

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
FIIT-182905-73915

Bc. Michal Hucko

Identifikácia zmätenia používateľa vo webovej aplikácii

Diplomová práca

Študijný program: Inteligentné softvérové systémy
Študijný odbor: 9.2.5 Softvérové inžinierstvo (hlavný),
9.2.8 Umelá inteligencia (vedľajší)
Miesto vypracovania: Ústav informatiky, informačných systémov a softvérového
inžinierstva
Vedúci práce: prof. Ing. Mária Bieliková, PhD
apríl 2019

Zadanie diplomovej práce

Meno študenta: **Bc. Michal Hucko**
Študijný program: Inteligentné softvérové systémy
Študijný odbor: Softvérové inžinierstvo – hlavný študijný odbor
Umelá inteligencia – vedľajší študijný odbor

Názov práce: **Identifikácia zmätenia používateľa vo webovej aplikácii**

Samostatnou výskumnou a vývojovou činnosťou v rámci predmetov Diplomový projekt I, II, III vypracujte diplomovú prácu na tému, vyjadrenú vyššie uvedeným názvom tak, aby ste dosiahli tieto ciele:

Všeobecný cieľ:

Vypracovaním diplomovej práce preukážte, ako ste si osvojili metódy a postupy riešenia relatívne rozsiahlych projektov, schopnosť samostatne a tvorivo riešiť zložité úlohy aj výskumného charakteru v súlade so súčasnými metódami a postupmi študovaného odboru využívanými v príslušnej oblasti a schopnosť samostatne, tvorivo a kriticky pristupovať k analýze možných riešení a k tvorbe modelov.

Špecifický cieľ:

Vytvorte riešenie zodpovedajúce návrhu textu zadania, ktorý je prílohou tohto zadania. Návrh bližšie opisuje tému vyjadrenú názvom. Tento opis je záväzný, má však rámcový charakter, aby vznikol dostatočný priestor pre Vašu tvorivosť.

Riadte sa pokynmi Vášho vedúceho.

Pokiaľ v priebehu riešenia, opierajúc sa o hlbšie poznanie súčasného stavu v príslušnej oblasti, alebo o priebežné výsledky Vášho riešenia, alebo o iné závažné skutočnosti, dospejete spoločne s Vaším vedúcim k presvedčeniu, že niečo v texte zadania a/alebo v názve by sa malo zmeniť, navrhnete zmenu. Zmena je spravidla možná len pri dosiahnutí kontrolného bodu.

Miesto vypracovania: Ústav informatiky, informačných systémov a softvérového inžinierstva, FIIT STU v Bratislave

Vedúci práce: **prof. Ing. Mária Bieliková, PhD.**

Termíny odovzdania:

Podľa harmonogramu štúdia platného pre semester, v ktorom máte príslušný predmet (Diplomový projekt I, II, III) absolvovať podľa Vášho študijného plánu

Predmety odovzdania:

V každom predmete dokument podľa pokynov na www.fiit.stuba.sk v časti:
home > Informácie o > štúdiu > harmonogram štúdia > diplomový projekt.

V Bratislave dňa 12. 2. 2018

**SLOVENSKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE**

Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4



prof. Ing. Pavol Návrat, PhD.
riaditeľ Ústavu informatiky, informačných systémov
a softvérového inžinierstva

Návrh zadania diplomovej práce

Finálna verzia do diplomovej práce ¹

Študent:

Meno, priezvisko, tituly: Michal Hucko, Bc.
Študijný program: Inteligentné softvérové systémy
Kontakt: hucko.miso@gmail.com

Výskumník:

Meno, priezvisko, tituly: Mária Bieliková, prof. Ing. PhD.

Projekt:

Názov: Identifikácia zmätenia používateľa vo webovej aplikácii
Názov v angličtine: Identification of user confusion in a web application
Miesto vypracovania: Ústav informatiky, informačných systémov a softvérového inžinierstva, FIIT STU, Bratislava
Oblasť problematiky: strojové učenie, modelovanie používateľa, vzory správania, interakcia človek počítač

Text návrhu zadania²

Webové aplikácie sú dnes už bežnou súčasťou našich životov. Pri ich používaní častokrát narazíme na situáciu, kedy máme problém sa ďalej pohnúť. Táto situácia je spravidla sprevádzaná stavom zmätenia používateľa. Tento stav je veľmi špecifický pre každého z nás. Jeho identifikáciou a predvídaním vieme uľahčiť prácu s webovými aplikáciami, napríklad prezentovaním nápovedy v správnom okamihu, čo nám umožní zefektívniť činnosť.

Analyzujte oblasť modelovania správania používateľa z implicitnej spätnej väzby pri interakcii s webovými aplikáciami. Sústreďte sa pri tom na signály, ktoré môžu indikovať rôzne stavy používateľa, špeciálne jeho zmätenie. Preštudujte metódy strojového učenia vhodné pre prácu s dátami z implicitnej spätnej väzby používateľa.

Navrhňte metódu odhaľovania zmätenia používateľa webovej aplikácii, ktorá umožní v reálnom čase tento stav odhaliť. Pokúste sa tento stav aj predikovať. Navrhňte pre riešenie využitie vhodných metód strojového učenia a ich prípadnú kombináciu. Navrhnutú metódu overte na reálnych dátach z používania vybratej webovej aplikácie. Vychádzajte z dát implicitnej spätnej väzby vybranej aplikácie. Kľúčovými v tomto prípade sú pohyby myšou, skrolovanie, kliky. Experimentujte a zvážte pri tom využitie dát zo sledovania pohľadu používateľa. Navrhnuté riešenie diskutujte aj v kontexte iných dostupných riešení pre daný problém.

¹ Vytlačiť obojstranne na jeden list papiera


² 150-200 slov (1200-1700 znakov), ktoré opisujú výskumný problém v kontexte súčasného stavu vrátane motivácie a smerov riešenia

Literatúra³


- THOMAS, P.. Using interaction data to explain difficulty navigating online. ACM Transactions on the Web (TWEB), vol. 8, no. 4, 2014, Article no. 24, 41 pages.
- TIKHONOV, A., PROKHORENKOVA, L., CEHLNOKOV, A., BOGATYY, I., GUSEV, G. . What can be Found on the Web and How: A Characterization of Web Browsing Patterns. In: Proceedings of the ACM Web Science Conference. ACM Press, 2015, Article no,14, 10 pages.

Vyššie je uvedený návrh diplomového projektu, ktorý vypracoval(a) Bc. Michal Hucko, konzultoval(a) a osvojil(a) si ho prof. Ing. Mária Bieliková, PhD. a súhlasí, že bude takýto projekt viesť v prípade, že bude pridelený tomuto študentovi.

V Bratislave dňa 10.1.2018



Podpis študenta



Podpis výskumníka

Vyjadrenie garanta predmetov Diplomový projekt I, II, III

Návrh zadania schválený: áno / nie⁴

Dňa: 12.2.2018



Podpis garanta predmetov

³ 2 vedecké zdroje, každý v samostatnej rubrike a s údajmi zodpovedajúcimi bibliografickým odkazom podľa normy STN ISO 690, ktoré sa viažu k téme zadania a preukazujú výskumnú povahu problému a jeho aktuálnosť (uveďte všetky potrebné údaje na identifikáciu zdroja, pričom uprednostnite vedecké príspevky v časopisoch a medzinárodných konferenciách)

⁴ Nehodiace sa prečiarknite

Čestné prehlásenie

Čestne vyhlasujem, že som túto diplomovú prácu vypracoval samostatne, iba s použitím uvedenej literatúry a podľa usmernení vedúceho diplomovej práce.

V Bratislave, apríl 2019

.....

Bc. Michal Hucko

Podakovanie

V prvom rade sa chcem poďakovať pani profesorku Bielikovej, ktorá svojimi nápadmi, časom a najmä vynikajúcim prístupom významne dopomohla k vypracovaniu tejto diplomovej práce. Taktiež ďakujem členom skupiny Pewe, ktorí si pravidelne našli čas a konzultovali so mnou môj postup a tak svojimi nápadmi posunuli kvalitu mojej práce vždy o kus ďalej. Ďakujem všetkým vo firme Seges, kde som mohol pracovať a vyvíjať svoju diplomovú prácu v produkčnom prostredí. Chcem tiež poďakovať spolužiakovi Matejovi Válkymu, s ktorým som pracoval na používateľskej štúdii a na všetkých implementačných ťažkostiach, ktoré nastali. Na záver ďakujem svojim priateľom, rodine a hlavne mojej priateľke Katke za ich podporu počas vypracovania tejto práce.

Bc. Michal Hucko

Anotácia

Slovenská technická univerzita v Bratislave

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Študijný program: Informačné systémy

Autor: Bc. Michal Hucko

Diplomová práca: Identifikácia zmätenia používateľa vo webovej aplikácii

Vedúci diplomovej práce: prof. Ing. Mária Bieliková, PhD

apríl 2019

Webové aplikácie sú denne navštevované každým z nás. Zastúpenie ich používateľov je pestré, od expertov až po začínajúcich. Mnohokrát sa stane, že pri napĺňaní svojej potreby je používateľ zmätený. Príkladom môže byť osoba, ktorá aplikáciu navštíví po prvýkrát. Zmätenie sa však môže týkať aj ľudí, ktorí aplikáciu poznajú. Napríklad v prípade aktualizácie alebo len skúšaní inej ako bežnej funkcionality. V tejto práci sa venujeme predikcii zmätenia používateľa webovej aplikácie s využitím dát z implicitnej spätnej väzby. Konkrétne nami navrhnutá metóda pracuje s interakčnými dátami používateľa z myši a okulografu. V práci sa tiež zameriavame na odhalenie momentu, kedy je vhodné používateľovi poskytnúť nápovedu, ktorá by mu pomohla pri naplnení svojho cieľa v aplikácii. Na natrénovanie modelu strojového učenia sme zostavili používateľskú štúdiu. Navrhnutú metódu sme na získaných dátach aj overili. Súčasťou práce je porovnanie modelu s využitím myši s modelom z dát z okulografu.

Annotation

Slovak University of Technology Bratislava

FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Degree course: Information Systems

Author: Bc. Michal Hucko

Master Thesis: Identification of user confusion in a web application

Supervisor: prof. Ing. Mária Bieliková, PhD

2019, Apríl

Web applications are daily visited by everyone of us. There is a variety of users from the novice ones to the experts. While fulfilling our needs we might be confused, many times. For example, it might be in the situation where we face the application for the very first time. But confusion might also occur dealing with application we are experienced with, while trying some new feature or after the update. These situations might be problematic. In this work we deal with the automatic identification of the user confusion in a web application using data from the user implicit feedback. Proposed method works with the interaction data from the mouse and the eyetracker. We focus on identification of the right moment to display a guide to the user. This guide may consist of hints which will explain how to work with the certain web application. We conduct a user study to train our machine learning model. We evaluate our method on data gathered at the study. In this work we also provide comparison between our approach and the model using only the eyetracking data.

Obsah

1	Úvod	1
2	Prístupy k detekcii stavov používateľa v aplikáciách na webe	3
2.1	Explicitná spätná väzba	3
2.2	Implicitná spätná väzba	4
2.3	Detekcia zmätenia používateľa	6
2.4	Zhrnutie a diskusia	8
3	Metriky správania používateľa na webe	11
3.1	Metriky myši	11
3.1.1	Metriky myši založené na pohybe	12
3.1.2	Metriky myši založené na klikoch a váhavosť	13
3.1.3	Metriky myši založené na pauzách	14
3.2	Metriky klávesnice	15
3.3	Histogramová reprezentácia črt	15
3.4	Sledovanie pohľadu	15
3.4.1	Okulograf	16
3.4.2	Zrenička	16
3.4.3	Oblasti záujmu	17
3.4.4	Metriky sledovania očí	17
3.5	Metriky HTTP serveru	17
3.6	Zhrnutie a diskusia	19
4	Predikčný model	21
4.1	Klasifikácia v reálnom čase	21
4.2	Logistická regresia	21
4.3	Náhodný les	22
4.4	Vyhodnotenie modelov	24
4.4.1	Presnosť a úplnosť	25
4.4.2	Plocha pod krivkou charakteristiky prijímajúceho operátora	25
4.4.3	Akeikeho informačné kritérium	26
4.5	Zhrnutie a diskusia	27
5	Návrh metódy predikcie zmätenia	29
5.1	Predspracovanie dát logeru myši a extrakcia črt	29
5.2	Predspracovanie dát okulografu	31
5.3	Trénovanie modelov a vyhodnotenie	32
6	Používateľská štúdia a overenie	33
6.1	Pilotná štúdia	33
6.1.1	Vyhodnotenie implicitnej spätnej väzby	34

6.1.2	Vyhodnotenie infraštruktúry logovania	34
6.2	Hlavná štúdia	35
6.2.1	Získavanie spätnej väzby	36
6.2.2	Mapovanie identít účastníka	37
6.3	Exploratívna analýza surových dát	38
6.3.1	Získané dáta	38
6.3.2	Tlačidlo zmätenia	38
6.4	Exploratívna analýza metrík	41
6.4.1	Metriky myši	42
6.4.2	Metriky okulografu	43
6.5	Výsledky predikcie	44
6.5.1	Výsledky nad celou úlohou	45
6.5.2	Výsledky nad časovými oknami	46
7	Zhodnotenie	49
	Literatúra	51
	Príloha A - Protokol pilotnej používateľskej štúdie	A-1
	Príloha B - Protokol používateľskej štúdie	B-1
	Príloha C - Loger myši a predikcia zmätenia v aplikácii YesElf	C-1
	Loger	C-2
	Predikcia v reálnom čase v aplikácii YesElf	C-3
	Príloha D - Opis digitálnej časti práce	D-1
	Príloha E - IITSRC článok a plagát	E-1
	Príloha F - UMAP článok (ACM Press, prijatý do sekcie Late Breaking Results)	F-1
	Príloha G - Návrh článku na konferenciu	G-1
	Príloha H - Zhodnotenie plánov práce	H-1
	Príloha I - Technická dokumentácia	I-1

1 Úvod

Webové aplikácie sú neodmysliteľnou súčasťou každého používateľa počítačov a iných výpočtových zariadení. Mnohé z nich denne navštevujú milióny ľudí. Môžeme cez ne nakupovať, pracovať, vykonávať bankové transakcie, študovať, registrovať sa alebo inak naplňať svoje potreby. Viaceré webové aplikácie poskytujú mnoho z týchto funkcií naraz. Používateľ po príchode na sídlo aplikácie mnohokrát netuší, čo všetko mu je poskytnuté. Jeho cieľom v danom čase je naplniť svoje potreby a to čo najrýchlejšie. Vynaloženie príliš veľkého úsilia zo strany používateľa môže viesť k strate záujmu o aplikáciu.

Jedným z hlavných cieľov webových aplikácií je vo všeobecnosti naplnenie potrieb návštevníka. Na to, aby sa tento cieľ podarilo uskutočniť musí byť vytvorené prostredie, ktoré je zrozumiteľné. Problémovou skupinou v tomto smere sú používatelia, ktorí navštevujú sídlo po prvýkrát. Takýto používatelia mnohokrát nevedia čo všetko im sídlo ponúka, a tak sa často nedokážu pohnúť ďalej alebo dokonca začať. Avšak spomínaný stav môže nastať aj v prípade skúsenejších používateľov. Príkladom môže byť objavovanie novej funkcionality alebo len interakcia s aplikáciou. Tu opäť zohráva dôležitú rolu čas potrebný na získanie požadovanej informácie. Ak je prekročená znesiteľná hranica zákazník mnohokrát aplikáciu opustí, čo môže viesť k strate z pohľadu vlastníka (používateľ si napríklad zakúpi inú službu). V práci [42] sa tento stav nazýva dezorientácia (angl. *disorientation*) alebo zmätenie (angl. *confusion*).

Zmätenie je spojené so zážitkom používateľa a je zjavne dôležité pracovať na jeho minimalizovaní. Z pohľadu používateľského zážitku (angl. *user experience*) existuje mnoho ciest ako dosiahnuť tento cieľ. Sú to napríklad dotazníky, rozhovory, ankety, kvalitatívne experimenty a iné metódy založené na individuálnom prístupe a zbieraní explicitnej spätnej väzby. Počas nich je správanie sledované pri práci s prostredím s cieľom identifikovať nedostatky aplikácie. Tieto metódy sa však veľmi ťažko škálujú. Ich analýzou potom vieme odstrániť problémové časti aplikácie. Treba si však uvedomiť, že častokrát nežiadúci stav (dezorientácia, zmätenie) ani nemusí byť dôsledkom nejakého problému v návrhu, ale komplexnosti aplikácie. Stav môže byť spôsobený slabšou webovou gramotnosťou používateľa alebo fyziologickými faktormi akým je napríklad únava, či len nešťastnou náhodou.

V našej práci sa venujeme identifikácii zmätenia používateľa webovej aplikácie. Prezentujeme metódu, ktorá je založená na automatizovanej predikcii tohto stavu. Metóda vychádza zo správania, ktoré je možné zaznamenať pri interakcii používateľa s aplikáciou. Pracujeme s udalosťami získanými z používania myši a okulografu. Jedná sa teda o implicitnú spätnú väzbu. Dáta z myši vieme technicky jednoducho získať, ak predpokladáme prístup ku zdrojovému kódu samotnej aplikácie. Pri navrhovaní metódy sme vychádzali z prác, ktoré sa tejto alebo podobnej problematike venujú [29, 28, 42, 36, 35].

Nami navrhnutá metóda môže byť použitá pri online pomocníkoch (angl. *guides*). Ich úlohou je vysvetliť používateľovi kroky s cieľom naplnenia jeho potrieb. Problémom pomocníkov je neznalosť stavu používateľa. Ich pravidlá sa dajú generalizovať na odporúčania tých istých krokov pre všetkých používateľov, prípadne nadviazať na spúšťač (časový, udalostný). Kvôli špecifickým

potrebám zákazníkov je však tento prístup nepostačujúci. Príkladom takéhoto pomocníka môže byť aplikácia *YeseEf*¹ od firmy *Brainware*, ktorá svojím jednoduchým rozhraním umožňuje používateľom navrhnuť týchto pomocníkov a jednoducho ich integrovať s už existujúcou webovou aplikáciou. V rámci tejto práce navrhujeme metódu predikcie zmätenia v reálnom čase, ktorú je možné priamo integrovať s nástrojom *YesElf*.

V kapitole 2 našej práce sumarizujeme prístupy v získavaní explicitnej a implicitnej spätnej väzby používateľa webovej aplikácie. V tejto kapitole tiež predstavujeme existujúce prístupy v identifikácii zmätenia používateľa. Ďalej sa v tejto kapitole zaoberáme sprievodcami ako nástrojmi, ktoré riešia situáciu zmätenia poskytnutím rady. V kapitole 3 poskytujeme prehľad metrík k zaznamenávaniu správania používateľa pomocou klávesnice, myši a okulografu. V kapitole 4 prezentujeme prístupy strojového učenia na riešenie klasifikačných problémov. Sumarizujeme v nej aj metriky vyhodnotenia takýchto modelov. Kapitola 5 predstavuje nami navrhnutú metódu predikcie zmätenia. V kapitole 6 predstavujeme používateľskú štúdiu, ktorú sme uskutočnili s cieľom overenia výsledkov. V tejto kapitole tiež dokumentujeme výsledky.

¹www.yeself.com/

2 Prístupy k detekcii stavov používateľa v aplikáciách na webe

Úspech tej ktorej webovej aplikácie závisí aj od toho ako sa podarí naplniť potreby používateľa. Na to, aby sme mohli identifikovať problematické miesta je potrebné získať od používateľa spätnú väzbu. V tejto kapitole sa venujeme získavaniu spätnej väzby s cieľom identifikovať stav zmätenia. Sumarizujeme niekoľko postupov dokumentovaných v prácach autorov, ktorí sa venujú práve problému s používateľským zážitkom a zmätením.

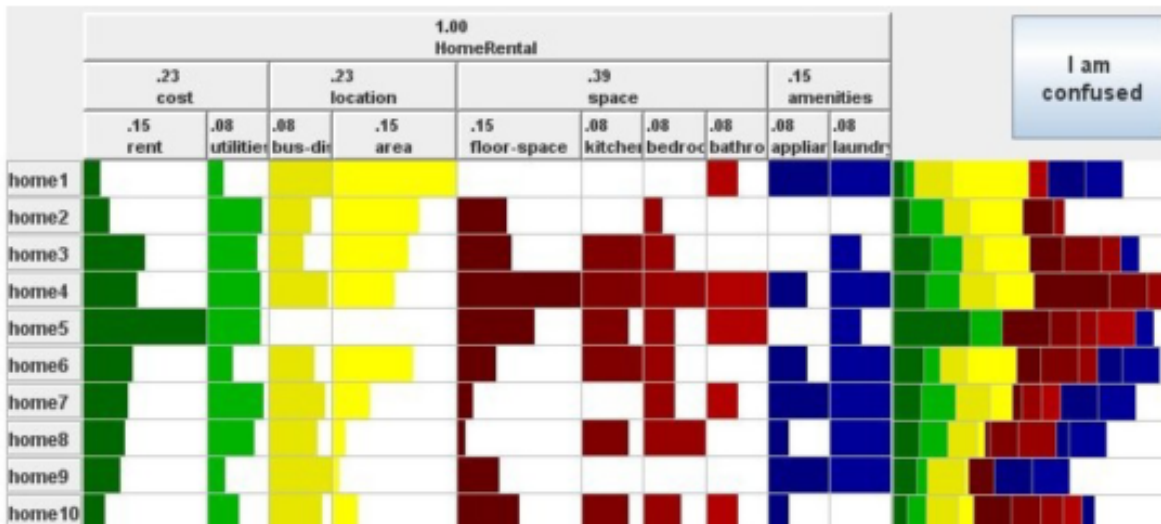
Detekcia stavov používateľa sa realizuje spracovaním jeho spätnej väzby. Tu rozlišujeme explicitnú spätnú väzbu (podkapitola 2.1) a implicitnú (podkapitola 2.2). Z možných stavov používateľa sa sústreďíme na stav zmätenia (podkapitola 2.3).

