

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

Adaptibilní systém pro doporučování obsahu

Bc. Jan Bouchner

Vedoucí práce: Ing. Jaroslav Kuchař

20. dubna 2014

Poděkování

Chci upřímně poděkovat všem, kteří mi věnovali čas, když jsem potřeboval pomoci při psaní této práce, především vedoucímu práce Jaroslavu Kuchaři za správné směřování, celkový vhled do technologií, cenné komentáře a nové nápady, které pomohly ke vzniku této práce. Děkuji také své rodině a přátelům za bezvýhradnou podporu během celých mých studií.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či spracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 20. dubna 2014

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2014 Jan Bouchner. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Bouchner, Jan. *Adaptibilní systém pro doporučování obsahu*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2014.

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Nahradte seznamem klíčových slov v angličtině oddělených čárkou.

Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

Klíčová slova Nahradte seznamem klíčových slov v češtině oddělených čárkou.

Obsah

Úvod	1
Základní pojmy	1
Motivace	4
Projekt	4
Struktura práce	4
1 Aktuální stav	5
1.1 Existující řešení	5
1.2 Sada algoritmů	5
2 Analýza	7
3 Ensemble	9
3.1 Online učení	9
3.2 Exploration vs. Exploitation	9
3.3 Multi-armed Bandit algoritmus	9
4 Principy a technologie	15
4.1 Vyhodnocovací technologie	15
4.2 Server	15
4.3 Api	15
4.4 Jak to udělat, aby to běželo rychle?	15
4.5 Principy a technologie technologií	15
4.6 Architektura řešení	16
4.7 Návrh komunikace	16
4.8 Parametrizace	16
5 Návrh a implementace	17
5.1 Aplikace	17
5.2 Formát zasílaných zpráv	17

5.3	Komunikační protokol	17
5.4	Nasazení	17
6	Experimenty a vyhodnocení	19
6.1	Testování různých způsobů chování	19
6.2	Experimenty	19
6.3	Zhodnocení aplikace	19
6.4	Budoucí práce	19
	Závěr	21
	Literatura	23
	A Seznam použitých zkratk	25
	B Obsah příloženého CD	27

Seznam obrázků

0.1	Příklad rozdělení zobrazující popularitu hodnocení. Zdroj: http://en.wikipedia.org/ [3]	2
-----	--	---

Úvod

“We are leaving the age of information and entering the age of recommendation”.

Základní pojmy

Hned na úvod práce jsem si dovolil vypůjčit citát z článku *The Long Tail* [1] (česky Dlouhý chvost) bývalého šéfredaktora časopisu Wired Chrise Andersona, který je též autorem stejnojmenné teorie. Ta je založena na tom, že díky internetu lze nabízet širokou škálu produktů, které by dříve mohly jen sotva slavit prodejní úspěchy, neboť poptávka po nich je příliš malá.

Co je míněno tvrzením „které by dříve mohly jen sotva slavit prodejní úspěchy“? Vysvětlení je k nalezení v několika následujících odstavcích, po kterémžto vysvětlení bude vše zasazeno do kontextu zmíněné teorie.

Lidstvo a informace

Význam informací a dat je pro lidstvo odjakživa nezpochybnitelný. Již před nástupem digitálního věku byl jejich objem velký, takže ke spoustě informací, na základě kterých by si byl jednotlivec utvořil vlastní názor, nebylo snadné se dostat. Lidé se často spoléhali na obecnou oblíbenost daného produktu (ať už se jednalo o zboží, článek, hudebního interpreta a jiné), která byla ale spíše než cokoliv jiného určena aktuálním společenským trendem nebo informace čerpali ze zkušeností svých několika známých na základě jejich doporučení.

S masovým rozvojem internetu pak začalo množství dostupných informací růst obřím tempem (organizace IDC došla v článku *Extracting Value from Chaos* [2] k závěru, že objem světových dat se každé dva roky zdvojnásobuje). Nové informace jsou produkovány takřka každou sekundu a na jejich setřídění máme mnohem méně času než dříve. Není již v lidských silách udržet si o všem přehled. Takovým vývojem se přirozeně změnil pohled na data. Cílem



Obrázek 0.1: Příklad rozdělení zobrazující popularitu hodnocení. Zdroj: <http://en.wikipedia.org/> [3]

není shromáždit co největší objem, neboť při tomto pokusu bychom se zanedlouho dostali do stavu informačního zahlcení. Ceněnou schopností je naopak vytěžit maximální množství užitečných informací a ty využít pro získání nových vědomostí a alespoň přibližně správnou odpověď na potřeby konkrétního jedince.

