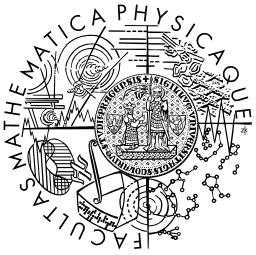


MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Karolína Kuchyňová

Ověřování identity uživatele založené na behaviorálních charakteristikách

Katedra softwarového inženýrství

Vedoucí diplomové práce: prof. RNDr. Tomáš Skopal, Ph.D.

Studijní program: Informatika

Studijní obor: Umělá inteligence

Praha 2020

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

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

V ..... dne ....

Podpis autora

Ráda bych poděkovala vedoucímu mé diplomové práce prof. RNDr. Tomáši Skopalovi, Ph.D. za čas, který mé práci věnoval, za jeho cenné rady a připomínky.

Dále bych chtěla poděkovat firmě Profinit za to, že mi umožnila sbírat data, ze kterých tato práce vychází, a jejímu Data Science týmu za nápady a rady ohledně jejich zpracování.

Největší díky pak patří mé rodině a příteli za jejich trpělivost a bezmeznou podporu v průběhu vzniku této práce.

Název práce: Ověřování identity uživatele založené na behaviorálních charakteristikách

Autor: Karolína Kuchyňová

Katedra: Katedra softwarového inženýrství

Vedoucí diplomové práce: prof. RNDr. Tomáš Skopal, Ph.D., Katedra softwarového inženýrství

Abstrakt: Ověřování identity uživatele přihlášeného do zabezpečeného systému je důležitým úkolem v oblasti informační bezpečnosti. Kromě hesla může být vhodné do procesu autentizace zahrnout i behaviorální biometriku. Ta sleduje chování uživatele, porovnává ho s jeho obvyklými akcemi, a může tak upozornit na podezřelé nesrovnalosti. Cílem této práce je prozkoumání možnosti vytvoření modelu pro ověření identity uživatele na základě jeho chování (stylu práce s myší a klávesnicí) ve webové aplikaci. Součástí práce je vytvoření vlastního datasetu pro studium dynamiky práce s klávesnicí a myší. Hlavní část práce se zabývá analýzou příznaků (charakteristik uživatelů), které je možné ze získaných dat extrahovat. Následně je změřena úspěšnost, které při ověřování identity uživatelů dosahují základní modely strojového učení využívající vybranou sadu příznaků.

Klíčová slova: ověřování identity uživatele, dynamika psaní na klávesnici, dynamika práce s myší, behaviorální biometriky

Title: User Identity Verification Based on Behavioral Characteristics

Author: Karolína Kuchyňová

Department: Department of Software Engineering

Supervisor: prof. RNDr. Tomáš Skopal, Ph.D., Department of Software Engineering

Abstract: Verifying the identity of a user logged into a secure system is an important task in the field of information security. In addition to a password, it may be appropriate to include behavioral biometrics in the authentication process. The biometrics-based system monitors the user's behavior, compares it with his usual actions, and can thus point out suspicious inconsistencies. The goal of this thesis is to explore the possibility of creating a user identity verification model based on his behavior (usage of mouse and keyboard) in a web application. The work includes creation of a new keystroke and mouse dynamics dataset. The main part of the thesis provides the analysis of features (user characteristics) which can be extracted from the obtained data. Subsequently, we report the authentication accuracy rates achieved by basic machine learning models using the selected set of features.

Keywords: user identity verification, keystroke dynamics, mouse dynamics, behavioral biometrics

Obsah

1. Úvod

V dnešní době je práce s počítačem nedílnou součástí života většiny lidí. Přes Internet běžně přistupujeme k webovým stránkám a aplikacím, ve kterých máme své účty a automaticky předpokládáme, že jsme jediní, kdo k nim má přístup. Může jít o aplikace s velmi citlivými daty jako například elektronické bankovnictví nebo intranet firmy, kam by měli mít přístup pouze oprávnění zaměstnanci. Pravděpodobně méně zásadní, ale také důležitá je pro uživatele například integrita profilů na sociálních sítích. Následky neoprávněného přístupu třetí strany k uživatelskému profilu se tak mohou různit od nepříjemností se zablokováním účtu na sociální síti až po ztrátu peněz na bankovním účtu nebo kompromitaci citlivých údajů firmy.

Proto je zcela zásadní otázkou, jak zaručit, že se do aplikace nebo systému dostanou pouze oprávnění uživatelé. Nejběžnějším postupem, jak toho docílit, je zavedení uživatelských jmen a hesel. Stále více aplikací nabízí i dvou nebo vícefázové ověřování. To může fungovat například tak, že uživateli po vyplnění hesla přijde ještě na mobilní telefon zpráva s kódem, který musí správně zadat, jinak se ke svému účtu nedostane. Tato varianta je typicky požadována od systémů, kde uživatel vnímá svá data jako velmi citlivá, jako je internetové bankovnictví nebo přístup k elektronické poště.

1.1 Ověřování identity uživatele

Obecně úloha ověření identity uživatele neboli jeho autentizace spočívá v tom, že uživatel deklaruje svoji identitu (například tím, že zadá svoje přihlašovací jméno nebo číslo účtu) a systém na základě splnění dalších požadavků (například správného zadání hesla) určí, zda věří, že osoba, která k němu přistupuje, je skutečně deklarovaný uživatel.

S úlohou autentizace těsně souvisí i otázka identifikace uživatele. Zde se ovšem scénář liší tím, že systém nemá předem informaci o tom, který uživatel by k němu měl přistupovat. Měl by sám na základě nějaké sledované charakteristiky určit, o kterého ze známých uživatelů se jedná, případně vyhodnotit, že žádného odpovídajícího uživatele nezná.

Identitu uživatele může systém ověřovat staticky, nebo průběžně. Statická varianta typicky spočívá v tom, že při přihlášení je uživatel jednorázově autentizován a poté může se systémem volně pracovat.

Oproti tomu při průběžném neboli dynamickém ověřování se v pravidelných intervalech kontroluje, zda se stále jedná o přihlášeného uživatele. Dynamicky autentizující systém by tak mohl včas odhalit situaci, kdy v započatém sezení uživatele pokračuje podvodník. K tomu může dojít například, pokud se útočník fyzicky dostane k počítači, ze kterého se oprávněný uživatel zapomněl odhlásit, nebo po síti pomocí tzv. únosu spojení (*session hijacking*).

Průběžná varianta autentizace může být požadována pro systémy, jejichž zabezpečení je kritické (například systémy používané armádou). Je pro ni zcela zásadní, aby autentizace uživatele co nejméně rušila při práci, ideálně aby ji vůbec nevnímal. (Například systém, který by vyžadoval opětovné zadání hesla každých

pět minut, by sice útočníkovi bez znalosti hesla mnoho šancí nedával, ale zároveň by byl velmi nepraktický i pro legitimní uživatele.)

Dynamicky autentizující systémy je možné rozdělit do dvou kategorií na proaktivní a reaktivní ([34]).

Proaktivní systém vyhodnocuje důvěryhodnost přihlášeného uživatele od okamžiku vstupu do systému průběžně a v momentě, kdy je chování uživatele příliš podezřelé, ho ze systému odhlásí. Toto je nutné, pokud systém obsahuje velmi důvěrná data.

Druhou variantou je reaktivní systém. Ten nechá po přihlášení uživatele volně pracovat a pouze na pozadí monitoruje jeho činnost. Až zpětně při nějaké konkrétní akci nebo opuštění systému je vyhodnocena uživatelova důvěryhodnost. Takovýto systém může fungovat například tak, že poté, co uživatel v elektronickém bankovnictví zadal příkaz k transakci, systém projde dosavadní chování uživatele a v případě nesrovnalostí označí transakci za podezřelou.

1.2 Metody autentizace

Standardně se způsoby, jakými uživatel může prokázat svoji identitu, dělí do tří kategorií.

První z nich je ověření na základě uživatelovy znalosti. Například že je schopen zadat heslo, které si zvolil, nebo mu bylo přiděleno, nebo správně zodpovědět danou kontrolní otázku.

Druhou kategorií jsou metody založené na tom, že uživatel vlastní určitou věc. Nejjednodušším příkladem je odeslání kódu v SMS zprávě na mobilní číslo uživatele. Pokud uživatel tento kód správně zadá, je to považováno za důkaz, že má mobil s daným číslem k dispozici a je tedy oprávněn k přístupu. Další ukázkou toho principu jsou různé přístupové karty nebo jiné tokeny.

Poslední kategorie ověřovacích metod není založena na tom, co uživatel zná nebo vlastní, ale na tom, čím je. Buď na jeho fyziognomii, nebo chování. Tyto metody se označují jako biometrické.

Použití biometrického ověřování identity přináší mnoho výhod. Jelikož si uživatel nemusí žádné údaje pamatovat, nehrozí, že je zapomene nebo se při jejich zadávání splete. Nemůže se stát, že by uživatel omylem prozradil kontrolní heslo, nebo se je útočník sám nějakým způsobem dozvěděl.

Na rozdíl od fyzického zařízení není možné uživateli biometriku snadno odcizit. Pro případného útočníka, který by se chtěl při přihlášení do systému za oprávněného uživatele vydávat, bude velmi obtížné nebo přímo nemožné věrně napodobit jeho fyzické rysy nebo přirozené chování.

Biometriky je možné dále rozdělit na fyziologické a behaviorální. Příklady fyziologických měření zahrnují otisky prstů, rozpoznávání obličeje nebo skenování sítnice.

Behaviorální biometriky jsou založeny na unikátním chování jedince. Řadí se sem mimo jiné dynamika práce s klávesnicí a myší, kterým se věnuje tato práce. Existuje však celé spektrum dalších behaviorálních charakteristik (pro příklady a detailnější rozbor mnoha dalších z nich viz [37]).

Ze srovnání fyziologických a behaviorálních metod vychází jako přesnější první z nich. Fyzické charakteristiky jsou totiž zpravidla více určující (lidé se v nich více odlišují) a především stabilnější.

Snadno se může stát, že se zcela změní styl uživatelovy práce s myší, pokud najednou pracuje místo optické myši s touchpadem, zatímco otisk jeho prstu nebo vzhled rohovky zůstává stále tentýž. Kromě podmínek prostředí mění behaviorální charakteristiky i čas. Například, když se uživatel postupně učí psát na klávesnici stále rychleji.

Behaviorální biometriky proto na rozdíl od fyziologických není možné využít přímo pro rozpoznání uživatele, ale mohou být s výhodou použity při ověřování jeho deklarované identity.

Na druhou stranu kvalitní naměření fyziologické biometriky vyžaduje sofistikované a drahé vybavení, zatímco zaznamenávání obvyklého chování jako práce s myší nebo klávesnicí žádné dodatečné požadavky na hardware počítače, kde uživatel pracuje, nemá. Další výhodou je, že tato data můžeme zaznamenávat na pozadí, aniž bychom tím jakkoli rušili uživatele při práci.

1.3 Náš scénář

Cílem této diplomové práce je prozkoumat možnosti vytvoření modelu pro ověřování identity uživatele na základě jeho chování ve webové aplikaci. Pracujeme zde s nejzákladnějšími a běžně dostupnými behaviorální charakteristikami - dynamikou práce s myší a klávesnicí.

V průběhu sezení uživatele zaznamenáváme, jak pohybuje myší, kdy a kam kliká a jak píše na klávesnici. Z těchto dat pak můžeme extrahovat charakteristické příznaky. Jedním z cílů práce je navrhnout a vyzkoušet sadu užitečných příznaků, které umožní dobře uživatele mezi sebou rozlišit. Je třeba, aby tyto rysy byly stabilní v rámci sezení jediného uživatele, ale při srovnání sezení různých uživatelů se co nejvíce lišily. Nakonec nad vybranými příznaky chceme vybudovat jeden nebo více modelů pro ověřování uživatelovy identity.

Práce probíhala v součinnosti s IT firmou Profinit EU, s.r.o. Ta umožnila sběr dat o chování svých zaměstnanců na vybraných stránkách interní webové aplikace firmy. Data jsme sbírali 8 měsíců od podzimu 2019 do léta 2020. Pro sledování jsme vybrali stránky, které používají zaměstnanci firmy nejvíce a kde současně nezadávají žádné citlivé údaje. Jde o stránky s jednoduchými formuláři nebo přehledovými tabulkami, které dobře odpovídají vzhledu obvyklých internetových aplikací.

Naším cílem není navrhnout co nejlepší model pro tuto konkrétní aplikaci, ale naopak chceme vytvořit dostatečně obecný model, který by šel snadno uplatnit v dalších podobně strukturovaných aplikacích, například v internetovém bankovnictví.

Hledaný model má za úkol na základě údajů o průběhu sezení uživatele a jeho deklarované identity určit, zda se jedná o oprávněný, nebo podvodný přístup. Jde tedy o reaktivní autentizaci. Získaná data by bylo možné použít i pro proaktivní průběžnou autentizaci, kdy by model dostával data postupně, jak byla zaznamenána v průběhu sezení. Tato varianta je však mimo rozsah práce.

Zaznamenávání aktivit uživatelů na stránkách by nemělo nijak ovlivňovat jejich běžné chování při práci s aplikací. Experiment probíhá v nekontrolovaném prostředí v tom smyslu, že uživatel se může k aplikaci přihlásit z libovolného počítače, (i když pravděpodobně ve většině případů pracuje vždy s tímtéž). Může se proto lišit hardwarové vybavení jako klávesnice a myš nebo další okolnosti při

práci. Změny prostředí verifikaci ztěžují, protože v jiných podmínkách se může chování uživatele měnit, ale současně tento scénář velmi dobře odpovídá tomu, jak lidé webové aplikace běžně používají.

V sesbíraných datech nemáme k dispozici žádné ukázky podvodných přihlášení. Při učení a vyhodnocování správnosti modelů tak jako příklady falešného přihlášení na účet uživatele použijeme náhodně vybraná sezení ostatních. Tento postup nemusí vždy úplně odpovídat realitě. Šlo by o tzv. „zero effort attack“, neboli situaci, kdy se útočník vůbec nesnaží maskovat své chování, například tak, aby se víc podobalo standardnímu chování uživatele.

1.4 Shrnutí cílů práce

Cíle této práce můžeme shrnout do tří bodů:

1. Vytvoření vlastního datasetu pro studium dynamiky práce s klávesnicí a myší. Dataset bude obsahovat informace o uživatelských interakcích na stránkách interní webové aplikace firmy Profinit. Půjde tak o záznam chování uživatelů při práci s běžně používanou aplikací v reálném prostředí. Získaná data by měla pokrývat dostatečně dlouhé časové období minimálně půl roku a výsledkem by tak měl být rozsáhlý dataset s informacemi o interakcích řádově stovek zaměstnanců firmy.

2. Navržení a experimentální evaluace sady příznaků, které lze extrahovat ze získaných dat o tom, jak uživatelé píší na klávesnici, klikají a pohybují myší. Cílem je vybrat příznaky, které mají potenciál umožnit vzájemné odlišení jednotlivých uživatelů a které je tedy vhodné použít při ověřování identity uživatelů.

3. Změření úspěšnosti modelů používajících vybrané příznaky pro rozhodování o pravosti uživatelských sezení. Testovány budou základní modely strojového učení jako rozhodovací stromy, naivní Bayesovský klasifikátor nebo SVM.

1.5 Struktura práce

První kapitola nabízí úvod do problematiky behaviorálních biometrik a představuje úlohu ověřování identity uživatele. Dále přibližuje situaci, ve které budeme tuto úlohu řešit, a jsou zde stanoveny vlastní cíle celé práce.

Druhá kapitola přibližuje dosavadní výsledky a postupy při studiu dynamiky práce s klávesnicí a myší. Zkoumá možné přístupy při sběru dat, strategie pro volbu vhodných příznaků, používané modely a jimi dosažené výsledky.

