to get more (or less) popular items by tweaking the value of popularity.*
- O18: What was my understanding of popularity of movie before participating in this study?
  - Most known by users

  - Best rated by users
  - Trending items
  - Other

- O19: What type of popularity behaved closest to what I expect from popularity?
  - Most known by users
  - Best rated by users
  - Don't know / Haven't noticed any difference

A.2.6 Calibration

- O20: The description of calibration was clear and sufficient.*
- O21: I was able to get more (or less) items of my favourite genres by tweaking the value of calibration.*

A.2.7 Diversity

- O22: The description of diversity was clear and sufficient.*
- O23: I was able to get more (or less) diverse items by tweaking the value of diversity.*
- O24: What was my understanding of diversity of movie before participating in this study?
  - Movies with different movie genres are diverse
  - Movies that are rated by different users are diverse
  - Movies that are different than the movies I rated are diverse
  - Other
- O25: What type of diversity behaved closest to what I expect from diversity?
  - Diversity based on coverage and minimal repetition of movie genres
  - Minimal similarity with any of previous items in list of recommendations
  - Average dissimilarity with previous items in list of recommendations
  - Don't know / Haven't noticed any difference

A.2.8 Novelty

- O26: The description of novelty was clear and sufficient.*
- O27: I was able to get more (or less) novel items by tweaking the value of novelty.*
- O28: Do you remember seeing this movie?^

  - Yes, I have rated this movie
  - Yes, this movie was recommended to me, but I haven't rated this movie
  - No, I don't remember seeing this movie

- O29: What was my understanding of movie's novelty before participating in this study?
  - Movies that are different than the movies I rated are novel
  - Movies that were more recently published are novel
  - Movies least known by users are novel
  - Movies that I previously did not know about are novel.
  - Other

- O30: What type of novelty behaved closest to what I expect from novelty?
  - Least known among all users
  - Minimal similiarity with any of movies I rated
  - Average dissimiliarity with movies I rated
  - Don't know / Haven't noticed any difference

A.2.9 Objectives overall

- O31: Being able to change the relevance, diversity, novelty, popularity and calibration ratios were useful for me.*

- O32: Overall, the effect of tweaking relevance, diversity, novelty, popularity and calibration fulfilled my expectations.*

- O33: Overall, recommendations improved after modifying the relevance, diversity, novelty, popularity and calibration ratios.*

A.2.10 Types of objectives filter

- O34: Which mechanism provided me with the most sufficient control over recommendations?
  - Sliders˜
  - Textboxes˜
  - Drag and drop˜
  - PLUS and MINUS buttons˜
  - Don't know / No preference

- O35: I have set the relevance value to 49. This is an attention check and you have to answer "Agree".*ˆ

- O36: Which mechanism was the most efficient in telling the system what I want?

  - Sliders˜
  - Textboxes˜
  - Drag and drop˜
  - PLUS and MINUS buttons˜
  - Don't know / No preference

- O37: Which mechanism was the most understandable and intuitive?
  - Sliders˜
  - Textboxes˜
  - Drag and drop˜
  - PLUS and MINUS buttons˜
  - Don't know / No preference

- O38: The objectives ratios were easiest to set with:
  - Sliders˜
  - Textboxes˜
  - Drag and drop˜
  - PLUS and MINUS buttons˜
  - Don't know / No preference

- O39: Overall, the mechanism that I find as best for tweaking the objectives is:
  - Sliders˜
  - Textboxes˜
  - Drag and drop˜
  - PLUS and MINUS buttons˜
  - Don't know / No preference

A.2.11 Overall

- O40: Overall, I am satisfied with the recommendations.*
- O41: Overall, I find the ability to tweak the objectives useful for getting better recommendations.*
- O42: I prefer getting recommendations without stating my preferences. (In other words I prefer the system recommending automatically without the need of setting the objectives.)*

A.2.12 Additional

- O43: I find the text-search on movie title helpful.
- O44: I find the movie filter on genres, directors, actors and release date helpful.
- O45: I find the ability to block movies directly ("Don't recommend this movie") helpful.
- O46: I find the ability to block movies on genre, director or actor helpful.
- O47: I appreciate the possibility of changing colours of the objectives.

A.3 Seznam akcí

Akce dělíme do skupin a platí, že každá akce ve skupině má stejnou prioritu. Zároveň je většinou každá skupina navázána na zobrazení alespoň jedné otázky z dotazníku. Nyní popíšeme a vyjmenujeme akce pro každou ze skupin a také vyjmenujeme otázky, jejichž zobrazení je na provedení všech akcí ze skupiny závislé.

A.3.1 Nastavení vah

Tato první skupina obsahuje jedinou akci zařizující, že uživatel se pokusil změnit váhy ve filtru kritérií kvality doporučování. Tato akce má nejvyšší prioritu 1.

