Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

FIIT-0000-00000

Michal Fašánek

Pokročilé spracovanie sekvenčných dát pomocou umelých neurónových sietí

Diplomová práca

Vedúci práce: Ing. Michal Barla, PhD

Január, 2016

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

FIIT-0000-00000

Michal Fašánek

Pokročilé spracovanie sekvenčných dát pomocou umelých neurónových sietí

Diplomová práca

Študijný program:	Informačné systémy
Študijný odbor:	Informačné systémy
Miesto vypracovania:	Ústav Aplikovanej Informatiky
Vedúci práce:	Ing. Michal Barla, PhD

Január, 2016

>>>> ASSIGNMENT <<<<
>>>> ZADANIE <<<<

Anotácia

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ

Študijný program:	Informačné systémy
Diplomová práca:	Pokročilé spracovanie sekvenčných dát pomocou umelých neurónových sietí
Autor:	Michal Fašánek
Vedúci práce:	Ing. Michal Barla, PhD

Január, 2016

Odporúčacie systémy slúžia na generovanie pridanej hodnoty z dát, ktoré sa vytvárajú pri používateľských prístupoch k online službám a produktom. Takéto odporúčania dokážu zlepšiť online biznis aj zákaznícky zážitok z používania systému. Najnovšie prístupy skúmajú možnosti aplikácie metód strojového učenia pre zlepšenie týchto odporúčaní. V tejto práci sa zameriavam na možnosti využitia rekurentných neurónových sietí s dlhou krátkodobou pamäťou(LSTM) pri analýze dát z elektronických portálov sprostredkujúcich produkty a služby online. Takáto analýza poskytuje náhľad do používateľského správania a z toho vyplývajúce možnosti spätnej väzby voči návštevníkom online portálov. Manažment biznisu orientovaného na zákazníkov si čoraz viac uvedomuje cenu verného zákazníka na konkurenčnom trhu. Je preto nutné pri generovaní odporúčaní ponúknuť len to najrelevantnejšie a nezahlcovať zákazníka, čo ho odrádza od opätovnej návštevy portálu.

Annotation

Slovak University of Technology Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Study program:	Informačné systémy
Master thesis:	Advanced processing of sequential data by artificial neural networks
Author:	Michal Fašánek
Supervisor:	Ing. Michal Barla, PhD
2016, January	

Recommender systems are used to generate additional value from data that is constantly created by user access to the online services and products. Such recommendations can improve online business and customer experience from accessing system. new methods research possibilities of applying machine learning to improve these recommendations. In this paper I focus on using recurrent neural networks with long short-term memory(LSTM) in data analysis from electronic portals providing online products and services. Such analysis lets us peer into the behavior of users and the possibility of feedback it creates. Management of customer-oriented business is realizing value of loyal customer more than ever before on this competitive market. It is therefore necessary to generate only relevant recommendations that do not annoy or spam customers, which might decrease their willingness for return.

Obsah

1 Úvod **1**

1.1 Použité skratky 2

2 Problémová oblasť **3**

2.1 Cross-selling 3

2.2 Analýza nákupného košíka 4

2.2.1 Support 4

2.2.2 Confidence 5

2.2.3 Lift 5

2.3 Odporúčacie systémy 5

2.3.1 Odporúčacie systémy založené na obsahu 6

2.3.2 Kolaboratívne odporúčacie systémy 6

2.3.3 Hybridné systémy 6

2.3.4 Problémy odporúčania 7

3 Dáta sprístupnené pre prácu **8**

3.1 Získavanie dát 8

3.1.1 Používateľská session 9

4 Neurónové siete **10**

4.1 Štruktúra 10

4.2 Učenie neurónovej siete 12

4.3 Optimalizácia 13

4.3.1 Gradient descent 13

4.4 Hyperparametre . . . 14
4.5 Aktivačná funkcia . . . 14
4.6 Rekurentné neurónové siete . . . 15
4.7 Siete s dlhou krátkodobou pamäťou - LSTM . . . 16

5 Výskum v danej oblasti **18**
5.1 RecSys Challenge 2015 . . . 18
5.1.1 Dáta . . . 18
5.1.2 Predikcia nákupov pomocou LSTM BiRNN . . . 20
5.1.3 Hlboká konvolučná neurónová sieť . . . 22

6 Dizajn **24**
6.1 Návrh . . . 24
6.1.1 Predspracovanie . . . 24
6.1.1.1 Vyvažovanie datasetu . . . 25
6.1.1.2 Normalizácia . . . 26
6.1.1.3 Architektúra siete . . . 26
6.2 Implementácia . . . 27

7 Výsledky **28**
7.1 Trénovanie . . . 28
7.2 Testovanie . . . 29

Literatúra **30**

A Technická dokumentácia **A-1**
A.1 Implementácia . . . A-1

B Používateľská dokumentácia **B-1**

C Elektronické médium **C-1**

D Plán letného semestra - DP3 **1**

Kapitola 1

Úvod

Ľudský mozog je zložitý a ešte v dnešných dobách z veľkej časti nepochopený orgán. Od prvého okamihu je trénovaný riešiť problémy, ktorých formálna špecifikácia presahuje naše možnosti. Snahou neurónových sietí je napodobniť takéto schopnosti a možnosti simuláciou architektúry mozgu.
Jednou z možných aplikácií takýchto schopností je prenikanie do mysle zákazníka na elektronickom trhu služieb a produktov. Online biznis má prostriedky, ktoré mu dovoľujú zhromaždiť obrovské množstvo dát o aktivite svojich zákazníkov. Na získanie pridanej hodnoty z týchto dát je však nutné identifikovať a pochopiť vzory, ktoré sa v týchto dátach nachádzajú. Takáto úloha je pre neurónové siete adekvátnou výzvou. Cieľom tejto práce je analyzovať správanie zákazníkov pri online nákupoch a generovanie odporúčaní produktov, o ktoré by mohol mať zákazník záujem pred tým, než ukončí svoj nákupu. Takéto poznatky smerujú k zlepšeniu biznisu a taktiež zákazníckej spokojnosti.

Použité skratky

- **NN, ANN** - Neurónová sieť (Neural Network, Artificial Neural Network)
- **FNN** - Dopredná neurónová sieť (Feedforward Neural Network)
- **RNN** - Rekurentná neurónová sieť (Recurrent Neural Network)
- **LSTM** -Dlhá krátkodobá pamäť (Long Short-Term memory)
- **CRM** - Manažment vzťahov so zákazníkmi (Customer Relationship Management)
- **VAW** - Web s pridanou hodnotou (Value Added Web)
- **IP** - Internetový protokol (Internet Protocol)

Kapitola 2

Problémová oblasť

Problémovú oblasť v tejto práci predstavuje výber produktov a služieb v internetovom predaji, ktoré majú maximálny potenciál zaujať zákazníkov pri práci s predajným portálom. Na takéto vzťahy medzi produktami a zákazníkmi alebo produktami navzájom môže mať vplyv množstvo faktorov ako napríklad ročné obdobie, vek a pohlavie nakupujúceho, špeciálne zľavy. Táto problematika zahŕňa podproblémy, ktorým sa venujeme ako analýza nákupného košíka(Market basket analysis - ďalej ako MBA) alebo cross-selling.