Třetí kapitola popisuje postup tvorby vlastního datasetu, stručně čtenáře seznamuje s firemní aplikací Profis, ve které byla data o práci s myší a klávesnicí sbírána, a shrnuje základní charakteristiky vytvořeného datasetu.

Čtvrtá kapitola se zabývá analýzou příznaků, které můžeme ze získaných dat odvodit. V páté kapitole jsou vybrané příznaky použity v modelech pro ověřování identity uživatelů a jsou zde prezentovány jimi dosažené výsledky. Tato kapitola zahrnuje také stručné představení všech použitých modelů.

Závěrečná kapitola pak zhodnocuje naplnění vytyčených cílů práce a navrhuje několik směrů, v nichž by bylo možno na ni navázat.

2. Teoretické základy a předchozí studie

Většina pozornosti se při studiu biometrik upírá na fyziologické biometriky jako jsou otisky prstů nebo skeny sítnice. Behaviorálními se vědci zabývají méně. Jednou z nejvíce a nejdéle studovaných behaviorálních biometrik je právě dynamika psaní na klávesnici. Ta se zabývá charakteristikami tempa psaní uživatele na klávesnici digitálních zařízení ([5]). Společně s dynamikou práce s myší jde pravděpodobně o nejvíce studované biometriky založené na interakci člověka a počítače.

Už v dobách druhé světové války byl zkoumán rytmus vysílání jednotlivých znaků při telegrafickém rádiovém provozu a používal se k identifikaci odesílatele zprávy. Od 80. let už se samostatně řešila dynamika psaní na klávesnici. V roce 1975 byla poprvé použita pro ověřování uživatele při zadávání hesla ([31]).

Obecně se nejprve studovala dynamika klávesnice při psaní předem známého fixního textu. To lze s výhodou použít pro verifikaci uživatele v průběhu zadávání přihlašovacího jména a hesla. Tento přístup však neumožňuje pasivní monitorování práce uživatele. K tomu je potřeba umět zpracovávat volný text. Se studiemi na něm se později začalo také.

Použití dynamiky práce s klávesnicí k ověřování identity uživatelů při zadávání hesla popisuje například [6]. Dalším příkladem studie, pracující s fixním textem, je [39]. Naopak jednou z nejčastěji citovaných studií zabývající se psaním volného textu je [14]. Využití stylu psaní na klávesnici pro identifikaci uživatelů popisují například [24] a [32]. Konečně přehled více než sta studií zabývajících se dynamikou klávesnice lze nalézt v [5].

Dynamika myši byla oproti klávesnici dlouho přehlížena a zkoumat se začala až na začátku 21. století. Lze ji definovat jako charakteristiku akcí získaných z myši jako vstupního zařízení od uživatele v průběhu jeho interakce s GUI ([3]).

Původně se pohyb myši zkoumal hlavně kvůli vylepšování grafického rozhraní aplikací, kdy se řešily uživatelsky přívětivé velikosti a polohy objektů na stránce, preferované úhly pohybu a podobně. Jedny z prvních studií, které se zabývají dynamikou myši jako biometrikou, pocházejí z let 2003 ([11]) a 2004 ([27]).

I u myši se nejprve zadávaly především fixní úkoly jako například spojování konkrétních bodů, které měly sloužit k ověření uživatelovy identity při přihlašování. Takový kontext popisuje například [28]. Také v této oblasti pak následovalo studium volného pohybu myši, umožňující kontinuální verifikace v průběhu sezení. Tou se zabývají mimo jiné studie [30] a [21]. Mezi často citované články na téma dynamiky myši patří [1] a [38].

Dalším krokem vývoje je pak spojení více behaviorálních biometrik do jediné tzv. multimodální biometriky. Jako nejpřirozenější se jeví právě kombinace dynamiky klávesnice a myši, neboť se jedná o ta nejzákladnější vstupní zařízení při práci s počítačem, dobře se doplňují a umožňují průběžné monitorování uživatele.

Studiem této kombinace behaviorálních biometrik se zabývá například [34], ve variantě pro průběžnou autentizaci uživatelů pak [13] a [23]. Poslední z uvedených se kromě autentizace zabývá i využitím dynamiky myši a klávesnice pro identifikaci uživatelů.

Přehled hlavních článků, ze kterých tato kapitola vychází, nabízí tabulka 2.1.

Studie	Autor a rok	Biometrika	Téma
[6]	Bartlow et al.(2006)	dynamika klávesnice	autentizace při zadávání hesla
[39]	Zhong et al.(2012)	dynamika klávesnice	autentizace při zadávání fixního textu
[14]	Gunetti et al.(2006)	dynamika klávesnice	autentizace na dlouhém volném textu
[24]	Modal et al.(2017)	dynamika klávesnice	identifikace
[32]	Tappert et al.(2010)	dynamika klávesnice	autentizace a identifikace na dlouhém textu
[5]	Banerjee et al.(2012)	dynamika klávesnice	přehled více než sta studií
[11]	Everitt et al.(2003)	dynamika myši	autentizace v online prostředí
[27]	Pusara et al.(2004)	dynamika myši	autentizace v online prostředí
[28]	Revett et al.(2008)	dynamika myši	speciální autentizační GUI
[30]	Shen et al.(2012)	dynamika myši	průběžná autentizace
[21]	Bours et al.(2013)	dynamika myši	průběžná autentizace
[1]	Awad et al.(2007)	dynamika myši	autentizace při běžné práci s počítačem
[38]	Zheng et al.(2011)	dynamika myši	autentizace při běžné práci s počítačem
[34]	Traore et al.(2012)	multimodální	autentizace v online prostředí
[13]	Fridman et al.(2015)	multimodální	průběžná autentizace
[23]	Bours et al.(2016)	multimodální	průběžná autentizace a identifikace

Tabulka 2.1: Přehled výchozích studií.

Obecný algoritmus

Fungování libovolného biometrického systému můžeme rozdělit do dvou fází. První z nich je registrace nového uživatele, kdy je pořízen jeden nebo více vzorků sledované biometriky, které jsou následně zpracovány a uloženy do referenční databáze. Druhou fází je pak verifikace nebo identifikace uživatele při jeho opětovném přihlášení, kdy se nová data, která žádající o přístup poskytne, porovnají se vzorem z registrace.

Pro práci s jakoukoli behaviorální biometrikou lze vyjít z obecného algoritmu navrženého v [37]:

1. Zvolíme chování, které budeme sledovat.
2. Chování rozdělíme na dílčí akce.
3. Spočítáme frekvence dílčích akcí pro jednotlivé uživatele.
4. Zkombinujeme výsledky do příznakového vektoru v profilu uživatele.
5. Porovnáme míru podobnosti uloženého vzoru a aktuálního chování uživatele.
6. Experimentálně určíme prahovou hodnotu, kdy ještě považujeme chování za dostatečně odpovídající referenčnímu vzoru.

7. Ověříme, nebo zamítneme uživatele na základě porovnání aktuálního skóre podobnosti a zvoleného prahu.

Aplikace

Standard Evropské unie EN 50133–1 požaduje od komerční biometriky nejvýše 0,001 % FAR (*false acceptance rate*) a 1 % FRR (*false rejection rate*) (pro vysvětlení těchto metrik viz 2.4). Tak dobrých výsledků zatím klávesnice a myš ani v kombinaci nedosahují, ale jde o vyvíjející se techniky, které zatím zcela jistě nedosáhly vrcholu svého potenciálu.

Přesto existuje široká nabídka komerčních řešení pro verifikaci identity uživatele využívající právě dynamiku klávesnice nebo myši. Příklady firem, které tato řešení poskytují, lze najít v [5]).

Z behaviorálních biometrik se zatím pro identifikaci osob ve větším měřítku používá pouze rozpoznávání hlasu nebo podpisu. Některé studie ovšem uvádějí, že by mohla být použita i dynamika při práci s klávesnicí za předpokladu, že je počet rozpoznávaných uživatelů omezen a psaný text je dostatečně dlouhý ([10]). Obecně se ale soudí, že zatím je pro dynamiku klávesnice i myši vhodnější aplikací pouze autentizace uživatelů.

2.1 Podmínky studií

Zásadní vliv na výsledky studií mají podmínky, za kterých sběr dat probíhal. V následující části se zaměříme na to, co přesně bylo úkolem účastníků, kolik se jich do studie zapojilo, v jakém prostředí a jak dlouho studie probíhaly. V závěrečné tabulce 2.2 vidíme shrnutí různých podmínek pro vzorek studií.

2.1.1 Úkol

Dynamiku psaní na klávesnici je možné zkoumat buď na volném, nebo na statickém, tedy předem daném textu. Jako fixní text slouží nejčastěji uživatelské jméno a heslo, ale může jít také o opisování zadané fráze nebo i delšího souvislého textu. V počátcích zkoumání dynamiky klávesnice se studie zaměřovali především na fixní text zadávaný při přihlašování uživatele tzv. *password hardening* ([6]). Tento přístup je výhodný pro aplikace, kde se kromě úvodního přihlášení neočekává mnoho dalších vstupů z klávesnice. Některé studie pracují jak s fixním, tak s volným textem a porovnávají úspěšnosti autentizace za použití každé z variant ([32]).

Otázkou je také volba minimálního požadovaného počtu znaků v analyzovaném textu. Výhodou dlouhého textu jsou podrobnější a spolehlivější statistiky pro jeho analýzu. Nevýhodou může být časová náročnost jeho vytváření a nepraktičnost v tom smyslu, že v mnoha aplikacích uživatel sám od sebe dostatečně dlouhý text na klávesnici nezadává. Například v [32] je minimální délka pro fixní i volný text více než 600 znaků. Výsledky pak ukazují, že pro dostatečnou úspěšnost by mohla stačit hranice 300 znaků. Další často citovanou studií zabývající se analýzou dlouhého textu je [14].

Pokud je úkolem systému průběžná autentizace, je nutné pracovat s volným textem. Opačná varianta, která by nutila uživatele neustále opakovaně zadávat

předem danou frázi, by byla pro většinu aplikací nepřijatelná.

Obdobou fixního textu pro dynamiku myši je použití speciálního autentizačního GUI, kde uživatel pracuje s myší a vykonává požadované akce ([28]). To přináší mimo jiné výhodu, že takto lze vynutit dostatečné množství akcí potřebné pro následnou analýzu.

Na začátku přípravy experimentu je třeba se rozhodnout, do jaké míry bude vymezeno zadání uživatelova úkolu na počítači. Zde je možné použít různé přístupy od úplně volného běžného užívání počítače, konkrétnější úkol jako práce s internetovým prohlížečem, až po striktní opisování zadaného textu.

Pokud je monitorováno neomezené používání počítače bez jakéhokoli konkrétního úkolu, hrozí, že následně budeme uživatele spíše než podle individuálního stylu práce s myší a klávesnicí rozlišovat podle toho, co na svém počítači obvykle dělají. Například bude pravděpodobně snadné určit uživatele, kteří počítač používají převážně k hraní her, oproti těm, kteří obvykle pracují s textovým editorem nebo těm, kteří převážně sledují filmy a videa.

Konkrétní příklady

Volné běžné používání počítače bylo analyzováno například v [38] a [3]. Uživatelé zde mohli bez omezení surfovat na webu, hrát hry nebo pracovat s textovým editorem a podobně. Konkrétnější úkol měli zadaný účastníci v [13]. Ti pracovali s daným prohlížečem a textovým editorem a měli k zadaným tématům nejprve najít na internetu informace a poté sepsat články. S omezením na akce při práci s webovým prohlížečem pracovaly i studie [27] a [19].

Vliv omezení na konkrétní aplikaci a úkol zkoumá [20]. Zde byly vytvořeny tři datasety. První z nich obsahoval všechny akce při práci s počítačem, druhý pouze ty akce, ke kterým došlo v prohlížeči souborů *File Explorer*, a třetí tyto akce zužoval pouze na ty, které se týkaly přímo práce se soubory.

Podobná trojice experimentů byla provedena i v [1]. Hlavní experiment sledoval běžné denní používání počítače. Druhý se omezoval na procházení webu a třetí se zaměřil na přesně definovaný úkol spočívající v klikání na obdélníky objevující se na obrazovce.

Je také možné pro potřeby experimentu vytvořit vlastní aplikaci. Nejčastěji jde o různá online fóra ([38]). Příkladem může být také jednoduchá síťová aplikace s přihlašovacím jménem a heslem simulující sociální síť z [34]. Zde byly přihlašovací údaje všech uživatelů veřejně známé a každý se tak mohl přihlásit jako kdokoli jiný. To umožnilo získat data o průběhu skutečných „podvodných“ sezení. To je relativně ojedinělý přístup, protože typicky se v rámci experimentu získávají pouze data ze sezení právoplatných uživatelů a za ukázky podvodů se prohlásí náhodně vybraná sezení ostatních.

Příkladem zcela fixního úkolu je „zámek pro myš“ (*mouse lock*) navržený ve [28]. Jde o rozhraní představující trezor s 24 obrázky po obvodu a uživatel má za úkol správným klikáním myší zadat heslo skládající se ze zvolené sekvence pěti obrázků.

2.1.2 Prostředí

Postupy, jakými účastníci experimentů svá data poskytují, můžeme rozdělit obecně do tří kategorií.

První z nich je stažení speciálního softwaru klienta na vlastní počítač ([32], [3], [21], [30], [23]). Program buď poskytuje vlastní rozhraní, se kterým uživatel pracuje, nebo pouze běží na pozadí a zaznamenává akce uživatele při práci s jinými aplikacemi. Klientská aplikace poté obvykle odesílá shromážděná data na server v laboratoři.

Druhou variantou je vytvoření webové aplikace fungující v prohlížeči ([38], [14]). Na webových stránkách se uživatelé přihlásí a mohou poskytovat své vzorky.

Třetí možnost spočívá ve využití prostředí laboratoře. To samozřejmě umožňuje nejlepší kontrolu podmínek, za nichž sběr dat probíhá. Některé studie se maximálně snaží všechny proměnné prostředí eliminovat a vytvořit úplně stejné podmínky pro všechny. Například při získávání dat v [13] byla vytvořena simulovaná kancelář, kde byly všechny počítače i vstupní zařízení identické. Cílem je nedetekovat prostředí, ale vlastní styl práce s myší nebo klávesnicí. Na druhou stranu přílišná míra kontroly prostředí nemusí odpovídat tomu, jak by autentizační systém měl být reálně používán.

Některé studie kombinují více přístupů tak, že vzorky menšího počtu uživatelů získají v kontrolovaném prostředí laboratoře a data od dalších pak sbírají ve volných podmínkách online ([1], [38], [5]).

Behaviorální systémy jsou často velmi citlivé na vstupní senzory. Pokud se například pokoušíme autentizovat uživatele píšícího na jiné klávesnici, než je ta, z níž pocházejí vzorová data, výrazně to zhorší výsledky systému ([32]). Pro smysluplně porovnatelné příznaky z práce s myší a potažmo dobré výsledky autentizace může být potřeba fixovat některé parametry jako rozlišení obrazovky, nastavení tlačítek a rychlosti myši a podobně. (Fixní nastavení monitoru a identickou myš používá například [20]). Alternativně se lze snažit volit příznaky tak, aby byly na vnějších podmínkách co nejméně závislé, například pracovat s histogramy nebo jiným relativním porovnáváním.

Právě nedostatečná kontrola proměnných prostředí experimentů je důvodem kritiky výsledků některých studií ([19]). Autoři kritiky tvrdí, že díky tomu, že uživatelé měli různá nastavené monitoru nebo rozložení klávesnic, bylo dosaženo přehnaně optimistických výsledků, zatímco kdyby všichni pracovali ve stejných podmínkách, model by je nebyl vůbec schopen rozeznat, nebo by se minimálně jeho úspěšnost výrazně zhoršila.

V některých studiích se věnuje pozornost i vlivu použitého vstupního zařízení, například tak že část vzorků je z klávesnice stolního počítače a druhá část používá klávesnici notebooku ([32]) nebo porovnává data při použití optické myši a touchpadu ([19]).