2.1 Explicitná spätná väzba

Získanie spätnej väzby od používateľa je predpokladom k odhaleniu chýb, ktoré v aplikácii vznikajú. Jedným z jej druhov je explicitná spätná väzba. V tomto prípade ide o priame vyjadrenie názoru na danú problematiku. Získanie veľkého množstva tejto spätnej väzby od jednotlivca môže byť veľmi náročné kvôli nadmernej kognitívnej záťaži, ktorá z tejto úlohy vyplýva. Taktiež, v prípade veľkého množstva používateľov môže byť jej získanie veľmi nákladné kvôli problematickej škálovateľnosti.

Výstupom explicitnej spätnej väzby sú zväčša odpovede na dotazníky, ktoré sú tvorené niekoľkými otázkami. Autori otázok sa zameriavajú primárne na rýchlosť pri vypĺňaní otázok. Príkladmi druhov otázok na získavanie explicitnej spätnej väzby sú:

- *Likertova škála.* [4] Ide o škálu na vyjadrenie intenzity názoru. Samotná škála vznikne spočítaním odpovedí pre jednotlivé prvky škály.
- *Hviezdičkový systém.* V tomto prípade sa jedná o iný druh vizualizácie škály. Hviezdičky majú široké využitie pri systémoch na odporúčanie produktov [9]. Používateľ vyjadří svoju spokojnosť priradením určitého množstva hviezdíčiek.
- *Binárne hodnotenie.* V tomto prípade ide o vyjadrenie súhlasu alebo nesúhlasu na otázku. Obrázok 1 ukazuje použitie tlačidla v pravom hornom rohu v práci [28] na vyjadrenie zmätenia (túto prácu analyzujeme v nasledujúcich kapitolách).



Obr. 1: Príklad binárneho hodnotenia na vyjadrenie zmätenia [28].

Jednou z hlavných výhod explicitnej spätnej väzby v porovnaní s implicitnou spätnou väzbou je presnosť. Používateľ totiž jasne definuje svoj názor. V práci [21] sa autori venujú jej charakteristikám v prípade odporúčacieho systému pre rádiové služby. Pracujú so systémom, ktorý používateľom dovolí označiť skladbu ako obľúbenú (angl. *loved*) alebo neželanú (angl. *banned*). V experimente zozbierali dáta od viac ako 10 000 používateľov so záznamami o viac ako milióno skladieb. Výsledky ich experimentu ukazujú, že použitie explicitnej spätnej väzby klesá s časom využitia aplikácie. Autori sa domnievajú, že tento jav môže byť zapríčinený tým, že používatelia nevidia zmysel v jej odovzdávaní.

V práci [43] sa autori venujú porovnaniu explicitnej a implicitnej spätnej väzby v prípade vyhľadávania na webe. Ich cieľom bolo zistiť či implicitná spätná väzba môže nahrádzať explicitnú. Zostrojili experiment so 16 účastníkmi, kde úlohou bolo ohodnotiť výsledky vyhľadávania pre daný dopyt. Systém vždy vrátil prvých 30 dokumentov vyhľadávачa podľa relevantnosti. V prípade explicitnej spätnej väzby, používateľ len zvolil či je daný výsledok relevantný. Pri implicitnej variante sledovali správanie používateľa pri interakcii s výsledkami. Za relevantný sa v tomto prípade považoval dopyt, pri ktorom si účastník vyžiadal detailnejšie zobrazenie. Výsledky ich experimentu hovoria o tom, že dáta explicitnej a implicitnej spätnej väzby neboli rozdielne. Hypotézu overili štatistickým testom.

2.2 Implicitná spätná väzba

Pri implicitnej spätnej väzbe sa dáta získavajú zo správania používateľa bez toho aby používateľ priamo poskytoval nejakú informáciu. Získanie interpretácie implicitnej spätnej väzby, t.j. jej vyhodnotenie však v mnohých prípadoch nie je triviálna záležitosť. Na jej úspešné vyhodnotenie potrebujeme mnoho dát o správaní používateľa. Dôležité je označiť a teda vyhodnotiť správa-

nie používateľa, ktoré tieto dáta opisujú. Mnohokrát sa na označenie správania používa práve explicitná spätná väzba [42, 29].

V práci [22] sa autori venujú vyhodnocovaniu dát z prechádzania webom a reformulácie dopytov vo vyhľadávaní na webe. Okrem toho sa venujú generovaniu trénovacej sady na učenie s použitím implicitnej spätnej väzby. Na základe používateľskej štúdie, kde účastníkom prezentovali zoznam položiek pre zadaný dopyt, zistili ako jednotlivci interagujú s výsledkami vyhľadávača. Na zaznamenanie správania účastníka používali okulograf. Výsledky potom prispeli k vytvoreniu stratégie získavania spätnej väzby z klikov. Podľa nej zhotovili používateľskú štúdiu, kde na overenie použili explicitnú spätnú väzbu z dotazníka na doménovú znalosť vyhľadávača. V prvej časti položili účastníkom 10 otázok. 5 bolo navigačných a 5 bolo informatívnych. Štúdia prebehla v dvoch fázach. V prvej ju absolvovalo 34 študentov (vek 18-23), kde všetci boli motivovaní extra bodmi. Druhá časť štúdie prebehla s tými istými otázkami, kde študenti boli vybraní tým istým spôsobom, len prezentácia výsledkov bola v troch rôznych poradiach:

- *Normálne poradie.* Výsledky vyhľadávača neboli upravené.
- *Zamenené poradie.* Prvé dva výsledky vyhľadávača podľa priority boli vymenené.
- *Opačné poradie.* 10 výsledkov vyhľadávača bolo v opačnom poradí.

Hlavné zistenie práce poukazuje na to, že spätná väzba z klikov je vychýlená najmenej v dvoch smeroch:

- *Vychýlenie na základe dôveryhodnosti* (angl. *trust bias*). Ľudia klikajú viac na odkazy, ktoré vyhľadávač umiestni na vyššiu pozíciu aj napriek tomu, že ich abstrakt nie je relevantný oproti ostatným prezretým odkazom.
- *Vychýlenie na základe kvality kontextu* (angl. *quality-of-context bias*). Autori ukazujú, že klikanie na odkazy nie je ovplyvnené len relevantnosťou abstraktu odkazu, ale aj odkazmi, ktoré ho obklopujú a ich abstraktami.

V článku [32] sa autori venujú spracovaniu logov zo serverov s cieľom identifikácie vzorov, ktoré sa v nich často vyskytujú. Na ich následnú analýzu používajú metódy strojového učenia. Nástroj, s ktorým pracujú sa nazýva *WebTool*. Cieľom tejto práce je na základe týchto vzorov správania reštruktúrovať stránku za účelom zväčšenia efektivity používateľov pri vyhľadávaní informácií. Ukazujú postup, v ktorom využívajú profily používateľov. Na automatické odhaľovanie profilov používajú asociačné pravidlá (angl. *association rules*) a sekvenčné vzory (angl. *sequential patterns*). Príklad asociačného pravidla: '50% návštevníkov, ktorí prišli na linky A a B navštívili aj C'. Príkladom sekvenčného vzoru môže byť: '60% klientov, ktorí navštívili A a B v jednej transakcii, taktiež navštívili C za nasledujúci mesiac'. Ich metóda ukazuje, že použitím analýzy sekvencií na zozbieraných logoch vedia odhaliť vzory v návštevách používateľa. Autori taktiež ukazujú využitie tejto informácie pri štruktúrovaní informácií na stránke.

2.3 Detekcia zmätenia používateľa

V tejto práci sa venujeme zmäteniu používateľa. Pred samotnou analýzou existujúcich prístupov detekcie zmätenia definujeme pojem zmätenie. Výkladové slovníky charakterizujú slovo zmätenie ako neistotu v tom, čo sa má diať alebo v tom, čo je požadované. Slovník *Webster*² charakterizuje zmätenie ako situáciu, v ktorej sú ľudia neistí v tom, čo robiť alebo v tom, ako situáciu pochopiť. Dôsledkom zmätenia na webe je neschopnosť používateľa plynule pokračovať v naplňaní svojho cieľa. Práve toto vedie k znechuteniu používateľa.

V práci [42] označujú tento stav slovom obtiažny (angl. *struggling*). Vo svojej práci prezentujú nástroj *LATTE* na automatické odhalovanie obtiažnych sedení používateľa. Cieľom nástroja je zlepšenie používateľského zážitku aplikácie, do ktorej je nasadený skrz odporúčania na vylepšenie. Na vytváranie modelu používajú celé používateľské sedenia, v ktorých identifikujú signály založené na správaní používateľov indikujúce obtiažne sedenia. Výstupom ich nástroja sú rady pre vlastníkov aplikácií, v ktorých identifikujú problémové miesta, prípadne mätúce linky aj iné nedostatky. Pri hľadaní signálov autori vychádzali z kontrolovanej štúdie, kde zaznamenávali výstupy myši, klávesnice a adresy stránok, ktoré účastníci navštívili. Dokopy mali 22 účastníkov. Počas štúdie sa sústredili primárne na jednoduché metriky, ktoré dokážeme na stránke zaznamenať bez použitia rozšírenia. Práve tieto metriky sa v ich výsledkoch javia ako najlepšie ukazovatele obtiažnych sedení. Menovite ide o celkový čas sedenia, celkový počet stránok sedenia a čas na stránku (všetkých presnosť lepšia ako 0,75 pri predikovaní obtiažneho sedenia). Ich kombináciou pomocou logistickej regresie dosiahli presnosť až 0,86 pri predikcii. Okrem toho skúmali aj predikciu s využitím metrick, ktoré vieme zaznamenať rozšírením aplikácie pomocou jazyka *javascript*. Tu sa najlepšie ukázala metrika počtu skrolov (presnosť až 0,84) a metrika pohybu myši (presnosť 0,74). Vo výsledkoch autori ukazujú, že je možné pre každú metriku vypočítať hraničnú hodnotu, ktorej presiahnutie by indikovalo obtiažne sedenie. Príkladom je, že pri presiahnutí 12 stránok na úlohu sa sedenie klasifikuje ako zmätené.

V práci [28] sa autori zameriavajú práve na identifikáciu zmätenia používateľa s využitím okulografu na stránke s vizualizáciami grafov. V ich prípade sa jednalo o grafy domov a ich atribútov. Pre každý atribút bolo zobrazené ako veľmi ovplyvňuje cenu domu. Autori zostavili štúdiu, ktorá pozostávala z úloh nad touto stránkou. Podarilo sa zozbierať záznamy zo 136 sedení, kde účastníci vypracovali sadu úloh. Jednalo sa o úlohy vyhľadávania vo vizualizácii (nájdite najlacnejší dom alebo nájdite najlepší dom podľa pomeru rozloha a cena). Dokopy každý účastník prešiel 40 úloh. Počas celého sedenia bol pohľad účastníka zaznamenávaný okulografom *Tobii T120*, v miestnosti bez okien (kvôli rovnakej veľkosti zreničky). Na označenie mätúcich úloh nechali používateľa stlačiť tlačidlo v prípade, že sa cítil zmätený, pričom pri jeho návrhu vychádzali z práce [7]. Pred experimentom inštruovali účastníkov o použití tohto tlačidla. Práca dokumentuje presnosť modelu 0,61.

V práci [36] autori predikujú zmätenie s využitím dát myši v počítačovej hre. Hra bola inšpirovaná adventným kalendárom a pozostávala z hracej plochy s 24 očíslovanými dlaždicami rôznej veľkosti a rôzneho umiestnenia. Cieľom hry bolo postupne odklikať dlaždice v číselnom poradí.

²www.merriam-webster.com/

Autori si proces riešenia hry prešli so štyridsiatimi štyrmi vybranými účastníkmi. Prechod od jednej dlaždice k druhej označili ako jeden pohyb. Po absolvovaní hry účastníci priradili každému z 24 pohybov hodnotu likertovej škály (od 1 do 7), ktorá reprezentovala stav zmätenia (7 pre maximálne zmätenie). Spolu zozbierali 1056 pozorovaní. Už pri úvodných pozorovaniach našli vysokú mieru korelácie (0,86 pearsonov korelačný koeficient) medzi štandardizovanými hodnotami času pohybu a priradenej hodnoty zmätenia. Keďže účastníci spätne neboli schopní označiť začiatok a koniec zmätenia, na výber časového úseku vzorky pre zmätenie použili nasledujúce pravidlá: pre nezmätené pozorovanie zobrali druhú polovicu pohybu a pre zmätené pozorovanie prvú polovicu (vychádzajúc z faktu, že pohyb vždy skončil v cieľi). Následne dáta myši použili na výpočet 33 črt založených na zmenách smerov pri pohybe. Na predikciu natrénovali 4 klasifikátory, kde najlepšiu metriku F1 skóre 0.946 dosiahol algoritmus podporných vektorov strojov (angl. *support vector machines algorithm*).

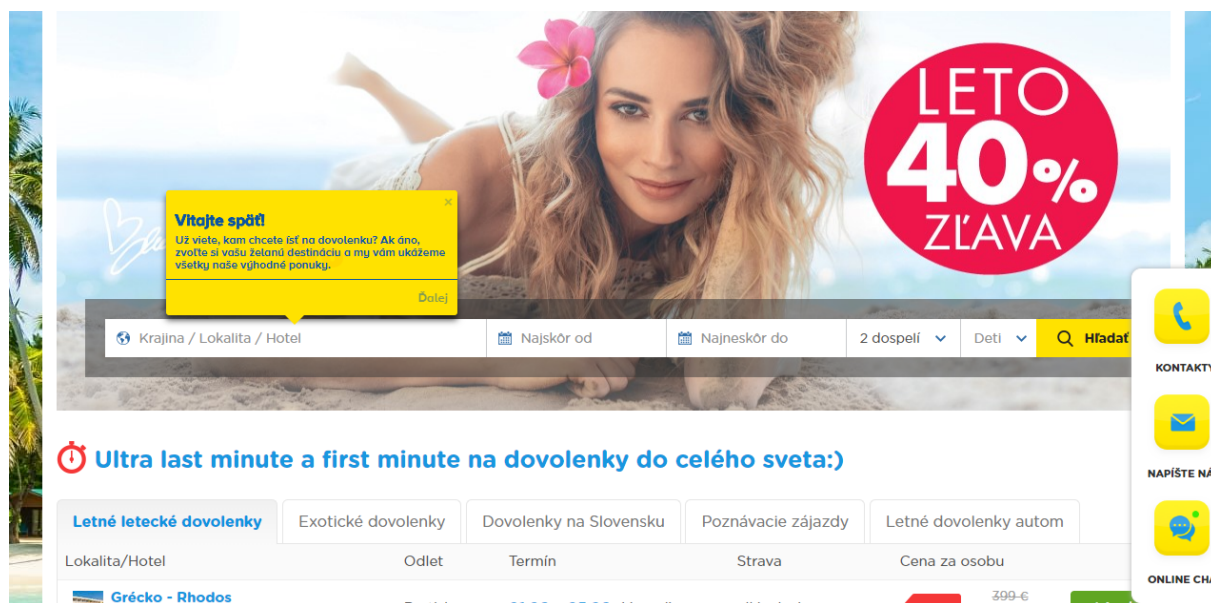
Tí istí autori analyzujú využitie predchádzajúceho prístupu založenom na zmenách smerov v správaní myši pri webovej aplikácii v práci [35] s využitím tej istej hry. V tomto článku využívajú oveľa väčšiu vzorku na overenie svojej hypotézy (916 hier od 282 používateľov dokopy 21984 záznamov). Na zaznamenanie skutočnej hodnoty zmätenia aplikujú protokol hlasného rozprávania (angl. *think aloud protocol*). V rámci protokolu nahrávajú hlas účastníka, ktorý opisuje postup riešenia úlohy a emócie s tým spojené. Z týchto nahrávok potom označovali úseky zmätenia. V tomto prípade sa im podarilo dosiahnuť najvyššiu hodnotu metriky F1 skóre 0,938 pri použití algoritmu náhodných lesov (angl. *Random forest algorithm*). V závere práce autori poukazujú na možnosť využitia tejto metódy na predikciu zmätenia pri prechádzaní jednotlivými elementami stránky, avšak upozorňujú na nefunkčnosť pri úlohách s čítaním rozsiahlejších textov. Tvrdia, že v takom prípade niektorí ľudia zvyknú nasledovať text kurzorom, čo môže vytvárať vzory rozpoznávané ako zmätenie. Ďalšou z limitácií ich riešenia bola implementácia logeru udalostí, ktorý zaznamenal pohyb len pri prekonaní vzdialenosti myšou viac ako 10 pixelov. Z toho vypláva, že nie všetky pohyby boli zaznamenané.

Identifikácia stavu zmätenia používateľa nám môže pomôcť odhaliť problém, ktorý má používateľ a tak buď vylepšiť webovú aplikáciu analýzou tohto problému, alebo vylepšiť používateľský zážitok v konkrétnom kontexte. Tento kontext však nie vždy vieme vopred identifikovať. V prípade práce [42] autori vytvorili nástroj, ktorý označí miesta, ktoré tento problém spôsobujú. Vlastník stránky potom môže tieto miesta vhodne opraviť. V spomínanej práci [32] zase autori poskytnú prehľad o ceste používateľa v stránke, čo tiež môže prispieť k reštrukturalizácii obsahu webu.

Iné riešenie môže byť poskytnutie nápovedy s inštrukciami pre daný problém. To by však znamenalo identifikovanie samotného problému, čo je mimo rozsah tejto práce. Tieto nápovedy však môžu byť všeobecného charakteru týkajúce sa domény stránky. Tomuto postupu sa venujú sprievodcovia (angl. *guides*). Uvedením správnej nápovedy v čase zmätenia používateľa by sme vedeli zabrániť negatívnej skúsenosti. Príkladom softvéru na vytváranie nápovedí je aplikácia *Yeself*³. Aplikácia umožňuje vloženie nápovedy do stránky pomocou jazyka *javascript*. Obrá-

³www.yeself.com

zok 2 poskytuje ukážku využitia produktu na stránke *Firotour*⁴ (stránka s dovolenkami). Bublina s obsahom, ktorý radí ako využiť vyhľadávacie pole na stránku je navrhnutá priamo doménovým expertom.



Obr. 2: Ukážka použitia nápovedy na webovej stránke.

2.4 Zhrnutie a diskusia

V tejto kapitole sme poskytli prehľad možností získavania explicitnej a implicitnej spätnej väzby používateľa s cieľom odhalenia zmätenia vo webovej aplikácii. Explicitná spätná väzba sa javí presnejšia. Ukázali sme však problém s jej získavaním a škálovaním. Naopak pri implicitnej spätnej väzbe nemusíme dosiahnuť očakávanú prenosnosť výsledkov. Taktiež pri nej čelíme problémom spojenými s jej interpretáciou. Ďalším z problémov pri explicitnej spätnej väzbe je jej získavanie. Ak je realizované formou nesprávne konštruovaných otázok môže to veľmi narušiť jej presnosť. Opýtaní si môžu mylne interpretovať znenie týchto otázok a preto nesprávne odpovedajú.

Zaujímavý prístup pri získavaní explicitnej spätnej väzby zmätenia používateľa poskytla práca [28]. Tlačidlo "som zmätený" však nemusí vrátiť očakávané výsledky v prípade, že používateľ nevie presne čo to zmätenie znamená. Autori identifikovali tento problém a riešili ho vhodnými inštrukciami. Taktiež po každom sedení si s účastníkmi prešli momenty, kedy bolo tlačidlo stlačené. Týmto postupom predišli nevhodnej interpretácii zaznamenaných dát. Ďalšie problémy s tlačidlom sa týkajú umiestnenia veľkosti a dizajnu samotného tlačidla, ktoré môžu ovplyvniť používateľa. Táto práca ukazuje využitie explicitnej spätnej väzby na natrénovanie predikčného modelu nad implicitnou z okulografu.

⁴www.firotour.sk/

Pri implicitnej spätnej väzbe v práci [22] môžeme vidieť, že autori poukázali na vychýlenosť výsledkov založených na klikoch myši. Môžeme teda predpokladať, že na vhodnú analýzu zmätenia používateľa potrebujeme viac ako len kliknutia, ktoré počas sedenia nastali. Na druhej strane však práce [42, 32] ukazujú, že na odhalenie stavu používateľa často-krát postačujú udalosti, ktoré vieme zaznamenať jednoduchým HTTP serverom.

V tejto kapitole sme taktiež analyzovali zmätenie ako také. Z existujúcich prác vidieť, že autori neposkytujú definíciu tohto pojmu alebo ho definujú každý po svojom. Taktiež sme si všimli, že autori označujú tento pojem ako obtiažnosť alebo neistotu.