Seznam akcí ve skupině:

- Použití filtru kritérií kvality doporučování

A.3.2 Typy mechanismů

Všechny akce z této skupiny reprezentují výběr daného mechanismu pro filtr kritérií kvality doporučování. Tyto akce mají prioritu 2.

Seznam akcí ve skupině:

- Výběr posuvníků
- Výběr textových polí
- Výběr tlačítek + a -
- Výběr drag and drop

Splněním všech těchto akcí je podmíněno zobrazení následujících otázek:

- O34: Který mechanismus mi poskytl nejdostatečnější kontrolu nad doporučeními?
- O36: Který mechanismus byl nejúčinnější pro nastavení toho, co chci?
- O37: Který mechanismus byl nejsrozumitelnější a nejintuitivnější?
- O38: Váhy kritérií se nejjednodušeji nastavovali pomocí:
- O39: Celkově lze říci, že mechanismem, který považuji za nejlepší pro specifikaci svých preferencí ke kritériím kvality doporučování, jsou:

A.3.3 Relevance

Všechny akce z této skupiny reprezentují nastavení metriky relevance a následné doporučení, kdy je skóre relevance počítáno na základě vybrané metriky. Tyto akce mají prioritu 2.

Seznam akcí ve skupině:

- Nastavení $EASE_{POS}$ jako metriky pro relevanci
- Nastavení $EASE_{NEG}$ jako metriky pro relevanci

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O14: Jaký typ relevance se choval nejblíže tomu, co očekávám od relevance?

A.3.4 Diverzita

Všechny akce z této skupiny reprezentují nastavení metriky diverzity a následné doporučení, kdy je skóre diverzity počítáno na základě vybrané metriky. Tyto akce mají prioritu 2.

Seznam akcí ve skupině:

- Nastavení intra-list diverzity jako metriky pro diverzitu
- Nastavení diverzity na základě maximální podobnosti jako metriky pro diverzitu
- Nastavení binomické diverzity jako metriky pro diverzitu

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O25: Jaký typ diverzity se choval nejblíže tomu, co od diverzity očekávám?

A.3.5 Novelty

Všechny akce z této skupiny reprezentují nastavení metriky novelty a následné doporučení, kdy je skóre novelty počítáno na základě vybrané metriky. Tyto akce mají prioritu 2.

Seznam akcí ve skupině:

- Nastavení intra-list distance-based novelty jako metriky pro novelty
- Nastavení distance-based novelty na základě maximální podobnosti jako metriky pro novelty
- Nastavení očekávaného doplňku popularity jako metriky pro novelty

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O30: Jaký typ novelty se choval nejblíže tomu, co od novelty očekávám?

A.3.6 Popularita

Všechny akce z této skupiny reprezentují nastavení metriky popularity a následné doporučení, kdy je skóre popularity počítáno na základě vybrané metriky. Tyto akce mají prioritu 2.

Seznam akcí ve skupině:

- Nastavení popularity dle známosti jako metriky pro popularitu
- Nastavení popularity na základě hodnocení jako metriky pro popularitu

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O19: Jaký typ popularity se choval nejblíže tomu, co od popularity očekávám?

A.3.7 Náhled explanations

Všechny akce z této skupiny reprezentují nastavení varianty náhledu vysvětlení. Tyto akce mají prioritu 3.

Seznam akcí ve skupině:

- Nastavení kompletního ohraničení barvami kritérií pro náhled vysvětlení
- Nastavení ohraničení barvami kritérií pouze vlevo pro náhled vysvětlení
- Nastavení názvu položky v barvě kritéria s nejvyšším skóre pouze vlevo pro náhled vysvětlení

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O7: Jaký typ náhledů vysvětlení byl nejužitečnější?

A.3.8 Vizualizace skóre kritéria

Všechny akce z této skupiny reprezentují nastavení varianty vizualizace skóre kritéria kvality doporučování. Tyto akce mají prioritu 3.

Seznam akcí ve skupině:

- Nastavení procentuálního podílu pro vizualizaci skóre
- Nastavení hrubého skóre pro vizualizaci skóre
- Nastavení zaokrouhleného skóre pro vizualizaci skóre
- Nastavení sloupcového grafu pro vizualizaci skóre

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O8: Jaký typ vizualizace skóre kritéria byl nejužitečnější?

A.3.9 Úroveň podrobnosti vysvětlení

Všechny akce z této skupiny reprezentují nastavení varianty úrovně podrobnosti vysvětlení. Tyto akce mají prioritu 3.
Seznam akcí ve skupině:

- Nastavení vysvětlení všech kritérií jako úrovně podrobnosti vysvětlení
- Nastavení vysvětlení jen ke kritériu s nejvyšším skóre jako úrovně podrobnosti vysvětlení
- Nastavení vysvětlení pouze kritérií s nadprůměrným skóre jako úrovně podrobnosti vysvětlení

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O9: Jaká úroveň podrobnosti vysvětlení byla nejužitečnější?