2.1.3 Účastníci

Častým nedostatkem studií zabývajících se dynamikou práce s klávesnicí nebo myší je relativně malý počet účastníků. Ten se obvykle pohybuje v řádu nižších desítek. Jeden z nejstarších a nejcitovanějších článků na téma dynamiky myši [27] pracuje například pouze s jedenácti uživateli. Výsledky získané na takto malém vzorku pak nemusí být považovány za dostatečně průkazné.

Kromě počtu účastníků je důležité také rozložení jejich věků, pohlaví a zkušenosti. Některé studie se cíleně snaží, aby v experimentu byly zastoupeny různé věkové skupiny, muži i ženy s různě širokou zkušeností s prací na počítači. Ide-

ální je hodně diverzní zastoupení očekávaných uživatelů biometrického systému. Záleží proto na kontextu, ve kterém se využití biometriky předpokládá.

Obvyklejší je ale situace, kdy účastníky jsou dobrovolníci z řad akademiků nebo studentů. Typicky tedy lidé zvyklí běžně s počítačem intenzivně pracovat. Nemusí proto jít o reprezentativní vzorek celé populace.

2.1.4 Doba trvání

Zajímavým aspektem je pro behaviorální biometriky jejich změna v čase. Může se stát, že chování uživatele o půl roku později nebude odpovídat vzorkům, které byly na začátku stanoveny jako referenční. Tuto skutečnost bohužel velká řada studií nemá možnost reflektovat. Typicky uživatelé veškerá data poskytnou v řádu několika týdnů, takže nemůžeme sledovat vliv časového odstupu sběru vzorků.

Časový interval, ze kterého data uživatelů vycházejí, se obvykle pohybuje od několika dní ([23]) po měsíce ([34], [1]).

Důraz na rozložení uživatelových sezení v čase kladla například studie [30]. Zde měli účastníci za úkol poskytnout 30 sezení po 30 minutách. Podmínkou ovšem bylo, že za den mohli absolvovat nejvýše jedno sezení. Díky tomu data jednoho uživatele představovala časové rozmezí od 30 do 60 dnů.

Výjimečnou je také studie v [32], která se mimo jiné zabývala i vlivem časového odstupu vzorků, na kterých se model trénuje, od těch, na kterých je následně testován. Srovnávaly se zde výsledky pro data získaná ve stejném týdnu, po uplynutí dvou týdnů, po čtyřech týdnech a po dvou letech. Podle očekávání se s časem chování uživatelů výrazně změnilo, což vedlo ke zhoršení úspěšnosti použitého modelu.

Kromě celého časového intervalu, z něhož pocházejí data od uživatele, se při experimentech také výrazně liší délka jednoho uživatelského sezení i to, kolik dat celkově každý účastník poskytuje. Obvykle je k dispozici více než 10 hodin zaznamenané práce na uživatele (10 v [3], 13 v [1] a 15 ve [20]). Délka uživatelských sezení se pak pohybuje od 30 minut ([19], [30]) až po několikahodinovou souvislou práci ([23]). Pokud by bylo sezení příliš krátké, může být obtížné posbírat dostatek dat pro ověření nebo určení identity uživatele. Přitom pokud sledujeme chování pouze v určité aplikaci, mohou být krátká sezení pro reálné uživatele typická.

2.1.5 Datasety

Často zmiňovaným problémem, s nímž se studium dynamiky myši i klávesnice potýká, je nedostatek standardních veřejných datasetů, na kterých by bylo možné nové modely ohodnotit. Neexistuje také jednotná konvence pro formát těchto dat. V důsledku toho je téměř nemožné výsledky jednotlivých studií porovnávat, protože každá z nich pracuje se zcela jinými daty.

Nejčastěji je při každé studii prováděno nové sbírání dat. To komplikuje situaci vědcům, protože pro svůj výzkum musí vytvářet nový dataset, který zpravidla zahrnuje pouze malý vzorek uživatelů. Obvykle jde o několik málo desítek účastníků experimentu, což neodpovídá reálnému systému, se kterým by jich mohly pracovat až stovky. Ne vždy se také podaří zajistit dostatečnou kontrolu podmínek při sběru dat. Někdy jsou datasety vytvořené v rámci konkrétní studie následně zveřejněny (například pro dynamiku myši v případě [1] nebo [30]). Pře-

Studie	Počet účastníků	Kontrola prostředí	Úkol
[34]	24	Ne	vytváření a sledování příspěvků na jednoduché sociální sítí
[23]	25	Ne	běžná práce na vlastním počítači
[28]	6	-	zadávání hesla na grafickém trezoru
[38]	30	Ano	běžná práce s počítačem
[38]	přes 1 000	Ne	aktivita na internetovém fóru
[19]	17	Ano	procházení zadaných webových stránek v prohlížeči Chrome
[30]	28	Ano	běžná práce s počítačem
[1]	22	Ne	běžná práce na vlastním počítači
[13]	67	Ano	psaní článků na zadaná témata, práce s prohlížečem a Wordem v simulované kanceláři
[20]	20	Ne	běžná práce na vlastním počítači a explicitně používání File Exploreru
[6]	16	Ne	zadávání vlastních a cizích přihlašovacích údajů do Java appletu v prohlížeči
[8]	15	Ano	zadávání vlastních a cizích přihlašovacích údajů a následně popis obrázku
[24]	64	Ne	napsání zkouškové eseje
[27]	18	Ne	prohlížení zadaných webových stránek v Internet Exploreru
[14]	40	Ne	volné delší psaní na libovolné téma
[25]	49	Ne	běžná práce na vlastním počítači
[3]	48	Ne	běžná práce na vlastním počítači
[32]	118	Ano	přepisování zadaného textu a volné psaní emailu v Java appletu

Tabulka 2.2: Ukázky podmínek různých studií.

hled některých veřejně dostupných datasetů pro dynamiku klávesnice uvádí [5] a pro práci s myší [20].

Důležitým požadavkem na použitelnost datasetu je dostatečně velký počet uživatelů, aby na něm bylo možné ověřit možnosti škálování navrhované modelu, které by bylo při reálném použití nutné. Pro testování robustnosti by se hodily i příklady v různé míře zašumělých dat. Specifikem behaviorálních biometrik je proměnlivost vzorků v čase, ideální dataset by to měl reflektovat a nabízet vzorky sezení rozložené v dostatečně dlouhém časovém horizontu. Z hlediska dynamiky klávesnice je praktické mít k dispozici datasety s ukázkami nejen fixního, ale i volného textu.

2.2 Použité příznaky

Při výběru příznaků je cílem najít takové, které jsou co nejvíce unikátní a zároveň stabilní pro každého uživatele. Jinými slovy takové, kde je malá variabilita v datech jedince, a naopak velké rozdíly při porovnávání s ostatními. Důležitým kritériem je také stabilita při změnách návyků v čase nebo různých podmínkách při práci.

2.2.1 Klávesnice

Co se týče surových dat zaznamenávaných z klávesnice, sledují téměř všechny studie totéž. A to informaci, kterou klávesu uživatel stiskl a v jakém čase, a kdy ji následně opět uvolnil. Někdy se explicitně zohledňuje i rozložení kláves na klávesnici a ke kódu klávesy se přidává i její souřadnice nebo aspoň relativní poloha vpravo, vlevo nebo uprostřed ([32]).

Nejběžněji používané příznaky odvozené od dynamiky práce s klávesnicí shrnuje tabulka 2.3.

Příznak	Popis	Použito v
doba držení klávesy	rozdíl času stisku a uvolnění klávesy	[34], [23], [5]
doba přechodu mezi klávesami (varianta *press-press*, PP)	rozdíl času stisku předchozí a následující klávesy	[13], [32]
doba přechodu mezi klávesami (varianta *release-press*, RP)	rozdíl času uvolnění předchozí a stisku následující klávesy	[32], [8]
doba přechodu mezi klávesami (varianta *release-release*, RR)	rozdíl času uvolnění předchozí a uvolnění následující klávesy	[23]
délka držení trigramu	rozdíl času stisků první a poslední z trojice kláves	[14], [34]
rychlost psaní	celková rychlost psaní ve znacích za čas	[34], [15], [5]
délka přechodu z klávesy Shift	rozdíl času stisku klávesy Shift a následující klávesy	[34], [6]

Tabulka 2.3: Ukázka nejčastěji používaných příznaků z klávesnice.

Doby držení a přechodů

Nejtypičtějším příznakem, který se z těchto dat extrahuje, je doba držení klávesy (tzv. *dwell time*) (používá ji například [34], [23], [5], [13]).

Dalším běžným postupem je sledování časů přechodů mezi dvěma klávesami. Zde jsou čtyři možnosti, jak přesně tento čas měřit. Může jít o dobu od stisku první klávesy do stisku druhé (varianta *press-press*, PP) ([13], [32]), od stisku první do uvolnění druhé (*press-release*, PR), eventuálně čas od uvolnění první do stisku druhé (*release-press*, RP)([32], [8]) nebo od uvolnění jedné do uvolnění druhé (*release-release*, RR) ([23]). Zatímco první dvě charakteristiky jsou nutně vždy kladné, RP i RR mohou být i záporné, pokud uživatel stiskl (eventuálně i uvolnil) druhou klávesu dříve, než uvolnil první. Někdy se pracuje s tím, jak často k tomuto jevu při uživatelově psaní přirozeně dochází ([34]).

Časy držení i přechodů mohou být sledovány v různé granularitě. V nejmenší míře podrobnosti máme pro všechny klávesy dohromady pouze jedinou průměrnou hodnotu. Při větší míře detailu je možné rozeznávat několik skupin kláves, například písmena, čísla, kontrolní klávesy, klávesy pro navigaci a tak dále. Časy držení nebo přechodů pak můžeme mít určeny zvlášť pro konkrétní skupinu. Charakteristický a často se opakující vzor může být třeba doba přechodu z kontrolní klávesy jako Shift na písmeno ([34], [6]). Při nejpodrobnějším zkoumání můžeme rozlišovat i jednotlivé klávesy a mít tak dobu držení resp. přechodu pro každou klávesu resp. konkrétní dvojici kláves ([14]).

Tento přístup ovšem vyžaduje dostatečně dlouhý vstupní text, protože jinak v něm nebudou mít některé klávesy nebo digramy dostatečné pokrytí. Jednu z možností, jak tomuto problému čelit navrhuje [32], zde jsou klávesy zorganizovány do hierarchie a pokud nemá příslušná klávesa dostatečně pokrytí, použije se místo ní údaj o hierarchicky nadřazené skupině. Alternativou je technika s aproximační maticí (*approximation matrix technique*) popsaná v [2], kde se zohledňují souřadnice kláves na klávesnici a pro nedostupný digram se použije hodnota digramu s co nejbližšími klávesami a lepším pokrytím.

Dobu přechodu lze samozřejmě zobecnit i na delší sekvence kláves než dvojice (například [14] pracuje i s trigramy). Zde se ovšem problém s nedostatkem výskytů některých n-gramů ještě prohloubí a byl by potřeba ještě delší text pro dostatečně přesnou analýzu. Tato překážka ovšem zcela odpadá v situaci, kdy pracujeme s fixním a nikoli volným textem. Pak sice můžeme sledovat pouze n-gramy, které se v něm vyskytují, ale máme zaručenu jejich konstantní četnost v každém ze vzorků.

Kromě průměrné doby držení klávesy nebo přechodu jsou často sledovány také směrodatné odchylky těchto veličin ([34]). Průměry i odchylky jsou tím méně zkreslené, čím delší je vstupní text ([32] uvádí jako dostatečnou délku alespoň 200 znaků).

Mnoho studií pracuje také s celkovou rychlostí psaní uživatele ve znacích nebo stiscích klávesnice na čas ([34], [15], [5]). Přitom je třeba odstraňovat příliš dlouhé pauzy v psaní, kdy uživatel pravděpodobně věnoval pozornost něčemu jinému [32].

Speciální přístupy

Pokud se rozhodneme rozdělit klávesy do určitých kategorií, umožňuje nám to sledovat procentuální zastoupení každé z nich ve vzorku (použito v [34] a

[32]). Pokud máme k dispozici i informace o poloze klávesy na klávesnici, lze pracovat například s předpoklady o tom, kterou ruku uživatel pro stisk dané klávesy používá ([32]).

Dále je možné se zaměřit na některé konkrétní vzory v práci uživatele s klávesnicí. Například jestli a jak používá klávesy pro navigaci na stránce jako Home, End, šipky, nebo klávesové zkratky jako Ctrl+C a Ctrl+V a podobně. Stejně tak je možné sledovat četnost chyb, tedy situací, kdy uživatel používá klávesy Delete nebo Backspace ([15]).

Další charakteristikou může být síla, kterou uživatel jednotlivé klávesy tiskne. To ale vyžaduje speciální klávesnici, na níž je možné tento tlak měřit ([5]).

Zajímavý přístup používá [14], který definuje tzv. stupeň neuspořádanosti (*degree of disorder*). Jde o srovnání relativních rychlostí přechodů mezi klávesami. To by mělo zvýšit robustnost systému v tom smyslu, že pokud se celková rychlost psaní zpomalí (například v důsledku únavy uživatele), zůstane zachováno relativní pořadí rychlostí psaní digramů, protože některé dvojice znaků se mu budou stále psát lépe než jiné.

2.2.2 Myš

Při studiu dynamiky práce s myší se monitoruje především pohyb myši a klikání jejími tlačítky. Pokud je součástí myši i kolečko, je možné sledovat i jeho otáčení.

Při kliknutí se standardně zaznamenává čas, kdy začalo a skončilo, tedy okamžik, kdy uživatel stiskl a uvolnil tlačítko myši. Dále se může ukládat poloha kliknutí na stránce, eventuálně i to, jestli bylo použito levé nebo pravé tlačítko myši.

Pro sledování pohybu myši je možné použít jeden ze dvou přístupů. První spočívá v zaznamenávání polohy myši v reakci na událost pohybu, kterou registruje operační systém nebo prohlížeč. Alternativou je sledovat a ukládat polohu myši v pravidelných časových intervalech nezávisle na tom, zda se hýbá, nebo ne (například v [27] je použit interval 100 ms).

Obvykle používané příznaky odvozené od dynamiky práce s myší shrnuje tabulka 2.4.

Jednoduché charakteristiky

Ze surových dat lze snadno získat elementární údaje o pohybu jako je uražená vzdálenost v pixelech, uplynulý čas nebo směr pohybu. Z nich se pak odvozují další příznaky jako rychlost pohybu, zrychlení nebo jeho obrácená varianta - podíl času a rychlosti ([21]), zakřivení nebo úhlová rychlost ([34]). Kromě průměrů těchto veličin je možné sledovat také směrodatnou odchylku, eventuálně i další momenty [27].

V některých studiích se používají také histogramy, například pro uraženou vzdálenost ([1], [21]) nebo uplynulý čas ([1]). Eventuálně se pomocí neuronové sítě aproximují vzájemné závislosti veličin jako čas nebo rychlost na uražené vzdálenosti ([1], [3]).

Podrobná data o pohybu nabízí možnost detailněji pracovat s trajektoriemi zaznamenaných pohybů. Například v [13] se pro každou zaznamenanou trojici

Příznak	Popis	Použito v
délka trvání kliknutí	rozdíl času stisku a uvolnění tlačítka myši	[34], [23], [13]
délka pauzy mezi akcemi	délka intervalu mezi zaznamenanými akcemi myši	[30], [34], [3]
rychlost	průměrná rychlost pohybu myši	[20], [30], [13]
úhlová rychlost	průměrná úhlová rychlost pohybu myši	[20],[12]
zrychlení	průměrné zrychlení při pohybu myši	[20], [30],[12]
zakřivení	úhel, který svírají tři po sobě jdoucí body polohy myši	[20], [38], [13]
histogram doby trvání pohybu	relativní četnosti pohybů určité délky trvání	[1], [3]
histogram uražených vzdáleností	relativní četnosti pohybů určité délky	[1] [21]
histogram typů akcí	relativní četnosti základních typů akcí (typy akcí jsou popsány níže)	[1], [3]
histogram směrů pohybu	relativní četnosti pohybu do každého z osmi základních směrů	[1], [34], [3]
rychlost v daném směru	průměrná rychlost pohybu v každém z osmi základních směrů	[1], [34]

Tabulka 2.4: Ukázka nejčastěji používaných příznaků odvozených od práce s myší.

bodů určuje jejich vzdálenost a zakřivení a pro delší sekvence se přidávají informace o úhlech a odchylkách od přímého směru.