3 Metriky správania používateľa na webe

Pri interakcii používateľa s aplikáciou vieme vyhodnotiť správanie pomocou záznamov aplikácie (angl. *logs*). Samotné záznamy však môžu byť uchovávané v rôznej forme. Preto sa pre popis správania používajú metriky alebo hodnoty, ktoré sú odvodené od nich. V tejto kapitole sa venujeme metrikám. Poskytujeme prehľad prístupov pri vytváraní metrík zo záznamov okulografu a logeru.

V podkapitole 3.1 sa venujeme myši, ako zdroju metrík. Umiestnením vhodného skriptu do stránky vieme jednoducho zbierať záznamy. Z tohto dôvodu vieme použiť tieto dáta aj pri reálne nasadenej aplikácii mimo kontrolovaného prostredia. Podobne sú na tom metriky uvedené v podkapitole 3.2 spojené s klávesnicou. Diskutujeme ich reprezentáciu pomocou histogramu, ktorá sa využíva v oblasti autentifikácie používateľa. Keďže pohľad používateľa je dôležitý zdroj implicitnej spätnej väzby a v našom laboratóriu máme prístup k zariadeniam na sledovanie pohľadu (okulograf), poskytujeme tiež prehľad metrík odvodených z pohľadu používateľa (3.4). Dobrým zdrojom dát pre webové aplikácie sú aj metriky *HTTP* serveru, ktoré krátko diskutujeme v podkapitole 3.5.

3.1 Metriky myši

Počítačová myš je jedným z hlavných zariadení pri interakcii človeka s webovou aplikáciou. Pomáha pri jednoduchom prechádzaní vrámci jednej alebo viacerých stránok sídla aplikácie. Medzi hlavné akcie myši patria:

- Pohyb myši (angl. *mouse movement*). Prejavuje sa presúvaním kurzora po obrazovke v rovnakom smere ako sa pohybuje ruka s myšou.
- Kliknutie myši (angl. *mouse click*). Akcia sa vykoná po interakcii používateľa s jedným z tlačidiel myši. V zásade môže ísť o ktorékoľvek z tlačidiel. Pri bežných myšiach rozlišujeme:
 - Lavý klik (angl. *left click*).
 - Pravý klik (angl. *right click*).
 - Dvojklik (angl. *double click*). Akcia, ktorá je vyvolaná dvomi kliknutiami za sebou (zväčša sa jedná o lavý klik), medzi ktorými je čas menší ako hraničná doba δ .
- Skrol myši (angl. *mouse scroll*). Akcia sa vykoná po interakcii používateľa so skrolovacím kolieskom myši.
- Ťahaj a pusť (angl. *drag and drop*). Akcia medzi stlačením tlačidla myši a pustením tlačidla myši (zväčša lavý klik), spojená s pohybom myši nad hraničnú dobu δ .

Keďže počas interakcie používateľ generuje veľké množstvo týchto záznamov (v prípade pohybov myši, môže ísť až o niekoľko desiatok za sekundu), v praxi sa používajú metriky, ktoré sa získajú pomocou výpočtov z týchto "hrubých dát" (angl. *raw data*).

3.1.1 Metriky myši založené na pohybe

Väčšina akcií myši pri interakcii používateľa s webovou aplikáciou je tvorená práve záznamami o pohybe myši. Je preto dôležité opísať tieto udalosti vhodnými metrikami, ktoré potom charakterizujú správanie používateľa v určitých časových úsekoch.

V prácach [15, 16] sa autori snažia autentifikovať používateľa na základe interakcie v počítačovej hre s dlaždicami. Počas hrania boli akcie myši zaznamenané na serveri. Následne na to boli zo záznamov extrahované pohyby myši používateľa. Pod pojmom pohyb autori rozumejú sadu koordinátov myši v časovom úseku kde rozdiel medzi jednotlivými časovými pečiatkami je menší ako hraničná hodnota γ (autori použili $\gamma = 120ms$). Každý z pohybov musí obsahovať viac ako 4 záznamy myši ináč sa vyradí. Následne bol pohyb charakterizovaný vektormi:

- $\mathbf{t} = \{t_i\}_{i=1}^n$ - Vektor časov jednotlivých vzoriek.
- $\mathbf{x} = \{x_i\}_{i=1}^n$ - Vektor x-ových súradníc jednotlivých vzoriek.
- $\mathbf{y} = \{y_i\}_{i=1}^n$ - Vektor y-ových súradníc jednotlivých vzoriek.

Kde $\delta x_k = x_{k+1} - x_k$ a $\delta y_k = y_{k+1} - y_k$ sú zmeny v jednotlivých súradniciach. Neskôr boli vypočítané metriky. V tabuľkách 1 a 2 môžeme vidieť metriky, ktoré autori používali.

Tabuľka 1: Základné metriky odvodené z pohybu myši [15, 16].

	Názov metriky	Popis	Výpočet
1	Horizontálna rýchlosť	Rýchlosť vzhľadom na x-ovú os	$v_x = \frac{\delta x}{\delta t}$
2	Vertikálna rýchlosť	Rýchlosť vzhľadom na y-ovú os	$v_y = \frac{\delta y}{\delta t}$
3	Rýchlosť	Prvá derivácia dráhy podľa času	$v = \sqrt{v_x^2 + v_y^2}$
4	Zrýchlenie	Druhá derivácia dráhy podľa času	$\dot{v} = \frac{\delta v}{\delta t}$
5	Trhnutie	Tretia derivácia dráhy podľa času	$\ddot{v} = \frac{\delta \dot{v}}{\delta t}$

Tabuľka 2: Iné črty odvodené z pohybov myši [15, 16].

	Názov metriky	Popis	Výpočet
1	Dĺžka	Celková suma dĺžok pohybov	$s = \sum_{i=1}^n \sqrt{(x_i^2 + y_i^2)}$
2	Trvanie	Celková suma trvania pohybov	$t = \sum_{i=1}^n \delta t_i$
3	Rovnosť	Rovnosť pohybu (angl. <i>straightness</i>)	$S = \frac{\sqrt{(x_1 - x_n)^2 (y_1 - y_n)^2}}{s}$
4	Chvenie		$J = s'/s$

Okrem toho, sa na namerané pohyby aplikovalo kubické vyhladzovanie (angl. *cubic spline smoothing*) čím sa vypočítali vyhladené hodnoty pre koordináty x', y' spolu s vyhladenou dráhou s' . Následne sa pre každú z nameraných metrick $x', y', v_x, v_y, v, \dot{v}, \ddot{v}$ vypočítali štatistické hodnoty pre každý pohyb:

- Maximum - Výber najvyššej hodnoty pre daný atribút.
- Minimum - Výber najnižšej hodnoty pre daný atribút.

- Priemer - Výber priemernej hodnoty pre daný atribút.
- Štandardná odchýlka - Štatistika charakterizujúca rozptyl hodnôt pre daný atribút.
- (Minimum - maximum) - Štatistika charakterizujúca rozdelenie hodnôt.

V práci [13] sa autori snažia odhaliť neautorizovaného používateľa aplikácie pomocou jeho správania sa prostredníctvom myši. Okrem vyššie spomenutých metrík použili aj ďalšie, ktorých výpočet a popis dokumentuje tabuľka 3. V práci [27] sú tieto metríky označené ako holistické (angl. *holistic*) a taktiež boli použité pri autentifikácii používateľa aplikácie.

Tabuľka 3: Holistické metríky odvodené z pohybu myši [13].

	Názov metriky	Popis	Výpočet
1	Centrum trajektórie masy	Suma čiastkových vzdialeností násobená prislúchajúcou dobou zmeny a predelená dĺžkou pohybu (angl. <i>Trajectory center of mass TCM</i>).	$TCM = \frac{1}{s_n} \sum_{i=1}^{n-1} t_{i+1} \sqrt{\delta x^2 + \delta y^2}$
2	Rozptylový koeficient	Suma čiastkových vzdialeností na druhú, násobená prislúchajúcou dobou zmeny a predelená dĺžkou pohybu na druhú (angl. <i>Scattering coefficient SC</i>).	$SC = \frac{1}{s_n} \sum_{i=1}^{n-1} t_{i+1}^2 \sqrt{\delta x^2 + \delta y^2} - TCM^2$
3	Tretí moment	TCM s tretou mocninou trvania zmeny (angl. <i>Third moment</i>).	$M_3 = \frac{1}{s_n} \sum_{i=1}^{n-1} t_{i+1}^3 \sqrt{\delta x^2 + \delta y^2}, k = 3$
4	Štvrtý moment	TCM so štvrtou mocninou trvania zmeny (angl. <i>Fourth moment</i>).	$M_4 = \frac{1}{s_n} \sum_{i=1}^{n-1} t_{i+1}^4 \sqrt{\delta x^2 + \delta y^2}, k = 4$
5	Zakrivenie zrýchlenia	Priemer z inverznej hodnoty zrýchlenia.	$VCrv = \frac{\ddot{v}}{(1+\ddot{v}^2)^{\frac{3}{2}}}$

3.1.2 Metriky myši založené na klikoch a váhavosť

Okrem pohybu myši je jednou z akcií, ktoré používateľ vykonáva klikanie. Tento jav nastane po stlačení tlačidla myši. Pozostáva z dvoch častí:

- *Stlačenia tlačidla*. Charakteristikou tohto javu sú hodnoty spojené s pozíciou kurzora x_{down}, y_{down} a časom udalosti t_{down} .
- *Uvoľnenie tlačidla*. Charakteristikou tohto javu sú hodnoty spojené s pozíciou kurzora x_{up}, y_{up} a časom udalosti t_{up} .

V práci [13] autori používajú tieto charakteristiky klikov na výpočet metrík, ktoré sú sumarizované v tabuľke 4. Opäť aj v tomto prípade sa vypočítajú hodnoty štatistík pre každú z metrík.

Ďalším zaujímavým pohľadom na spracovanie dát o klikoch je analýza váhavosti používateľa. V práci [38] autori identifikujú váhavosť používateľa webovej aplikácie na základe dĺžky páuz v správaní pred klikmi. Ich hypotéza je založená na tvrdení, že čím je dĺžka pauzy pred klikom

Tabuľka 4: Metriky založené na klikoch [13].

	Názov metriky	Popis	Výpočet
1	Trvanie	Čas medzi stlačením a uvoľnením tlačidla myši (angl. Duration CD).	$t_{up} - t_{down}$
2	Dĺžka	Dĺžka prejdená medzi stlačením a uvoľnením tlačidla (angl. Length CL).	$\sqrt{(x_{up} - x_{down}) + (y_{up} - y_{down})}$
3	Pauza pred kliknutím	Pauza medzi uvoľnením tlačidla a ďalším kliknutím.	

Tabuľka 5: Metriky založené na nečinnosti [15].

Názov metriky	Popis	Výpočet
Počet páuz	Celkový počet páuz u používateľa	$NP = \sum_{i=1}^n p_i, p_i = \begin{cases} 1, & t_i > \beta \\ 0 & \end{cases}$
Čas strávený nečinnosťou	Súčet dĺžok páuz u používateľa	$PT = \sum_{i=1}^n p_i t_i, p_i = \begin{cases} 1, & t_i > \beta \\ 0 & \end{cases}$
Pomer času nečinnosti	Podiel času nečinnosti s celkovým časom.	$PTR = \frac{PT}{t_n}$

väčšia tým je váhavosť používateľa vyššia. Pod pojmom pauza rozumejú žiadnu činnosť s myšou ani s klávesnicou. Na základe informácie o váhavosti autori identifikujú nedostatky v používateľskom rozhraní. Nazbierané hodnoty váhavosti boli porovnané s analýzou sedení používateľským expertom, ktorý označil body, kde nastala váhavosť účastníka. Pri vyhodnocovaní výsledkov autori experimentovali s normalizáciou hodnoty na základe priemernej pauzy používateľa. Ich výsledky ukazujú ako posúvanie tejto hodnoty ovplyvňuje celkovú presnosť pri vyhodnocovaní používateľských problémov rozhrania.

Podobný prístup na predikciu používateľských problémov v aplikácii pomocou váhavosti z páuz používateľa bol navrhnutý v práci [33] už v roku 1995, kde autori merali váhavosť na základe dĺžky páuz medzi zadáním úlohy a prvého úderu klávesnice. V tomto prípade šlo o aplikáciu knižničného katalógu založenú na príkazovom riadku (angl. *command line based application*). Na overenie hypotézy zostrojili experiment s 12 účastníkmi a pri vyhodnocovaní boli odhalené všetky problémy použiteľnosti.

3.1.3 Metriky myši založené na pauzách

Jednou z ďalších zaujímavých vlastností používateľa, spojenou s myšou, je nečinnosť alebo pauzy. V spomínanej práci [15] autori využili aj metriky odvodené od tejto vlastnosti. Tabuľka 5 poskytuje prehľad s opisom spolu so vzorcami.

Pri týchto metrikách je dôležité spomenúť definíciu pauzy. V prácach [15, 13, 27] je tento jav definovaný ako nečinnosť po dobu dlhšiu ako β .

3.2 Metriky klávesnice

Okrem počítačovej myši používateľ pri interakcii s webovou aplikáciou používa klávesnicu. Typickými príkladmi interakcie sú textové polia vo formulároch, vyhľadávacie polia v katalógoch alebo klávesové skratky. Údaje o interakcii používateľa s klávesnicou môžu byť jednou z jeho charakteristík. Svedčí o tom práca [27], kde autori využívajú takéto dáta k autentifikácii používateľa. Jednou z týchto charakteristík je dynamika úderov klávesov (angl. *keystroke dynamics*). Je definovaná časom od stlačenia klávesy až po jej uvoľnenie. Takéto údaje súvisia s jedným úderom. V prípade páru (angl. *digraf*) môžeme pracovať so štyrmi hodnotami. Časom od stlačenia po stlačenie, od uvoľnenia po uvoľnenie, od stlačenia po uvoľnenie a od uvoľnenia po stlačenie.

Autori práce [14] sa venovali autentifikácii používateľa pomocou dynamiky úderov už v roku 1980. Pracujú so záznamami o písaní šiestich pisárov, ktorých úlohou bolo prepísať tri texty. Následne na to pre každý pár klávesov vypočítali štatistické hodnoty pre jednotlivé časy. Ich metóda bola vyhodnotená na texte s 300 slovami a dosiahla 4% chybovosť. To znamená, že len v 4 percentách prípadov by systém autentifikoval neoprávneného používateľa (angl. *false acceptance rate FAR*).

3.3 Histogramová reprezentácia črt

Črty ako také môžu veľa prezrádzať o správaní používateľa. Svedčí o tom mnoho prác spomínaných v predchádzajúcich kapitolách. Ich získanie je však mnohokrát spojené s agregovaním iných hodnôt. Vezmime si napríklad prácu [15], kde autori najprv rozdelili získané dáta na pohyby, následne extrahovali črty myši pre každý pohyb zvlášť a vo výsledku vypočítali agregované hodnoty. Takto vytvorený model používateľa však stráca informácie o pohyboch zvlášť. Problém je načrtnutý aj v práci [27], kde sa autori venujú biometrii.

Ako riešenie poskytujú histogramovú reprezentáciu črt pohybov myši. Podrobnejšie je riešenie rozobrané v práci [2], kde sa autori venujú biometrii správania používateľa založenej na pohyboch myši s použitím neurónovej siete. Spomínaný problém má však niekoľko obmedzení. Jedným z nich je určenie počtu stĺpcov histogramu. Riešenie tohto problému môže byť spojené s natrénovaním niekoľkých modelov s rôznym počtom stĺpcov a následným výberom najlepšieho modelu.

Druhý problém je spojený s hraničnými hodnotami pre stĺpce histogramu. V práci [27] autori zvolili postup, kde priestor medzi minimálnou a maximálnou hodnotou danej črty lineárne rozdelia. Výsledky však môžu byť ohrozené vybočujúcimi hodnotami (angl. *outlayers*). Iné riešenie môže byť spojené s lineárnym rozdelením priestoru v medzikvartilovom rozpätí hodnôt.

3.4 Sledovanie pohľadu

Každý používateľ pri interakcii s webovou aplikáciou spracováva informácie primárne pomocou zrakového vnemu. Údaje o pohybe očí človeka počas interakcie s webovou aplikáciou môžu ho-

vorit o jeho zmätení. Svedčia o tom spomínané práce [29, 28]. V našej práci sa snažíme porovnať prístupy predikcie zmätenia založené na dátach z okulografu s našou metódou založenou na dátach z myši. Z toho dôvodu uvádzame existujúce metriky okulografu popisujúce správanie používateľa. Podľa práce [10] ľudské oko pri svojich pohyboch používa šesť svalov. V práci sa uvádza, že rozlišujeme niekoľko základných pohybov:

- *Sakády*. Sú zároveň dobrovoľné a reflexné pohyby oka. Ich trvanie je v intervale od 10 do 100ms čo je príliš krátke na zachytenie obrazu.
- *Hladké nasledovanie* (angl. *Smooth pursuit*). Ide o pohyb oka, kedy je prenasledovaný objekt pohľadom.
- *Fixácie*. Je jav, počas ktorého je stabilizovaná sietnica. Počas neho je pozornosť upriamená na objekt. V práci [10] sa uvádza, že fixácie sú zväčša spojené s miniatúrnymi pohybmi očí.

3.4.1 Okulograf

Na to, aby sme mohli zaznamenávať pohyby očí potrebujeme okulograf (angl. *eyetracker*). Ide o zariadenie, ktoré mapuje tieto pohyby do súradnicovej sústavy. Podľa práce [10] poznáme dva základné druhy takýchto zariadení. Prvý je hardvér pripevnený na hlavu (angl. *head-mounted hardware*). Ide o zariadenie podobné okuliarom, ktoré priamo zachytáva pohyby oka z bezprostrednej blízkosti. Druhou skupinou je hardvér pripevnený na stole (angl. *table-mounted hardware*). V tomto prípade ide o zariadenia, s ktorými sa počas experimentu nemanipuluje. Príkladom sú produkty firmy *Tobii* [41].

Jednou z kľúčových vecí pri nahrávaní pohľadu človeka pomocou okulografu je frekvencia zariadenia. Ide o počet záznamov za sekundu, ktoré je nástroj schopný zaznamenať. Štandardne sa pracuje s frekvenciou 60 až 300Hz [10].

Podľa práce [10] je pred samotným nahrávaním pohľadu s okulografom dôležitá kalibrácia. V tomto kroku je účastníkovi experimentu nastavované zariadenie na sledovanie pohľadu. V prípade modelov pripevnených na stole sa pracuje s výškou stoličky a vzdialenosťou okulografu od hlavy. Následne sa prechádza kalibračná sekvencia. Ide o obrazovku, na ktorej účastník pohľadom nasleduje guľôčku postupne po všetkých rohoch obrazovky. V prípade zariadení pripevnených na hlavu ide o veľmi podobné sekvencie nasledovania objektov pohľadom.

3.4.2 Zrenička

Súčasťou ľudského oka je aj zrenička. Okulografy okrem dát o bode pohľadu dokážu poskytnúť aj informácie o jej veľkosti. Ich spracovaním vieme získať informácie spojené s mentálnou činnosťou používateľa. V práci [19] sa autori zaoberajú dilatáciou zreničky. Dilatácia zreničky je zmena veľkosti. Zostavili experiment s dvanástimi účastníkmi, kde sa riešili úlohy rôznej mentálnej obtiažnosti. Ich výsledky ukazujú, že obtiažnosti týchto úloh boli spojené so zmenou veľkosti

zreničky. Taktiež v práci [5] autori ukazujú vzťah medzi kognitívnou záťažou človeka a jeho zreničkou.

Pri týchto dátach však treba byť opatrný. Dilatácia zreničky je ovplyvnená svetelnými podmienkami prostredia. To znamená, že jej veľkosť môže byť odlišná v závislosti od svetelných podmienok alebo farieb monitoru. Vplyvu efektu svietivosti displeja na zreničku sa venujú autori práce [23], kde pre tento účel zostavili model. Jeho úspešnosť overili na experimente so 73 účastníkmi.

3.4.3 Oblasti záujmu

Oblasti záujmu (angl. *areas of interest* AOI) sú jednou z dôležitých vecí v experimente so sledovaním pohľadu. V prípade experimentu s webovou aplikáciou sa jedná o časti rozhrania, na ktoré sa chceme zamerať. Môže ísť napríklad o časti html kódu stránky. Oblasti záujmu sa však dajú vymedziť aj pri experimentoch s desktopovou aplikáciou. Vždy je však potrebné vyznačiť časti, na ktoré sa zameriavame.

Sledovanie pohľadu používateľa v oblastiach záujmu môže veľa prezrádzať o jeho správaní. V knihe [10] sa metriky odvodené od AOI označujú ako bežné. V práci [29] autori pomocou takýchto metrick odhaľujú problémové správanie používateľov pri práci s vizualizáciami. V ich experimente sa snažia predikovať učiacu krivku (angl. *learning curve*). Taktiež v práci [17] využívajú tieto metriky pri vyhodnocovaní nástrojov na kreslenie.

3.4.4 Metriky sledovania očí

V tejto časti chceme zhrnúť metriky pohľadu používané v spomínaných dielach. Prehľad sumarizujeme tabuľkou 6. Okrem metrick prác z predchádzajúcich podkapitol poskytujeme aj prehľad základných metrick odvodených z fixácii a sakád. V prácach [24, 25] ich autori využívajú pri generovaní modelu na podporu študentov pri práci s interaktívnymi simuláciami. Vo väčšine prác sa však pracuje so štatistikami týchto metrick. Vypočítajú sa hodnoty priemeru, minima, maxima a štandardnej odchýlky a následne sú použité v modeli.