Cross-selling

Cross-selling reprezentuje generický názov pre snahu predať prídavné produkty existujúcemu zákazníkovi. V snahe neodradiť zákazníka nezaujímavými ponukami sa kladie dôraz na výber najrelevantnejších produktov, keďže je dôležité zobraziť zákazníkovi čo najmenej ponúk. Zahltenie zákazníka ponukami, ktoré pre neho nie sú relevantné totiž často vedie k nepríjemnému zážitku zákazníka a prejavuje sa ako na nákupe tak aj na šanci, že zákazník sa znovu rozhodne využiť konkrétny portál pre elektronický nákup v budúcnosti. Cross-selling je často charakterizovaný v zmysle „Ako predstavíme správny produkt

správnemu zákazníkovi v správnom čase za pomoci správneho komunikačného kanálu pre zaistenie dlhodobého úspechu".

Analýza nákupného košíka

Populárna technika využívaná pre cross-selling sa nazýva MBA. Hlavná idea spočíva v tom, že produkty, ktoré si už zákazník vybral v aktuálnom nákupe obsahujú cenné informácie o smerovaní odporúčania pre zákazníka. MBA využíva tri základné metriky pre počítanie súvislostí medzi položkami nákupov vzhľadom na dostupné historické dáta.

- Support
- Confidence
- Lift

Support

Neobvyklé udalosti či položky často predstavujú informácie, ktoré nemajú dostatočný význam pre ich sledovanie. Support predstavuje spôsob, ako ich ignorovať. Pre konkrétny produkt/službu je sledovaný výskyt v dátach. Pokiaľ nedosahuje stanovenú úroveň, položka je ignorovaná ako nezaujímavá pre analýzu.

Položky, ktoré sú takýmto spôsobom zredukované, sú následne analyzované podľa apriori algoritmu. **Apriori algoritmus** združuje často nakupované položky podľa ich spoločného výskytu v nákupoch. Algoritmus pracuje iteratívne - najprv vytvára dvojice, potom trojice, atď., až kým neexistujú skupiny, ktoré by dosahovali potrebné minimum spoločných výskytov nato, aby boli vyhodnotené ako skupina.

Confidence

Confidence vyhodnocuje podmieňenú pravdepodobnosť výskytu jednej položky alebo skupiny položiek(RHS - right hand side) za predpokladu, že sa už v košíku nachádza iná položka alebo skupina(LHS - left hand side).

$conf(LHS \rightarrow RHS) = P(RHS|LHS) = \frac{P(RHS \cap LHS)}{P(LHS)} = \frac{supp(LHS \cap RHS)}{supp(LHS)}$

Lift

Lift predstavuje mieru zlepšenia výskytu hodnotenej položky za predpokladu položky v košíku. Je to pomer pravdepodobností výskytu oboch položiek v pomere k pravdepodobnosti výskytu položky, ktorú sa chystáme odporučiť(RHS)

$lift(LHS \rightarrow RHS) = \frac{P(RHS \cap LHS)}{P(RHS)} = \frac{conf(LHS \rightarrow RHS)}{supp(RHS)}$

Odporúčacie systémy

Odporúčacie systémy(ang. *recommenders*) predstavujú skupinu softvérových nástrojov pre odporúčanie položiek používateľom. Ako najčastejšie aplikácie v súčasnosti vystupuje odporúčanie v online nakupovaní, hudbe alebo zdrojoch informácií [17]. Odporúčacie systémy sú obľúbenou témou výskumu kvôli svojmu potenciálu pri aplikácii v biznis sfére.

Hlavnou úlohou odporúčacieho systému je poskytnúť odporúčania - zoradený zoznam položiek, kde nás zaujímajú položky s najvyšším hodnotením. Hodnotenie(angl. *rating*) predstavuje relevantnosť alebo zaujímavosť, ktorú pre používateľa daná položka má. Ako jednoduchý príklad hodnotenia môžeme uviesť skóre alebo počet hviezdičiek, ktoré používateľ priradí filmu, reštaurácii alebo hotelu. Odporúčacie systémy sa snažia odhaliť charakteristické črty cieľového používateľa a vygenerovať zoznam, ktorý zodpovedá jeho potrebám alebo záujmom [1].

Odporúčacie systémy založené na obsahu

Táto podmnožina odporúčacích systémov je založená na myšlienke, že používateľovi sa budú páčiť položky, ktoré zdieľajú charakteristické črty s položkami, o ktoré prejavil záujem v minulosti. Úlohou systému je teda evidencia charakteristických čŕt položiek a generovanie matíc vzdialeností medzi jednotlivými položkami. Výber odporúčaných položiek je následne realizovaný ako zoradený zoznam položiek podľa najkratšej vzdialenosti od množiny položiek, ktoré používateľ ohodnotil pozitívne a zároveň najväčšej vzdialenosti od negatívne hodnotených položiek [1].

Kolaboratívne odporúčacie systémy

Narozdiel od predošlého prístupu, kolaboratívne systémy sa spoliehajú na nepriame spojitosti medzi položkami. Analýzou predošlých preferencií sa vytvorí profil používateľa, ktorý je následne porovnávaný s profilmi ostatných používateľov. Vytvorený zoznam odporúčaní predstavuje najlepšie ohodnotené položky od používateľov s najväčšou zhodou používateľského profilu [1].

Hybridné systémy

Kombinácie prístupov sa snažia prekonať nedostatky oboch prístupov vzájomným dopĺňaním sa. Existuje niekoľko prístupov, ktorými je možné kombinovať prístupy [5]:

- Váhovanie - kombinácia výstupov oboch systémov do jedného odporúčania
- Prepínanie - vyberanie výstupu podľa situácie
- Miešanie - zobrazenie viacerých výstupov separátne
- Kombinovanie - spájanie podčastí do jediného systému
- Kaskáda - systém B schvaľuje položky vybrané iným systémom
- Augmentácia - systém B odporúča výber z výstupu systému A

- Meta-úroveň - model systému A je vstupom pre systém B

Problémy odporúčania

Slabinou odporúčacích systémov sú nedostatočné historické dáta. Problém predstavujú hlavne pre nových používateľov, ktorí si nevybrali dostatok položiek aby im bol vygenerovaný používateľský profil. V takomto prípade nie je možné robiť porovnanie s ostatnými používateľmi ani generovať na základe už vybraných položiek. Je to problémom kolaboratívneho aj obsahového odporúčania. Podobný problém predstavuje ohodnotená vzorka položiek. Je možné generovať odporúčania podobných položiek, ale bez ohodnotenia svojej skúsenosti predchádzajúcim používateľom nie je nijaká záruka, že odporúčanie bude vhodné [1].