Informace a The Long Tail Theory

Vrátím se nyní ještě jednou k teorii Dlouhého chvostu. Pro účely této práce ji nemá význam vysvětlovat dopodrobna, ale je vhodné zmínit alespoň hlavní myšlenku, která vychází z úvah výše. Nejdůležitějším poznatkem je to, že – z pohledu obchodního – s nárůstem informací dochází k fenoménu snižování prodejů dřívějších hitů (tedy těch několika málo produktů, které se na daném trhu dostanou k masovému publiku, a které určuje především trend) ve prospěch prvků nacházejících se v takzvaném dlouhém chvostu. Uživatel se tak může dostat k informacím, ke kterým by se obyčejným hledáním přes jeden z vyhledávačů třeba nikdy nedostal (pokud by tedy opravdu přesně nevěděl, co hledá). Bližší představu si lze utvořit při pohledu na obrázek 0.1. Na vodorovné ose leží jednotlivé produkty, svislá osa pak znázorňuje objem prodeje (či popularitu). Žlutá část pod grafem vpravo reprezentuje dlouhý chvost (zhruba 80 procent produktů, po kterých je na trhu poptávka v malých objemech). Zelená část nalevo pak znázorňuje těch několik dominujících produktů (označuje se též jako *hlava*). Tento fenomén si lze dost dobře ilustrovat na příkladu hudebního průmyslu, kde díky obrovským databázím interpretů nedochází k selekci a prodeji jen těch potenciálně nejúspěšnějších, ale uživatelům se dostává mnohem větší výběr.

Všechny výše zmíněné faktory zapříčinily rozvoj přístupu zvaného *informační filtrování*. Odtud už je jen krok k doporučování na míru, které se v

posledních několika letech rozmohlo zaváděním tzv. *doporučovacíh systémů*.

Doporučovací systémy

Doporučovací systémy pracují s předem známými daty a mohou uživateli pomoci v tom smyslu, že z obrovské množiny informací různými způsoby vyfiltrují relevantní podmnožinu. Například sledováním a vyhodnocováním historie používání služby uživatelem (doposud nakoupené produkty, seznam všech jím ohodnocených článků či žebříček preferovaných hudebních interpretů) nebo porovnáváním kontextu uživatele s ostatními lidmi využívajícími tu samou službu. Oběma způsoby je možné objevit pro uživatele zajímavé a jeho vkusu vyhovující položky.

Použití

Kamkoliv se v dnešní době na internetu podíváme, máme velkou šanci, že na takovéto systémy narazíme. Mohou mít sice různá jména, například *like*, *lidé, které můžete znát* nebo třeba *uživatelé kupující produkt X kupují též produkt Y*, ale všechny mají stejný význam, a to zaujmout či upozornit na něco či někoho konkrétního. Spousta elektronických obchodů, renomovaných aukčních domů, ale též serverů se zábavou na nich doslova staví svá podnikání. Sami jejich zákazníci navíc o doporučování stojí. Jsou vděční za navigaci a existuje vysoká pravděpodobnost, že dle ní budou provádět i své nákupy. Uživatelé ale nejsou hloupí – když dostávají špatná doporučení, odmítají je a ve výsledku odmítají celý systém.

Jako příklady úspěšně nasazených systémů lze uvést například doporučování zboží na serveru Amazon.com¹, nabídka filmů na serveru Netflix² či různých článků z databází na internetu (CNN, Google News).

Typy

Doporučovací systémy lze dle způsobu práce rozdělit do dvou skupin.

TODO

behavioral (založeny na interakci uživatelů se články) - most popular, collabo. filter., item to item a content (založeny na člancích) - content similarity, latest item zmínit nejznámější doporučovací algoritmy, collaborative filtering, most popular, text similarity atd. ze treba pro potreby doporučovani novinek je lepsi kolaborativni filtrovani a ze pro doporučovani obsahu je lepsi text similarity

¹Dle TODO je 35 procent prodaného zboží z předchozího doporučení.

²Dle TODO jsou 2/3 půjčených filmů z doporučení.