Často se řeší otázka, jak příznaky o pohybu myši zvolit tak, aby byly co nejvíce nezávislé na platformě a nezáleželo u nich například na rozlišení obrazovky a podobně. Jako příklad příznaků spojených s konkrétním nastavením uvádí [38] třeba rychlost i zrychlení pohybu, naopak za nezávislé považuje směry, úhly nebo zakřivení.

U samotného klikání se nejčastěji pracuje s průměrnou dobou držení tlačítka myši ([34], [23], [13]). Dále můžeme zohlednit i její směrodatnou odchylku, nebo se zvlášť zabývat dobou trvání dvojitého kliknutí podobně jako u digramů na klávesnici ([23]).

Typy akcí a směry

Často se elementární akce myši v rámci přípravy příznaků agregují do komplexnějších akcí vyšší úrovně. Ty mohou představovat například zamíření a kliknutí (*point & click*), což je pohyb myši a následné kliknutí, dále přenášení (*drag & drop*), kdy jde o stisknutí tlačítka myši, pohyb a až pak následuje uvolnění, nebo volný pohyb myši bez klikání ([30], [3]).

Užitečnou charakteristikou mohou být i délky intervalů bez jakékoli akce myši, sloužící jako přirozená pauza mezi akcemi uživatele ([30], [34], [3]). Zde lze určovat průměrnou dobu trvání intervalu nebo třeba histogram různých délek ticha v průběhu sezení ([2]). Je samozřejmě zásadní uvažovat pouze krátké přirozené pauzy v souvislé práci s myší a nikoli situace, kdy uživatel věnuje pozornost něčemu zcela jinému. Například ve [2] jsou zkoumané intervaly ticha omezeny

délkou 20 s.

V mnoha studiích se uplatňuje rozdělování pohybů do osmi základních směrů. To umožňuje počítat hodnoty vztažené ke každému z nich zvlášť. Používá se například průměrná rychlost v každém ze směrů ([1], [3]), histogram zastoupení pohybů v jednotlivých směrech, procenta akcí, času nebo uražené vzdálenosti pro pohyb v daném směru ([34]). Podobně lze přistoupit i k samotným akcím a sledovat například histogram typů akcí nebo průměrnou rychlost pohybu pro daný typ akce ([1], [3]).

Pro spolehlivé výsledky o statistikách typů pohybu a akcích je potřeba dostatečně dlouhý vstupní vzorek chování. V případě průběžné autentizace se tyto statistiky berou pouze přes několik po sobě jdoucích akcí a je otázka, kolik přesně jich je k rozhodnutí zapotřebí. Zde je potřeba vyvažovat dostatečnou spolehlivost statistiky, která vyžaduje co nejvíce akcí, a to, že vyhodnocení chceme typicky provádět co nejrychleji.

Obecně statistiky akcí a pohybů nefungují jako příznak dobře v situaci, kdy všichni uživatelé plní zcela stejný úkol. V takovém případě je lepší použít detailnější příznaky jako rychlosti nebo křivky ([20]).

Další přístupy

Pokud se zkoumá dynamika myši pouze při práci s konkrétním grafickým rozhraním, je možné použít vysoce specifické příznaky. Například v [21] se u autentizačního rozhraní v podobě klikacího obrázkového trezoru pracuje s časy potřebnými k výběru jednotlivých správných znaků zámku a celé „otevírající“ kombinace.

Celkově může použití fixního grafického rozhraní umožnit rychlé a spolehlivé porovnání charakteristik jako je zrychlení, zakřivení nebo úhlová rychlost ([20]). Problémem je, že takový přístup je pro uživatele obtěžující a neumožňuje pasivní monitorování.

Zajímavý přístup, založený na „mikronávycích“, neboli vzorech chování běžných pro nějaký úkol, popisuje [30]. Tyto vzory se definují jako souvislé sekvence akcí, které se v zaznamenaném chování objevují dostatečně často. Dále se v rámci daného vzoru sledují běžné příznaky jako doba klikání, rychlost pohybu, zrychlení nebo pozice nejrychlejšího pohybu v rámci trajektorie. To by mělo vést k dosažení přesnějších a lepších výsledků, než když se tyto údaje určují pro veškeré chování jako celek.

2.3 Modely

Po rozhodnutí o volbě příznaků je třeba je ze surových dat extrahovat a data pročistit. Zásadní je zde odstranění odlehlých hodnot. Ty mohou být způsobeny selháním hardwaru nebo softwaru, ale nejčastěji k nim dojde přirozeně v důsledku chování uživatele. Například pokud sledujeme délku držení klávesy, ale uživatel chce napsat velké písmeno a drží klávesu Shift cíleně dlouho, nebo zaznamenáváme délky přirozené pauzy mezi akcemi uživatele, ale on na chvíli odejde od počítače. Proto je důležité pro každý z příznaků určit interval přípustných hodnot.

Předložení příznaků modelu typicky předchází jejich normalizace. V případě velkého množství použitých příznaků může být vhodné použít některou z technik

pro redukci dimensionality jako například ISOMAP nebo Fisherova diskriminační analýza ([20]).

Nejčastěji používané modely a jejich obvyklé značení shrnuje tabulka 2.5.

Název modelu	Značení	Studie používající model
Rozhodovací strom	DT (*decision tree*)	[21], [20]
Naivní Bayesovský klasifikátor	NB (*naive Bayes classifier*)	[21], [13]
Algoritmus k nejbližších sousedů	KNN (k *nearest neighbors*)	[21], [20], [30], [32]
Support vector machines	SVM	[21], [38], [20]
Neuronové sítě	ANN (*artificial neural networks*)	[23], [1], [30], [21], [3]

Tabulka 2.5: Obvykle používané modely.

Obecné rozdělení biometrických modelů do čtyř základních kategorií nabízí [5]. První skupinu tvoří statistické modely, které pracují s hlavně s průměry a směrodatnými odchylkami příznaků a vyhodnocují míru podobnosti vzorků v některé ze standardních metrik. Například v [39] autoři doporučují kombinaci Mahalanobisovy metriky pro normalizaci a dekorelaci vstupních příznaků a Manhattanské metriky, která by měla omezit vliv odlehlých hodnot.

Druhá kategorie obsahuje modely založené na použití neuronových sítí. Zde je pro každého uživatele natrénována vlastní neuronová síť a jejím výstupem je poté míra jistoty, že jde o deklarovaného uživatele. Při tréninku se síť učí ohodnocovat vzorky uživatele 100 % a vzorky podvodníků 0 %. Ve většině studií jsou použity mělké neuronové sítě, obvykle s jedinou skrytou vrstvou ([23], [1], [30], [21], [3]).

Nejčastěji studie používají třetí kategorii modelů, což jsou běžné modely strojového učení jako naivní Bayesovský klasifikátor ([21], [13]), rozhodovací stromy ([21], [20]), algoritmus k nejbližších sousedů ([21], [20], [30], [32]), nebo SVM ([21], [38], [20]). Všechny tyto modely jsme se rozhodli pro ověřování identity uživatelů vyzkoušet i my. Bližší seznámení s nimi nabízí sekce 5.1. V naprosté většině studií, jež některé z těchto modelů porovnávaly, bylo dosaženo nejlepších výsledků za použití SVM.

Poslední kategorie modelů zahrnuje heuristiky a různé kombinace předchozích přístupů. Příkladem může být využití evolučních algoritmů pro hledání parametrů některých z výše jmenovaných modelů.

Nejčastěji je pro každého uživatele natrénován vlastní model. Lze použít i více různých modelů a pro finální výstup kombinovat jejich výsledky ve většinovém hlasování ([38]). Pro učení modelu je typicky potřeba předložit vzorky legitimních i podvodných sezení. Protože téměř nikdy nejsou k dispozici skutečná podvodná sezení, používají se většinou jako ukázky podvodů náhodně vybraná sezení ostatních. Někdy je možné vystačit si pouze s pravými vzorky uživatele a k objevení podvodníka použít techniky detekce anomálií, tento úkol se označuje jako unární klasifikace ([30], [39]). Alternativou použitou v [8] nebo [23] pro identifikaci uživatelů je trénování klasifikátoru rozhodujícího mezi každou dvojicí uživatelů zvlášť.

Průběžná autentizace

Pokud je cílem průběžná autentizace v reálném čase, je potřeba předchozí systémy dále rozšířit. Typickým řešením je zavedení modelu, který neustále monitoruje aktuální pravděpodobnost, že se jedná o legitimního uživatele na základě dosavadního průběhu sezení ([21], [23]).

Každá akce uživatele je následně předána tomuto modelu, který ji vyhodnotí a na základě výsledku upraví aktuální věrohodnost uživatele. Pokud nová akce odpovídá očekávanému chování uživatele, míra důvěry se zvýší. Pokud je naopak akce vyhodnocena jako nepravděpodobná, důvěra v pravost uživatele klesá. Ve chvíli, kdy důvěryhodnost uživatele klesne pod krajní mez, může být uživatel prohlášen za podvodníka a ze systému odhlášen. Práh úrovně důvěry je možné stanovit globálně pro všechny, nebo nastavit individuálně pro každého z uživatelů podle stability jeho chování tak, aby se předešlo situaci, kdy bude neoprávněně odhlášen.

2.4 Používané metriky a dosažené výsledky

Základními metrikami, které se pro ohodnocení úspěšnosti autentizačních systémů používají, jsou FAR (*false acceptance rate*) a FRR (*false rejection rate*).

FAR vyjadřuje podíl, v kolika případech bylo podvodné sezení považováno za oprávněné, ku počtu všech předložených falešných sezení. Někdy se dále rozlišují *zero-effort* FAR a *active-imposter* FAR ([17]). První z nich uvažuje sezení, kdy se útočník nijak nesnaží o to, aby jeho vzorek napodoboval vzorek deklarovaného uživatele, druhá naopak pracuje se sezeními, kdy útočník takovou snahu vyvíjí.

FRR se zaměřuje na opačný problém. Jde o pravděpodobnost toho, že systém vzorek oprávněného uživatele označí za podvodný. Vztah mezi FAR a FRR je často založen na kompromisu. Autentizační model vypočítá pravděpodobnost, že předložený vzorek náleží danému uživateli, a záleží na dalším nastavení, jaká bude mezní hodnota pravděpodobnosti pro přijetí nebo zamítnutí přístupu uživatele.

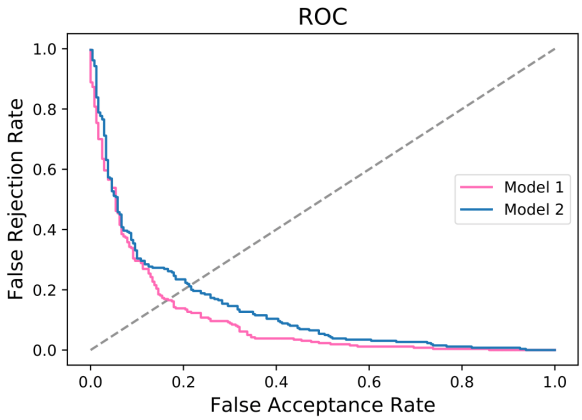
Pokud bude pro autentizaci uživatele vyžadována vysoká míra jistoty, že jde o oprávněný přístup, sníží se FAR, ale pravděpodobně se zvýší také počet případů, kdy bude odmítnut i oprávněný uživatel. Naopak při nižších požadavcích na shodu s očekávaným chováním klesá FRR, ale i pro útočníka je snadnější autentizací úspěšně projít.

Většinou je upřednostněno snížení FAR na úkor navýšení FRR, protože mírné uživatelské nepohodlí je méně zásadní než neoprávněné vniknutí do zabezpečeného systému. V praxi se pak volba kompromisu mezi FAR a FRR odvíjí od konkrétní aplikace, kde by měl být autentizační mechanismus použit.

Obvykle se testuje více různých nastavení rozhodovací hranice pro přijetí nebo zamítnutí uživatele a naměřené příslušné hodnoty FAR a FRR se zanáší do grafu. Tím vzniká tzv. ROC (*receiver operating characteristic*) křivka (ukázka na obrázku 2.1). Speciálním místem na této křivce je bod, kde se FAR a FRR rovnají. Ten se označuje jako EER (*equal error rate*). Čím nižší je tato hodnota, tím lepší je testovaný autentizační systém.

Doba autentizace

Pokud je úkolem systému průběžná verifikace identity uživatele, je důležitým kritériem i to, jak rychle nebo po kolika akcích je možné jednotlivá vyhodnocení věrohodnosti provádět. Jako metriku je možné použít například průměrný počet akcí podvodníka (ANIA - *average number of imposter actions*) předtím, než ho systém odhalí, a průměrný počet akcí oprávněného uživatele (ANGA - *average*



Obrázek 2.1: Ukázka ROC křivky.

number of genuine actions) ([21]). Ideální samozřejmě je, aby ANGA bylo nekonečně velké, tedy aby systém oprávněného uživatele nikdy neprohlásil za útočníka a neodhlásil. Stejně jako u FAR a FRR i mezi ANIA a ANGA je antagonistický vztah, daný nastavením hranice pro odhlášení uživatele ze systému.

Pokud nás zajímá přesný čas, který systém pro autentizaci vyžaduje, můžeme sledovat tzv. *mean time to alarm* ([1]), což je součet doby, jež je potřeba pro získání dostatečného množství uživatelských dat, s časem nutným pro jejich zpracování a vlastní rozhodnutí o verifikaci.

I zde je potřeba balancovat protichůdné požadavky na přesnost a rychlost systému. Je nutné získat dostatek dat od uživatele, aby při jeho následné analýze bylo možné extrahovat vypovídající příznaky. Na druhou stranu, čím delší čas na sběr dat systém potřebuje, tím delší je příležitost pro útočníka volně pracovat v zabezpečené aplikaci, než bude odhalen. Častým problémem navrhovaných systémů je právě nedosažení praktických časů verifikace. Pro systém s kritickým přístupem by mělo být možné odhalit útočníka nejvýše po několika málo minutách ([19]).

Další možná kritéria

Při použití v reálné situace je důležité sledovat také FTA (*Failure to acquire*), což je procento pořízených vzorků, které systém nemůže zpracovat, protože z nich není možné extrahovat potřebné příznaky ([9]).

Kromě objektivních kritérií by se mělo přihlížet i k subjektivnímu hodnocení uživatelů. Zde je důležitá otázka přijatelnosti, důvěra ve správné fungování systému nebo snadnost použití, jež se typicky odvíjí od použitých vstupních senzorů a celkového času, který systém pro autentizaci potřebuje.

Při použití biometrického systému v praxi mohou hrát roli ještě mnohé další faktory jako jeho cena, robustnost nebo škálovatelnost. Důležitá jsou také právní omezení, pokud by uživatelské vzorky nějakým způsobem obsahovaly jejich osobní údaje.

Výsledky studií

Tabulka 2.6 ukazuje výběr výsledků některých předchozích studií. Je třeba zdůraznit, že většina z nich je dosažena na vlastních datasetech, různě velkých a vytvořených za různých podmínek, a není proto možné je vzájemně porovnávat.