Kapitola 3

Dáta sprístupnené pre prácu

Pre túto prácu boli sprístupnené dáta z používateľských prístupov na stránky e-shopu ponúkajúceho produkty a služby tretích strán za výhodnejších podmienok. Partneri(tretie strany) uverejňujú svoje akciové ponuky na portáli, ktorý im ponúka zákaznícky atraktívne prostredie. Zákazníci sú motivovaní navštevovať portál kvôli tomu, že portál uverejňuje iba akciové ponuky. Spoločnosti získavajú zákazníkov bez potreby nadmernej reklamnej kampane tým, že sa ich ponuka objaví na portáli s vysokou navštevovanosťou. Pre túto prácu disponujeme s niekoľkomesačnými používateľskými dátami, ktoré predstavujú zhruba desať miliónov záznamov od milióna používateľov.

Získavanie dát

Pri všetkých používateľských prístupoch a aktivitách portál eviduje potenciálne dôležité dáta o týchto prístupoch pre možnosti dátovej analytiky a rozpoznávania trendov v správaní zákazníkov pre generovanie pridanej hodnoty pomocou prispôsobovania sa a individuálneho prístupu k zákazníkom. Takéto biznis stratégie vedú k maximalizovaniu zisku z predaja a zákazníckej spokojnosti pri práci s portálom. Dáta sú kategorizované a inak spracúvané do použiteľnej podoby. Výsledkom je dataset, ktorý slúži ako základ pre prácu

odporúčacích systémov, ktoré táto práca skúma.

Používateľská session

Snahou dátového zberu je schopnosť rozpoznať zákazníka, ktorý pristupuje k portálu. Jeden ucelený prístup k portálu sa nazýva **používateľská session**. Charakterizuje ju používateľ a jeho navigácia hierarchiou stránok portálu. Jedinečný používateľ je charakterizovaný unikátnym identifikátorom - **cookie**, ktorá je sprostredkovaná prehliadačom používateľa portálu. Problematika cookie spočíva v tom, že jeden používateľ môže vystupovať pod rôznymi cookie identifikátormi. Takáto situácia môže byť spôsobená prístupom z rôznych zariadení alebo cieleným používateľským zásahom do cookie - zmazaním, po ktorom používateľ odosiela cez prehliadač odlišnú cookie identifikáciu. Session ohraničujúca jeden prístup je rozdeľovaná pokiaľ používateľ presiahne časový rámec vyhradený pre jednu session, ktorý štandardne predstavuje 30 minút [24].

Medzi najdôležitejšie dostupné údaje z platobného portálu patria:

- Cookie
- Používateľský účet
- Čas aktivity
- Prehliadaný obsah
- Druh aktivity
- Dodávateľ tretej strany
- ...

Z uvedených údajov je možné roztriediť cookies a analyzovať jednotlivé prístupy, pohyb po portáli a položkách(produkty a služby) a nákupy.

Kapitola 4

Neurónové siete

Koncept neurónových sietí vznikol v 40. rokoch minulého storočia inšpiráciou biologickými neurónovými sieťami v mozgu [16]. Cieľom bolo prekonať bariéru medzi tým, čo je pre ľudský mozog ľahko riešitelné ale ťažko formálne definovatelné matematickými pravidlami. Tieto problémy, ktoré riešime intuitívne, pri pokuse o formálnu špecifikáciu ukazujú, aké množstvo znalostí používame v každodennom živote. Ako vhodný príklad slúži vizuálne rozoznávanie objektov, ktoré je pre osobu samozrejmé, no až v posledných rokoch zaznamenávame prvé úspechy v tejto problematike pri použití NN [12].

Štruktúra

Podobne ako v mozgu, základ neurónovej siete tvoria neuróny a prepojenia medzi nimi. Neuróny sú organizované vo vrstvách, ktoré sa delia na 3 základné typy.

Vstupná vrstva - reprezentuje dáta, ktoré podsúvame sieti pre interpretáciu. Dáta musia byť pred posunutím vstupnej vrstve často predspracované, aby bola sieť schopná interpretovať ich. Počet neurónov na vstupnej vrstve je ovplyvnený množstvom dát, ktoré máme na vstupe. V sieti existuje iba jediná vstupná vrstva.

Výstupná vrstva - interpretácia dát sieťou. Výstupnú vrstvu je možné nazvať „výsledok"siete.

Skrytá vrstva - nachádzajú sa medzi vstupnou a výstupnou vrstvou. Ich počet určuje hĺbku siete. NN nemusí mať ani jednu skrytú vrstvu, no takáto sieť dokáže modelovať iba lineárnu závislosť. Všeobecne platí, že čím viac skrytých vrstiev má sieť, tým zložitejšie vzťahy dokáže simulovať. Zvyšujú sa však aj nároky na učenie a výpočtové nároky. Jediná skrytá vrstva vytvára pozoruhodný rozdiel v aplikovatelnosti modelu, keďže prekonáva hranicu lineárnej závislosti funkcie, ktorú model pokrýva. Pri vysokej zložitosti modelu je možné naraziť na problém preučenia, ktorý bráni sieti korektne generalizovať. Neexistuje nijaký spoľahlivá metóda pre správny počet alebo veľkosť skrytých vrstiev. Empiricky sa vyvinulo niekoľko odhadov, ale v praxi je nutné overovať správnosť modelu praktickou evaluáciou. Odhadové pravidlá najčastejšie padajú na neschopnosti integrovať vo svojom rozhodnutí komplexitu úlohy a redundanciu v tréningových dátach [12].

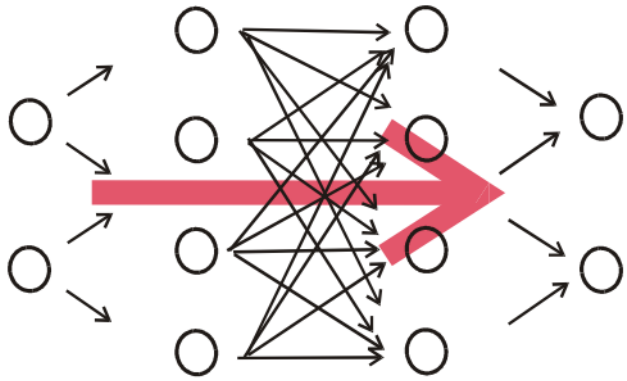
Prepojenia - Váhované prepojenia medzi neurónmi fungujú ako pamäť neurónovej siete. V jednoduchom modeli neurónovej siete sú prepojenia iba medzi neurónmi navzájom susediacich vrstiev. Prepojenie existuje medzi každým neurónom *n*-tej do *n+1* vrstvy. Neuróny jednej vrstvy pritom medzi sebou nie sú prepojené. Signál sa šíri týmito prepojeniami od vstupnej vrstvy smerom k výstupnej vrstve v jednom smere, ako je to ilustrované na obr. 4.2. Takéto siete sa volajú *dopredné*. Hlavný účel prepojenia je niesť váhu. Váha prepojenia určuje, aký významný je vzťah medzi dvomi danými neurónmi, ktoré spája. Korektná váha daného prepojenia je na začiatku neznáma, jej korektné nastavenie je výsledkom procesu učenia [12].