Motivace

Představme si nyní jakéhosi rádce, který dokáže v každém okamžiku rozhodnout, co je za daných okolností nejvhodnější. Rádce v klidu vyčkává, ale v okamžiku, kdy je tázán uživatelem na to, jakou metodou si má nechat doporučit obsah, aby z toho měl maximální užitek, je schopen ihned a bez dlouhého rozmýšlení odpovědět. Rádce dokáže i pružně reagovat na situaci, kdy náhle dojde ke změně preferencí či vyvstanou jiné nečekané události, které mají za následek doporučování nevhodného obsahu.

Projekt

Cílem práce tedy je navrhnout adaptibilní systém, který bude automaticky vhodně volit a kombinovat algoritmy. Bude využívat zpětnou vazbu ohledně kvality doporučení, která principem odměny za dobré doporučení nebo trestu za špatné ovlivní preference při kombinování. Zároveň bude zodpovědná za podporu velkého množství dotazujících se uživatelů.

Tato diplomová práce dokumentuje návrh a vývoj takového systému a s ním spolupracujících komponent.

Toto je seznam modulů, které byly pro potřeby práce vyvinuty:

- **Adaptibilní systém pro doporučování obsahu.** The first item
- **Sada základních algoritmů určených k doporučování.** The second item
- **Rozhraní pro algoritmy a systému pro kombinování.** The third etc ...

Struktura práce

Tato práce je strukturována do X různých kapitol a popisuje celý vývojový cyklus systému. Kapitoly jsou řazeny v tom samém pořadí, v jakém probíhaly vývojové fáze.

Kapitola Experimenty a vyhodnocení The first item

Kapitola 6 The second item

Kapitola The third etc ...

Aktuální stav

Seznamte se s aktuálním stavem a proveďte rešerši.

1.1 Existující řešení

1.2 Sada algoritmů

Identifikujte sadu základních algoritmů určených pro kombinování. Ty implementujte nebo použijte existující implementace.

Analýza

2.0.1 Real time

live read + live write = real time recommendation <http://www.slideshare.net/d0nut/realtime-recommender-with-redis-hands-on>

Adaptibilní systém

Navrhněte, implementujte a otestujte systém, který bude dynamicky kombinovat algoritmy a optimalizovat celkovou kvalitu algoritmů.

3.1 Strojové učení

Vzhledem k povaze řešeného problému je nutné zaměřit se na metody tzv. *strojového učení*. Jedná se o vědeckou disciplínu (jednu z větví oboru umělé inteligence), která se zabývá tím, jak se má počítač přizpůsobit určité situaci, aniž by byl pro danou situaci explicitně naprogramován. Metody strojového učení, které hledají generalizaci libovolných dat nebo slouží pro adaptaci existujícího systému na změny okolí, se používají ve všech oblastech informačních věd od analýzy snímků, analýzy DNA a textu, až po simulace chování člověka.

Typickým příkladem je vytvořit z dostupných dat model, který dokáže:

- predikovat cenu akcií za 6 měsíců z aktuální výkonnosti společnosti a ekonomických dat
- rozpoznat spam od regulérního e-mailu
- u pacienta hospitalizovaného s infarktem predikovat riziko dalšího infarktu
- napomoci společnostem zabývajících se internetovou reklamou v rozhodování se, kterou reklamní strategii použít k maximalizaci zisků Používá to ale třeba Google analytics! při správě online experimentů https://support.google.com/analytics/answer/2844870?hl=cs&ref_topic=2844866

Algoritmy strojového učení mohou být děleny do taxonomie (nadtříd a podtříd) založené na požadovaném výsledku algoritmu nebo typu vstupu, který je k dispozici během trénování stroje. Algoritmů je celá řada, bude tedy vhodné zmínit zde alespoň ty nejtypičtější.

Supervised learning ³ TODO

Unsupervised learning ⁴ TODO

Reinforcement learning ⁵ TODO

TODO zmínit, že pro naše potřeby je nejvhodnější reinforcement learning.

Algoritmus začne ve stavu nevědomí (ignorant state), kdy neví nic o daných okolnostech a začíná nabývat vědomosti tím, že testuje systém. Tím, jak vstřebává data a vyhodnocuje výsledky, učí se, jaké chování je nejlepší.

Z psychologického hlediska se jedná o následující úvahu: Jakým způsobem dokáže trest a odměna ovlivňovat naše budoucí chování? Na základě čeho se (nejen) lidstvo učí novým věcem?