3.5 Metriky HTTP serveru

Jednou z najbežnejších softvérových architektúr v prípade webových aplikácii je architektúra klient server. V rámci nej klientská časť kódu vo forme *html* a *css* je poslaná zo serveru priamo do prehliadača cieľového používateľa pri dopyte na cieľovú url. Pri prechode používateľa medzi jednotlivými pod-stránkami aplikácie sa na server posielajú dopyty. Server následne odpovedá opäť formou *html* a *css* kódu pre odosielateľa. Na väčšine moderných serverov sa celá komunikácia loguje do súborov. Tieto logy vedia poskytnúť cenné informácie o správaní používateľa a zároveň ich získavanie nie je nákladné.

Tabuľka 6: Prehľad metrík sledovania pohľadu z literatúry.

Názov	Anglický názov	Popis	Zdroje
Odvođené od sakád			
Dĺžka sakády	<i>Saccade duration</i>	Vzdialenosť prejdená medzi dvomi po sebe idúcimi fixáciami.	[24, 25]
Uhol sakády (relatívny)	<i>Relative saccade angle</i>	Uhol sakády k predchádzajúcej sakáde.	[24, 25, 17]
Uhol sakády (absolútne)	<i>Absolute saccade angle</i>	Uhol sakády k x-ovej osi.	[24, 25, 17]
Počet sakád	<i>Number of saccades</i>	Počet sakád pre pozorovanie.	[12]
Trvanie sakády	<i>Saccade duration</i>	Trvanie sakády v milisakundách	[17]
Pomer sakád	<i>Saccade ratio</i>	Pomer počtu sakád k celkovému počtu časovej doby.	[17]
Odvođené od fixácii			
Počet fixácii	<i>Number of fixations</i>	Počet fixácii pre pozorovanie	[24, 25, 17]
Frekvencia fixácii	<i>Fixation rate</i>	Je to počet fixácii za jednotku času	[24, 25, 17]
Dĺžka trvania fixácie	<i>Fixation duration</i>	Je to časová doba fixácie	[24, 25, 17]
Priestorová hustota fixácii	<i>Spatial density</i>	Hustota fixácii v priestore	[17]
Čas do prvej fixácie	<i>Time to first fixation</i>	Je to čas, ktorý uplynie pred prvou fixáciou	[8, 37]
Odvođené od AOI			
Počet fixácii pre oblasť	Fixation rate on AOI	Je to množstvo fixácii na konkrétnu oblasť.	[29, 17]
Dĺžka fixácie pre oblasť	Fixation length on AOI	Dĺžka fixácie pre oblasť.	[29, 17]
Počet prechodov z danej oblasti inam	Number of transitions	Počet jednotlivých prechodov z vybranej oblasti do inej oblasti.	[29, 17]
Odvođené od zreničky			
Šírka zreničky	Pupill width	Šírka zreničky pre záznam.	[19]

Tabuľka 7: Prehľad metrík odvodených z HTTP serveru z práce [42].

Metrika	Popis
Celkový čas	Doba od prvého záznamu sedenia až po posledný záznam.
Počet stránok	Počet unikátnych url navštívených počas sedenia.
Čas na stránku	Priemerný čas pre navštívenú stránku v rámci sedenia.
Počet cyklov	Cyklom sa označuje taká cesta stránok, pri ktorej sa užívateľ vráti k už raz navštíveneju stránke.
Počet zámen (angl. swap)	Zámenou sa označuje taká postupnosť stránok, v rámci ktorej používateľ prejde na stránku inej domény a následne sa vráti (napr. využitie vyhľadávča).

V už spomínanej práci [42] autori využívajú metriky odvodené z týchto logov na predikciu obtiažnych sedení. Prehľad metrík poskytuje tabuľka 7. O širšom využití dát z web logov svedčí aj práca [31], kde autori poskytujú prehľad možností ich sekvenčného spracovania. Autori v článku prešli široké využitie týchto algoritmov a výstupom práce je taxonómia, ktorá delí algoritmy spracovania sekvencií z logov do troch skupín.

3.6 Zhrnutie a diskusia

V tejto kapitole sme opísali metriky myši, klávesnice a pohľadu vhodné na modelovanie správania používateľa. Na to aby sme vedeli tieto metriky extrahovať a ďalej s nimi pracovať, je potrebné aby sme mali dáta v požadovanej forme. Napríklad, v prípade metrík myši potrebujeme aby sme mali dáta vo forme pohybov (skupina záznamov súradníc myši v čase, kde časový rozdiel medzi jednotlivými záznamami je menší ako hraničná hodnota). Tu sa nám otvára priestor pre experimentovanie s hraničnou hodnotou, jej posúvaním totiž vieme získať iné rozdelenie súradníc do pohybov a teda aj iné hodnoty metrík.

V prípade okulografu potrebujeme záznamy koordinátov rozdeliť do fixácií a sakád. Na toto rozdelenie vieme použiť jeden z už existujúcich implementácií fixačných filtrov. Keďže týchto implementácií je viacero opäť sa tu otvára priestor na experimentovanie. V našom prípade použijeme fixačný filter, ktorý je súčasťou infraštruktúry, ktorú použijeme pri používateľskej štúdií.

Je vhodné spomenúť, že pri metrikách myši sme sa venovali najmä tým, ktoré sa využívajú pri autentifikácii používateľa na webe. Dôvod je slabé využitie metrík myši pri predikcii zmätenia. Avšak ako sme spomínali tieto dva problémy sú veľmi podobné, keďže pri oboch sa snažíme namodelovať daného používateľa na základe jeho správania. Pri autentifikácii ide o klasifikáciu do viacerých tried, zatiaľ čo pri predikcii zmätenia nás zaujímajú hlavne dve triedy a to zmätený a nezmätený. Poskytli sme zaujímavý pohľad na spracovanie metrík formou histogramov. Tu chceme pripomenúť, že na to aby bolo toto použitie efektívne je potrebné, aby pre každý histogram používateľa bolo dostatok dát. Nemá totiž zmysel využiť takúto reprezentáciu, keď chceme rozdeliť len pár dát do histogramov používateľa (napr. menej ako 10).

Problémom metrík pohľadu je dostupnosť okulografov. Bežný používateľ nedisponuje takýmto zariadením v domácnosti. Aplikácia modelu založenom na takýchto metrikách môže byť proble-

matická. Avšak ich použitie na kontrolovanom experimente je ideálne. Využitím dát z pohľadu môžeme lepšie pochopiť akcie používateľa. Využijeme ich na porovnanie modelu založenom len na správaní z myši a klávesnice.

Stále sa tu však otvára diskusia spojená s ochranou osobných údajov. Pri prípadnom produkčnom využití nášho riešenia je potrebné na to pamätať. V takomto prípade môže byť problém pri zaznamenávaní správania klávesnice, ktoré so sebou nesie citlivé informácie o používateľovi. Z tohto dôvodu sa budeme ďalej sústrediť na dáta z myši a okulografu.

4 Predikčný model

Po získaní dát a extrahovaní črt z pohľadu a myši je potrebné natrénovať model predikcie zmätenia. V tomto prípade sa bude jednať o binárnu klasifikáciu a učenie s učiteľom. Je preto vhodné zvoliť model a následne vybrať metriky na porovnávanie medzi netrénovanými modelmi.

Učenie s učiteľom (angl. *supervised learning*) je metóda strojového učenia (angl. *machine learning*), pri ktorej sa model učí z vopred označených dát. Následne vo fáze predikcie model využije nadobudnuté informácie [26]. Okrem toho poznáme ešte učenie bez učiteľa (angl. *unsupervised learning*), kde dáta nie sú vopred označené a učenie s odmenou a trestom (angl. *reinforcement learning*).

V našom prípade sa bude jednať o klasifikáciu dát do dvoch skupín zmätený, nezmätený. Klasifikačný problém je taký, kde predikujeme kategorickú premennú a nevieme povedať vzdialenosť medzi jej jednotlivými hodnotami. Opakom je regresný problém, kde predikovaná premenná je taká pri ktorej vieme určiť vzdialenosť medzi dvomi jej hodnotami. Príkladom regresného problému môže byť predikcia ceny domu [11].

4.1 Klasifikácia v reálnom čase

Jedným zo zámerov našej práce je využitie detekcie zmätenia, t.j. výsledkov klasifikácie v reálnom čase. Snahou tejto práce je vedieť odhaliť moment, kedy stav zmätenia u používateľa nastal tak, aby bolo možné čo najskôr na identifikovaný stav reagovať. V prístupe práce [42] autori používajú pri predikcii obtiažnych sedení dáta z celého sedenia. Takýto prístup klasifikácie by sa dal využiť v našom prípade vtedy, keď by sme chceli označiť sedenia, kde nastalo zmätenie po ich uplynutí. Čo však prípady, kde chceme predikovať zmätenie už počas sedenia?

V už spomínaných prácach [29, 28] autori predikujú zmätenie v reálnom čase na základe časových okien. Pod pojmom časové okno (angl. *time window*) autori rozumejú časový interval rovnakej veľkosti (napríklad niekoľko sekúnd), pomocou ktorého sa rozdelia záznamy o sedení. Na samotné delenie autori využili algoritmus posuvného okna (angl. *sliding window algorithm*), ktorý na začiatku extrahuje okno od časovej pečiatky prvého záznamu. Následne sa toto okno posúva o n sekúnd a extrakcia pokračuje až kým sa nedosiahne posledný záznam. Každé okno predstavuje jeden záznam datasetu. Predmetom výskumu v tomto prístupe je určenie veľkosti okna. Autori práce pracovali hneď s niekoľkými nastaveniami (5 a 10 sekúnd).

4.2 Logistická regresia

Ako jedným zo základných klasifikačných algoritmov je logistická regresia. V našom prípade, predikcie len dvoch hodnôt sa javí ako jedna z najvhodnejších možností. Výstupom predikcie logistickej regresie v prípade dvoch hodnôt je pravdepodobnosť s akou pozorovanie o patrí do jednej z tried. Označením $P(\text{zmätený} = \text{áno} \mid \text{črta})$ skrátene $P(\text{črta})$ budeme rozumieť pravdepodobnosť s akou bude pozorovanie klasifikované ako zmätené vzhľadom k črte. Štandardne sa

pozorovanie klasifikuje vždy vtedy, keď je táto hodnota väčšia ako 0.5, avšak hraničná hodnota sa dá upraviť v závislosti od problému.

Model logistickej regresie vychádza z lineárnej regresie, ktorá slúži na riešenie regresných problémov. Pri nich je predikovaná hodnota kvantitatívna, a teda vieme určiť vzdialenosť medzi predikovanými hodnotami. Na výpočet modelu lineárnej regresie sa používa vzorec [39]

$$\varnothing(x_1, x_2, \dots, x_n) = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n, \quad (1)$$

kde označením x_1 až x_n rozumieme zložky vektoru pozorovania (v našom prípade hodnoty vybraných metrik pre konkrétne pozorovanie) a pod označením β_1 až β_n rozumieme parametre modelu.

Ak by sme model aplikovali na predikciu kategorických dát je jasné, že výstupom funkcie by boli hodnoty mimo intervalu $[0;1]$. Aby sme sa vyhli tomuto problému musíme použiť funkciu, ktorá mapuje tieto výsledky do intervalu $\langle 0;1 \rangle$. V prípade logistickej regresie sa používa *logistická funkcia* [34]

$$\varnothing(x_1, x_2, \dots, x_n) = \frac{e^{\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n}}{1 + e^{\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n}}. \quad (2)$$

Získaná funkcia má tvar písmena S a vždy nadobudne hodnoty z intervalu $\langle 0;1 \rangle$. Jej úpravou vieme získať [34] vzorec

$$\log\left(\frac{\varnothing(x_1, x_2, \dots, x_n)}{1 - \varnothing(x_1, x_2, \dots, x_n)}\right) = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n, \quad (3)$$

kde ľavá časť sa nazýva *logit*.

Súčasťou tréningu modelu Logistickej regresie je učenie sa parametrov $\beta_0 \dots \beta_n$. Ako je spomenuté v knihe [34] na ich učenie sa používajú tréningové dáta a maximalizovanie funkcie (angl. *maximum likelihood*)

$$l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=1} (1 - p(x_{i'})). \quad (4)$$

Kde parameter β_0 sa špeciálne označuje slovom zachytenie (angl. *intercept*).

4.3 Náhodný les

Algoritmus náhodný les (angl. *Random forest*) je ďalšou možnosťou v prípade klasifikácie. V práci [29] bol použitý pri binárnej predikcii zmätenia nad dátami z okulografu. Jednou z jeho výhod je nelineárny prístup k riešeniu problémov. Algoritmus má aj svoju modifikáciu na riešenie regresných problémov. Náhodný les je jedným z algoritmov súborového učenia (angl. *ensemble learning*) [44]. Klasifikátory súborového učenia sú založené na aplikácii viacerých klasifikátorov súčasne. V prípade náhodného lesu ide o kombináciu viacerých rozhodovacích stromov (angl. *Decision tree*).

Rozhodovací strom je algoritmus založený na sérii pravidiel začínajúcich v koreni stromu. Listy,

alebo terminálne uzly stromu, tvoria predikované hodnoty algoritmu. Miesta vo vnútri stromu, kde sa vetvy delia sa nazývajú interné uzly a vytvárajú ďalšie vetvy. Každá z vetiev je zakončená hodnotou (listom) alebo ďalším interným uzlom. Podľa knihy [20] tréning algoritmu regresie pomocou rozhodovacieho stromu pozostáva z dvoch krokov:

1. Rozdelenie priestoru pozorovaní X_1 až X_N do J neprekrývajúcich sa regiónov R_1 až R_J .
2. Hodnota regiónu je reprezentovaná priemernou cieľovou hodnotou všetkých pozorovaní v regione.

Delenie na regióny je s cieľom minimalizovať sumu štvorcových chýb (pri regresie). Avšak vo väčšine prípadov existuje veľké množstvo kombinácií delení. Z tohto dôvodu sa používa chamtivý prístup zhora na dol (angl. *top-down greedy approach*) označovaný ako rekurzívne binárne delenie (angl. *recursive binary splitting*). Začína sa v koreni stromu, kedy všetky pozorovania patria do jedného regiónu. V každom kroku delenia sa vytvorí vždy dva nové regióny. Chamtivým sa označuje tento prístup preto, že výber rozdelenia na danej úrovni je ovplyvnený minimalizáciou chyby len pre danú úroveň stromu bez ohľadu na možné zlepšenia pri ďalších úrovniach. Takýto algoritmus však vyprodukuje veľké stromy. V praxi sa potom tieto stromy orezávajú (angl. *pruning*) na menšie stromčeky. Výber menších stromčekov sa realizuje na základe chyby na krížovej validácii.

Pri klasifikácii pomocou rozhodovacieho stromu sa pre daný región vyberie najčastejšie vyskytovaná trieda (angl. *most commonly occurring class*), následne je však potrebné región ohodnotiť. V knihe [20] sa uvádza niekoľko spôsobov ako vyhodnocovať výber regiónov:

1. *Klasifikačná chyba* (angl. *classification error*)

$$err = 1 - \max_k(p_{mk}), \quad (5)$$

Kde p_{mk} vyjadruje proporciu počtu triedy k v m -tom regione. Avšak táto chyba nie je citlivá na rast stromu, preto sa vo väčšine problémov nevyužíva.

2. *Gini index*

$$gini = \sum_{k=1}^K p_{mk}(1 - p_{mk}), \quad (6)$$

ktorý nadobúda nízke hodnoty v prípade, že všetky hodnoty p_{mk} sú blízke k 0 alebo 1. Malé hodnoty tohto indexu hovoria o tom, že v regione dominuje jedna z tried.

3. *Entropia*

$$entr = - \sum_{k=1}^K p_{mk} \log(p_{mk}), \quad (7)$$

ktorá nadobúda rovnaké hodnoty ako Gini index.

Pri algoritme náhodného lesu sa trénuje n stromov súčasne. Na vytvorenie odlišných tréningových sad pre každý zo stromov sa využíva metóda vzorkovania (angl. *bootstrapping*), kedy sa celková tréningová sada rozdelí náhodným výberom vzoriek.

4.4 Vyhodnotenie modelov

Po natrénovaní modelov nad dátami je potrebné ich vyhodnotiť. Pred samotnou aplikáciou metrík vyhodnotenia je vhodné si dataset rozdeliť na tréningovú a testovaciu sadu. Štandardne je tréningová sada niekoľkonásobne väčšia ako testovacia. Používajú sa pomery ako 80 ku 20 percentám, alebo 70 ku 30 percentám. Pri klasifikácii je však potrebné, aby sa zachoval aj pomer v početnosti jednotlivých tried pre tréningovú a testovaciu sadu.

Pri mnoho problémoch je však dataset vysoko nevyvážený [29, 42]. V takomto prípade je vhodné ho vyvážiť pomocou techniky vzorkovania. Poznáme dve techniky vzorkovania:

1. Pod-vzorkovanie (angl. *undersampling*). Je technika, kedy sa z majoritnej triedy vzoriek vyjme len určitá menšia reprezentatívna vzorka. V práci [30] autori ukazujú priaznivý vplyv pod-vzorkovania na metriku plocha pod priamkou (opísanú v ďalšej podkapitole).
2. Nad-vzorkovanie (angl. *oversampling*). Je opakom pod-vzorkovania. Kedy sa snažíme zväčšiť početnosť minoritnej triedy v datasete. Tento postup je mnohokrát spojený s generovaním umelých dát odvodených od existujúcich.

V práci [1] autori predstavujú symetrickú techniku nad-vzorkovania minoritnej triedy (*Smote*). Technika je založená na kombinácii pod-vzorkovania a nad-vzorkovania vzhľadom k minoritnej triede datasetu. Aj tu autori opisujú priaznivé výsledky pri klasifikácii vzhľadom na metriku plochy pod priamkou s použitím klasifikátora rozhodovacích stromov. Počas vzorkovania sa vytvárajú syntetické reprezentácie minoritnej triedy na základe k -najbližších susedov a náhody. Zároveň počas toho je pod-vzorkovaná majoritná trieda v rovnakom pomere s použitím náhodného výberu. Táto technika je tiež použitá v práci [29] pri predikcii zmätenia s použitím okulografu.

Po vyvážení datasetu a natrénovaní modelu na tréningovej sade vzoriek pristupujeme k časti predikcia. Vezmeme testovacie vzorky $x_{1..n}$ a necháme model predikovať ich triedy $y_{1..n}$. Ich skutočné triedy budeme označovať $d_{1..n}$. V prípade binárnej klasifikácie vieme, že $y_i, d_i \in \{0, 1\}$. Špecifickejšie pre náš prípad 1 znamená zmätený a 0 nezmätený. Po predikcii vieme identifikovať 4 možné množiny [40]:

1. Správne pozitívne pozorovania (angl. *true positive* TP). Sú také pozorovania z testovacej množiny, kde $y_i = 1$ a zároveň $d_i = 1$.
2. Správne negatívne pozorovania (angl. *true negative* TN). Sú také pozorovania z testovacej množiny, kde $y_i = 0$ a zároveň $d_i = 0$.
3. Nesprávne pozitívne pozorovania (angl. *false positive* FP). Sú také pozorovania z testovacej množiny, kde $y_i = 1$ a zároveň $d_i = 0$.
4. Nesprávne negatívne pozorovania (angl. *false negative* FN). Sú také pozorovania z testovacej množiny, kde $y_i = 0$ a zároveň $d_i = 1$.

Tieto množiny vieme lepšie zobrazit do matice zmätenia (angl. *confusion matrix*), zobrazenej v tabuľke 8.

Tabuľka 8: Všeobecná matica zmätenia, y_i je predikovaná hodnota a d_i je očakávaná hodnota [40].

	$d_i = 1$	$d_i = 0$
$y_i = 1$	TP	FP
$y_i = 0$	FN	TN

V ďalších častiach opíšeme metriky vyhodnocovania modelov klasifikácie odvodené pomocou matice zmätenia.

4.4.1 Presnosť a úplnosť

Podľa práce [40] označujú autori metriky presnosť (angl. *precision*) a úplnosť (angl. *recall*) ako jedny z najpopulárnejších metrick na vyhodnotenie klasifikačných modelov. Tieto metriky boli taktiež použité v práci [42] pri porovnaní modelov na predikovanie obtiažnych sedení. Úplnosť bola použitá v práci [29] na porovnanie modelov na predikciu zmätenia používateľa z jeho pohľadu.