Neurón - predstavuje základnú stavebnú jednotku neurónovej siete. Skladá sa z *aktivačnej funkcie* a *prahovej hodnoty*. Prahová hodnota neurónu ϑ_i^{k+1} je odpočítaná od sumy vstupných váhovaných hodnôt $w_{ij}^k \,.\, o_j^k$.
Na výsledok o_i^{k+1} sa následne aplikuje aktivačná funkcia f podľa obr. 4.1. Takýto výstup je následne prepojeniami posielaný do ďalších neurónov. Špeciálny prípad je neurón vstupnej a výstupnej vrstvy. Na vstupe totiž neurón hodnotu iba posiela ďalej a na výstupe po spracovaní nie je zasielaná nikam -

predstavuje výsledok siete.

$$o_i^{k+1} = f\left(\sum_{j=1}^{N} w_{ij}^k \,.\, o_j^k \;-\; \vartheta_i^{k+1}\right)$$

Obr. 4.1: *Výstupná hodnota neurónu [15]*



Obr. 4.2: *Štruktúra doprednej neurónovej siete (FNN) [13]*

Učenie neurónovej siete

Učenie predstavuje kľúčovú aktivitou pre schopnosť siete produkovať požadované výsledky. Spočíva vo vystavovaní neurónovej siete tréningovým dátam, ktoré sa sieť snaží interpretovať.

Učenie s učiteľom je metóda, pri ktorej je dostupná sada tréningových dát „označená". Pri interpretovaní výsledku je možné okamžite určiť, aká chyba nastala a následne ju propagovať do siete. Na toto sa využíva tzv. *spätná propagácia*(backpropagation), ktorá upravuje váhy siete v rozsahu chyby, ktorá nastala - rozdiel medzi správnym výsledkom pre daný vstup a samotným výsledkom siete.

Učenie bez učiteľa predstavuje alternatívnu metódu, pri ktorej tréningové dáta nemajú dostupné výsledky. Neurónová sieť sa sama učí rozhodnúť, čo je pre ňu relevantné. Učenie bez učiteľa predstavuje možnosť ako získať takmer neobmedzené množstvá tréningových dát tam, kde učenie s učiteľom vyžaduje manuálne a kvôli časovej náročnosti nedostupné označovanie.

Optimalizácia

Snahou optimalizácie je hľadať ideálne riešenie v často obrovskom prehľadávanom priestore. V prípade neurónových sietí je to hľadanie optimálneho nastavenia siete, ktorá následne dokáže meniť vstup na predpokladaný výstup. Asi najčastejšou úlohou optimalizačných techník v neurónových sieťach je minimalizovať stratu(angl. *loss*). Tá naznačuje, ako ďaleko sa naša sieť nachádza od konvergencie k ideálnemu riešeniu. Je definovaná ako súčet chýb, ktoré boli dosiahnuté v tréningových alebo validačných množinách. Pri učení sa snažíme optimalizovať stratu, ale nie nulovať ju. V bežných prípadoch pri zašumených dátach totiž nulová chyba znamená preučenie - kopírovanie datasetu namiesto odhaľovania vzorov v ňom.

Gradient descent

Gradient descent predstavuje stratégiu spätnej propagácie v ktorej je vypočítaná derivácia chyby a tá je následne prenesená do parametrov siete ich upravovaním pre minimalizáciu chyby v nasledujúcej iterácii.

Obľúbená varianta je **stochastický gradient descent**, ktorý je síce menej presný, ale prenáša chybu po každej iterácii, narozdiel od originálneho stochastického gradientu, ktorý upravuje parametre až po celej sade. Ajkeď teda SGD nedosiahne presnosť GD, dostane sa do blízkosti riešenia oveľa rýchlejšie a bude tam oscilovať. Táto vlastnosť ho uprednostňuje pri väčších datasetoch.

Pri používaní gradient descent algoritmu je problematické správne odhadnúť

hyperparametre. Bežný postup je preskúmať široké okolie a sledovať ako vplýva zmena hyperparametrov na schopnosť minimalizácie chyby [23].

Hyperparametre

Nastavenia, pomocou ktorých kontrolujeme správanie neurónových sietí sa nazývajú *hyperparametre*. Tieto hodnoty nie sú získané učením siete pokiaľ nemodelujeme vnorený systém za týmto účelom. Príkladom hyperparametra je počet skrytých vrstiev NN. Pri nízkom počte nebude model schopný naučiť sa funkciu definovanú problémom. Pri vysokom počte je možné, že sieť v sebe uloží menší tréningový dataset, nazývané tiež ako problém *preučenia*(overfitting). Pri preučení sieť nezíska schopnosť generalizácie problému kvôli sledovaniu tréningového datasetu. Je zjavné, že zvolenie správnych hyperparametrov má pre výsledky metódy kľúčovú úlohu [12]. Medzi ďalšie významné hyperparametre patria:

- Šírka jednotlivých vrstiev - ovplyvňuje koľko prepojení existuje a mení tak nároky siete na učenie ako aj zložitosť vzorov, ktoré sa sieť dokáže naučiť.
- Rýchlosť učenia - rozhoduje o tom, v ako rozsahu bude upravovaná sieť pri spätnej propagácií. Vystupuje ako kvantifikátor aplikovanej zmeny váh.
- Momentum - predstavuje riešenie pre problém lokálneho minima aj pre osciláciu pri stochastickom gradient descente. Vysoké momentum je schopné prekonať lokálne minimum a pri oscilácii okolo globálneho riešenia sa spomaľuje [2].

Aktivačná funkcia

Aktivácia je matematická operácia aplikovaná na výstupy z predchádzajúcej vrstvy. Mení vstupné hodnoty na výstupný signál.

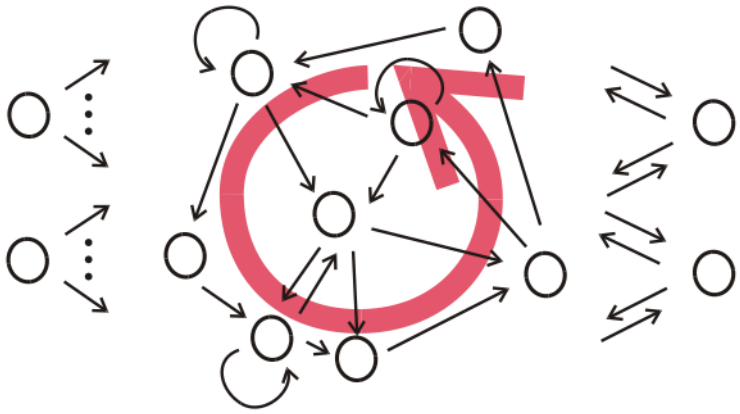
Sigmoidná funkcia - Produkuje signál v kladnom rozsahu $< 0, 1 >$. Najefektívnejšia je pre dáta ktoré sú na vstupe v rovnakom rozsahu [18].

ReLU - Usmernená lineárna jednotka, predstavuje najjednoduchšiu derivovateľnú nelineárnu funkciu. Nesaturuje výstup a vďaka tomu dosahuje dobré výsledky pre hlboké neurónové siete - nevzniká efekt miznúceho gradientu.