3.1.1 Online učení

Navrhovaná strategie je též nazývána jako *online učení*. Nutno zmínit, že slovem online zde není míněno něco ve smyslu internetu, ale ve smyslu neustále se vyvíjející aktualizace dat. Učící algoritmus v každém kole vykoná nějakou akci, přijme zpětnou vazbu a přepíše si daný zisk či ztrátu.

Z matematického hlediska má online učení propojení na klasické online algoritmy, teorii (opakovaných) her a teorii pravděpodobnosti. Díky těmto znalostem tak můžeme navrhovat pravděpodobností dynamické systémy, kterými lze modelovat složitá průmyslová zařízení nebo třeba výherní automat známý jako mnohoruký bandita (Multi-Armed Bandit).

3.2 Multi-armed Bandit algoritmus

The Multi-Armed Bandit Problem je jedním z klasických problémů online učení. Tato herní strategie je podobná tradičnímu hernímu automatu, který představuje one-armed strategii, ale mnohoruká varianta má více herních pák *V každém kole si lze pro hru vybírat mezi N automaty.*

3.2.1 Princip algoritmu

Strategii si lze představit tak, že stojíme před N výherními automaty (též nazývány jako bandité) a v každém kole máme možnost vybrat si jeden z nich, na kterém budeme hrát. Pravděpodobnosti výher u jednotlivých automatů jsou pro nás neznámé.

Formálně lze strategii popsat jako skupinu výnosových (reward) distribučních funkcí $B = R_1, \dots, R_K$, kde K je počet banditů. Každý bandita má tedy přiřazenu jednu distribuční funkci, jež vyjadřuje pravděpodobnost úspěchu.

³učení s učitelem

⁴učení bez učitele

⁵zpětnovazební učení nebo též učení posilováním

Zpočátku nemá hráč žádnou informaci o průběhu hry ani o rozložení pravděpodobnosti úspěchu napříč bandity. Tím, že v každém kole vybíráme vždy jen jednoho banditu, se snažíme navrhnout strategii pro maximalizaci celkové výhry.

Samozřejmě, že kdybychom znali banditu s největší pravděpodobností výhry, potom bychom jej vždy vybírali, čímž bychom maximalizovali výhry. Naším úkolem bude tedy nalézt nejlepšího banditu a to co nejrychleji, jak je to jen možné.

K návrhu strategie nám pomáhá to, že jednotlivé bandity nejdříve testujeme, abychom získali nutné znalosti. Poté už se lze zaměřovat na páky, které nám poskytují největší zuzitkované znalosti.

Úkol trochu komplikuje stochastická povaha banditů. Suboptimální bandita může vracet spoustu výher, což by nás mohlo přimět uvěřit, že právě tento bandita je velice štědrý. Podobně ale naopak nejlepší bandita může vracet spoustu proher. Měli bychom tedy pořád zkoušet i lůzry, nebo se na ně vykašlat?

Co je dalším problémem, tak pokud najdeme banditu, který vrací “docela dobré” výsledky, měli bychom toto zachovat a nadále rozvíjet naše “docela dobré” skóre, nebo bychom měli zkoušet i další bandity v naději, že nalezneme ještě lepšího? Tomuto se říká exploration vs. exploitation dilemma.

NEMOJE Necht u_1, \dots, u_K jsou průměrné hodnoty výnosových distribučních funkcí. Hráč aktivně zatáhne za páku každé kolo a sleduje příslušný výnos. jeho úkolem je maximalizovat součet těchto výnosů. Oželení (regret) po T kolech je definováno jako rozdíl mezi součtem výnosů příslušných výnosových funkcí optimální strategie a součtu získaných výnosů (VZOREC VIZ kybernář)

3.2.2 Exploration vs. Exploitation

NEMOJE Explorace Nacházení nových oblastí hledání - náhodné procházky, nevyužívá předchozích znalostí Exploatace Využití stávajících znalostí, uvíznutí v lokálních extrémech, rigidita

3.3 Bayesian Bandits

Ukazuje se ale, že nalézt optimální řešení je neuvěřitelně obtížné. A může trvat léta, než je celkové řešení vyvinuto. Existuje totiž také spousta přibližně optimálních řešení, které jsou docela dobré.

Právě to jedno z mnoha takových řešení (které lze velice dobře škálovat) je známo jako Bayesian Bandits a bude se jednat o online algoritmus, o kterém už padla řeč v sekci ?? a má přímou souvislost s zpětnovazebním učením (reinforcement learning).

NEMOJE PRUBEH ALGORITMU Bayesovské řešení začíná stanovením pravděpodobností výhry pro každého banditu z předchozích zkušeností. V našem případě jsou tyto pravděpodobnosti od 0 do 1.