Studie	Biometrika	Počet účastníků	Dosažený výsledek
[32]	dynamika klávesnice	118	FAR 3,9 %, FRR 15,7 %
[6]	dynamika klávesnice	16	EER 2 %
[14]	dynamika klávesnice	40	FAR $<$ 0,005 %, FRR $<$ 5 %
[28]	dynamika myši	6	FAR 2–6 %, FRR 0–8 %
[38]	dynamika myši	30	FAR 2,96 %, FRR 0,86 %
[30]	dynamika myši	28	FAR 0,37 %, FRR 1,12 %
[1]	dynamika myši	22	FAR 2,46 %, FRR 2,46 %
[20]	dynamika myši	20	FAR 6 %, FRR 5 %
[21]	dynamika myši	49	ANIA 94
[27]	dynamika myši	18	FAR 0,43 %, FRR 1,75 %
[12]	dynamika myši	25	EER 1,01 %
[3]	dynamika myši	48	FAR 2,6 %, FRR 2,5 %
[34]	kombinace myši a klávesnice	24	EER 8,21 %
[13]	kombinace myši a klávesnice	67	FAR 0,4 %, FRR 1 %

Tabulka 2.6: Výsledky dosažené v různých studiích.

3. Sběr dat

Následující kapitola popisuje získaná data o uživatelských interakcích a proces jejich sběru. Začíná bližším seznámením s prostředím, z nějž data pocházejí, tedy s interní firemní aplikací Profis. Vzhled sledovaných stránek aplikace je klíčový pro to, jak uživatel může se stránkou interagovat, a potažmo pro to, jaká data o jeho chování budeme získávat.

Dále stručně nastíníme proces získávání dat od detekce události v prohlížeči až po uložení záznamu o ní do databáze. Tato část nabízí také přehled všech použitých databázových tabulek.

V následující části představíme dataset vytvořený na základě získaných dat. Nakonec se blíže podíváme na množství záznamů, které se v období od konce října 2019 do konce května 2020 podařilo shromáždit.

3.1 Uživatelské interakce

Podoba vstupních dat o uživatelských interakcích se odvíjí od toho, co na monitorovaných stránkách uživatelé běžně dělají a jak přesně jejich akce budeme zaznamenávat.

Sbírané informace pocházejí od zaměstnanců firmy Profinit EU, s.r.o. v rámci jejich pracovní činnosti a firma souhlasila s jejich zpracováním a zveřejněním omezené verze vytvořeného datasetu v rámci této diplomové práce. (Tento souhlas lze nalézt v příloze.)

3.1.1 Popis monitorovaných stránek

Tato část nabízí základní seznámení s firemní aplikací Profis, na jejíchž stránkách sbíráme data o uživatelských aktivitách. S aplikací běžně pracuje každý z několika set zaměstnanců firmy. Aplikace má dvě verze - intranetovou, která je přístupná pouze z interní sítě firmy, a verzi s omezenou funkcionalitou dostupnou z Internetu.

Aplikace nabízí několik různých modulů, z nichž většina je přizpůsobena přímo potřebám konkrétních oddělení firmy. My jsme pro monitorování zvolili pouze modul Vykazování, který je přístupný všem zaměstnancům. Nachází se zde stránky pro vykazování pracovních činností a podávání žádanek o dovolenou. Rozhodli jsme se tak proto, že tyto stránky patří k nejnavštěvovanějším a také proto, že zde uživatelé nevyplňují žádné své osobní nebo jiné citlivé údaje. Všechny sledované stránky jsou přístupné v internetové i intranetové verzi aplikace.

Sekce pro Vykazování odpracovaných hodin se skládá ze čtyř stránek:

- Nová činnost (obrázek 3.1). Zde uživatel vytváří nový záznam o odvedené práci. Stránka má podobu jednoduchého formuláře. Některé položky se uživateli automaticky předvyplní, ostatní pak musí doplnit ručně. Je také možné použít některou z šablon vytvořených uživatelem. V tom případě se položky formuláře vyplní automaticky.

- Detail činnosti (obrázek 3.2). Jde o stránku, která poskytuje informace o již

uložené činnosti. Vzhledem se nijak neliší od stránky pro zadání nové činnosti. Uživatel zde může podle potřeby dříve vyplněné údaje upravit.

- Seznam vykázaných činností (obrázek 3.3). Stránka poskytuje základní přehled o všech vykázaných činnostech uživatele. Z tabulky je možné přejít do detailů záznamů, mazat je nebo vytvářet nové jako kopie již existujících. V horní části se nachází sekce s mnoha možnostmi, jak zobrazované záznamy filtrovat.

- Kvartální přehled vykázané činnosti (obrázek 3.4). Stránka nabízí přehled toho, kolik hodin zaměstnanec odpracoval jednotlivé dny zvoleného kvartálu. Po kliknutí na příslušný den je možné se na vykázané činnosti podívat v detailu nebo vytvořit nový záznam pro daný den.

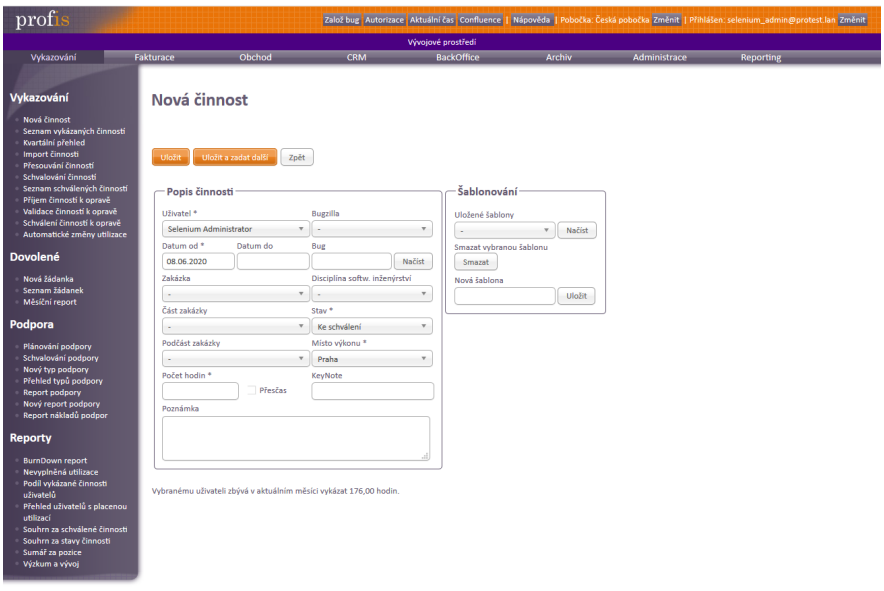
Obdobnou strukturu mají i stránky týkající se žádostí o dovolené:

- Nová žádanka o dovolenou (obrázek 3.5). Jedná se o krátký formulář s informacemi o žádané dovolené, kde je většina položek předvyplněna a uživatel obvykle doplňuje pouze časové rozmezí.

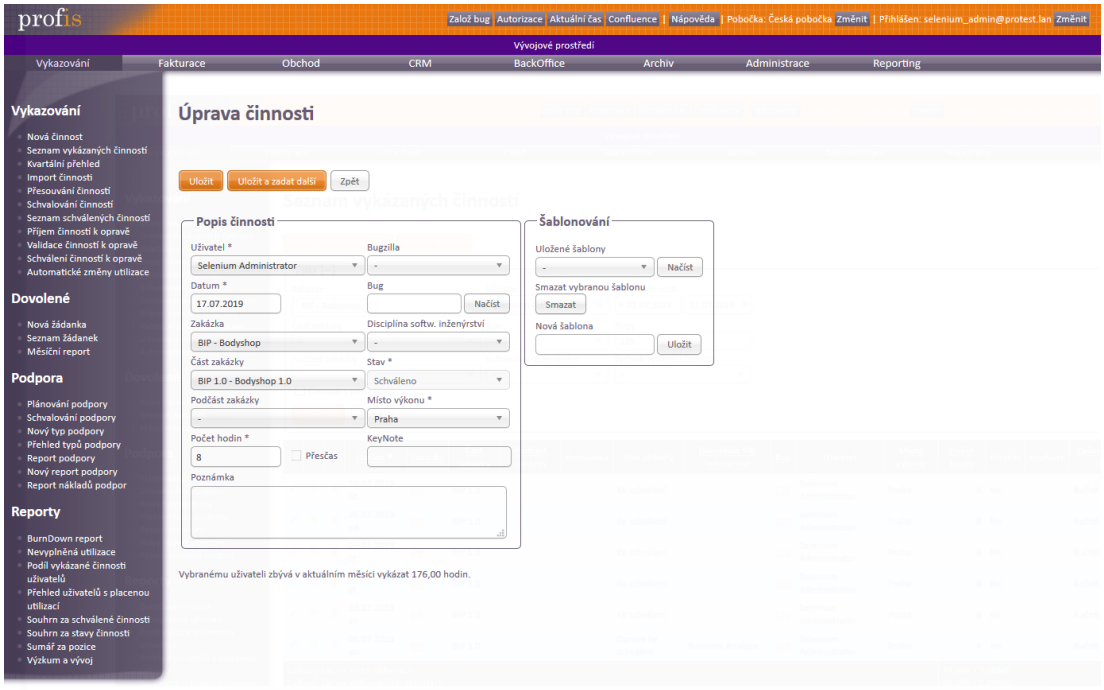
- Detail žádanky o dovolenou (obrázek 3.6). Opět jde o stránku k již vytvořené žádance, kde je možné některé informace dále upravovat a přibývá sekce týkající se schvalování a historie žádosti.

- Seznam žádanek (obrázek 3.7). Stránka obsahuje přehled existujících žádanek odpovídajících kritériím nastaveným ve filtru v horní části. Z přehledu je možné přejít do detailu jednotlivých žádanek a oprávnění zaměstnanci je zde také mohou schvalovat.

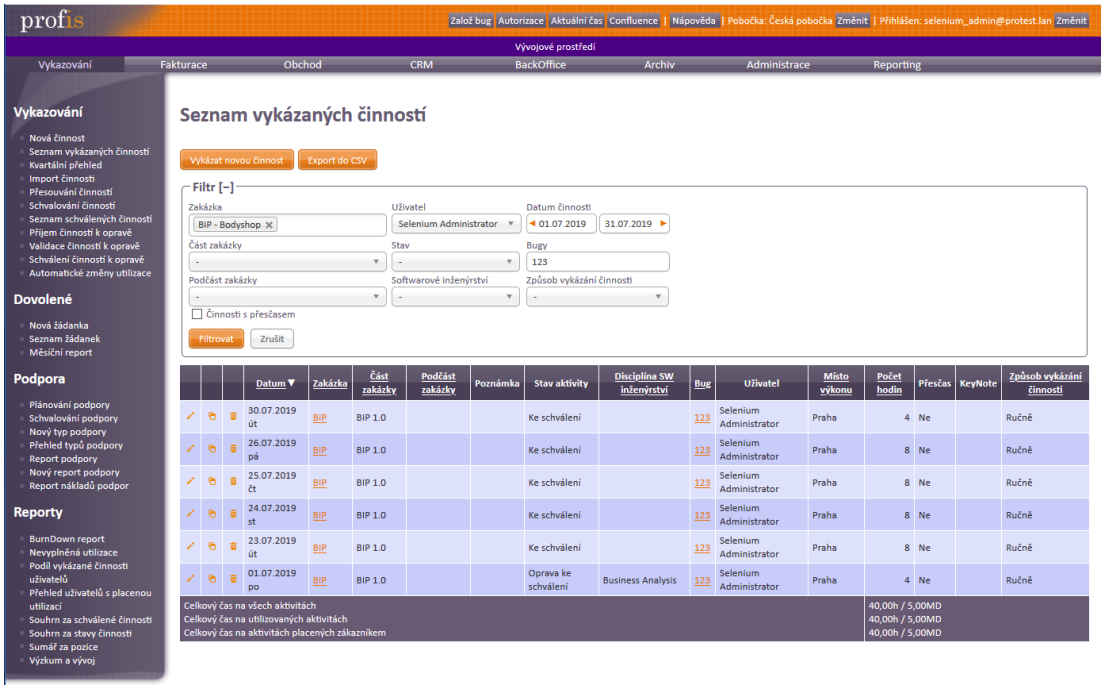
- Report dovolených (obrázek 3.8). Stránka nabízí jednoduchý přehled toho, jak uživatel ve zvoleném měsíci pracoval a čerpal dovolenou.



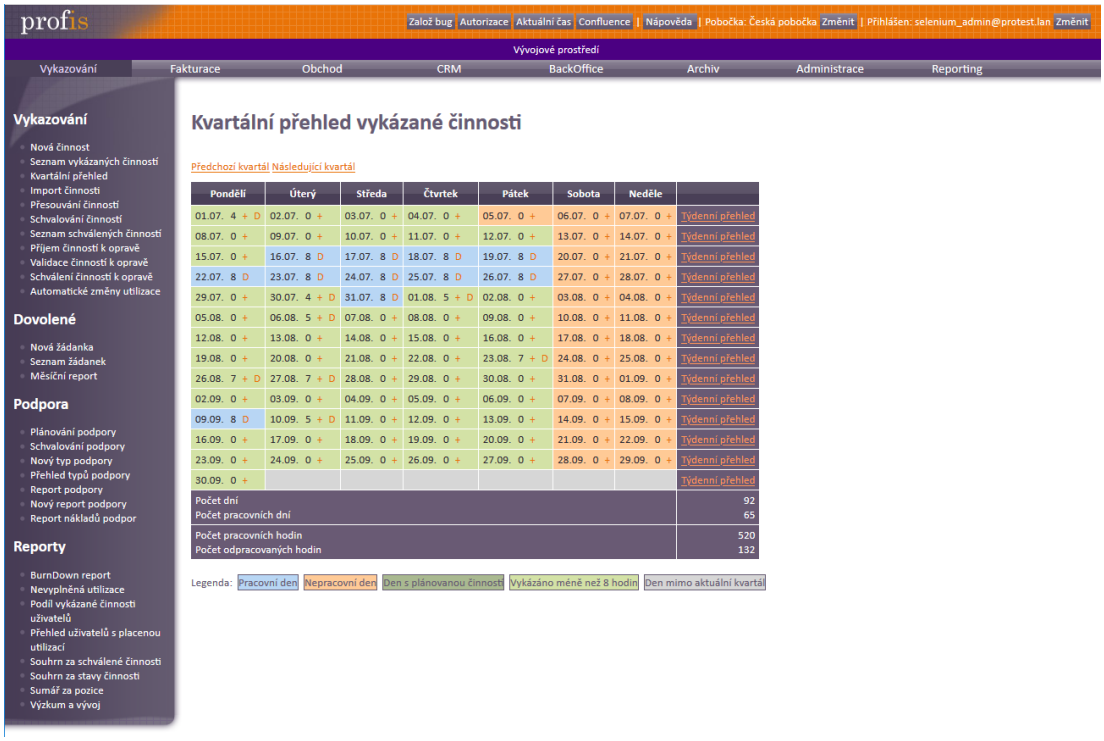
Obrázek 3.1: Stránka pro vytvoření nového záznamu o činnosti.