Presnosť označuje úspešnosť klasifikátora pri označení, že testovacia vzorka patrí do danej triedy. Jej vzorec je [40]

$$precision = \frac{TP}{TP + FP}. \quad (8)$$

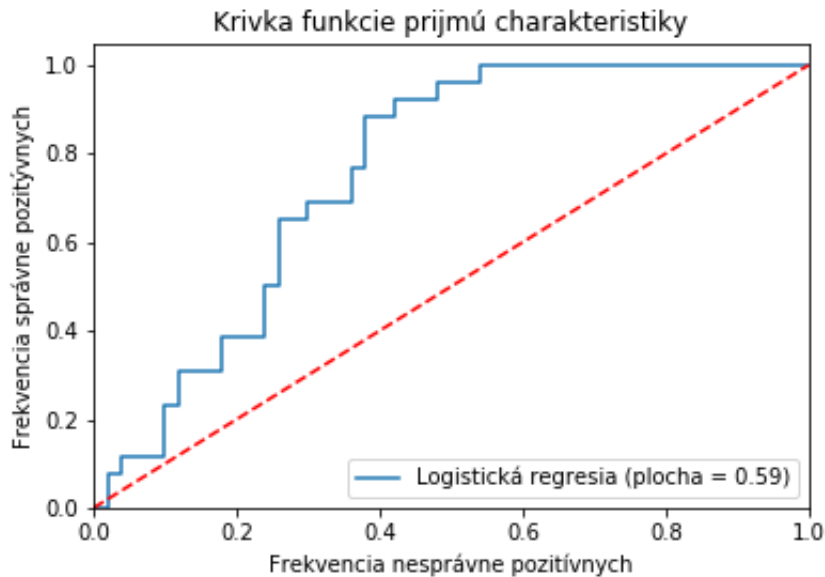
Na druhej strane tu máme úplnosť, ktorá hovorí o tom, ako veľmi vieme pomocou natrénovaného modelu predikovať všetky výskyty danej triedy. Jej vzorec je [40]

$$recall = \frac{TP}{TP + FN}. \quad (9)$$

4.4.2 Plocha pod krivkou charakteristiky prijímajúceho operátora

Plocha pod krivkou (angl. *area under curve* AUC) je metrika, ktorá sa počíta vyjadrením obsahu pod krivkou funkcie prijímú charakteristiky (angl. *Reception Operating Characteristic* ROC). ROC krivka nám zase hovorí o schopnosti modelu rozlišovať triedy na testovacej sade [18]. Je založená na rastúcom vzťahu medzi TP a FP meraniami. Hodnota AUC je následne len výpočtom plochy pod touto krivkou. Čím je hodnota AUC vyššia tým je model lepší.

Obrázok 3 poskytuje ukážku ROC krivky s prislúchajúcim výpočtom AUC. Na osách grafu môžeme vidieť hodnoty relatívnych počtov pre TP a FP . Taktiež je na grafe vyznačený prerušovanou čiarou najhorší možný model.



Obr. 3: Ukážka *ROC* krivky (plná čiara). Prerušovanou čiarou je označená najhoršia možná situácia. Zdroj grafu sú prvotné experimenty na dátach z myši.

V práci [40] *AUC* označujú ako jednu z najpopulárnejších metrík na vyhodnotenie klasifikačných problémov. Metrika je použitá na porovnanie modelov aj v spomínaných prácach [28, 42]. Taktiež autori uvádzajú vzorec na výpočet

$$AUC = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right). \quad (10)$$

4.4.3 Akeikeho informačné kritérium

Akeikeho informačné kritérium (angl. *Akaike information criterion* AIC). Metrika pomenovaná podľa slávneho štatistika Hirotugu Akaikeho, ktorá určuje, ktorý z dvoch modelov je lepší. Na vyhodnotenie okrem chyby na testovacej sade (angl. *sum of squares error* SSE), berie do úvahy aj zložitosť modelu, ktorá negatívne vplyva na hodnotu tejto metriky. Vzorec na výpočet *AIC* potom vyzerá nasledovne

$$AIC = N * \ln\left(\frac{SSE}{N}\right) + 2 * k, \quad (11)$$

kde N označuje počet meraní, SSE označuje sumu chýb štvorcov a k označuje počet parametrov modelu [3]. Čím nižšia je hodnota *AIC*, tým je model považovaný za lepší.

Okrem toho poznáme aj Bayesovo informačné kritérium (angl. *Bayesian information criterion* BIC), ktoré ešte viac penalizuje počet parametrov modelu

$$BIC = N * \ln\left(\frac{SSE}{N}\right) + k * \ln(N). \quad (12)$$

4.5 Zhrnutie a diskusia

V tejto kapitole sme sa venovali predikčným modelom pre problém binárnej klasifikácie. Pred samotným predstavením možných modelov sme prešli možnosti predikcie v reálnom čase. V produkcii totiž nemôžeme predikovať na celej úlohe, keďže nevieme určiť začiatok a koniec. Využitie spomínaných okien rovnakej veľkosti sa javí ako možné riešenie. Otvára sa tu však priestor na experimentovanie s veľkosťou okna. Výber príliš malých okien vedie k slabej úspešnosti. Svedčí o tom najmä práca [29]. Na druhej strane výber príliš veľkých okien robí problém v produkcii, kde mnohokrát potrebujeme predikovať zmätenie okamžite po príchode používateľa, nemôžeme čakať, kým sa okno naplní.

Ďalej sme predstavili klasifikátor logistickej regresie a klasifikátor náhodného lesu. Prihliadnuc k literatúre [29, 28, 42, 20, 44] sa tieto algoritmy ukazujú ako ideálne na riešenie problémov binárnej klasifikácie. Algoritmus logistickej ukazuje lepšie výsledky pri lineárne separovateľných problémoch zatiaľ čo náhodný les naopak. Pri ich trénovaní sa otvára priestor na experimentovanie s hyper parametrami. Napríklad v prípade náhodného lesu vieme pracovať s množstvom a hĺbkou stromov a v prípade logistickej regresie zase s váhami predikovaných tried.

Opísali sme aj metriky, ktoré je možné využiť pri vyhodnotení modelov. Hlavná myšlienka našej práce je, že plánujeme nasadiť naše riešenie priamo do produkčného prostredia. Tu je veľmi dôležitá presnosť pri predikcii triedy zmätenia a teda počty jej správne pozitívnych pozorovaní a nesprávne pozitívnych pozorovaní. Z tohto dôvodu sa sústredíme práve na metriku presnosti a na samotnú maticu zmätenia pri modeloch. Identifikácia zmätenia v produkcii bude sprevádzaná ponúknutím rady zmätenému používateľovi. Chceme teda predísť zbytočnému zaťažovaniu nezmätených používateľov.

5 Návrh metódy predikcie zmätenia

V našej práci sa venujeme identifikácii stavu zmätenia používateľa vo webovej aplikácii. Táto kapitola predstaví návrh metódy klasifikácie. Predpokladáme dva zdroje dát na predikciu. Prvým zdrojom je myš, a jej údaje získané pri interakcii s webovou aplikáciou. Druhým zdrojom dát je okulograf a údaje získané zo sledovania pohľadu používateľa. Dáta z okulografu poslúžili na porovnanie výsledkov klasifikácie.

Našu metódu sme rozdelili do dvoch scenárov. V prvom sa snažíme detekovať zmätenie na základe dát z celého sedenia. Podobný postup zvolili autori v práci [42]. V druhom scenári sa snažíme odhaliť presný bod zmätenia v reálnom čase interakcie užívateľa s webovou aplikáciou. Pod pojmom presný bod rozumieme moment, kedy používateľ nevie pokračovať v napĺňaní svojich potrieb v aplikácii vo vlastnej réžii. Tradične je tento bod spojený s využitím nápovedy, zákazníckej podpory alebo externej služby (vyhľadávača). Taktiež sa môže stať, že je tento moment spojený s rezignáciou, kedy používateľ aplikáciu opustí. Keďže v našom prípade využijeme metódu s nápovedami sústredíme sa na taký moment, kedy je používateľ ochotný priznať si, že takúto nápovedu naozaj potrebuje. Môže sa teda jednať aj o interval, ktorý takémuto bodu predchádza.

Našu metódu predikcie zmätenia sme rozdelili do niekoľkých krokov:

1. *Predspracovanie surových dát.* V tomto kroku čistíme surové dáta (angl. *raw data*). To znamená, že odstraňujeme duplikáty a neúplné záznamy. Okrem toho pri dátach z myši extrahujeme pohyby a pri okulografe fixácie a sakády. V prípade predikcie v reálnom čase taktiež extrahujeme okná rovnakej časovej dĺžky. Experimentovali sme s oknami dĺžky 5 a 10 sekúnd podobne ako v prácach [28, 29].
2. *Extrakcia črt.* V tomto kroku pristúpime k extrakcii črt z oboch zdrojov dát. V prípade vysoko nevyváženého datasetu je potrebné vyvážiť minoritnú triedu. Jednou z techník v tomto prípade je vzorkovanie (angl. *oversampling*).
3. *Natréňovanie klasifikátora.* V tomto kroku pristúpime k trénovaniu vybraných klasifikátorov. V prípade viacerých sa vyberie ten najúspešnejší na základe zvolených metrick: presnosť a úplnosť pri predikcii zmätenia, plocha pod krivkou charakteristiky prijímajúceho operátora a akaikeho informačné kritérium v prípade porovnaní viacerých nastavení rovnakého klasifikátora.
4. *Klasifikácia.* Po natréňovaní modelu prechádzame do predikcie. V prvom prípade pôjde o predikciu nad dátami z celého sedenia, zatiaľ čo v druhom predikujeme nad oknami rovnakej veľkosti.

5.1 Predspracovanie dát logeru myši a extrakcia črt

Z analýzy existujúcich riešení vyplýva, že forma surových dát získaných pri zaznamenávaní správania nie je vhodná pre tréňovanie predikčných modelov. Je preto potrebné dáta upraviť.

Tabuľka 9 poskytuje prehľad záznamu, ktorý bol zaznamenaný naším logerom. Opis jednotlivých polí:

1. **Šírka okna prehliadača** (*displayWidth*) - nastavená šírka displeja pri logovaní. Celočíselná hodnota.
2. **Výška okna prehliadača** (*displayHeight*) - nastavená výška displeja pri logovaní. Celočíselná hodnota.
3. **Zachytená udalosť** (*event*) - udalosť zachytená pri logovaní. Refazec môže obsahovať jednu z nasledujúcich hodnôt:
 - (a) Pohyb myši (*mousemove*).
 - (b) Skrolovanie myši (*scroll*),
 - (c) Stlačenie myši (*mousedown*).
 - (d) Uvoľnenie myši (*mouseup*).
 - (e) Stlačenie klávesy (*keydown*).
 - (f) Uvoľnenie klávesy (*keyup*).
 - (g) Klik (*click*).
4. **Identifikátor používateľa** (*userId*) - jedinečný identifikátor.
5. **X-ová súradnica myši** (*xpos*) - súradnica myši v rámci dokumentu. Celočíselná hodnota.
6. **Y-ová súradnica myši** (*ypos*) - súradnica myši v rámci dokumentu. Celočíselná hodnota.
7. **Xpath elementu pod kurzorom** (*xpath*) - xpath elementu vo formáte reťazca.

Tabuľka 9: Príklad surových dát z logera.

display Height	display Width	event	userId	timestamp	xpos	ypos	xpath
4624	1418	mousemove	3c36dbbe-0e2e-4356-a618-8f5077f2b145	2018-07-20 16:26:41.481	1142	321	<xpath>

V nasledujúcom kroku rozdelíme záznamy na časové okná (dĺžky 5 a 10 sekúnd). Pre každé okno extrahujeme pohyby. Vychádzame pritom z prác [15, 16, 27], kde je pohyb definovaný ako sekvencia po sebe idúcich aspoň 4 koordinátov myši, kde časová zmena medzi dvomi po sebe idúcimi koordinátami je menšia ako 120 ms. Je vhodné spomenúť, že situácia kedy hranica okna delí pohyb je vyhodnotená ako dva pohyby, pre každé z okien jeden. Takáto situácia totiž nastala v zanedbateľnom množstve prípadov.

Následne pre každý pohyb vypočítame črty:

1. *Čas*. Celkový čas pohybu myši v milisekundách.

2. *Vzdialenosť.* Vzďialenosť prejdená pohybom v pixeloch.
3. *Rýchlosť.* Priemerná rýchlosť pohybu.
4. *Zrýchlenie.* Priemerné zrýchlenie pohybu.
5. *Hybný moment.* Priemerný hybný moment pohybu.
6. *Počet záznamov.* Počet záznamov o pohybe myši pre daný pohyb.
7. *Zmena času.* Priemerná zmena času medzi záznamami.
8. *Horizontálna rýchlosť.* Rýchlosť voči x-ovej osi.
9. *Vertikálna rýchlosť.* Rýchlosť voči y-ovej osi.

Keďže časové okno môže a nemusí pozostávať z pohybov pridali sme aj nasledujúce črty:

1. *Skroly.* Počet udalostí typu skrol pre dané okno.
2. *Kliky.* Počet udalostí typu klik pre dané okno.
3. *Pohyby.* Počet udalostí pohybov pre dané okno.
4. *Stlačenie klávesy.* Počet udalostí typu stlačenie klávesy pre dané okno.
5. *Uvoľnenie klávesy.* Počet udalostí typu uvoľnenie klávesy pre dané okno.

V prípade klasifikácie nad celým sedením extrahujeme tie isté črty, pričom za okno považujeme celé sedenie. Taktiež na celé sedenie aplikujeme predspracovanie pomocou histogramovej reprezentácie používateľa opísanej v podkapitole 3.3. Ako počet stĺpcov sme zvolili pevnú dĺžku 4. Na výpočet hraničných hodnôt stĺpcov sme použili lineárne rozdelenie rozsahu minima a maxima z rozpätia črt všetkých používateľov. Histogramovú reprezentáciu nemá zmysel aplikovať v prípade predikcie reálnom čase, kde okno správania obsahuje v priemere menej ako 3 pohyby.

5.2 Predspracovanie dát okulografu

V prípade okulografu sú dáta o pohľade veľmi podobné tým z myši. Ide o polohy očí doplnené o údaje o veľkosti zreničky a vzdialenosti od monitora v čase. Hlavný rozdiel oproti myši je v tom, že ide o dvojicu koordinátov pre každé oko zvlášť. Aj v tomto prípade sa jedná o veľké množstvo dát, ktorých objem je podmienený frekvenciou zariadenia.

V prvom kroku predspracovania odfiltrujeme nevalidné dáta [10]. Mnohokrát sa totiž stáva, že okulograf nesníma oči a generuje šum. Celý proces filtrovania pozostáva z identifikácie nevalidných záznamov pomocou validačného kódu zariadenia, ktorý je generovaný pre každý záznam oka zvlášť.

Ďalej je potrebné z dát získať informácie o fixáciách a sakádach. Samotné surové dáta sú totiž len polohy očí v čase. Ako sme uviedli v kapitole 3, pri rýchlych pohyboch oka náš mozog nezaznamenáva obraz. Preto je dôležité získať informáciu o fixáciách a sakádach. Na toto slúžia

fixačné filtre. Existuje viacero druhov fixačných filtrov v závislosti od použitého okulografu. Je preto veľmi dôležité zjednotiť parametre filtra. Následne vypočítame metriky:

1. *Počet fixácií.* Hodnota vyjadruje počet fixácií v rámci časového okna.
2. *Počet sakád.* Hodnota vyjadruje počet sakád v rámci časového okna.
3. *Počet fixácií.* Hodnota vyjadruje počet fixácií v rámci časového okna.
4. *Štatistika dĺžky fixácií.* Hodnoty minima, maxima, priemeru, kvantilov a štandardnej odchýlky dĺžok fixácií pre dané okno.
5. *Štatistika zmeny veľkosti zreničky.* Hodnoty minima, maxima, priemeru, kvantilov a štandardnej odchýlky zmeny veľkosti zreničky oproti priemeru účastníka pre dané okno.
6. *Štatistika vzdialenosti očí od monitora.* Hodnoty minima, maxima, priemeru, kvantilov a štandardnej odchýlky zmeny vzdialenosti očí od monitora pre dané okno.

5.3 Trénovanie modelov a vyhodnotenie

Po extrahovaní metrík natrénujeme model. Pre predikciu porovnáme klasifikátor logistickej regresie a náhodného lesu. Modely otestujeme v dvoch scenároch. V prvom budeme predikovať zmätenie nad dátami z celej úlohy. V tomto kroku porovnáme predikciu s využitím dát z okulografu a z myši. Cieľom je overiť úspešnosť predikcie s využitím jednotlivých metrík a histogramovej reprezentácie.

V druhom kroku pôjde o predikciu v reálnom čase s využitím okien rovnakej veľkosti. Na porovnanie použijeme okná veľkosti 5 a 10 sekúnd podobne ako v práci [29]. V tomto kroku bude potrebné dáta vyvážiť. Je totiž jasné, že ak označíme za zmätené len okná pred klikom na tlačidlo zmätenia a ako nezmätené okná v úlohách kde kliknutie nenastalo, počet zmätených okien bude podstatne nižší. Z tohto dôvodu aplikujeme algoritmus *SMOTE*.

Na vyhodnotenie modelov použijeme metriku presnosti a pokrytia nad predikovanými triedami. Identifikovanie stavu zmätenia bude v produkcii spojené s poskytnutím sprievodcov s cieľom pomôcť používateľovi. Z toho dôvodu je veľmi dôležité predikovať zmätenie s čo najväčšou presnosťou. Na základe týchto metrík na testovacej sade vyberieme najúspešnejší model.

6 Používateľská štúdia a overenie

Pri návrhu štúdie sme riešili hneď niekoľko otázok spojených so získaním spätnej väzby o zmätení používateľa. Týkali sa hlavne toho ako odhaliť presný čas kedy zmätenie nastalo. Okrem týchto otázok sme riešili aj mnoho otázok spojených s funkčnosťou nami implementovanej infraštruktúry na zber dát myši používateľa. Na zodpovedanie týchto otázok a overenie návrhu štúdie sme zostavili pilotnú štúdiu počas prvej časti nášho diplomového projektu. Túto štúdiu opisujeme v prvej podkapitole.

Pilotná štúdia odhalila drobné nedostatky infraštruktúry, ktoré sme okamžite opravili. Všetky získané poznatky sme potom aplikovali pri hlavnej štúdii, ktorá sa uskutočnila v druhej časti nášho diplomového projektu a sumarizujeme ju v druhej podkapitole.

6.1 Pilotná štúdia

Pred návrhom štúdie na overenie našich hypotéz sme zostavili pilotnú používateľskú štúdiu, v rámci ktorej sme mali hneď niekoľko cieľov:

- *Nájdenie vhodného spôsobu získavania implicitnej spätnej väzby.* Naším cieľom bolo navrhnúť metódu, ktorou by sme vedeli identifikovať čas zmätenia používateľa. V tomto prípade sme sa sústredili na odhalenie presného času na základe explicitnej spätnej väzby.
- *Overenie infraštruktúry logovania aktivity myši a klávesnice.* Cieľom tohto kroku bolo overenie implementácie logera priamo na dátach zo správania používateľa. Keďže sme v rámci štúdie využili aj okulograf, dáta z neho poslúžili pri validovaní dát myši.

Celá štúdia prebiehala na fakulte informatiky a informačných technológií Slovenskej technickej univerzity v Bratislave v Centre používateľského zážitku a interakcie (*UXI Centrum*) [6]. Na nahrávanie pohľadu sme použili okulograf *Tobii TX300* s frekvenciou 300Hz a softvér *Tobii Studio*⁵. Celú štúdiu sme realizovali v spolupráci so študentom inžinierskeho štúdia fakulty Bc. Matejom Váľkym. Spolu s ním sme navrhli a realizovali celú štúdiu. Jeho práca sa zameriavala na interakciu s pomocníkmi vo webovej aplikácii z toho dôvodu boli niektoré úlohy prispôbené jeho výskumným otázkam. Na zaznamenávanie pohybu myši sme využili skript jazyka *javascript* vložený priamo do prehliadača *Mozilla Firefox*⁶. Ten generoval udalosti, ktoré posielal na serverovú časť napísanú nad rámcom *Django Rest* jazyka *Python*, ktorý ich ukladal do databázy. V miestnosti boli závesy a osvetlenie, ktoré bolo nastavené na konštantnú hodnotu počas celej štúdie.

Štúdie sa zúčastnili šiesti účastníci z personálu fakulty a študentov, ktorí boli náhodne vybraní. Každý absolvoval sedenie v plnom rozsahu 20 minút, počas ktorého boli v miestnosti moderátor a zapisovateľ. Pred samotným začiatkom experimentu bol každý z účastníkov oboznámený s

⁵www.tobii.com

⁶www.mozilla.org

celým postupom štúdie. Moderátor sedel celú dobu pri účastníkovi. Pred plnením úloh prebehla kalibrácia pohľadu.

Celý experiment prebiehal nad elektronickým obchodom Alza⁷ a pozostával z piatich úloh, ktorých obsah sa týkal vyhľadania ceny špecifického produktu, alebo zodpovedania otázok doménového typu. Kompletné znenie úloh a protokol štúdie nájdete v prílohe A. Po každej úlohe nasledovala jedno-výberová otázka na obtiažnosť úlohy s týmito možnosťami:

- Veľmi ľahká.
- Pomerne ľahká, ale myslím si, že sa dá vyriešiť efektívnejšie.
- Dost ľahká, ale mal/a som drobný problém, čo ma zdržalo.
- Pomerne náročná.
- Veľmi náročná.

Ďalej nasledovala otvorená otázka so znením: Opíšte priebeh úlohy pričom sa zamerajte na problémové časti. Po prejdení všetkých úloh a zodpovedaní poslednej otázky moderátor spolu s účastníkom spoločne prešli záznam z *Tobii Studio*. Účastník mal za úlohu opisovať kroky, ktoré vykonával a zamerať sa na zmätenie. Všetko bolo dokumentované zapisovateľom.