A.3.10 Textové vyhledávání

Tato skupina obsahuje jedinou akci s prioritou 4.
Seznam akcí ve skupině:

- Použití textového vyhledávání na základě názvu filmu

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O43: Textové vyhledávání dle názvu filmu považuji za užitečné.

A.3.11 Podrobnější filtr

Tato skupina obsahuje jedinou akci s prioritou 4.
Seznam akcí ve skupině:

- Použití podrobnějšího filtru k vyhledávání filmu dle žánrů, režiséra, herce, nebo data vydání

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O44: Podrobnější filtr umožňující vyhledávání podle žánrů, režisérů, herců a data vydání považuji za užitečný.

A.3.12 Přímé blokování filmu

Tato skupina obsahuje jedinou akci s prioritou 4.
Seznam akcí ve skupině:

- Nastavení bloku přímo na film

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O45: Možnost přímého blokování filmů považuji za užitečnou.

A.3.13 Blokování filmu na základě vlastností

Tato skupina obsahuje jedinou akci s prioritou 4.
Seznam akcí ve skupině:

- Nastavení blokovacího pravidla na žánr, režiséra nebo herce

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O46: Možnost blokovat filmy podle žánru, režiséra nebo herce považuji za užitečnou.

A.3.14 Zobrazení detailu

Tato skupina obsahuje jedinou akci s prioritou 4.
Seznam akcí ve skupině:

- Kliknutí na náhled položky a zobrazení jejího detailu

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O6: Informace poskytnuté při kliknutí na detail doporučeného filmu (plakát, popis, žánry, trailer, režisér, herci) byly dostatečné k tomu, abych mohl posoudit, zda se mi film bude líbit.

A.3.15 Změna barev

Tato skupina obsahuje jedinou akci s prioritou 5.
Seznam akcí ve skupině:

- Nastavení barev pro kritéria kvality doporučování

Splněním všech těchto akcí je podmíněno zobrazení následující otázky:

- O47: Oceňuji možnost měnit barvy kritérií kvality doporučování.

A.4 Text nápovědy ke kritériím kvality doporučování

- Relevance
  - Relevance zařizuje doporučení filmů, které nejvíce odpovídají vašim preferencím.
  - Příklad: Pokud byste ohodnotili pouze prvních 6 filmů o Harrym Potterovi, nejrelevantnějšími filmy by byly zbývající 2 filmy o Harrym Potterovi.
- Diverzita
  - Cílem diverzity je omezit doporučování podobných filmů v seznamu doporučení.

  - Příklad: Pokud je prvním doporučeným filmem Toy Story : Příběh hraček, druhý film by se měl lišit od prvního, takže druhým doporučením by neměl být jiný animovaný film.

- Novelty
  - Cílem novelty je doporučovat filmy, které by pro vás mohly být nové. To znamená, že seznam doporučení by měl obsahovat filmy, které neznáte, nebo filmy, které by vás nenapadly.

- Popularita
  - Cílem popularity je doporučovat známé filmy.

- Kalibrace
  - Cílem kalibrace je doporučit seznam filmů, které mají podobný poměr žánrů jako má seznam filmů, které jste hodnotili.
  - Příklad: Pokud se vaše hodnocené filmy sestávají z 60 % akčních filmů a 40 % romantických filmů, tak film, který poměr žánrů v seznamu doporučení nejvíce připodobní poměru 60 % akčních filmů, 40 % romantických filmů a 0 % ostatních žánrů, bude mít největší skóre kalibrace.

A.5 Seznam textů explanations česky

- $EASE_{POS}$
  - Uživatelům, kterým se líbí podobné filmy jako vám, se velmi líbí tento film.
  - Uživatelům, kterým se líbí podobné filmy jako vám, se obvykle líbí tento film.
  - Uživatelům, kterým se líbí podobné filmy jako vám, se obvykle nelíbí tento film nebo jej neznají.
  - Uživatelům, kterým se líbí podobné filmy jako vám, se opravdu nelíbí tento film nebo jej neznají.

- $EASE_{NEG}$
  - Uživatelům, kterým se líbí a nelíbí podobné filmy jako vám, se velmi líbí tento film.
  - Uživatelům, kterým se líbí a nelíbí podobné filmy jako vám, se obvykle líbí tento film.
  - Uživatelům, kterým se líbí a nelíbí podobné filmy jako vám, se obvykle nelíbí tento film nebo jej neznají.
  - Uživatelům, kterým se líbí a nelíbí podobné filmy jako vám, se opravdu nelíbí tento film nebo jej neznají.