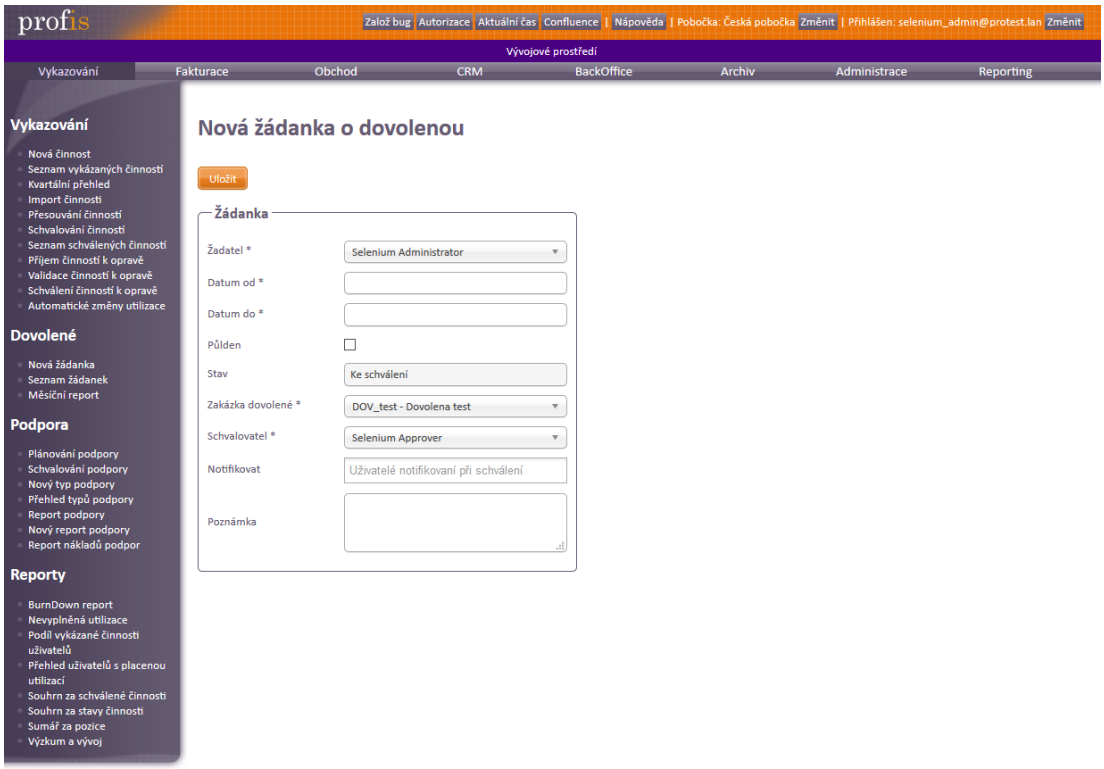
Obrázek 3.2: Detail záznamu o činnosti.



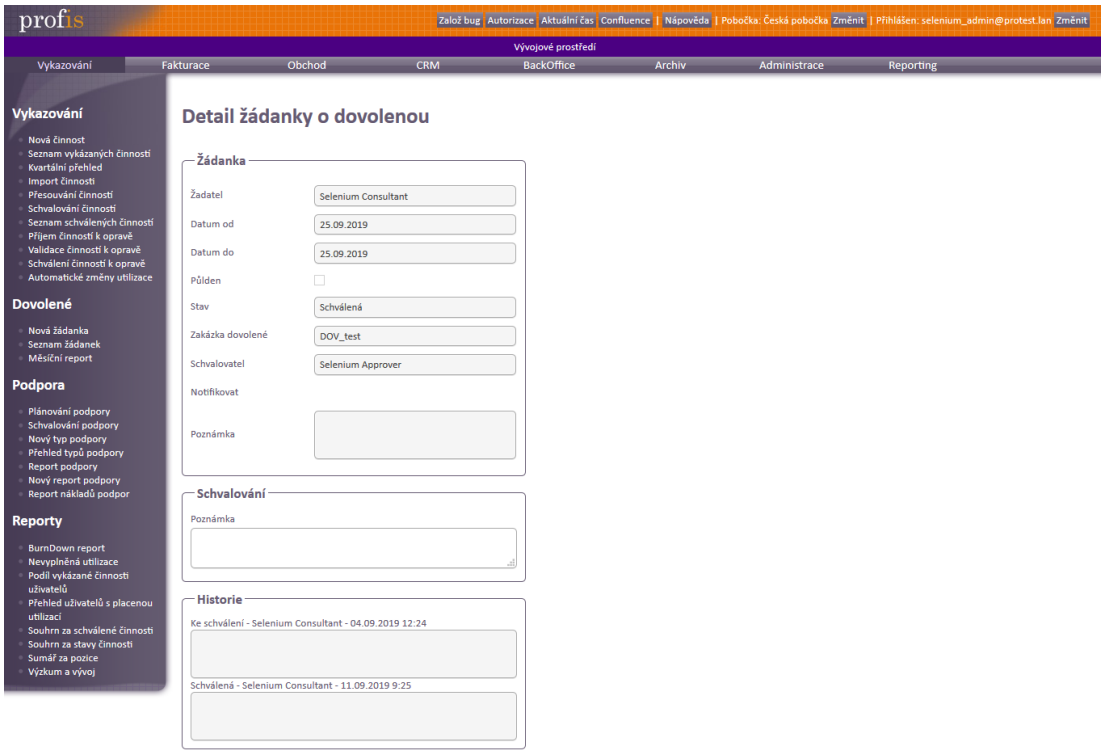
Obrázek 3.3: Seznam vykázaných činností.



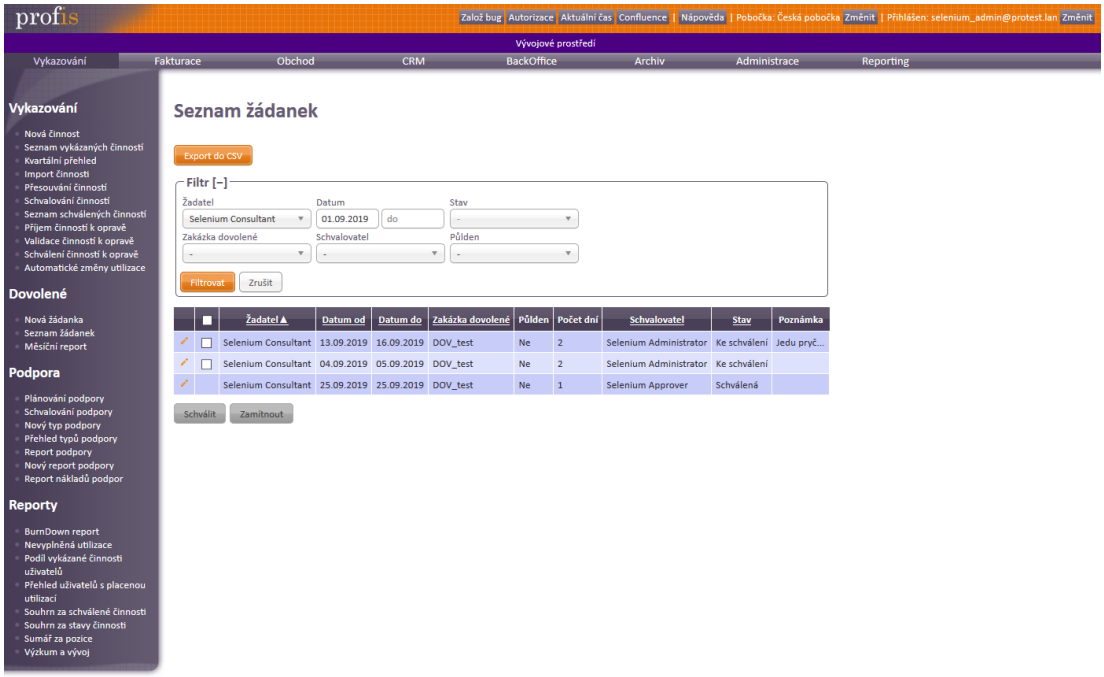
Obrázek 3.4: Kvartální přehled vykázaných činností.



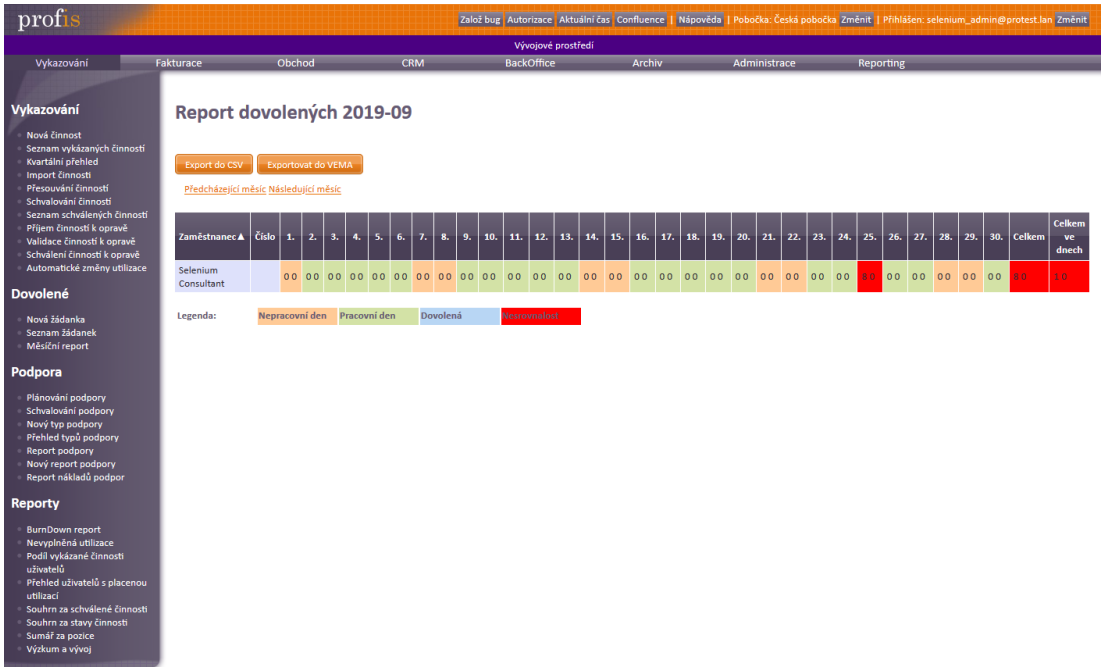
Obrázek 3.5: Stránka pro vytvoření nové žádanky o dovolenou.



Obrázek 3.6: Detail žádanky o dovolenou.



Obrázek 3.7: Seznam žádanek o dovolenou.



Obrázek 3.8: Report dovolených.

Prvky stránek

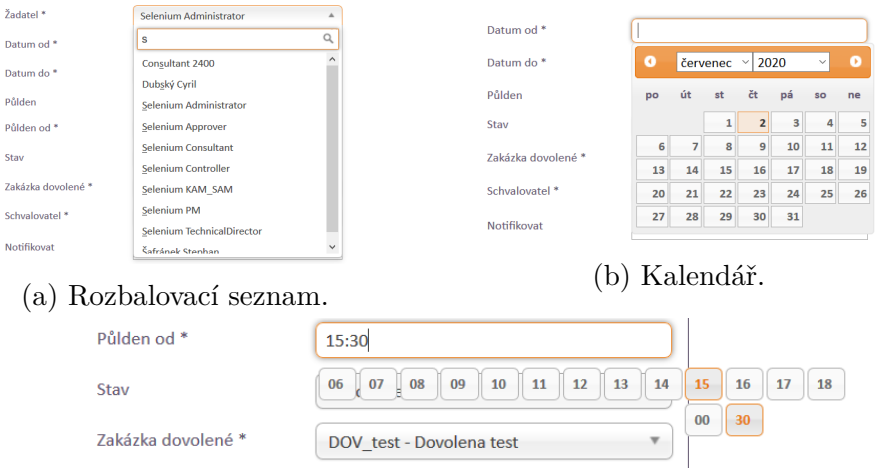
Nyní stručně popíšeme typy prvků, se kterými se uživatelé na sledovaných stránkách setkávají. Ty totiž hrají zásadní roli v tom, jakým způsobem uživatelé se stránkou interagují. Při výběru příznaků může být například užitečné blíže se zaměřit na to, jak uživatel pracuje s určitým konkrétním typem prvku.

Nejčastěji zastoupeným prvkem ve formulářích i filtrech je rozbalovací seznam (*dropdown*), ve kterém uživatel vybírá z předpřipravených položek (na obrázku 3.9a). V seznamu se dá pohybovat pomocí posuvníku nebo šipek klávesnice. Je zde také podpora vyhledávání podle zadaných počátečních písmen položky.

Důležitou skutečností pro získaná uživatelská data je, že na všech stránkách se nachází relativně málo textových polí. Navíc do většiny těchto polí uživatel píše velmi krátký text jako například počet odpracovaných hodin. Jediným místem, kde lze očekávat delší vstupní text, jsou pole pro poznámky. Důsledkem toho je, že pro naprostou většinu uživatelských sezení máme k dispozici pouze minimální vstup z klávesnice.

Pro výběr data nebo konkrétního času ve formuláři nebo nastavení filtru mají uživatelé k dispozici speciální prvky zobrazující kalendář nebo výběr hodin (na obrázcích 3.9b a 3.9c). Dalším typem prvku, na který je možné v aplikaci narazit, jsou zaškrtávací pole.

K navigaci může uživatel používat odkazy buď přímo na stránce, nebo v levém bočním menu. Pro potvrzování vyplněných údajů, přechody mezi stránkami a podobně slouží v aplikaci nejrůznější tlačítka.



(a) Rozbalovací seznam.

(b) Kalendář.

(c) Výběr času.

Obrázek 3.9: Ukázka speciálních prvků v aplikaci.

3.1.2 Zaznamenávané akce

Na vybraných stránkách zaznamenáváme pomocí JavaScriptu informace o následujících akcích, které zachycuje prohlížeč:

- Stisknutí a uvolnění tlačítka myši (`mousedown & mouseup`). Přitom ukládáme identifikátor prvku stránky, nad kterým ke kliknutí došlo, a také relativní a absolutní polohu kurzoru při akci vzhledem k levému hornímu rohu prvku. Jako identifikátor prvku používáme jeho xPath ([36]), absolutní polohu počítáme v pixelech a relativní v procentech vzhledem k rozměrům prvku.

- Stisknutí a uvolnění klávesy (`keydown & keyup`). Zde sledujeme opět xPath prvku, který byl aktivní, když uživatel klávesu použil, a dále také identifikátor stisknuté klávesy.

- Pohyb myši (`mousemove`). U něj nás zajímá pouze x-ová a y-ová souřadnice kurzoru na obrazovce v okamžiku zaznamenání události. Protože typicky i při krátké interakci dochází k velmi mnoha událostem spojeným s pohybem myši, rozhodli jsme se omezit maximální počet zaznamenaných pohybů na 50. Aby omezený počet událostí reprezentoval delší vzorek interakce, záznamy provádíme s periodou 7. To znamená, že pouze každá sedmá událost pohybu myši se zaznamená.

U všech výše zmíněných akcí zaznamenáváme také jejich typ (`mousedown / mouseup / keydown / keyup / mousemove`), datum a čas, kdy k nim došlo, a časovou značku toho, kolik milisekund uplynulo od načtení stránky do okamžiku záznamu akce.

Ve srovnání s přístupy použitými ke sběru dat v jiných studiích nám prostředí webového prohlížeče umožňuje všechny sledované události zachycovat v něm a není potřeba získávat je přímo od operačního systému. Pro úplnost doplňme, že prohlížeč umožňuje sledovat i další události spojené s myší a klávesnicí například `keypress` (akce napsání znaku z klávesnice) a `click` (akce kliknutí na

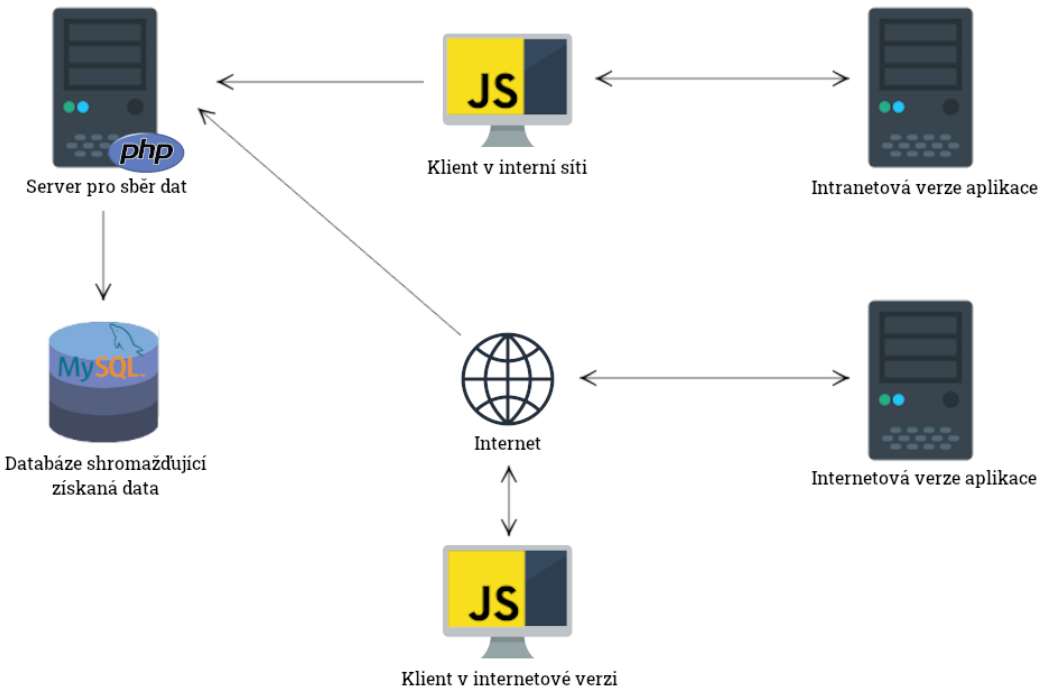
určité místo), které nijak nevyužíváme, protože nepřinášejí nové nebo relevantní informace.