Softmax - Funkcia najčastejšie využívaná pri klasifikačných úlohách na výstupe. Škáluje výsledné neuróny tak, aby spoločný výstupný signál dosahoval hodnotu 1. [20]

Rekurentné neurónové siete

Do popredia výskumu sa v súčasnosti dostávajú pokročilé modely, ktoré už nie sú obmedzené na jednoduchý dopredný prístup. Vďaka rapídnemu zvyšovaniu výkonu grafických kariet sa čoraz častejšie aplikujú *rekurentné modely neurónových sietí*(RNN) [13]. Špecializáciou rekurentných sietí je práca so sekvenčnými dátami. Tieto siete predstavujú generalizáciu dopredných modelov ich rozšírením o cyklické prepojenia [12]. Takýmto spôsobom je možné využiť súčasnú hodnotu premennej na ovplyvnenie vlastnej hodnoty v budúcnosti. Cyklický charakter rekurentného modelu je zobrazený na obr. 4.3.



Obr. 4.3: *Štruktúra rekurentnej neurónovej siete [13]*

Siete s dlhou krátkodobou pamäťou - LSTM

LSTM predstavuje vylepšený model RNN. Vnútorná štruktúra ako doplnok ku externej rekurencii medzi jednotlivými neurónmi obsahuje aj *internú rekurenciu*, zobrazenú v štruktúre LSTM neurónu na obr. 4.4. Medzi najdôležitejšie súčasti tohto modelu patria sigmoidné brány, ktoré rozhodujú o tom, ako sa signál bude šíriť. LSTM tak prekonáva problém strácajúceho sa gradientu, ktorým trpí klasická RNN architektúra [11].

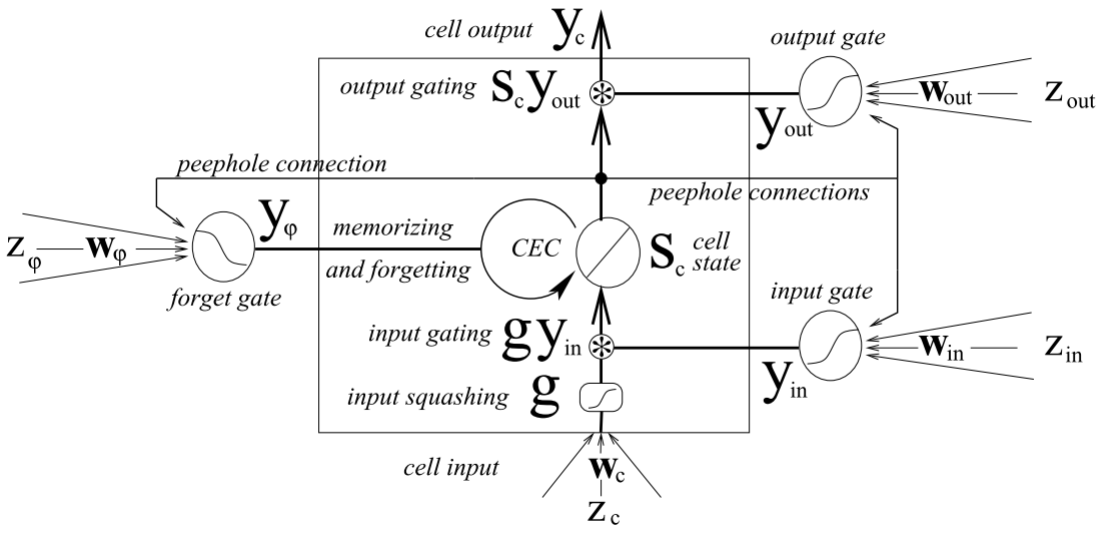
Brána zabudnutia ovplyvňuje, či nastáva vnútorná rekurencia neurónu. Stav tak môže ale nemusí byť faktorom ovplyvňujúcim nasledujúcu iteráciu výpočtu v sieti. Významné zlepšenie v LSTM sieťach prišlo s myšlienkou *kontextom podmieneného zabúdania*. Takýto model sa ukazuje extrémne výhodným pri riešení problémov zahŕňajúcich *časové pauzy*(lags) [6]. Dôležitý prvok na obr. 4.4 predstavuje čierna kocka. Označuje pauzu o veľkosti jednej iterácie. Hodnota signálu tak ovplyvňuje nasledujúcu iteráciu, tj. vplýva na neskoršie udalosti.

Nazeracie diery(peepholes) predstavujú vylepšenie LSTM. Rieši problémy,

ktoré vznikajú na základe faktu, že brána nedostáva priame informácie o stave jadra LSTM bloku(CEC). Táto situácia nastáva, keď je výstupná brána zatvorená. *Nazeranie* predstavuje techniku váhovaného prepojenia CEC s bránami bloku daného jadra. Prepojenia sú štandardné s výnimkou časovej pauzy.

LSTM siete v praxi dokázali svoje schopnosti pri aplikácií na rôzne netriviálne dátové problémy. Pozornosť je kladená na frekventovanú časovú závislosť v dátach:

- Rozoznávanie rukopisu [9]
- Rozoznávanie reči [8]
- Označovanie obrázkov [14]



Obr. 4.4: *Schéma nazerania v LSTM bloku [7]*

Kapitola 5

Výskum v danej oblasti

V tejto časti sa zaoberáme štúdiami skúmajúcimi aplikáciu data-miningu pri generovaní odporúčaní a aplikáciou neurónových sietí v dátovej analytike. Tieto štúdie vedú k optimalizácii poskytovania online produktov a služieb.

RecSys Challenge 2015

RecSys Challenge 2015 bola súťaž v data-miningu zameranom na odporúčacie zariadenia vyhlásená organizáciou ACM. Súťažiaci mali možnosť implementovať riešenie, ktoré by dokázalo s čo najväčšou úspešnosťou predpovedať na testovacej množine nákupy v online obchode [3].

Dáta

Pre túto súťaž boli poskytnuté dáta od vlastníka väčšieho európskeho portálu pre elektronický predaj. Dáta reprezentujú zhruba 6 mesačnú aktivitu používateľov na portáli. Dokopy súbory obsahujú vyše 33 miliónov záznamov z klikov zoskupených do 9.5 milióna sessions. Z toho bolo predaných 1.1 milióna položiek. Dáta sú rozdelené do dvoch súborov:

- yoochoose-clicks.dat
  - ID session, čas, ID položky, kategória položky

- yoochoose-buys.dat
  - ID session, čas, ID položky, cena, množstvo

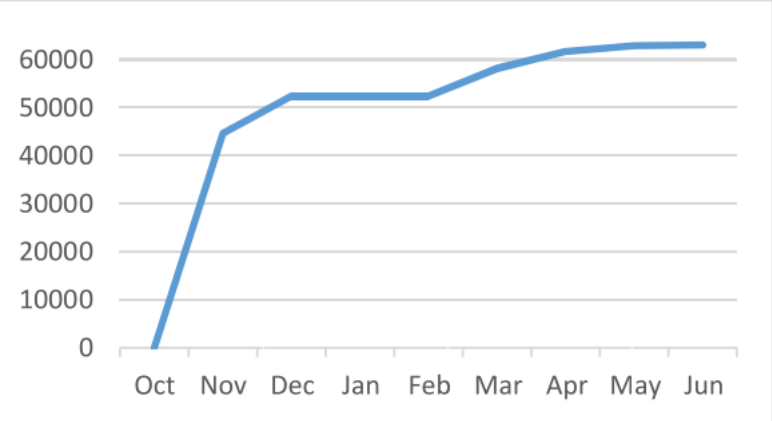
Cieľom súťaže bolo predpovedať, ktoré používateľské sessions budú nákupné a ktoré z prehliadaných položiek budú kúpené. Metrikou pre úspešnosť bol vzorec:

$$score(Sl) = \Sigma_{\forall S \in Sl} = \begin{cases} if s \in Sb \rightarrow \frac{Sb}{S} + \frac{|AS \cap BS|}{|AS \cup BS|} \\ else \rightarrow -\frac{Sb}{S} \end{cases}$$

kde platí:

- Sl - počet session vložených ako riešenie
- S - počet všetkých sessions v teste
- s - session v teste
- Sb - počet nákupných sessions v teste
- As - sada predikovaných nákupov v session s
- Bs - sada kúpených položiek v session s

Perfektné skóre dosiahnuteľné pre tento dataset s danou metrikou je 135176, pričom víťazný tím dosiahol skóre 63102. Na obrázku 5.1 možno sledovať najlepšie riešenia v priebehu mesiacov.



Obr. 5.1: *Najlepšie priebežné výsledky počas trvania súťaže [3]*

Predikcia nákupov pomocou LSTM BiRNN

Štúdia [22] využíva rekurentnú neurónovú sieť s krátkou dlhodobou pamäťou(LSTM) pri predikcií YooChoose datasetu vzorov v klikaní na stránkach internetového predajného portálu. Prístup oddeľuje dve separátne problematiky - predikciu nákupných sessions a následne predikciu kúpených položiek.

Dostupné dáta 5.1 sú predspracované na vstup do neurónovej siete. Tá následne po tréningu predpovedá výsledky pre testovacie dáta. Po predspracovaní vyzerajú jednotlivé vstupy do siete nasledovne:

Predpovedanie položiek
Mesiac
Deň v týždni
Čas dňa(minúty)
Počet klikov na položku
Počet nákupov položky
Aktuálna cena položky
Maximálna cena položky
Minimálna cena položky
Položka na predaj
Trvanie kliku
Kategória

Predpovedanie nákupnej session
Počet kikov v session
Počet unikátnych položiek v session
Počet klikov za minútu
Minimálny počet klikov v session
Maximálny počet klikov v session
Priemerný počet klikov
Trvanie session
Priemerné trvanie kliku
Minimálna dĺžka kliku
Maximálna dĺžka kliku
...

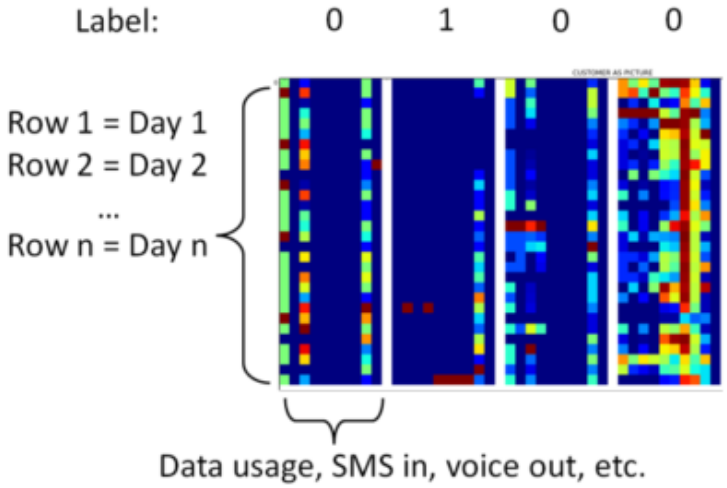
Táto práca sa porovnala s niekoľkými ďalšími metódami ako gradient-boosting regresia a hlboká neurónová sieť (DNN). Skóre sa odvíja od metriky pre yoochoose dataset 5.1.1.

	Session Score	Item Score	Total Score T
Item Features (II)			
BiRNN	-13117	59459	**47342**
Item Features (I)			
GBR	-17239	55677	38438
DNN	-17238	55094	37856
Session-Item Features (I)			
GBR	-14217	59582	45366
DNN	-13915	59097	45182
Session Features / Item Features (II)			
GBR	-12577	53773	41196
DNN	-12192	53013	40821
Session Features / Session-Item Features (II)			
GBR	-13217	60771	**47554**
DNN	-13016	59974	46958

Obr. 5.2: *Vyhodnotenie*

Hlboká konvolučná neurónová sieť

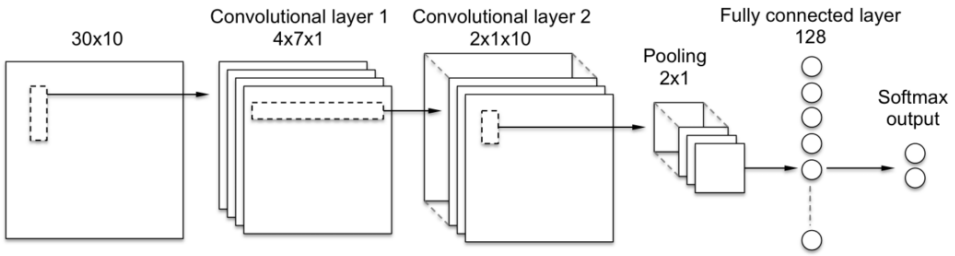
Pokročilé modely neurónových sietí ako hlboké konvolučné siete dosahujú veľmi dobré výsledky pri problémoch spracovania obrazu [19]. Za účelom využitia týchto vlastností štúdia skladá z používateľskej aktivity obrazovú mapu - dvojrozmerné pole normalizovaných pixelov. Za účelom učenia má každý obraz dostupné označenie, ktoré hovorí či daný zákazník prešiel ku konkurencii alebo nie. Obr. 5.3 zobrazuje ukážku aktivity zákazníka v dostupných službách za posledných n dní [21].



Obr. 5.3: *Aktivita zákazníka v mape pixelov. Hodnota pixelov sa zvyšuje od modrej k červenej [21].*

Experiment uvažuje 30-dňové okno predikcie, z ktorého sieť usudzuje aktivitu zákazníka. Okno sa nachádza 14 dní pred posledným registrovaný telefonátom. Pokiaľ sa posledný registrovaný telefonát nekonal v posledných 14 dňoch od aktuálneho dátumu, považujeme zákazníka za neaktívneho a neberieme ho do úvahy.
Po vytvorení obrazového datasetu z dostupných záznamov boli dáta podsunuté konvolučnej neurónovej sieti na obr. 5.4. Táto sieť má architektúru podobnú iným sieťam určeným pre spracovanie obrazu. Sieť analyzuje týždňové vzory v aktivite pomocou 7x1 filtra prvej konvolučnej vrstvy. Na konci

siete je pomocou binárneho softmax klasifikátora vyhodnotený výsledok.