6.1.1 Vyhodnotenie implicitnej spätnej väzby

Pri prechádzaní videí s účastníkmi sme odhalili, že neboli schopní určiť bod, kedy zmätenie začalo. Každý vedel opísať miesto, kde zmätenie bolo, ale nevedeli určiť začiatok. Tieto určenia boli v zhode s otázkou ohľadom obtiažnosti úlohy, to znamená, že vždy keď účastník spätne slovné komentoval prítomnosť zmätenia počas vyhodnotenia videa tak tú istú úlohu označil ako náročnú/pomerne náročnú. *Pearsonov korelačný koeficient* v tomto prípade dosiahol hodnotu 0,6 na 30 vzorkách.

Záverom je zistenie, že používatelia nie sú schopní spätne anotovať začiatok stavu zmätenia. Avšak sú schopní spätne označiť úlohu, kde zmätenie nastalo.

6.1.2 Vyhodnotenie infraštruktúry logovania

V tejto časti sme prechádzali záznamy zozbierané pomocou našej implementácie logera. Sústredili sme sa na úspešnosť zaznamenávania našej infraštruktúry. Spätne sme modelovali pozície kurzora myši v čase pričom sme zmeny sledovali spoločne s video záznamom účastníkov. V tomto prípade šlo len o vizuálnu kontrolu a žiadne formálne výpočty presnosti neprebehli.

Záverom tejto časti bolo zistenie, že pomocou našej infraštruktúry sme schopní zaznamenávať reálne správanie myši používateľa. Odhalili sme problém spojený s nedoposielaním údajov o správaní pri tvrdej navigácii prehliadača (angl. *hard navigation*). Tento nedostatok sme napravili.

⁷www.alza.sk

6.2 Hlavná štúdia

Štúdia sa uskutočnila na Fakulte informatiky a informačných technológií Slovenskej technickej univerzity v Bratislave v laboratóriu používateľského zážitku pre skupinovú štúdiu UXI sentra (UX Class) [6]. Zmena laboratória bola podmienená typom štúdie, na rozdiel od pilotnej štúdie teraz išlo o kvantitatívnu štúdiu a vybraná miestnosť disponuje dvadsiatimi počítačmi vybavenými okulografom. Štúdie sa zúčastnilo 59 študentov fakulty, ktorí boli rozdelení do štyroch skupín, kde dve skupiny mali sedenie ráno od deviatej a dve vo večerných hodinách od siedmej. Študenti prišli na štúdiu dobrovoľne a každý z nich podpísal informovaný súhlas o spracovaní osobných údajov v zhode so zákonmi Slovenskej republiky.

Štúdia trvala dve hodiny a pozostávala zo šiestich úloh na dovolenkovom portáli *FiroTour*⁸. Zmena portálu štúdie bola spôsobená tým, že v dobe štúdie sme úzko spolupracovali s firmou, ktorá portál *FiroTour* vlastní. Tým pádom sme mali priamo prístup k zdrojovému kódu aplikácie, kde sme mohli náš logger jednoducho integrovať. Celý priebeh dokumentuje protokol štúdie v prílohe A. Na zaznamenávanie pohľadu sme použili okulograf *Tobii X2-60*, s frekvenciou 60Hz. Na zaznamenávanie pohybov myši a interakciu s klávesnicou sme použili náš logger. Účastníci po celú dobu interagovali s produkčnou verziou dovolenkového portálu (žiadne reklamy neboli odstránené).

Priebeh štúdie bol vytvorený v softvéri *iMotions*⁹. Ide o softvér na vytváranie používateľských štúdií s využitím okulografu. Dôvod zmeny softvéru na štúdiu bol v návrhu štúdie, ktorého súčasťou boli otvorené otázky po každej z úloh. Pôvodný softvér *Tobii Studio* neposkytoval možnosť jednoducho zakomponovať takýto druh otázky. Celá štúdia v *iMotions* bola tvorená dvomi časťami. V prvej prebehla kalibrácia okulografu, ktorú tvorila štandardná kalibračná sekvencia okulografu a kalibračná sekvencia zreničky účastníka. Druhá časť pozostávala z plnenia šiestich úloh. V *iMotions* sme vytvorili 6 blokov, ktoré tvorili hlavnú sekvenciu. Každý z blokov pozostával z troch stimulov:

1. *Stimul s inštrukciami*. Štandardný text so znením úlohy. Na konci textu bolo vždy spomenuté tlačidlo f10, ktorým sa dalo medzi jednotlivými časťami štúdie prechádzať. Zvolili sme biely text na sivom pozadí.
2. *Stimul s url*. Po prechode na tento stimul sa účastníkovi sám otvoril prehliadač a bol presmerovaný na url, kde začínala aktuálna úloha. Ako prehliadač sme zvolili *Google Chrome*¹⁰. Dôvod zmeny prehliadača bol spojený s implementáciou logera, ktorá pracuje spoľahlivejšie práve na prehliadači *Google Chrome*. Používateľ mal po dokončení stlačiť tlačidlo f10 na prechod do tretej časti bloku.
3. *Stimul s dotazníkom*. Posledná časť bloku bola tvorená stimulom s tromi otázkami. Prvá bola určená na odpoveď na úlohu. Otázka plnila validačnú funkciu pri neskoršom vyhodnotení štúdie. Druhá bola jedno-výberová otázka o stlačení tlačidla počas úlohy. Opäť úloha

⁸www.frotour.sk/

⁹www.imotions.com

¹⁰www.google.com/chrome/

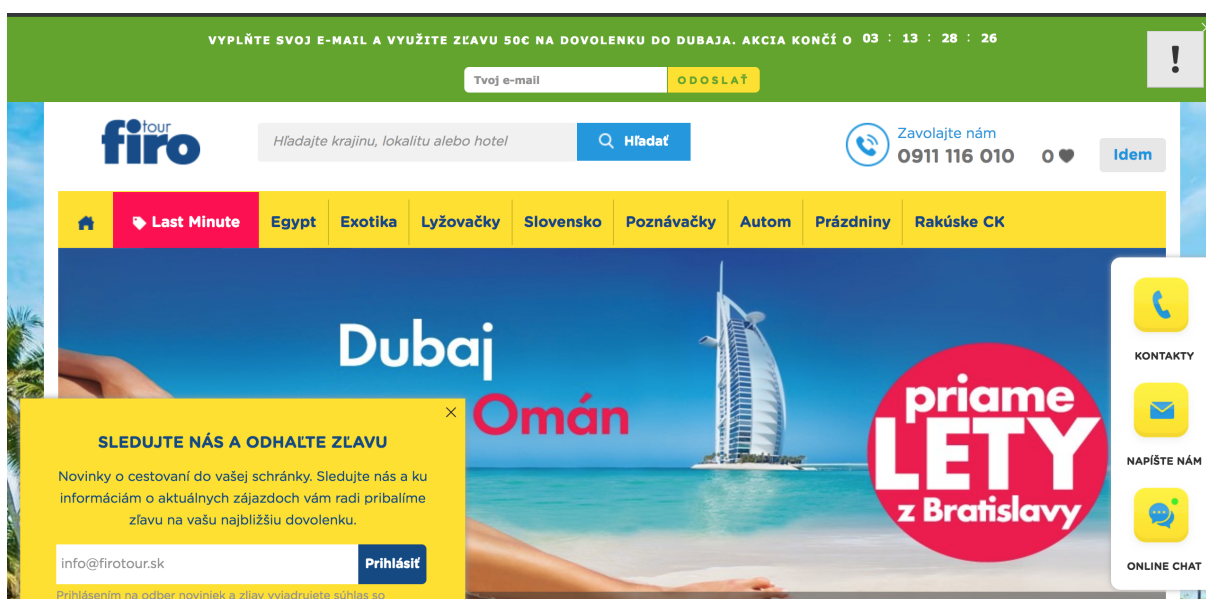
s validačnou funkciou, či používateľ naozaj stlačil tlačidlo. Posledná otvorená otázka slúžila na opis úlohy a opis problémov, ktoré počas riešenia nastali. Po dokončení bolo opäť potrebné stlačiť f10.

Softvér *iMotions* umožňuje náhodné prerozdelenie blokov v rámci spustenej sekvencie. Vďaka tomu bolo zaručené, že každý účastník mal úlohy v náhodnom poradí. Výber úloh sme riešili priamo s cestovnou kanceláriou *Firo Tour*. Stretli sme sa so zamestnancami zákazníckeho servisu, ktorí nás informovali o najčastejších problémov aplikácii.

6.2.1 Získavanie spätnej väzby

Kľúčovou časťou používateľskej štúdie bolo zachytávanie spätnej väzby používateľa. Bolo potrebné vystihnúť správny moment, kedy bol používateľ zmätený. Počas pilotnej štúdie sme na určovanie použili dotazník po každej úlohe, kde sme sa pýtali na problémy s úlohou. Tento postup sa však ukázal ako neefektívny. Účastníci mnohokrát zabudli na problémy, ktoré počas úlohy vznikli, a teda ich nespomenuli v dotazníku po úlohe. Navyše sme zistili, že účastníci nie sú schopní spätne odhadnúť začiatok zmätenia.

V hlavnej štúdii sme navrhli a implementovali softvérové tlačidlo do pravého horného rohu stránky. Vychádzali sme pritom z práce [28], ktorú sme analyzovali v kapitole 2.3. Namiesto textu autorov "som zmätený" (angl. *I am confused*) sme však použili symbol "!". Dôvodom bolo nevyvolávať pocit chyby na strane používateľa. Tlačidlo bolo súčasťou logera. Po kliknutí naň podobne ako v citovanej práci nenastala žiadna spätná väzba pre používateľa. Namiesto toho sa odoslala udalosť na server o stlačení, ktorá bola uložená do databázy. Obrázok 4 poskytuje ukážku umiestnenia a dizajnu tlačidla.



Obr. 4: Ukážka použitia tlačidla na získavanie spätnej väzby zmätenia. Tlačidlo sa nachádza v pravom hornom rohu a je označené symbolom "!".

Tlačidlo sme opísali nasledujúcim textom: "Tlačidlo "!": V pravej hornej časti obrazovky prehliadača uvidíte tlačidlo s textom "!". Prosíme vás o to, aby ste na toto tlačidlo klikli vždy vtedy, keď budete potrebovať pomoc s úlohou. Tlačidlo má simulovať stav, kedy by ste chceli stránku opustiť z dôvodu, že by ste nevedeli kde hľadať potrebnú informáciu na splnenie úlohy. Taktiež je na toto tlačidlo vhodné kliknúť vtedy, ak by ste chceli na splnenie úlohy použiť inú aplikáciu ako <doména> (napríklad google). Po stlačení tlačidla nenastane žiadna udalosť zo strany systému. Následne môžete v úlohe pokračovať alebo prejsť ďalej tlačidlom F10."

6.2.2 Mapovanie identít účastníka

Pri experimente sme pracovali s dvomi hlavnými zdrojmi dát o správaní používateľa. Prvým z nich bol okulograf. Identita pri dátach účastníka v tomto prípade bola riešená cez *iMotions*, kde pred začatím štúdie bolo potrebné uviesť identifikátor. Následne uvedený identifikátor bol priradený každému záznamu okulografu. Problém však bol v prípade loggeru myši na stránke FiroTour. Logger nemohol vedieť o identite používateľa, ktorý na ňom pracoval. Pri štúdiu sme totiž použili nami vyvinutú produkčnú verziu skriptu na logovanie dát, ktorá logovala všetkých používateľov na stránke dovolenkového portálu bez nutnosti identity, len na základe *visitorId*. Tento identifikátor bol vygenerovaný pri prvej návšteve a uložený do lokálnej pamäte prehliadača (angl. *local storage*).

Na vyriešenie situácie sme použili mapovanie ip adresy počítačov v učebni, v ktorej štúdia prebiehala, na identifikátor v štúdiu. Počítač si teda vždy pri prvom navštívení url vypýtal identifikátor z externej služby. Následne tento identifikátor pridal do každého záznamu o používaní myši. Ako externú službu sme využili *Bridge*¹¹. Ako identifikátor sme zvolili 6-členný reťazec písmen a čísel. Jeho aplikáciou sme prispeli k anonymizovaniu dát.

¹¹www.bridge.buddyweb.fr/en

6.3 Exploratívna analýza surových dát

6.3.1 Získané dáta

Každý z 59 účastníkov úspešne absolvoval všetkých 6 úloh štúdie. Menšie technické komplikácie sa prejavili pri práci so softvérom, určeným na priebeh štúdie. Z dôvodu týchto komplikácií sme prišli o odpovede po úlohách u troch účastníkov. To znamená, že celkovo sme zozbierali 336 odpovedí na dotazníky po úlohe.

Pri zaznamenávaní správania myši používateľa sme odhalili menšie technické problémy. Tie sa prejavili pri rozlišovaní úlohy, na ktorej používateľ práve bol. Rozlišovanie úloh sme riešili skrytým identifikátorom na úvodnej url úlohy. Pri 7 úlohách sa však stalo, že sa účastník preklikol na podstránku bez identifikátora skôr ako mohla byť informácia so skrytým identifikátorom zaznamenaná na serveri. Z tohto dôvodu sme pri týchto účastníkoch neboli schopní zrekonštruovať poradie úloh a s istotou priradiť záznamy myši k jednotlivým úlohám. Vyriešili sme to vylúčením týchto účastníkov zo vzorky.

Pri dotazníku na konci každej úlohy sme sa účastníkov pýtali na to, či počas riešenia stlačili tlačidlo '!'. Táto otázka plnila úlohu spätnej kontroly tlačidla. Porovnaním prijatej udalosti na strane serveru a odpoveďou na túto otázku sme odhalili technické problémy pri 22 úlohách datasetu. Vždy sa jednalo o problém, kedy účastník v dotazníku označil, že tlačidlo stlačil avšak na server táto informácia nedošla. Nenašli sme žiadny prípad, kedy by to bolo naopak. Následne sme sa pozreli na príčinu týchto nedostatkov. Jedno z možných vysvetlení je omyl na strane účastníka, keďže niektoré úlohy boli veľmi podobné. Iný dôvod problému mohol byť na strane serveru, ktorý túto informáciu zle spracoval. Pre korektnosť výsledkov sme tieto úlohy vyradili z datasetu.

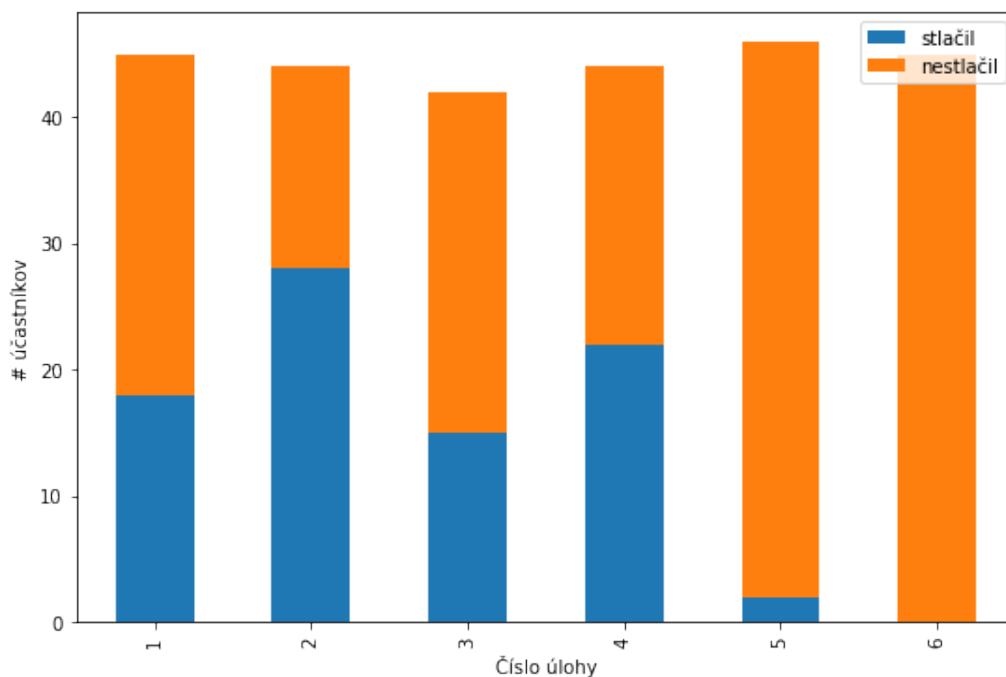
Po aplikácii spomenutých filtrov sme získali dataset o veľkosti 266 záznamov o riešení. Rozdelenie počtov záznamov medzi úlohy reprezentuje tabuľka 10.

Tabuľka 10: Prehľad počtov záznamov jednotlivých úloh v očistenom datasete.

Číslo úlohy	1	2	3	4	5	6
Počet záznamov	45	44	42	44	46	45

6.3.2 Tlačidlo zmätenia

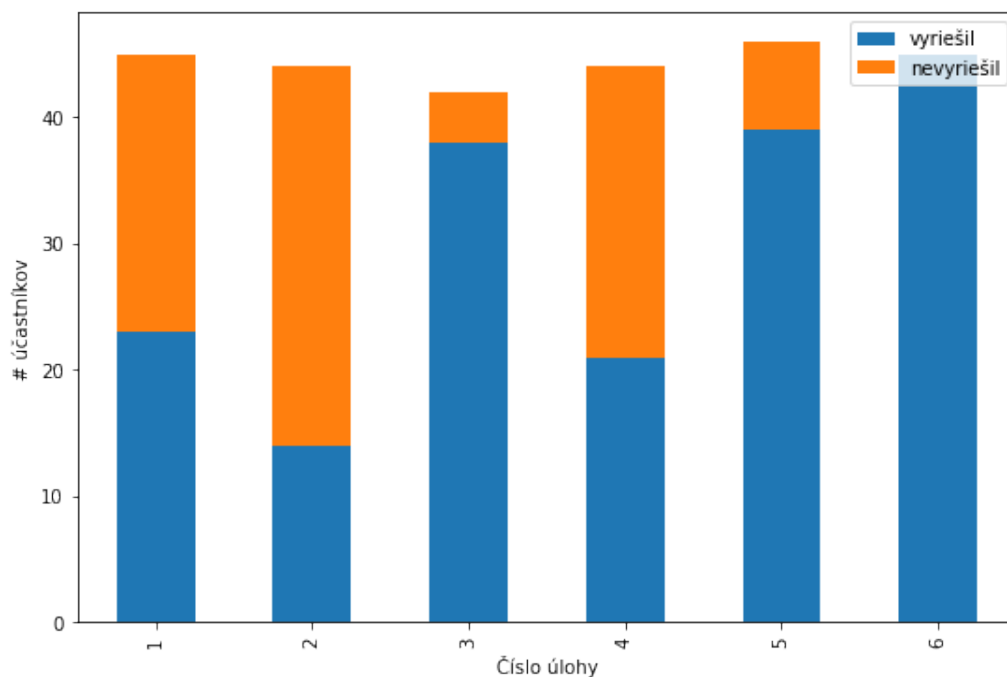
Po dôkladnom čistení datasetu sme analyzovali zmätenie používateľa. Pri návrhu úloh sme dbali na to, aby nazbierané dáta preukázali výskyt zmätenia v úlohách. Obrázok 5 ukazuje na počty stlačenia tlačidla '! ' pri jednotlivých úlohách.



Obr. 5: Graf rozdelenia stlačenia tlačidla '?' pre jednotlivé úlohy.

Na grafe si môžeme všimnúť, že s úlohami 1 až 4 mali účastníci značné problémy. Úloha 2 bola dokonca tak náročná, že viac ako polovica účastníkov stlačila tlačidlo '?' počas jej riešenia. Na druhej strane tu máme úlohy 5 a 6, kde účastníci nemali skoro žiadne problémy pri riešení. Išlo o úlohu jednoduchého vyhľadania informácie, kde stačilo prejsť na inú podstránku a informáciu vyhľadať. Tieto úlohy boli vybrané s cieľom kontrolnej vzorky správania, kedy používateľ nemá žiaden problém.

V ďalšom kroku sme analyzovali odpovede na úlohy. Po každej úlohe nasledoval dotazník, kde účastníci mohli napísať odpoveď na úlohu. Ručne sme anotovali tieto odpovede. Pri anotácii sme sa sústredili na to, či účastník úlohu vyriešil. Teda nešlo nám o správnosť odpovede na úlohu. Za nevyriešené sme brali iba prípady, kedy pole na odpoveď zostalo nevyplnené alebo iné prípady, kedy účastník explicitne vyjadril, že úlohu nevie splniť alebo informácia, ktorú tam napísal nesúvisela s úlohou. Obrázok 6 ukazuje na počty vyriešených záznamov pri jednotlivých úlohách.



Obr. 6: Graf rozdelenia vyriešených záznamov pri jednotlivých úlohách.

Na grafe si môžeme všimnúť očakávané výsledky v prípade úlohy 6, kedy všetci tí čo nestlačili tlačidlo '!' zároveň úlohu vyriešili. Na ostatných grafoch mohol nastať jeden z nasledujúcich javov:

1. Účastník stlačil tlačidlo '!' a úlohu nevyriešil. Toto je očakávaný jav.
2. Účastník nestlačil tlačidlo '!' a úlohu vyriešil. Toto je opäť očakávaný jav.
3. Účastník stlačil tlačidlo '!' a úlohu vyriešil. Toto je jav kedy používateľ, potom čo stlačil tlačidlo "!" úlohu nevzdal a pokračoval v nej.
4. Účastník nestlačil tlačidlo '!' a úlohu nevyriešil. Toto je neočakávaný stav v správaní. Jedným z vysvetlení je, že účastník zabudol kliknúť na tlačidlo pred tým ako sa prepol na inú úlohu. Prípadne, že zmätenie nenastalo, účastník jednoducho úlohu nevedel a neriešil to.