Oproti ostatním studiím jsme se rozhodli zaznamenávat pohyb myši prostřednictvím příslušné události. Daleko obvyklejší je periodické zaznamenávání polohy myši nezávisle na tom, jak nebo zda se pohybuje. Tuto možnost jsme nezvolili vzhledem k tomu, že chceme omezit počet akcí myši, který zaznamenáme, a proto sledujeme pouze okamžiky, kdy se myš skutečně pohybuje.

Vzhledem k výsledkům příznaků extrahovaných ze záznamů o pohybu myši (viz sekce 4.4) nelze jednoznačně doporučit postup, kdy uložíme pouze každou sedmou událost pohybu a jejich celkový počet omezíme na 50. Jediný užitečný příznak, který jsme z těchto dat extrahovali, byly časy mezi záznamy o pohybu. Průměrná rychlost pohybu myši odvozená z těchto dat se ukázala jako nevhodná pro odlišení uživatelů. Otázkou samozřejmě je, zda bychom dosáhli lepších výsledků, kdybychom ukládali všechny události o pohybu a jejich počet uměle neomezovali.

3.2 Proces získávání dat

Sběr dat o uživatelských interakcích s myší a klávesnicí probíhá následovně. Uživatel na svém počítači pracuje na monitorovaných stránkách v intranetové, nebo internetové verzi aplikace. Prostřednictvím skriptu v jazyce JavaScript jsou zaznamenávány jeho akce. Tato data jsou následně odeslána na server, kde jsou dále zpracována a uložena do databáze. Celé schéma ilustruje obrázek 3.10.



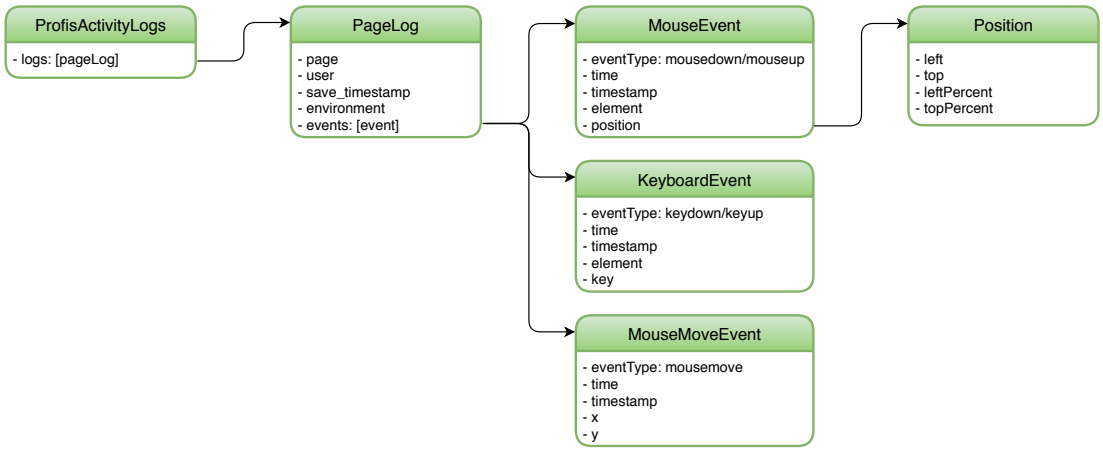
Obrázek 3.10: Schéma průběhu získávání dat.[1]

3.2.1 Záznam a odesílání dat na server

K záznamu všech výše popsaných uživatelských interakcí (viz sekce 3.1.2) používáme JavaScript. Příslušný skript je vložen do zdrojového kódu všech sledovaných stránek aplikace. Shromážděná data se odesílají na server ve formátu JSON.

Pro každou navštívenou stránku se vytváří nový objekt logu. Ten kromě seznamu akcí, ke kterým při uživatelově interakci došlo, obsahuje také URL adresu stránky, identifikátor uživatele, čas opuštění stránky a prostředí aplikace (zda jde o produkci, integraci, testovací verzi pro vývoj a podobně). Příchod na novou stránku aplikace i přenačtení aktuální stránky způsobí založení nového logu.

Data objektu, ve kterém se údaje o uživatelských aktivitách odesílají ve formátu JSON na server, ilustruje obrázek 3.11. Aby posílání dat při odchodu ze stránky uživatele nezdržovalo, je objekt logu uložen lokálně v prohlížeči a na server se odesílá až při načtení nové stránky aplikace. Pro odeslání je použita metoda *post* objektu AJAX([33]).



Obrázek 3.11: Formát objektu s daty odesílaného na server.

Na serveru, kam jsou odesílána získaná data ve formátu JSON, běží jednoduchý php skript. Jeho úkolem je přijatá data uložit do SQL databáze provozované na stejném serveru. K tomu je potřeba nejprve otevřít spojení s databází, pomocí připravených příkazů do ní data vložit, následně spojení ukončit a eventuálně do chybového souboru zaznamenat problémy, ke kterým v průběhu zpracování dat došlo.

3.2.2 Persistence dat

Pro ukládání záznamů o akcích uživatelů používáme relační databázi MySQL. Data jsou zde ukládána do čtyř tabulek.

[1]Použité ikony pocházejí z následujících zdrojů:
`https://freeicons.io/essential-collection-2/database-icon-icon-2`,
`https://freeicons.io/regular-life-icons/device-computer-icon-17798`,
`https://freeicons.io/games-and-technology-icons/computer-icon-1865`,
`https://freeicons.io/free-business-icons/globe-icon-5507`,
`https://www.php.net/download-logos.php`.

- První z nich, `page_log`, obsahuje obecné informace o návštěvě stránky jako je její URL nebo identifikace uživatele.
- Tabulka `mouse_events` obsahuje záznamy o klikání myši, tedy detailní informace k akcím stisknutí a uvolnění tlačítka myši. Jde především o údaje o poloze kliknutí v rámci prvku a časech událostí.
- Do tabulky `keyboard_events` se zaznamenávají data o vstupech z klávesnice. Důležitý je především identifikátor použité klávesy a čas akce.
- Poslední je tabulka `mousemovement_events`, kam se ukládají informace o polohách kurzoru při pohybu myši.

Detailní popis formátu použitých tabulek v databázi ukazují tabulky 3.1, 3.2, 3.3 a 3.4.

Název sloupce	Datový typ	Popis
`id`	INT	identifikátor záznamu generovaný databází
`user`	VARCHAR	identifikátor uživatele
`page`	VARCHAR	URL navštívené stránky
`savetime`	BIGINT	datum a čas opuštění stránky
`environment`	ENUM	prostředí aplikace (produkce, integrace a podobně)

Tabulka 3.1: Formát tabulky `page_log` s informacemi o návštěvě stránky.

3.3 Vytvoření datasetu

Poté, co jsme popsali, jaká data od uživatelů budeme mít v databázi k dispozici, můžeme rozhodnout o tom, jak přesně je při tvorbě autentizačního modelu použijeme.

Model by měl fungovat tak, že po zpracování informací o uživatelském sezení a deklarované identitě uživatele, vyhodnotí přístup buď jako oprávněný, nebo podvodný. V našich datech ovšem nemáme žádné implicitní rozdělení do sezení, pouze záznamy o jednotlivých návštěvách stránek.

3.3.1 Definice sezení

Možným řešením by bylo při odesílání dat na server přidat informaci o identifikátoru sezení z prohlížeče. To ovšem není vhodný přístup pro naši situaci, kdy většina uživatelů se na svém pracovním počítači z aplikace explicitně neodhlašuje a dlouhodobě tak pokračuje v rámci jediného sezení. Kdybychom proto použili sezení tak, jak je vidí prohlížeč, počet sezení by byl u většiny uživatelů velmi malý, přestože jsme data o nich sbírali déle než půl roku.

Problém neexistujících sezení proto řešíme tak, že uživatelovy návštěvy stránek rozdělíme do sezení explicitně sami. Řekneme, že dva po sobě jdoucí záznamy o stránkách, které uživatel navštívil, patří do stejného sezení, pokud je nedělí větší než hodinový časový odstup. Tento interval by měl být dostatečně velký vzhledem k tomu, že uživatel typicky s jednou stránkou pracuje pouze několik minut. Je tak krajně nepravděpodobné, že by při jeho souvislé práci trval přechod mezi

Název sloupce	Datový typ	Popis
`id`	INT	identifikátor záznamu generovaný databází
`log_id`	INT	reference na příslušný záznam o návštěvě stránky
`event`	ENUM	příznak, zda šlo o stisknutí, nebo uvolnění tlačítka myši
`datetime`	DATETIME	datum a čas akce
`t_stamp`	INT	časová značka od načtení stránky v ms
`element`	VARCHAR	xPath elementu stránky, kde k akci došlo
`position_left_abs`	SMALLINT	absolutní horizontální poloha kurzoru vzhledem k prvku
`position_top_abs`	SMALLINT	absolutní vertikální poloha kurzoru vzhledem k prvku
`position_left_perc`	TINYINT	relativní horizontální poloha kurzoru vzhledem k prvku v procentech
`position_top_perc`	TINYINT	relativní vertikální poloha kurzoru vzhledem k prvku v procentech

Tabulka 3.2: Formát tabulky `mouse_events` s informacemi o klikání myší.

Název sloupce	Datový typ	Popis
`id`	INT	identifikátor záznamu generovaný databází
`log_id`	INT	reference na příslušný záznam o návštěvě stránky
`event`	ENUM	příznak, zda šlo o stisknutí, nebo uvolnění klávesy
`datetime`	DATETIME	datum a čas akce
`t_stamp`	INT	časová značka od načtení stránky v ms
`element`	VARCHAR	xPath elementu stránky, kde k akci došlo
`pressed_key`	VARCHAR	identifikátor použité klávesy

Tabulka 3.3: Formát tabulky `keyboard_events` s informacemi o vstupech z klávesnice.

Název sloupce	Datový typ	Popis
`id`	INT	identifikátor záznamu generovaný databází
`log_id`	INT	reference na příslušný záznam o návštěvě stránky
`datetime`	DATETIME	datum a čas akce
`t_stamp`	INT	časová značka od načtení stránky v ms
`x`	SMALLINT	x-ová souřadnice kurzoru v pixelech
`y`	SMALLINT	y-ová souřadnice kurzoru v pixelech

Tabulka 3.4: Formát tabulky `mousemovement_events` s informacemi o pohybu myši.

některými dvěma stránkami déle než hodinu. Připomeňme, že i nové načtení aktuální stránky se považuje za přechod mezi stránkami a vede k vytvoření nového záznamu o návštěvě stránky.

Výše popsaným postupem tak rozdělíme všechny uživatelské návštěvy stránek do sezení s tím, že delší než hodinová pauza mezi záznamy vynucuje vždy začátek nového sezení. Jako jednoznačný identifikátor sezení použijeme identifikátor prvního záznamu o návštěvě stránky, který k sezení patří.

3.3.2 Formát datasetu

Z dat tabulek (popsaných v sekci 3.2.2) jsme vytvořili trojici navazujících datasetů. První z nich obsahuje informace o práci s klávesnicí, druhý poskytuje údaje o klikání uživatelů a poslední je odvozen od tabulky se záznamy o pohybu myši. Záznamy ve všech třech datasetech propojují identifikátory uživatelů, jejich sezení a stránek navštívených v jejich průběhu. Přesný formát jednotlivých datasetů shrnují tabulky 3.5, 3.6 a 3.7.

Název sloupce	Popis
`user_id`	identifikátor uživatele
`session_id`	identifikátor sezení
`page_log_id`	identifikátor záznamu o návštěvě stránky
`keydown_time`	čas stisku klávesy (v ms od načtení stránky)
`keyup_time`	čas uvolnění klávesy (v ms od načtení stránky)
`key`	označení klávesy
`key_presses_count`	počet zaznamenaných stisků klávesy před jejím uvolněním

Tabulka 3.5: Struktura datasetu práce s klávesnicí.

Název sloupce	Popis
`user_id`	identifikátor uživatele
`session_id`	identifikátor sezení
`page_log_id`	identifikátor záznamu o návštěvě stránky
`start_time`	čas stisku tlačítka myši (v ms od načtení stránky)
`end_time`	čas uvolnění tlačítka myši (v ms od načtení stránky)
`element`	xPath elementu stránky, na který uživatel klikl
`x_rel`	x-ová pozice kliknutí relativně k velikosti elementu (v procentech)
`y_rel`	y-ová pozice kliknutí relativně k velikosti elementu (v procentech)

Tabulka 3.6: Struktura datasetu klikání.

3.3.3 Srovnání s jinými studiemi

V porovnání s podmínkami, za kterých se sbírala data v jiných studiích, popsanými v sekci 2.1, je náš dataset mezi ostatními relativně ojedinělý z několika důvodů.

Zaprvé počtem uživatelů, jejichž data máme k dispozici. Celkově dataset obsahuje záznamy od více než 500 uživatelů, což je řádově více než ve většině ostatních studií. Objem dat od každého z nich se ovšem výrazně liší. Blíže se počty sezení od jednotlivých uživatelů budeme zabývat v části 3.4.2.

Název sloupce	Popis
`user_id`	identifikátor uživatele
`session_id`	identifikátor sezení
`page_log_id`	identifikátor záznamu o návštěvě stránky,
`page_name`	URL navštívené stránky
`time`	čas záznamu (v ms od načtení stránky)
`x`	x-ová poloha myši (v pixelech)
`y`	y-ová poloha myši (v pixelech)

Tabulka 3.7: Struktura datasetu pohybu myši.

Druhým specifikem našich dat je dlouhá doba, po kterou se od uživatelů sbírala. Jde o více než sedm měsíců od konce října 2019 do konce května 2020. V takto dlouhém intervalu by už mělo být možné sledovat vliv změn chování uživatelů v čase.

Výjimečná je také situace, kdy data pocházejí z konkrétní aplikace, která není vytvořena pouze pro účely studie, ale uživatelé s ní reálně běžně pracují. Z popisu sledovaných stránek aplikace (sekce 3.1.1) je možné udělat si představu o tom, jakým způsobem je uživatelé používají.

Díky tomu, že sledujeme uživatelské interakce s aplikací, již detailně známe, můžeme zaznamenávat rozšiřující informace o uživatelských akcích jako například, s kterým konkrétním prvkem stránky pracují, kam v rámci něj klikají nebo to, jak vypadá stránka, na které uživatel pohybuje myší. V příznacích se tak můžeme zaměřovat třeba na jeden typ prvku, konkrétní prvek, nebo sledovat chování pouze na vybrané stránce.