- Intra-list diverzita

  - Tento film je odlišný od všech předchozích filmů v tomto seznamu.
  - Tento film je docela odlišný od všech předchozích filmů v tomto seznamu.
  - Tento film je docela podobný předchozím filmům v tomto seznamu.
  - Tento film je velmi podobný předchozím filmům v tomto seznamu.
- Diverzita na základě maximální podobnosti
  - Žádný předchozí film v tomto seznamu není podobný tomuto filmu.
  - Žádný předchozí film v tomto seznamu není velmi podobný tomuto filmu.
  - Některé předchozí filmy v tomto seznamu jsou podobné tomuto filmu.
  - Některé předchozí filmy v tomto seznamu jsou velmi podobné tomuto filmu.
- Binomická diverzita
  - Tento film se žánrově liší od většiny předchozích filmů v tomto seznamu.
  - Tento film se docela žánrově liší od většiny předchozích filmů v tomto seznamu.
  - Tento film se žánrově příliš neliší od většiny předchozích filmů v tomto seznamu.
  - Tento film se žánrově vůbec neliší od většiny předchozích filmů v tomto seznamu.
- Očekávaný doplněk popularity
  - Tento film většina uživatelů nezná.
  - Tento film nepatří mezi nejznámější mezi uživateli.
  - Tento film je docela známý mezi uživateli.
  - Tento film uživatelé znají.
- Distance-based novelty na základě maximální podobnosti
  - Žádný vámi hodnocený film není podobný tomuto filmu.
  - Žádný vámi hodnocený film není velmi podobný tomuto filmu.
  - Tento film je docela podobný některým vámi hodnoceným filmům.
  - Tento film je velmi podobný některým vámi hodnoceným filmům.
- Intra-list distance-based novelty
  - Tento film v průměru není podobný vámi hodnoceným filmům.
  - Tento film v průměru není velmi podobný vámi hodnoceným filmům.
  - Tento film je v průměru docela podobný vámi hodnoceným filmům.
  - Tento film je v průměru velmi podobný vámi hodnoceným filmům.

- Popularita dle známosti
  - Tento film většina uživatelů zná.
  - Tento film je známý mezi uživateli.
  - Tento film není příliš známý mezi uživateli.
  - Tento film uživatelé neznají.
- Popularita na základě hodnocení
  - Tento film obdržel od uživatelů velmi dobrá hodnocení.
  - Tento film obdržel od uživatelů dobrá hodnocení.
  - Tento film nepatří mezi nejlépe hodnocené.
  - Tento film není dobře hodnocen.
- Kalibrace
  - Tento film pomáhá připodobnit poměr žánrů v tomto seznamu poměru žánrů mezi filmy, které jsou vámi pozitivně hodnoceny.
  - Tento film docela pomáhá připodobnit poměr žánrů v tomto seznamu poměru žánrů mezi filmy, které jsou vámi pozitivně hodnoceny.
  - Tento film příliš nepomáhá připodobnit poměr žánrů v tomto seznamu poměru žánrů mezi filmy, které jsou vámi pozitivně hodnoceny.
  - Tento film vůbec nepomáhá připodobnit poměr žánrů v tomto seznamu poměru žánrů mezi filmy, které jsou vámi pozitivně hodnoceny.

B. Samostatné přílohy

B.1 Implementace

Součástí odevzdání je mimo text ještě implementace webové aplikace a doporučovacího systému. V hlavní složce jsou následující části:

B.1.1 Soubor README.md

V tomto souboru jsou velmi stručně představeny komponenty a způsob spuštění celého řešení.

B.1.2 Soubor docker-compose.yml

V tomto souboru je definována konfigurace všech komponent, služeb atd., které jsou součástí výsledného docker kontejneru, v rámci něhož lze celé řešení spustit.

B.1.3 Adresář Database

Adresář Database obsahuje zálohovací soubor celé databáze a skript, který po spuštění serveru provede obnovu databáze ze zálohovacího souboru.

B.1.4 Adresář moo-as-voting-fast

Tento adresář obsahuje upravenou a rozšířenou implementaci multi-objective doporučovacího systému včetně dokumentace spuštění doporučovacího systému, popisu komunikace se systémem a vygenerované dokumentace upravených a přidaných částí. Naše implementace je k dispozici také na GitHubu[1] stejně jako implementace původní.[2]

B.1.5 Adresář WebAppForMORecSys

V tomto adresáři je implementace webové aplikace, a to včetně stručné dokumentace popisující datový model a komponenty řešení a generované dokumentace. Implementace webové aplikace je také k dispozici na vyžádání na GitHubu.[3]

[1] https://github.com/patmach/moo-as-voting-fast
[2] https://github.com/pdokoupil/moo-as-voting-fast
[3] https://github.com/patmach/WebAppForMORecSys