Prior a posterior by Verča "před provedením pokusu"- např. pravděpodobnost, že hodím sudý číslo na kostce je $1/2$, a "aposteriori" je po provedení pokusu - nedřív uděláš pokus - to se používá třeba u Bayesova vzorce. ale tý tabulce upřímně moc nerozumím... to prior má být asi jakože to, co předpokládáš, že bys měl mít a to posterior je, jak to jakože "odhadneš" třeba zdat - jakože pokusem

V každém kole: Vzorkování náhodné veličiny X_b Výběr bandity s největším vzorkem (např. vyber banditu $B = \operatorname{argmax} X_b$) Pozorujme výsledek vybrání bandity B a updatujme priors (takový ty prostě věci z dřívějšího vyhodnocování - pokusy a výhry) Návrat na 1)

A to je vše. Z výpočetního hlediska algoritmus zahrnuje vzorkování z N distribucí. Počáteční priors jsou $\text{Beta}(\alpha = 1, \beta = 1)$ (uniformní rozdělení) a pozorovaná náhodná veličina X (výhra či prohra, zakódována jako 1, respektive 0) je Binomialní, proto posterior (po provedení pokusu) je $\text{Beta}(\alpha = 1 + X, \beta = 1 + 1 - X)$

Takže pokud bychom měli odpovědět na otázku z dřívějška (zda vybírat i ty lůzry), tak tento algoritmus navrhuje to, abychom nevyřazovali lůzry, ale měli bychom je vybírat s klesajícím tempem, jakmile nashromáždíme dost jistoty, že existují lepší bandité. Vyplývá to z podstaty, že vždy existuje nenulová šance, že lůzr dosáhne statusu B , ale pravděpodobnost této události se snižuje s tím, jak hrajeme více kol.

TODO možná nějaký obrázky (grafy) jak se to pulls vyvíjí (všechno je v článku na campdb)

Všimněme si, že se zase až tolik nestaráme o nějaké závěry nad skrytými pravděpodobnostmi, spíše se pro tento problém zabýváme o výběr nejlepšího bandity (nebo přesněji slovy, stále jistějšího během vybírání banditů)

To je třeba vidět v tom grafu níže, že distribuce červeného bandity je velice široká (což představuje neznalost o tom, jaká by skrytá pravděpodobnost vůbec mohla být), ale jsme si docela dobře jistí, že není nejlepší, algoritmus se tedy rozhodne tohoto banditu ignorovat.

ZPETNA VAZBA Multi-armed bandit je příklad Single-stage typu Reinforcement learningu (viz https://cw.felk.cvut.cz/wiki/_media/courses/a3m33ui/prednasky/files/ui-2010-p11-reinforcement_learning.pdf). Tedy snažíme se po jedné akci ihned uplatňovat feedback. Každá akce v multi-armed je nazývána jako jedna hra. Po každé hře at obdržíme (stochastický) reward t : $E[r_t|a_t] = Q^*(a_t)$ a z dlouhodobého hlediska je právě cílem maximalizovat reward. "To solve the multi-armed bandit problem, one must explore a variety of actions and exploit the best of them"

$Q^*(a)$ je "action-value odhad - v čitateli součet rewardu z jednotlivých výběrů / počet výběrů

Pak je ještě "greedy akce" ve hře t - $a_t^* = \operatorname{argmax} Q_t(a)$

Zajímavé poznámky V případě jakéhokoliv úspěchu doporučení by se měla navýšit někde hodnota pravděpodobnosti, se kterou bude algoritmus znovu vybrán. V případě neúspěchu pak by se tato pravděpodobnost měla exponenciálně snížit. V další iteraci už se bude systém rozhodovat s touto pravděpodobností mezi objevováním a zužitkováním. V případě zužitkování se vybírá z algoritmů, co již předtím něco vynesly. To je ta strategie learnable - u banditů to funguje tak, že poměr mezi objevováním a zužitkováním se mění v čase a ne v závislosti na předchozích výsledcích. Algoritmy jsou rozděleny ve svých dimenzích a každé té dimenzi je přiřazena určitá pravděpodobnost definující míru úspěšnosti nalezení toho, že tenhle algoritmus je fakt nejlepší. V případě, že uživatelé zužitkovávají tyto znalosti, tak si poté vybírají algoritmus, který má maximální pravděpodobnost z dané množiny. *epsilon greedy!!!!* - varianta *epsilon decreasing* Jak to má kybernetik: Je-li pirát ve stavu objevování, tak v případě útoku na loď nalezne příslušný prostor, který obsahuje místo útoku, a zvýší mu hodnotu pravděpodobnosti. Když je ve stavu zužitkování znalostí, tak pluje do prostoru, kde hledá transportní loď. V případě, že za celý cyklus nenalezne žádnou loď, tak se exponenciálně sníží hodnota pravděpodobnosti daného prostoru. Nalezne-li loď, tak zvýšení pravděpodobnosti proběhne jako ve stavu objevování.