Obr. 5.4: *Architektúra konvolučnej siete pre klasifikáciu zákazníkov z pixelovej mapy aktivity [21].*

Pomocou metódy *oblasti pod krivkou*(AUC) bolo zistené, že konvolučná sieť dosahuje lepšie výsledku ako model CHAID rozhodovacieho stromu. AUC vyhodnocuje pravdivé aj nepravdivé pozitívne výsledky [10] [4].

Kapitola 6

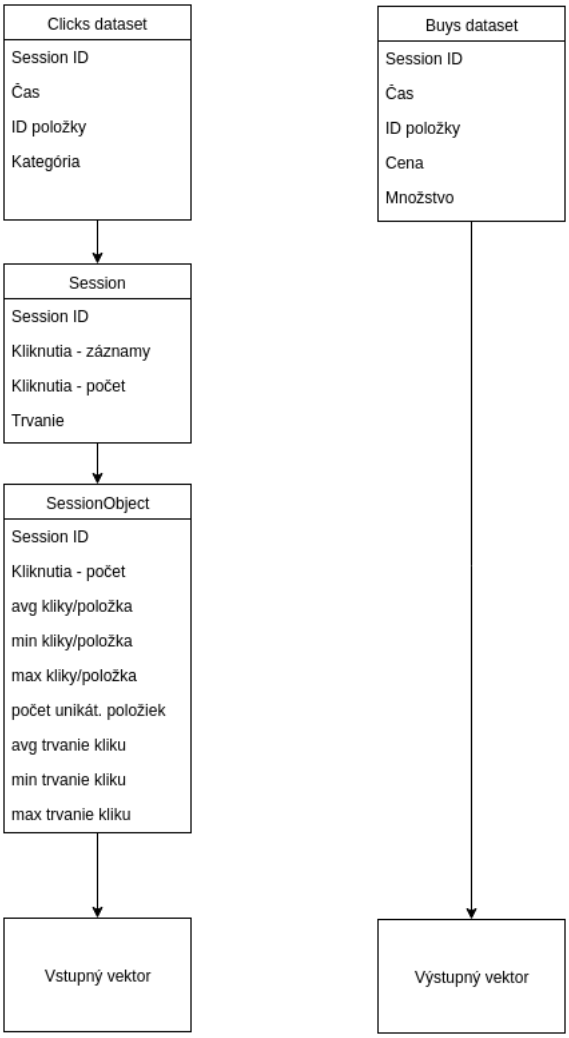
Dizajn

Návrh

Navrhli sme doprednú neurónovú sieť na klasifikáciu používateľských session pre YooChoose dataset. Na vstupe do nej vchádzajú predspracované dáta z používateľských aktivít a na výstupe je určená pravdeposobnosť pre to, do ktorej skupiny patrí.

Predspracovanie

Na obrázku 6.1 je znázornený proces predspracovania dát zo zdrojových súborov na vektory. Záznamy o nákupoch sú využité na generovanie tried pre tréningovú množinu a množinu správnych odpovedí vo validácií a v teste. Záznamy o klikoch generujú vstupné vektory ktoré sú klasifikované.



Obr. 6.1: *Na diagrame je znázornené predspracovanie dát*

Vyvažovanie datasetu

Pre korektné učenie klasifikačných úloh je nutné poskytnúť neurónovej sieti v datasete vyváženú reprezentáciu jednotlivých tried. Prvotné pokusy ukázali, že pri nevyváženom rozdelení datasetu(v našom prípade 1:9 pomer nákupných sessions a nenákupných sessions) sa neurónová sieť môže naučiť favorizovať vysoko zastúpenú triedu.
Pri našom pomere je štatisticky výhodnejšie určiť, že session nie je nákupná a

dosiahnuť v priemere 80-90% úspešnosť. Takáto informácia však nemá nijakú aplikáciu v praxi a preto preferujeme aj nižšiu úspešnosť so schopnosťou predpovedať obe triedy.

Pre odstraňovanie nerovností v klasifikácií existujú dva prístupy:

- oversampling - generovanie opakovaných dát z existujúcich pre triedy, ktorých zastúpenie je nedostatočné
- undersampling - orezávanie tried s prebytočným počtom vzoriek

Pre naše účely si vyberáme oversampling. Predspracovanie datasetu je výpočtovo náročná úloha a rozdiel medzi oversamplingom a undersamplingom v našom datasete predstavuje 20-násobok potrebnej dátovej vzorky.

Normalizácia

Interpretácia dát pre neurónovú sieť prechádza normalizáciou. Signál pre aktivačné funkcie je ľahšie interpretovateľný, pokiaľ sa nachádza v normálovom rozložení $< -1, 1 >$ alebo $< 0, 1 >$ Preto každý parameter prechádza normalizáciou podľa svojho rozsahu.

Architektúra siete

Testujeme architektúru doprednej siete s jednou skrytou vrstvou. Skrytá vrstva obsahuje ReLU aktivačné funkcie. Na vstupe sa nachádza 9 neurónov prijímajúcich normalizované hodnoty vo vstupnom vektore podľa obr. 6.1. Na výstupe sa nachádzajú 2 výstupné neuróny na ktoré je aplikovaná softmax aktivácia pre škálovanie súčtu výstupných hodnôt pre jednotlivé triedy na 1. Na výstupe tak máme percentuálnu šancu pre danú triedu.

Implementácia

Projekt je realizovaný v jazyku **Python**, ktorý je ideálny pre účely dataminingu ideálnou voľbou. Poskytuje rozsiahle balíky určené pre prácu vývojárov a dátovo zameraných výskumníkov ako napríklad *numpy*, *pandas* alebo *matplotlib*.
Python bol voľbou aj kvôli frameworku **Tensorflow** od spoločnosti Google. Tensorflow obsahuje rozsiahlu podporu a nástroje pre implementáciu strojového učenia, špeciálne neurónových sietí.
Pre efektivitu práce je v projekte využitý **Jupyter**. Poskytuje notebooky, v ktorých je možné upravovať a spúšťať Python skripty po jednotlivých častiach v bunkách, udržiava stav premenných a poskytuje pracovné rozhranie v prostredí internetového prehliadača. Uľahčuje tak prácu so serverom, na ktorom sú realizované výpočty.

Kapitola 7

Výsledky

Trénovanie

Trénovanie prebiehalo za použitia optimalizačnej stratégie ADAM. Počiatočná rýchlosť učenia je 0.001. Počet neurónov skrytej vrstvy je 128.
Pracujeme s podmnožinou o veľkosti 100 tisíc klikov, z ktorej je vygenerovaných 20 tisíc vstupných vektorov pre tréningovú množinu. Validačná a testovacia množina obsahujú 2,6 tisíc vektorov, tj. 10 % datasetu pre každú. Po vyvážení tried prebieha učenie na vzorke 18 tisíc vstupných vektorov pre triedu nákupných aj nenákupných sessions. Spolu je teda sieť učená na datasete 37 tisíc vzoriek. Vzorky sú pred vstupom zamiešané a rozdelené do úsekov. Po každom úseku je evaluovaná sieť voči validačnej množine.
Tréning je zastavovaný, keď validačná množina prestáva zaznamenávať zlepšovanie oproti predchádzajúcej epoche. Keďže je štatisticky možné aby sa objavilo jedno zhoršenie počas tréningu, sieť je zastavovaná ak na validačnej množine nedojde k zlepšeniu väčšiemu ako 0.25 % 3x po sebe.