Tabuľka 11 poskytuje početnosti jednotlivých stavov v datasete. Zamerali sme sa na problematické prípady, kedy účastník nestlačil tlačidlo a zároveň nenašiel informáciu k úlohe. Keďže po každej úlohe bol dotazník, kde účastníci mohli opísať priebeh riešenia úlohy, prešli sme tieto odpovede a zistili sme, že vo väčšine prípadoch šlo o to, že používateľ si pomýlil úlohu alebo len nič nenapísal.

Tabuľka 11: Prehľad stavov účastníkov vzhľadom k stlačeniu tlačidla a vyriešeniu úlohy.

Stav	Stlačil a nevyriešil	Nestlačil a vyriešil	Stlačil a vyriešil	Nestlačil a nevyriešil
Počet záznamov	69	154	26	27

6.4 Exploratívna analýza metrík

Po preskúmaní získanej explicitnej spätnej väzby používateľa sme generovali metríky myši pre predikciu. V prvom kroku sme pripravovali dáta na predikciu nad celou úlohou. Tu sme zobrali dáta z úlohy a v prípade, že počas riešenia nastalo stlačenie tlačidla označili sme ich ako zmätené, ak nenastalo označili sme ich ako nezmätené. V tomto kroku sme aj napárovali dáta z okulografu pre jednotlivé úlohy. Pre zaujímavosť spomenieme, že v dátach sme našli aj 4 prípady, kedy bolo tlačidlo zmätenia stlačené viac ako jeden krát počas úlohy (časová vzdialenosť medzi kliknutiami bola väčšia ako 3 sekundy, teda nešlo o dvojklik).

V druhom kroku sme dáta rozdelili na časové okná rovnakej veľkosti s sekúnd. Pri generovaní okien sme ako zmätené označili len okná, ktoré predchádzali prvej udalosti stlačenia tlačidla v úlohe. Na generovanie nezmätených okien sme použili len úlohy, kde stlačenie tlačidla nenastalo. Takéto okná sme vytvorili pomocou už spomínaného algoritmu posúvajúceho sa okna, pričom okno sa vždy posúvalo o jednu sekundu. Pri overovaní predikcie v reálnom čase sme zvolili okná veľkosti 5 sekúnd a 10 sekúnd podobne ako v spomínaných prácach [29, 28].

Pokračovali sme extrakciou pohybov myši z dát. V oboch prípadoch (pri predikcii nad celou úlohou aj pri predikcii v reálnom čase) sme koordináty zoskupili do pohybov. Pohyb je definovaný ako súbor minimálne 4 súradníc, kde rozdiel časov medzi súradnicami je menší ako 120 ms [15]. Pri predikcii v reálnom čase sme vypočítali metríky uvedené v návrhu metódy 5.1. Pri predikcii nad celým sedením sme pre každú z črt myši vypočítali histogram jej hodnôt pre pohyby. Histogramy sme opisovali v podkapitole 3.3. Stanovili sme pevný počet štyroch stĺpcov pre každú črtu a hranice sme určili na základe lineárneho rozdelenia minima a maxima hodnôt danej premennej z celého datasetu. To znamená, že ak by sa časy pohybov pre všetky pozorovania pohybovali od 0 do 100, hraničné hodnoty histogramových stĺpcov by boli nasledovné $\langle 0;25 \rangle$, $\langle 25;50 \rangle$, $\langle 50;75 \rangle$, $\langle 75;100 \rangle$. V prípade 5 a 10 sekundových okien takéto spracovanie nemá zmysel, keďže okná obsahujú priemerne menej ako 3 pohyby.

V práci porovnávame našu metódu predikcie nad celou úlohou s už spomínanou prácou [42], kde autori predikovali obtiažnosť sedenia na základe logov *HTTP* serveru. Na to aby sme porovnanie zrealizovali bolo potrebné vypočítať črty zhrnuté v podkapitole 3.5. Keďže sme po štúdií nemali prístup k *HTTP* logom vypočítali sme tieto metríky priamo z dát myši. Logy myši totiž v sebe niesli informáciu o url, na ktorej daná udalosť nastala. Vďaka tomu sa nám podarilo vypočítať metríky celkový čas, počet stránok, čas na stránku a počet cyklov. Naša infraštruktúra však neposkytovala informácie o návšteve iných stránok z toho dôvodu sme nevedeli vypočítať metriku počet zámen. Tieto metríky boli vypočítané len pre predikciu nad celou úlohou, keďže nemalo zmysel ich rátať pre okná malých veľkostí.

Pokračovali sme spracovaním metrík okulografu. Väčšiu časť predspravovania z metódy za nás vyriešila práve zvolená infraštruktúra *iMotions*. Vygenerovaný export obsahoval informácie o sakádach a fixáciách pre každý záznam. Softvér za nás aj odfiltroval nevalidné záznamy. V rámci generácie črt bolo potrebné dopočítať črty zreničky a vzdialenosti od monitora. Tieto metriky boli vypočítané len pre predikciu nad celou úlohou, kvôli technickým komplikáciám pri mapovaní okien okulografu na okná myši.

6.4.1 Metriky myši

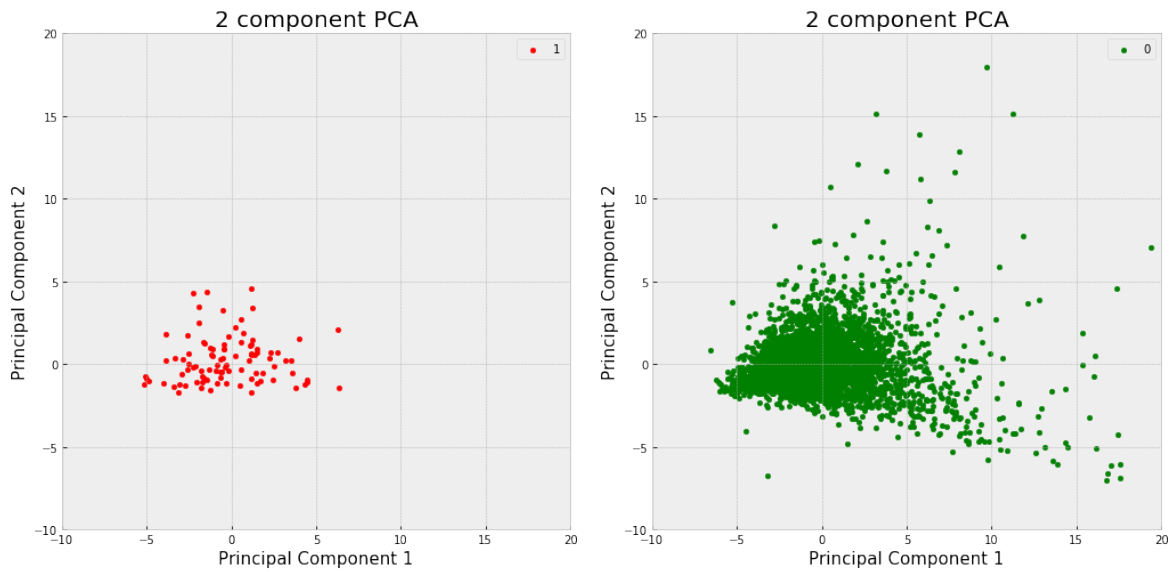
V prvom kroku sme sledovali koreláciu každej z vypočítaných metrík voči zmäteniu v danom okne (voči celému sedeniu v prípade datasetu určenom na predikciu nad celým sedením). V prípade okien veľkosti 5 a 10 sekúnd sme nenašli žiadnu značnú koreláciu s využitím *Pearsonovho koerelačného testu*. Všetky korelácie boli menšie ako 0,1 najvyššiu koreláciu dosahovali metriky rýchlosti a štandardizovanej rýchlosti (hodnoty okolo 0,04). Z týchto výsledkov sa dalo usúdiť, že v dátach nie je žiadna značná lineárna závislosť voči predikovanému stĺpcu. Toto tvrdenie sme sa snažili potvrdiť aj pomocou analýzy principiálnych komponentov (angl. *principal component analysis*), ďalej iba *PCA*. Pomocou *PCA* sme redukovali dimenzionalitu datasetu len na dva principiálne komponenty. Následne sme dáta vizualizovali pomocou grafov v dvojrozmernom priestore.

Obrázok 7 poskytuje ukážku vizualizácie pre okná veľkosti 10 sekúnd. Môžeme si na ňom všimnúť, že zmätené pozorovania (červenou) sú centrované okolo bodu $[0,0]$ podobne ako nezmätené. Jediný rozdiel je v rozptyle pozorovaní, ktorý je pri nezmätených vzorkách väčší. Taktiež z grafu vidieť, že ani po redukcii nejde o lineárne separovateľný problém. Podobnú vizualizáciu sme dostali aj v prípade okien veľkosti 5 sekúnd a pri predikcii nad celou úlohou. Je vhodné spomenúť, že samotnému kroku redukcii predchádzala analýza normálneho rozdelenia jednotlivých metrík. Tá zahŕňala vizualizáciu histogramov jednotlivých metrík. Každý zo stĺpcov spĺňal predpoklady normálneho rozdelenia. Každú z metrík sme normalizovali s využitím štandardného škálovania.

Pri predikcii nad celou úlohou s využitím histogramovej reprezentácie črt sme zaznamenali vyššie korelácie a to v prípade zrýchlenia, kde *Pearsonov koerelačný koeficient* dosiahol hodnotu viac ako 0,18. Zaujímavý výsledok korelácie sme dosiahli v prípade HTTP metriky počet cyklov nad celou úlohou, kde táto metrika dosahovala hodnotu koeficientu až 0,46. Taktiež metrika počtu stránok dosiahla hodnotu 0,41. Na druhej strane metrika času na úlohe dosiahla hodnotu len 0,09 z čoho vyplýva, že zmätenie nekorelovalo s dĺžkou riešenia úlohy.

Pokračovali sme zhotovením jednoduchých prediktorov nad jednotlivými HTTP metrikami. Šlo o klasifikátor logistickej regresie, ktorý predikoval zmätenie vždy len na základe jednej z metrík. Podobný prístup zvolili autori práce [42]. Cieľom týchto prediktorov bolo odsledovať hraničné hodnoty, ktoré sa klasifikátory na dátach naučia.

V tabuľke 12 môžeme vidieť výsledky. Zaujímavý je posledný stĺpec, ktorý slovne interpretuje naučené hranice. Je vidieť, že v prípade počtu cyklov, ak nastali viac ako 2, prediktor sa naučil, že ide o stav zmätenia. Podobný jav nastal aj v prípade počtu stránok.



Obr. 7: Vizualizácia datasetu vzoriek nad 10 sekundovými oknami pomocou redukcie dimenzionality s využitím PCA nad črtami myši. Zmätené pozorovania sú označené červenou a nezmätené pozorovania zelenou

Tabuľka 12: Jednoduché prediktory s využitím individuálnych črt HTTP serveru pri predikcii nad dátami z celej úlohy.

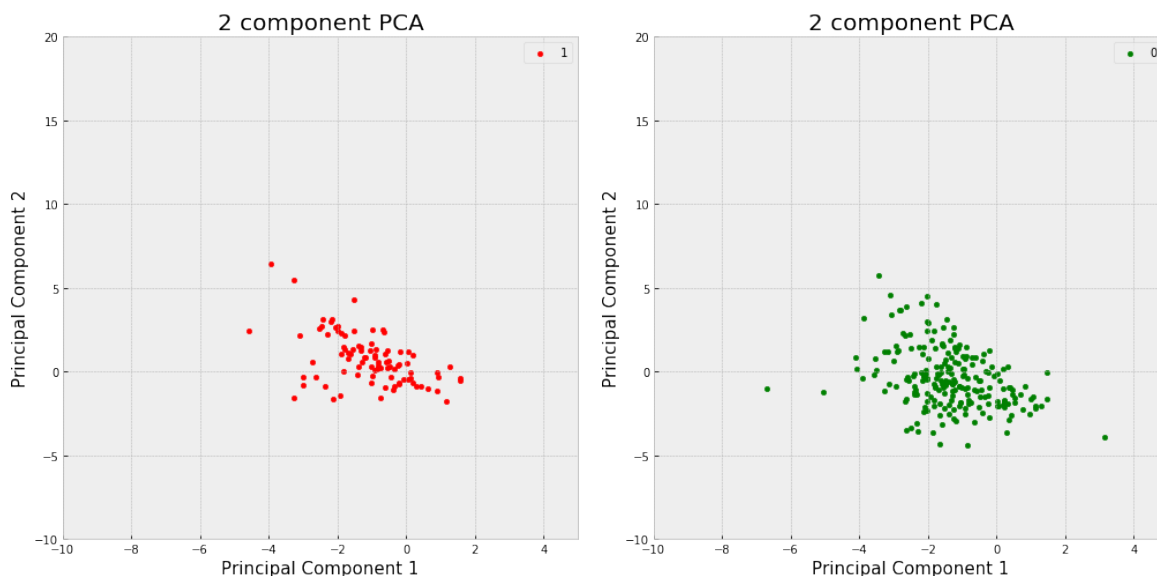
Črta	Presnosť	Úplnosť	AUC	AIC	Naučená hranica
Celkový čas	N/A	0	0.5	258.82	Nikdy nepredikuje zmätenie
Počet stránok	0.71	0.46	0.68	258.74	>9 stránok
Čas na stránku	N/A	0	0.5	258.82	Nikdy nepredikuje zmätenie
Počet cyklov	0.77	0.38	0.66	258.45	>2 cykly
Najlepšia kombinácia	0.72	0.50	0.7	226.43	

6.4.2 Metriky okulografu

Kvôli problémom s mapovaním dát z okulografu na okná veľkosti 5 a 10 sekúnd sme tieto metriky využili iba pri predikcii nad dátami z celej úlohy. Navyše cieľom tejto práce je navrhnúť metódu na predikciu v reálnom čase s využitím v produkčnom prostredí. Keďže okulograf nie je dnes ešte bežnou súčasťou hardvérového vybavenia návštevníkov webových aplikácií, využitie len dát z myši v produkčnom prostredí, t.j. pri predikcii zmätenia v reálnom čase, je naším úmyslom.

V prípade metrick okulografu sme postupovali rovnako ako pri metrikách myši. V prvom kroku sme sa pozreli na normálne rozdelenie metrick. Opäť aj v tomto prípade už samotný vizuálny test toto rozdelenie potvrdzoval. V nasledujúcom kroku sme sledovali koreláciu voči zmäteniu. Najvyššiu hodnotu 0,33 dosiahla metrika počtu fixácií. Všetky ostatné metriky mali hodnoty podstatne nižšie. Pre zaujímavosť sme skúsili opäť vizualizáciu v dvojrozmernom priestore pomocou redukcie dimenzionality algoritmom *PCA*. Obrázok 8 poskytuje prehľad, na ktorom si

môžeme všimnúť, že po aplikovaní algoritmu nejde o lineárne separovateľný problém.



Obr. 8: Vizualizácia datasetu vzoriek nad celou úlohou pomocou redukcie dimenzi-
onalita s využitím PCA nad črtami okulografu. Zmätené pozorovania sú označené
červenou a nezmätené pozorovania zelenou.

6.5 Výsledky predikcie

Všetky výpočty a vizualizácie sme realizovali pomocou knižnic jazyka *python* v prostredí *jupyter lab*¹². Na prácu s dátami sme využili knižnice *pandas*¹³ a *numpy*¹⁴. Na vizualizáciu dát sme použili knižnice *matplotlib*¹⁵ a *seaborn*¹⁶. Na trénovanie predikciu a výpočet metrík pri overovaní sme použili knižnicu *scikit-learn*¹⁷ a na balancovanie datasetu sme použili knižnicu *imblearn*¹⁸. Dáta z logeru sú uchovávané v databáze *elasticsearch*¹⁹, zatiaľ čo dáta z okulografu sme uchovali vo formáte súborov hodnôt oddelených čiarkou (angl. *comma separated values*).

Pri predikcii sme v prvom kroku dáta rozdelili s využitím vrstviacej troj-skupinovej krížovej validácie (angl. *stratified 3-fold cross validation*). V rámci každej skupiny boli dáta rozdelené v pomere 80 percent objemu pre trénovaciu sadu ku 20 percentám pre testovaciu. Následne pre každú trénovaciu sadu sme natrénovali klasifikátor s využitím prehľadávania mriežky (angl. *grid search*), ktorým sme sa snažili nájsť optimálne parametre pri maximalizovaní metriky presnosť. V rámci prehľadávania mriežky sme opäť použili vrstviacu troj-skupinovú krížovú validáciu. Vybrali sme vždy optimálnu možnosť mriežky, ktorej výsledky predikcie, nad testovacou sadou,

¹²www.jupyterlab.readthedocs.io/en/stable/index.html

¹³www.pandas.pydata.org/

¹⁴www.numpy.org/

¹⁵www.matplotlib.org/

¹⁶www.seaborn.pydata.org/

¹⁷www.scikit-learn.org/stable/

¹⁸www.imbalanced-learn.readthedocs.io/en/stable/index.html

¹⁹www.elastic.co/

sme zapísali do spoločného pola. Toto sme opakovali pre každú skupinu. Na konci sme vypočítali metriky nad týmto polom predikcií.

6.5.1 Výsledky nad celou úlohou

Ako prvé poskytujeme výsledky predikcie nad dátami z celej úlohy. Dôvod výberu tohto scenáru bolo porovnanie s existujúcim prístupom uvedeným v práci [42] na predikciu obtiažnych sedení na stránke. Pri predikcii sme využili histogramovú reprezentáciu základných metrík myši, metriky *HTTP* serveru odvodené z metrík myši a metriky okulografu.

Tabuľka 13 poskytuje výsledky rôznych experimentov. Môžeme si všimnúť pomerne dobré výsledky v prípade predikcie s využitím metrík *HTTP* serveru. Vysoká presnosť spolu s vysokou hodnotou *AUC* svedčia v prospech tohto výsledku. Potvrdila sa vysoká korelácia týchto metrík z predchádzajúcej podkapitoly. Ďalej je zaujímavé porovnanie výsledkov metrík myši a okulografu. Ako si môžeme všimnúť metriky myši majú len o niečo horšie výsledky.

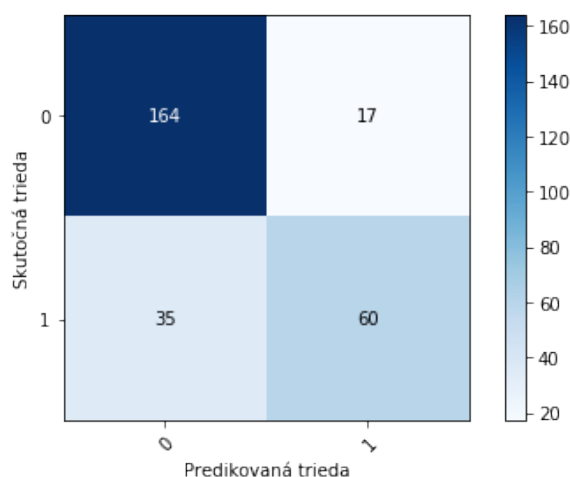
V experimentoch sme pokračovali ďalej a to s využitím metrík myši a *HTTP* serveru. Výber týchto metrík bol podmienený využitím v reálnej produkcii, keď okulograf ako taký momentálne nie je štandardným hardvérovým zariadením počítačov. Tu sme dosiahli najlepšie výsledky s presnosťou až 0,82. Taktiež je vhodné vyzdvihnúť vysokú úplnosť s hodnotou až 0,83.

V tabuľke 13 taktiež poskytujeme výsledky v prípade predikcie s využitím vyvažovania datasetu pomocou *SMOTE* algoritmu [1]. Šlo o nastavenie, kedy sa počet zmätených pozorovaní vyrovnal počtu nezamätených. Môžeme sledovať mierne zlepšenie v prípade predikcie nad dátami *HTTP* serveru. V ostatných prípadoch zlepšenie nenastalo alebo nastalo zhoršenie. Pre úplnosť uvádzame maticu zmätenia pre najlepší z klasifikátorov na obrázku 10. Najväčším problémom je 29 mylne klasifikovaných pozorovaní, kedy boli zmätené pozorovania označené ako nezamätené. Pri reálnom využití prediktora toto môže spôsobovať problém, kedy sa nepodarí odhaliť používateľov s problémami.

Záver tohto scenára je taký, že s využitím dát z myši a *HTTP* serveru sme schopní predikovať zmätenie na celej úlohe s vysokou presnosťou a úplnosťou. Takto natrénovaný klasifikátor sa dá využiť pri spätnej analýze sedení s cieľom odhaliť problémové sedenia, prípadne problémy použiteľnosti v aplikácii. Rovnaký postup predikcie obtiažnych sedení bol použitý v spomínanom článku [42], kde autori dosiahli úspešnosť 0.86 na dátach z *HTTP* serveru.

Tabuľka 13: Porovnanie predikcie pomocou rôznych kombinácií metrík s využitím algoritmu Logistickej regresie. Stĺpce označené slovom *SMOTE* prezentujú výsledky klasifikácie pri použití vyvažovania algoritmom SMOTE [1]. Výsledky boli získané aplikáciou vrstviacej troj-skupinovej krížovej validácie.