ZAVEREM Algoritmus je velice jednoduchý, proto je také jednoduché jej rozšířit. V mém případě bylo potřeba přidat learning rates - předpokládejme, že to prostředí se prostě může měnit v čase. Technicky by se tedy standardní Bayesian Bandit algoritmus self-updatoval tím, že by se učil tím, že to, co si myslel, že se zdá jako nejlepší, selhává nejčastěji. Také lze docílit toho, aby se algoritmus učil měnícím se prostředím rychleji. Potřebuje pro to jedinou věc - přidat rate napříč updatováním.

Pokud rate menší než 1, algoritmus bude zapomínat předchozí výsledky rychleji a tlak na neznalost bude směrem dolů. Naopak rate větší než 1 implikuje to, že náš algoritmus se bude chovat více riskantně a bude vsázet na dřívější výhry častěji a bude tedy více odolný proti změnám měnícího se prostředí.

3.3.0.1 Důraz na zpětnou vazbu

počet kliků, objednávky, čas na stránce...

3.3.0.2 Jednoduchý průzkum

mezi price/time my plýtváme na offline evaluaci a price je kvůli špatným doporučením ztracena

Principy a technologie

Analýza a návrh systému pro kombinování metod pro doporučování obsahu v reálném čase

4.1 Vyhodnocovací technologie

Výpočty. kvůli kombinování budeme počítat s floaty

4.2 Server

Je třeba nějaký server, na kterém to bude běžet

4.3 Api

4.4 Jak to udělat, aby to běželo rychle?

musíme zpracovávat data velice rychle rychlý webový server, rychlý síťový protokol, rychlou frontu zpráv, rychlé úložiště

4.5 Principy a technologie technologií

Sekce se zabývá výběrem vhodných technologií REST, Aplikační server, NoSQL, ZeroMQ, Hadoop, Apache mahout

`http://contest.plista.com/wiki/example`

4.6 Architektura řešení

4.7 Návrh komunikace

router, dealer, worker thready

4.8 Parametrizace

volba parametrů, ať už těch, co ovlivňují tresty nebo třeba úložiště (redis...)

Návrh a implementace

Realizace toho systému, jak jsem ty části propojil dohromady a tak

5.1 Aplikace

5.2 Formát zasílaných zpráv

5.3 Komunikační protokol

5.4 Nasazení

samotné aplikace, potom ještě rest api

Experimenty a vyhodnocení

kecy o testování

6.1 Testování různých způsobů chování

viz jak jarda vymyslel těch zhruba 5 příkladů, co mohou nastat

6.2 Experimenty

6.3 Zhodnocení aplikace

Slovní zhodnocení

6.4 Budoucí práce

bude li nějaká

Závěr

Literatura

- [1] Anderson, C.: The Long Tail. [online], říjen 2004. Dostupné z: <http://archive.wired.com/wired/archive/12.10/tail.html>
- [2] John Gantz, D. R.: Extracting Value from Chaos. [online], červen 2011. Dostupné z: <http://www.emc.com/collateral/analyst-reports/idc-extracting-value-from-chaos-ar.pdf>
- [3] Wikipedia: Long tail — Wikipedia, The Free Encyclopedia. 2014, [Online; stav z 19. dubna 2014]. Dostupné z: http://en.wikipedia.org/w/index.php?title=Long_tail&oldid=603431399

Seznam použitých zkratk

GUI Graphical user interface

XML Extensible markup language

Obsah přiloženého CD

	readme.txt.....	stručný popis obsahu CD
	exe	adresář se spustitelnou formou implementace
	src	
	impl.....	zdrojové kódy implementace
	thesis	zdrojová forma práce ve formátu L ^A T _E X
	text	text práce
	thesis.pdf	text práce ve formátu PDF
	thesis.ps	text práce ve formátu PS