Testovanie

Popri samotnej validačnej množine sú vyhodnocované aj metriky tried klasifikácie:

- True Positive
- True Negative
- False Positive
- False Negative

Literatúra

[1] Gediminas Adomavicius and Alexander Tuzhilin. 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, 2005.

[2] Nii O Attoh-Okine. Analysis of learning rate and momentum term in backpropagation neural network algorithm trained to predict pavement performance. *Advances in Engineering Software*, 30(4):291–302, 1999.

[3] David Ben-Shimon, Alexander Tsikinovsky, Michael Friedmann, Bracha Shapira, Lior Rokach, and Johannes Hoerle. Recsys challenge 2015 and the yoochoose dataset. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 357–358. ACM, 2015.

[4] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[5] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

[6] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *Neural computation*, 12(10):2451–2471, 2000.

[7] Felix A Gers, Nicol N Schraudolph, and Jürgen Schmidhuber. Learning

precise timing with lstm recurrent networks. *The Journal of Machine Learning Research*, 3:115–143, 2003.

[8] Alan Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. pages 6645–6649, 2013.

[9] Klaus Greff, Rupesh Kumar Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *arXiv preprint arXiv:1503.04069*, 2015.

[10] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[11] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[12] Yoshua Bengio Ian Goodfellow and Aaron Courville. Deep learning. Book in preparation for MIT Press, 2016.

[13] Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and theëcho state networkäpproach*. GMD-Forschungszentrum Informationstechnik, 2002.

[14] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[15] Vladimír Kvasnička, L'ubica Beňušková, Jiří Pospíchal, Igor Farkaš, Peter Tiňo, and Andrej Král'. *Úvod do teórie neurónovỳch sietí*. 1997.

[16] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[17] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Introduction to recommender systems handbook*. Springer, 2011.

[18] P Sibi, S Allwyn Jones, and P Siddarth. Analysis of different activation

functions using back propagation neural networks. *Journal of Theoretical and Applied Information Technology*, 47(3):1264–1268, 2013.

[19] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. pages 1–9, 2015.

[20] László Tóth. Phone recognition with deep sparse rectifier neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6985–6989. IEEE, 2013.

[21] Artit Wangperawong, Cyrille Brun, Olav Laudy, and Rujikorn Pavasuthipaisit. Churn analysis using deep convolutional neural networks and autoencoders. *arXiv preprint arXiv:1604.05377*, 2016.

[22] Zhenzhou Wu, Bao Hong Tan, Rubing Duan, Yong Liu, and Rick Siow Mong Goh. Neural modeling of buying behaviour for e-commerce from clicking patterns. In *Proceedings of the 2015 International ACM Recommender Systems Challenge*, page 12. ACM, 2015.

[23] Tong Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116. ACM, 2004.

[24] Huaqiang Zhou, Hongxia Gao, and Han Xiao. Research on improving methods of preprocessing in web log mining. In *The 2nd International Conference on Information Science and Engineering*, pages 1472–1474. IEEE, 2010.

Príloha A

Technická dokumentácia

Implementácia

Program je implementovaný v jazyku Python. Je upravený pre prácu s obomi verziami Python (2.X aj 3.X). Hlavný program je implementovaný v Python Jupyter Notebooku.

Modul DataHandler obsauje metódy určené na prácu s dátami. Nachádzajú sa tu metódy na:

- načítanie dát
- oversample a undersample
- časové údaje
- výpisy
- počítanie úspešnosti učenia
- normalizácia

Modul ModelInput obsahuje definície tried pre dátové objekty reprezentujúce dáta využívané v predspracovaní. Obsahuje tiež metódy, ktoré súvisia s predspracovaním dát, agregáciou a zmenou na vstupné vektory.

Triedy:

- Session - Inštancie tejto triedy predstavujú agregované dáta z klikov patriacich pod unikátnu používateľskú session.
- SessionObject - Inštancie sú predspracovaním dát, obsahujú údaje, ktoré sú vkladané na vstup do neurónovej siete po zakódovaní na vstupné vektory.

Príloha B

Používateľská dokumentácia

Pre spustenie programu je nutné mať:

- Python verzia 2.7 alebo 3.5
- Jupyter notebooks
- Zdrojové súbory programu

Program je spustený nasledovnými krokmi:

1. Spustenie Jupyter notebooku - pre spustenie je nutné zadať príkaz 'jupyter-notebook' v konzole. Po štarte Jupyter oznámi na akom porte pracuje. Štandardne sa snaží dostať na port 8888, pokiaľ nie je dostupný, hľadá najbližší voľný port v poradí.
2. Otvorenie notebooku s projektom - Jupyter poskytuje prehliadačové rozhranie pre prácu s notebookmi. Pre otvorenie rozhrania je nutné zadať 'localhost://portnumber' do adresy prehliadača. 'portnumber' reprezentuje číslo prideleného portu, teda štandardne 8888. Následne stačí otvoriť súbor notebooku v prehliadači súborového systému.
3. Spustenie vykonatelného kódu - Pre spustenie jednotlivých buniek stačí kliknúť na bunku a stlačiť Ctrl+Enter. Bunky sú usporiadané v poradí, v akom sa majú spúšťať. Bunky sú číslované podľa poradia spúšťania buniek. Bunka ktorá stále pracuje, má miesto čísla hviezdičku.
   Pre prepnutie datasetu na menší alebo väčší súbor stačí prepísať názov

súboru v prvej bunke. Relatívne cesty sú nastavené tak, aby sa do nich nemuselo zasahovať. Dostupné súbory sú v zložke 'data'.
Kód vypisuje informácie pod bunku, v ktorej aktuálne beží. Mnoho metód má definovaných parameter 'info', ktorého nastavenie na 'True' alebo 'False' ovplyvňuje množstvo výpisu.

Príloha C

Elektronické médium

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat:

- Aplikácia
  - Zdrojové kódy
  - Dataset
- Dokumentácia
  - Diplomová práca spolu s anotáciami v slovenskom a anglickom jazyku
  - LaTeX zdrojové súbory dokumentácie
  - Obrázky
  - Súbory dokumentácie BibTeX
- read.me - popis obsahu média

Príloha D

Plán letného semestra - DP3

- December/Január
  - Otestovanie LSTM verzie
  - Predikcia session items
- 1. a 2. týždeň
  - Spracovanie vlastného datasetu
- 3. - 5. týždeň
  - Migrácia riešenia
- 5. a 6. týždeň
  - Testovanie a ladenie riešenia
- 7. a 8. týždeň
  - Implementácia alternatívneho riešenia
- 9 týždeň
  - Porovnávanie riešení
- 10. - 12. týždeň
  - Dokumentácia