Črta	Presnosť	Presnosť (SMOTE)	Úplnosť	Úplnosť (SMOTE)	AUC	AIC
Najlepšia kombinácia metrik HTTP serveru	0.72	0.76	0.71	0.74	0.7	226.43
Metriky myši	0.59	0.59	0.63	0.58	0.58	268.86
Metriky okulografu	0.69	0.74	0.70	0.73	0.59	174.03
Najlepšia kombinácia bez okulografu	0.82	0.82	0.83	0.82	0.86	207.36
Všetky metriky	0.82	0.81	0.83	0.81	0.88	240.41



Obr. 9: Matica zmätenia pri klasifikátore Logistickej regresie pri predikcii zmätenia nad dátami z celej úlohy s využitím vrstviacej troj-skupinovej krížovej validácie (angl. *stratified 3-fold cross validation*). Pri tréningu a predikcii sa použili metriky odvodené z myši a z HTTP serveru. Zmätenia má označenie "1".

6.5.2 Výsledky nad časovými oknami

V nasledujúcom scenári sme predikovali v reálnom čase s využitím časových okien rôznej veľkosti. Kvôli časovej zložitosti výpočtu prezentujeme výsledky okien veľkosti 10 a 5 sekúnd. Pri predikcii sme využili črty dokumentované v kapitole 5.1.

Počas predikcie sme riešili problém s nevyváženým datasetom. Zmätených pozorovaní bolo podstatne menej ako tých nezmätených. V prípade okien veľkosti 10 sekúnd bolo nezmätených pozorovaní viac ako 36 krát viac a v prípade okien veľkosti 5 to bolo viac ako 50 krát viac. Ako riešenie tohto problému sme využili algoritmus *SMOTE* [1], určený na vyvažovanie datasetu. Použili sme implementáciu vyvažovania na základe piatich najbližších susedov. Pričom sme zvolili hneď niekoľko nastavení pri navyšovaní minoritnej triedy a to o 200, 500 a 2000 percent.

Podobné experimenty využili autori v práci [29].

V tabulke 14 prezentujeme výsledky predikcie na oknách veľkosti 10 sekúnd a v tabulke 15 výsledky pre okná veľkosti 5 sekúnd. Ako klasifikátor sme použili logistickú regresiu (LR), avšak pre porovnanie poskytujeme aj prehľad výsledkov pre klasifikátor náhodného lesu (angl. *Random forrest*, *RF*). V jednotlivých riadkoch si môžeme tiež všimnúť výsledky pre rôzne nastavenia vyvažovania (označenie SMOTE). Pri tréňovaní prediktorov pomocou prehľadávania mriežky sme sa sústredili na maximalizovanie presnosti. Dôvod bol taký, že v produkcii sa sústredíme na to, aby sme si boli čo najistejší, keď označíme niekoho za zmäteného. Chceme sa vyvarovať zbytočnému obťažovaniu ľudí, ktorí reálne zmätení nie sú. Z toho dôvodu sa ako najlepší preukázal klasifikátor logistickej regresie pri vyvažovaní minoritnej triedy o 200 percent.

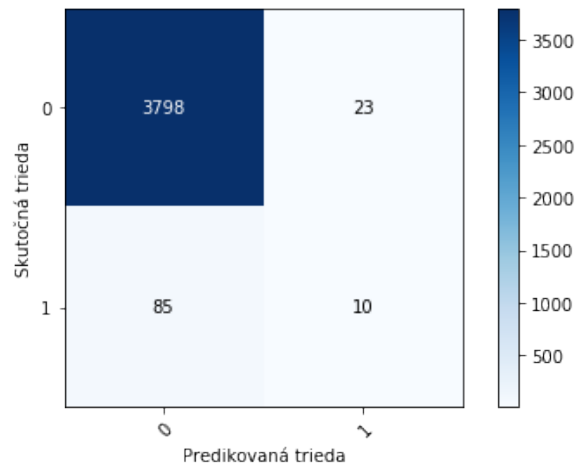
Tabuľka 14: Porovnanie predikcie pomocou algoritmu logistickej regresie *LR* a náhodného lesa *RF* na oknách veľkosti 10 sekúnd. Riadky označené slovom *SMOTE* prezentujú výsledky klasifikácie pri použití vyvažovania algoritmom SMOTE [1]. Výsledky boli získané aplikáciou vrstviacej troj-skupinovej krížovej validácie.

	Presnosť(0)	Presnosť(1)	Úplnosť(0)	Úplnosť(1)
LR	0.98	0.30	0.99	0.11
LR SMOTE 200%	0.98	0.36	0.09	0.09
LR SMOTE 500%	0.98	0.24	0.99	0.11
LR SMOTE 2000%	0.98	0.15	0.96	0.28
RF	0.98	0	1	0
RF SMOTE 200%	0.98	0.25	1.00	0.02
RF SMOTE 500%	0.98	0.27	1	0.07
RF SMOTE 2000%	0.98	0.08	0.95	0.25

Tabuľka 15: Porovnanie predikcie pomocou algoritmu logistickej regresie *LR* a náhodného lesa *RF* na oknách veľkosti 5 sekúnd. Riadky označené slovom *SMOTE* prezentujú výsledky klasifikácie pri použití vyvažovania algoritmom SMOTE [1]. Výsledky boli získané aplikáciou vrstviacej troj-skupinovej krížovej validácie.

	Presnosť(0)	Presnosť(1)	Úplnosť(0)	Úplnosť(1)
LR	0.97	0	0.98	0
LR SMOTE 200%	0.97	0.03	0.97	0.03
LR SMOTE 500%	0.97	0.02	0.95	0.03
LR SMOTE 2000%	0.97	0.03	0.88	0.13
RF	0.97	0	1	0
RF SMOTE 200%	0.97	0	1	0
RF SMOTE 500%	0.97	0	1	0
RF SMOTE 2000%	0.97	0.02	0.97	0.02

Porovnaním výsledkov predikcie s využitím okien 10 a 5 sekúnd môžeme povedať, že zväčšovaním okna získavame presnejšie výsledky. Svedčia o tom aj výsledky na celej úlohe, kde je presnosť predikcie najvyššia. Pri predikcii s využitím okien sa najlepšie javí klasifikátor logistickej regresie s 200 percentným vzorkovaním, kde sme dosiahli presnosť na predikcii zmätenia až 0,98. Pre úplnosť poskytujeme maticu zmätenia s využitím tohto klasifikátoru.



Obr. 10: Matica zmätenia pri klasifikátore logistickej regresie a 200 percentného vzorkovania minoritnej triedy pri predikcii zmätenia nad dátami o veľkosti okien 10 sekúnd, s využitím vrstviacej troj-skupinovej krížovej validácie (angl. *stratified 3-fold cross validation*). Pri trénovaní a predikcii sa použili metriky odvodené z myši. Predikcia zmätenia má označenie "1".

7 Zhodnotenie

V tejto práci sme predstavili metódu predikcie zmätenia používateľa vo webovej aplikácii. Navrhli sme ako natrénovať predikčný model s využitím histogramovej reprezentácie črt správania myši, štatistického vyhodnotenia dát z okulografu a iných metrík opisujúcich správanie používateľa vo webovej aplikácii. Na zhotovenie modelu sme potrebovali zozbierať označený dataset s udalosťami, kedy reálne nastalo zmätenie používateľa. Pre tento prípad sme pripravili používateľskú štúdiu so šesťdesiatimi účastníkmi, počas ktorej účastníci plnili úlohy na dovolenkovom portáli. Celý priebeh pozostával zo šiestich úloh, kde sme zaznamenali pohľad a zároveň myš. Na zaznamenanie myši a klávesnice sme použili nami implementovanú infraštruktúru v rámci projektu *YesElf* firmy *Brainware*.

Na predikciu zmätenia sme porovnali klasifikátory logistickej regresie a náhodného lesu. Je zaujímavé, že logistická regresia preukázala lepšie výsledky pri predikcii aj napriek nelinearite riešeného problému. Ukázali sme, že použitie základných črt správania používateľa akými sú napríklad dĺžka sedenia, počet navštívených stránok a počet cyklov v sekvencii stránok v rámci sedenia, môže byť užitočné pri predikcii zmätenia nad celou úlohou. Takéto metriky vieme jednoducho zaznamenať pomocou štandardného HTTP serveru bez nutnosti rozšírenia stránky. Rovnaké zistenie bolo prezentované aj v práci [42], kde autori využili spomínané metriky na klasifikáciu obtiažnych sedení.

Úspešnosť nášho klasifikátora však porástla s využitím histogramovej reprezentácie črt myši. Pri predikcii na celej úlohe sme využili aj dáta z okulografu. Výsledky predikcie na metrikách z okulografu sa ukázali porovnateľné s výsledkami predikcie len na metrikách myši. Tento výsledok je veľmi cenný keďže naznačuje, že na predikciu zmätenia nám bohato postačujú metriky myši. V súčasnej dobe totiž bežný počítač nie je vybavený zariadením ako je okulograf.

Významnou časťou našej práce a príspevkom k poznaniu je predikcia zmätenia v reálnom čase s využitím dát z myši. Naším cieľom bolo vyvinúť klasifikátor predikcie v reálnom čase, ktorý by vedel odhaliť presný bod zmätenia. Ako sme spomínali pod pojmom presný bod rozumieme moment, kedy používateľ nevie pokračovať v napĺňaní svojich potrieb v aplikácii vo vlastnej réžii. Tradične je tento bod spojený s využitím nápovedy, zákazníckej podpory alebo externej služby (vyhľadávača). Ak by sme vedeli identifikovať zmätenie v reálnom čase, mohla by byť používateľovi poskytnutá efektívna pomoc, napr. vo forme sprievodcov (prostredníctvom aplikácie *YesElf*). V tomto prípade sme využili okná rovnakej dĺžky (5 a 10 sekúnd). Dôvodom voľby časových okien pred celou úlohou bolo produkčné prostredie, kde nemáme možnosť získať dáta z celej úlohy používateľa, kvôli nejasnosti v jej začiatku a konci. Taktiež v takomto prípade potrebujeme zmätenie predikovať čo najskôr po príchode používateľa do aplikácie a z toho dôvodu sa veľkosti 5 a 10 sekúnd javia ako ideálne. Aj v tomto prípade sme využili algoritmy logistickej regresie a náhodného lesu. Podstatne vyššiu presnosť pri predikcii zmätených sedení dosiahla predikcia s využitím logistickej regresie pri oknách veľkosti 10 sekúnd. Metrika presnosti predikcie nad zmätenými pozorovaniami je v prípade produkcie kľúčová, keďže sa chceme vyvarovať prílišnému poskytovaniu sprievodcov používateľom, ktorý ich reálne nepotrebujú. Takéto zobra-

zovanie môže viesť k frustrácii. Výsledky oboch častí overenia potvrdzujú našu hypotézu, že s využitím dát z myši sme schopný predikovať zmätenia používateľa.

V rámci overenia našej metódy sme spolupracovali s firmou *Brainware*, ktorá vyvíja produkt *YesElf*. Spolu s kolegom Matejom Váľkym sme implementovali logger udalostí myši (pozri prílohu C). Veľký dôraz sme kládli na kvalitu softvérovej realizácie logera tak, aby sa dal použiť v produkcii a tak umožnil získanie rozsiahlych dát z používania. Logger bol vyvinutý s dôrazom na jednoduché škálovanie v produkcii a je v súčasnosti využívaný firmou *Brainware* na zaznamenávanie dát o správaní používateľa. Autor tejto práce pre nástroj *YesElf* implementoval aj infraštruktúru na predikciu zmätenia používateľa v reálnom čase. Aj táto infraštruktúra je jednoducho škálovateľná. Umožní spolu s loggerom overenie výsledkov tejto práce na rozsiahlejších produkčných dátach.

V rámci práce sme napísali článok na univerzitnú konferenciu *IIT.SRC* (článok s plagátom v prílohe E). Okrem toho sme publikovali článok na medzinárodnej vedeckej konferencii *UMAP* (článok v prílohe F).

V budúcnosti plánujeme našu metódu overiť produkčne na systéme *Aegon*. V tomto smere už pripravujeme používateľskú štúdiu so systémom s cieľom natrénovať predikčný model. Natrénovaný model nasadíme priamo do systému *YesElf* určeného na vytváranie interaktívnych sprievodcov, ktorý je integrovaný s aplikáciou firmy *Aegon*. Vždy keď u používateľa predikujeme zmätenie, systém *YesElf* mu ponúkne sprievodcov. Detaily aplikácie *YesElf* a predikcie v reálnom čase uvádzame v prílohe C. Okrem testovania v produkcii sa otvára možnosť využitia dát z klávesnice. V našej práci sme len ponúkli prehľad extrakcie metrík z tohto zdroju. V metóde sme však tieto dáta nezahrnuli z dôvodu krátkosti času a taktiež z dôvodu výberu domény *FiroTour*, kde sa klávesnica zväčša nepoužíva. V týchto dátach sa však môžu nachádzať informácie indikujúce zmätenie používateľa.

Literatúra

- [1] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall a W Philip Kegelmeyer. “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research* Vol. 16 (2002), s. 321–357.
- [2] Ahmed Awad E Ahmed a Issa Traore. “A new biometric technology based on mouse dynamics”. In: *IEEE Transactions on dependable and secure computing* Vol. 4. No. 3 (2007), s. 165–179.
- [3] Hirotugu Akaike. “A new look at the statistical model identification”. In: *IEEE transactions on automatic control* Vol. 19. No. 6 (1974), s. 716–723.
- [4] Gerald Albaum. “The Likert scale revisited”. In: *Market Research Society. Journal.* Vol. 39. No. 2 (1997), s. 1–21.
- [5] Jackson Beatty. “Task-evoked pupillary responses, processing load, and the structure of processing resources.” In: *Psychological bulletin* Vol. 91. No. 2 (1982), s. 276.
- [6] Maria Bielikova, Martin Konopka, Jakub Simko, Robert Moro, Jozef Tvarozek, Patrik Hlavac a Eduard Kuric. “Eye-tracking en masse: Group user studies, lab infrastructure, and practices”. In: *Journal of Eye Movement Research* Vol. 11. No. 3 (2018), s. 6.
- [7] Cristina Conati, Enamul Hoque, Dereck Toker a Ben Steichen. “When to Adapt: Detecting User’s Confusion During Visualization Processing.” In: *UMAP Workshops*. Citeseer. 2013.
- [8] Laura Cowen, Linden Js Ball a Judy Delin. “An eye movement analysis of web page usability”. In: *People and Computers XVI-Memorable Yet Invisible*. Springer, 2002, s. 317–335.
- [9] Simon Dooms, Toon De Pessemier a Luc Martens. “An online evaluation of explicit feedback mechanisms for recommender systems”. In: *7th International Conference on Web Information Systems and Technologies (WEBIST-2011)*. Ghent University, Department of Information technology. 2011, s. 391–394.
- [10] Andrew T Duchowski. “Eye tracking methodology”. In: *Theory and practice* Vol. 328 (2007).
- [11] Robin A Dubin. “Predicting house prices using multiple listings data”. In: *The Journal of Real Estate Finance and Economics* Vol. 17. No. 1 (1998), s. 35–59.
- [12] Claudia Ehmke a Stephanie Wilson. “Identifying web usability problems from eye-tracking data”. In: *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI... but not as we know it- Volume 1*. British Computer Society. 2007, s. 119–128.
- [13] Clint Feher, Yuval Elovici, Robert Moskovitch, Lior Rokach a Alon Schclar. “User identity verification via mouse dynamics”. In: *Information Sciences* Vol. 201 (2012), s. 19–36.
- [14] R Stockton Gaines, William Lisowski, S James Press a Norman Shapiro. *Authentication by keystroke timing: Some preliminary results*. Tech. spr. Rand Corp Santa Monica CA, 1980.
- [15] Hugo Gamboa a Ana Fred. “A behavioral biometric system based on human-computer interaction”. In: *Biometric Technology for Human Identification*. Zv. 5404. International Society for Optics a Photonics. 2004, s. 381–393.

- [16] Hugo Gamboa a Ana Fred. “A User Authentication Technic Using a Web Interaction Monitoring System”. In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer. 2003, s. 246–254.
- [17] Joseph H Goldberg a Xerxes P Kotval. “Computer interface evaluation using eye movements: methods and constructs”. In: *International Journal of Industrial Ergonomics* Vol. 24. No. 6 (1999), s. 631–645.
- [18] Mithat Gönen et al. “Receiver operating characteristic (ROC) curves”. In: *SAS Users Group International (SUGI)* Vol. 31 (2006), s. 210–231.
- [19] Shamsi T Iqbal, Piotr D Adamczyk, Xianjun Sam Zheng a Brian P Bailey. “Towards an index of opportunity: understanding changes in mental workload during task execution”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2005, s. 311–320.
- [20] Gareth James, Daniela Witten, Trevor Hastie a Robert Tibshirani. *An introduction to statistical learning*. Zv. 112. Springer, 2013.
- [21] Gawesh Jawaheer, Martin Szomszor a Patty Kostkova. “Characterisation of explicit feedback in an online music recommendation service”. In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM. 2010, s. 317–320.
- [22] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke a Geri Gay. “Accurately interpreting clickthrough data as implicit feedback”. In: *ACM SIGIR Forum*. Zv. 51. 1. Acm. 2017, s. 4–11.
- [23] Tomas Juhaniak, Patrik Hlavac, Robert Moro, Jakub Simko a Maria Bielikova. “Pupillary Response: Removing Screen Luminosity Effects for Clearer Implicit Feedback.” In: *UMAP (Extended Proceedings)*. 2016.
- [24] Samad Kardan a Cristina Conati. “Comparing and combining eye gaze and interface actions for determining user learning with an interactive simulation”. In: *International Conference on User Modeling, Adaptation, and Personalization*. Springer. 2013, s. 215–227.
- [25] Samad Kardan a Cristina Conati. “Exploring gaze data for determining user learning with an interactive simulation”. In: *International Conference on User Modeling, Adaptation, and Personalization*. Springer. 2012, s. 126–138.
- [26] Sotiris B Kotsiantis, I Zaharakis a P Pintelas. “Supervised machine learning: A review of classification techniques”. In: *Emerging artificial intelligence applications in computer engineering* Vol. 160 (2007), s. 3–24.
- [27] Peter Krátky. “Biometric User Model for Recognition on the Web”. In: *Information Sciences and Technologies* Vol. 9. No. 2 (2017), s. 19–24.
- [28] Sébastien Lallé, Cristina Conati a Giuseppe Carenini. “Predicting Confusion in Information Visualization from Eye Tracking and Interaction Data.” In: *IJCAI*. 2016, s. 2529–2535.
- [29] Sébastien Lallé, Dereck Toker, Cristina Conati a Giuseppe Carenini. “Prediction of users’ learning curves for adaptation while using an information visualization”. In: *Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM. 2015, s. 357–368.

- [30] Xu-Ying Liu, Jianxin Wu a Zhi-Hua Zhou. “Exploratory undersampling for class-imbalance learning”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* Vol. 39. No. 2 (2009), s. 539–550.
- [31] Nizar R Mabroukeh a Christie I Ezeife. “A taxonomy of sequential pattern mining algorithms”. In: *ACM Computing Surveys (CSUR)* Vol. 43. No. 1 (2010), s. 3.
- [32] Florent Masseglia, Pascal Poncelet a Maguelonne Teisseire. “Using data mining techniques on web access logs to dynamically improve hypertext structure”. In: *ACM sigweb Newsletter* Vol. 8. No. 3 (1999), s. 13–19.
- [33] Roy A Maxion et al. “Dependability at the user interface”. In: *Twenty-Fifth International Symposium on Fault-Tolerant Computing. Digest of Papers*. IEEE. 1995, s. 528–535.
- [34] John Neter, Michael H Kutner, Christopher J Nachtsheim a William Wasserman. *Applied linear statistical models*. Zv. 4. Irwin Chicago, 1996.
- [35] Avar Pentel. “Employing think-aloud protocol to connect user emotions and mouse movements”. In: *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*. IEEE. 2015, s. 1–5.
- [36] Avar Pentel. “Patterns of Confusion: Using Mouse Logs to Predict User’s Emotional State.” In: *UMAP Workshops*. 2015.
- [37] Alex Poole a Linden J Ball. “Eye tracking in HCI and usability research”. In: *Encyclopedia of human computer interaction* Vol. 1 (2006), s. 211–219.
- [38] Robert W Reeder a Roy A Maxion. “User interface defect detection by hesitation analysis”. In: *International Conference on Dependable Systems and Networks (DSN’06)*. IEEE. 2006, s. 61–72.
- [39] George AF Seber a Alan J Lee. *Linear regression analysis*. Zv. 329. John Wiley & Sons, 2012.
- [40] Marina Sokolova a Guy Lapalme. “A systematic analysis of performance measures for classification tasks”. In: *Information Processing & Management* Vol. 45. No. 4 (2009), s. 427–437.
- [41] Tobii Studio. “1.2 User Manual”. In: *Tobii Technology AB* (2008).
- [42] Paul Thomas. “Using interaction data to explain difficulty navigating online”. In: *ACM Transactions on the Web (TWEB)* Vol. 8. No. 4 (2014), s. 24.
- [43] Ryen W White, Joemon M Jose a Ian Ruthven. “Comparing explicit and implicit feedback techniques for web retrieval: Trec-10 interactive track report”. In: *Proceedings of the Tenth Text Retrieval Conference (TREC-10)*. 2002, s. 534–538.
- [44] Cha Zhang a Yunqian Ma. *Ensemble machine learning: methods and applications*. Springer, 2012.