Prostředí, v němž uživatelé svá data poskytovali, nebylo nijak kontrolováno ve smyslu fixace použitých vstupních ani výstupních zařízení. Uživatelé tak mohli pracovat s myší nebo touchpadem, různými druhy klávesnice nebo různě nastaveným rozlišením displeje. Ani data jednoho uživatele nemusí pocházet vždy ze stejného zařízení. Nejpravděpodobnější je, že ve většině případů uživatel používá aplikaci vždy ze stejného pracovního notebooku, ale nic mu nebrání připojit se k ní odkudkoli.

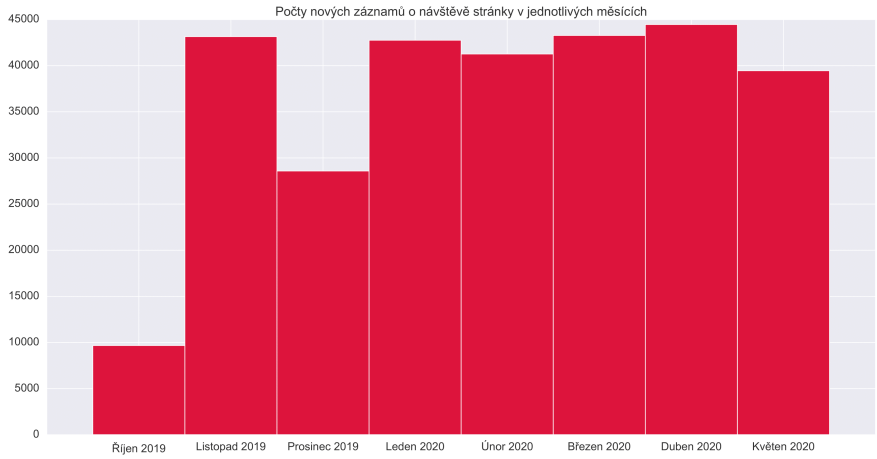
3.4 Statistiky získaných dat

V této části budeme analyzovat množství dat shromážděných v průběhu sběru od konce října 2019 do konce května 2020. Nejprve se budeme zabývat pouze záznamy v jednotlivých tabulkách. Ve druhé části budeme zkoumat počty sezení, jak jsme je definovali v sekci 3.3.1.

3.4.1 Počty záznamů

Nejdříve se zaměříme na to, jak se vyvíjel počet záznamů o návštěvách stránek uživateli v průběhu sledovaného období. To ukazuje graf na obrázku 3.12. Z něj vidíme, že kromě října, kdy sběr začal až koncem měsíce, je měsíční přírůstek počtu záznamů stabilní, s průměrnou hodnotou přes 40 tisíc záznamů. Jediný výraznější propad nastal v prosinci a byl zapříčiněn hlavně Vánocemi a častějším

čerpáním dovolené. Za sledované období jsme shromáždili celkem 292 568 těchto záznamů.

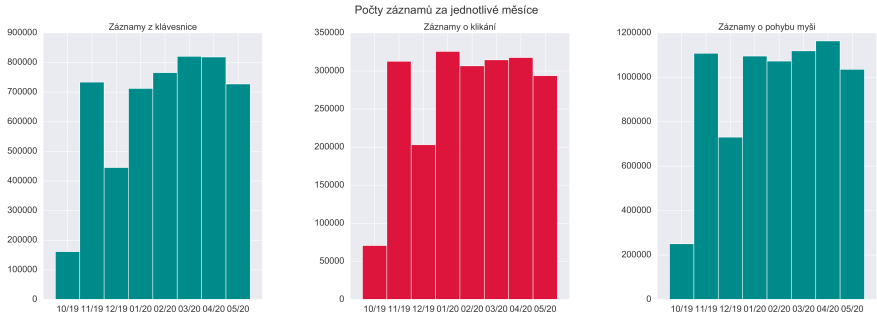


Obrázek 3.12: Záznamy o návštěvě stránek po měsících.

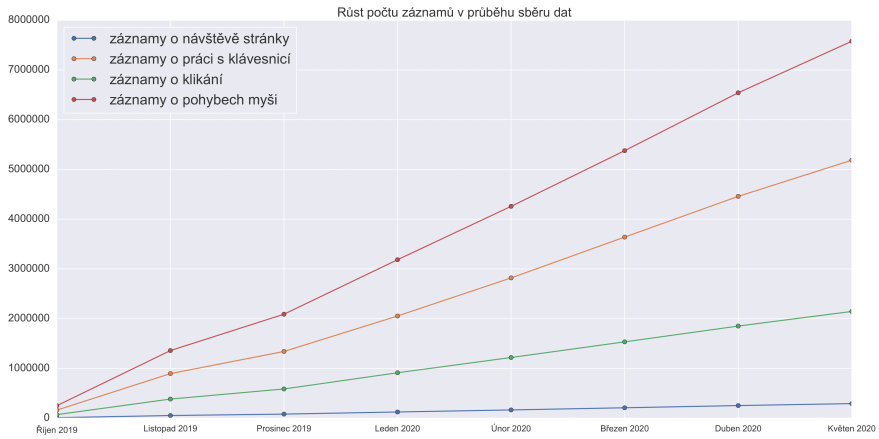
Stejnou charakteristiku pro všechny ostatní tabulky databáze zobrazují grafy na obrázku 3.13. I zde můžeme vidět stejný trend jako v předchozím případě. Počty záznamů jsou vyrovnané s výjimkou měsíce října a menšího poklesu v prosinci. Průměrný měsíční nárůst pro záznamy z klávesnice činí asi 718 tisíc. Pro záznamy o klikání myší je tento počet 300 tisíc. Podle očekávání nejvíce přibývá záznamů o pohybu myši. Měsíčně jde o průměrně více než milion nových záznamů.

V grafu 3.14 můžeme vidět vývoj celkového počtu záznamů ve všech tabulkách databáze. Nutně nejpomaleji narůstá počet záznamů o návštěvách stránek. Průměrně na jednu návštěvu stránky vychází 18 vstupů z klávesnice, 7 záznamů o klikání a 26 zaznamenaných pohybů myši. V případě interakcí s klávesnicí a při klikání vznikají většinou při jedné akci dva záznamy (jeden o stisku a druhý o uvolnění tlačítka myši nebo klávesy), reálný počet kliknutí nebo použitých kláves je tedy poloviční oproti počtu záznamů v příslušných tabulkách.

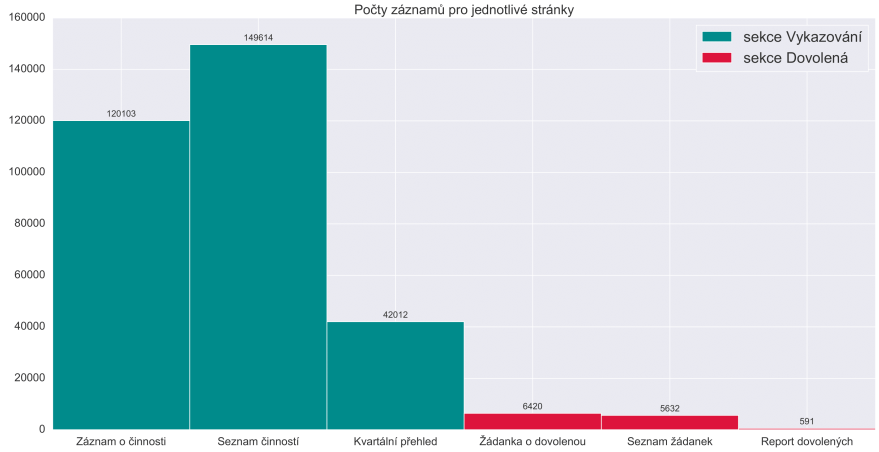
Rozdělení stránek, ze kterých záznamy pocházejí, shrnuje graf na obrázku 3.15. Ukazuje se, že drtivá většina navštívených stránek je v sekci Vykazování a pouze 4 % záznamů se týkají stránek s žádankami o dovolenou. Nejnavštěvovanější stránkou je Seznam činností. Odsud pochází téměř polovina (46 %) všech záznamů. Tato skutečnost může být částečně zapříčiněna tím, že na této stránce



Obrázek 3.13: Počty ostatních záznamů v databázi po měsících.



Obrázek 3.14: Růst počtu záznamů ve všech tabulkách databáze.



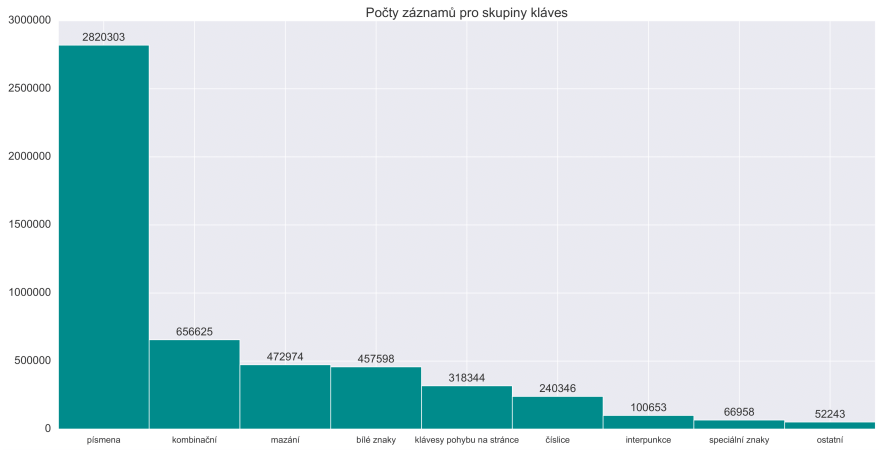
Obrázek 3.15: Rozdělení záznamů o stránkách.

mají uživatelé k dispozici nastavení filtrování položek v seznamu a každá aplikace tohoto filtru způsobí přenačtení stránky a s ním spojené vytvoření nového záznamu o její návštěvě.

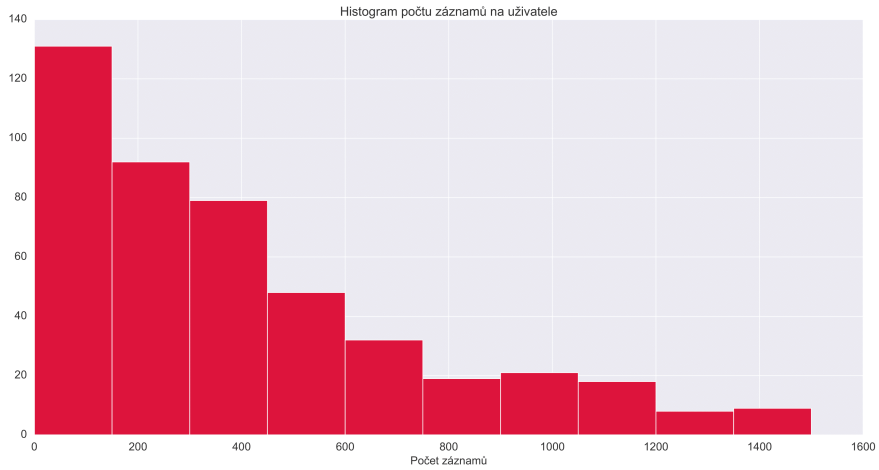
V tabulce se záznamy o práci s klávesnicí může být zajímavé podívat se, jaké druhy kláves uživatelé nejčastěji používají. Tyto informace zprostředkovává graf na obrázku 3.16. Nepřekvapivě jsou nejvíce používanou skupinou kláves písmena. Mezi 15 nejčastěji používanými klávesami jsou kromě nich pouze čtyři jiné klávesy, a to Backspace, Shift, Control a mezerník. Nejméně zastoupenými skupinami jsou pak interpunkční znaménka a další speciální znaky.

Pořadí mezi ostatními druhy kláves už tak nasnadě není. Druhou nejfrekventovanější skupinou jsou kombinační klávesy (Shift, Control a Alt), dále klávesy pro mazání (Backspace a Delete) a bílé znaky. Následují klávesy pohybu na stránce (například šipky, PageUp nebo Home) a číslice.

Posledním bodem, na který se zaměříme, je počet uživatelů, od kterých záznamy pocházejí. Celkově máme k dispozici data 518 různých uživatelů, ovšem počty zaznamenaných navštívených stránek se mezi jednotlivými uživateli diametrálně liší. Jejich histogram ukazuje obrázek 3.17. Vidíme, že největší část uživatelů má záznamů relativně málo. Do skupiny s méně než 200 záznamy jich patří 160. Střední počet záznamů od 200 do 1 000 máme k dispozici od 255 uživatelů.



Obrázek 3.16: Četnosti různých typů kláves mezi záznamy.



Obrázek 3.17: Počty záznamů uživatelů.

A pouze asi pětina (103) uživatelů má více než 1 000 záznamů.

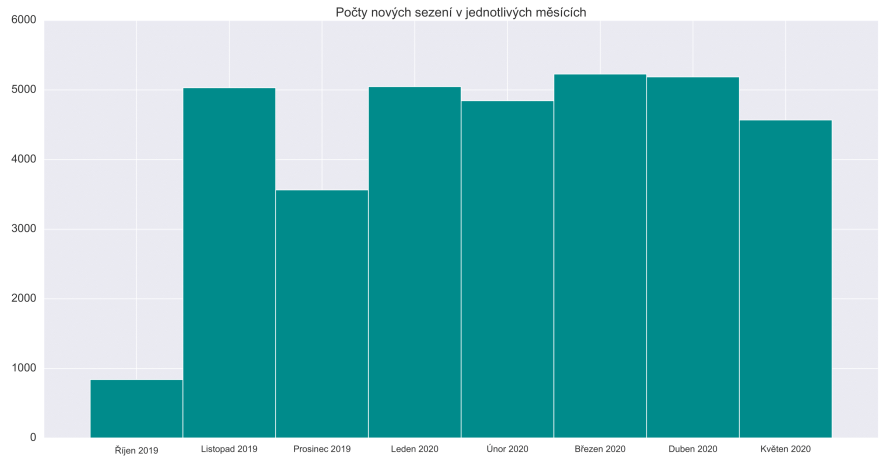
3.4.2 Počty sezení

Nyní budeme ve statistikách zohledňovat rozdělení záznamů do příslušných sezení tak, jak jsme je zadefinovali v sekci 3.3.1.

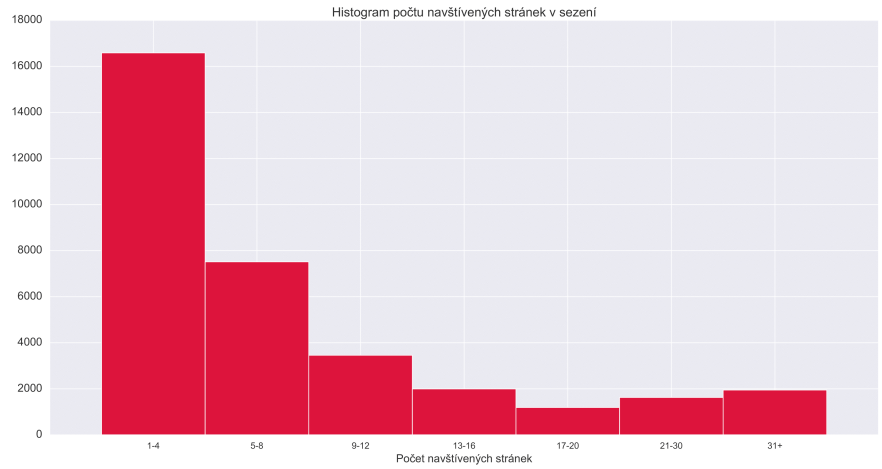
Nejprve se podíváme na přírůstky v počtech sezení v průběhu sběru dat, které ukazuje obrázek 3.18. Na první pohled tento graf vypadá velmi podobně jako v případě počtu záznamů o navštívených stránkách (3.12). V tomto případě je ovšem průměrný měsíční přírůstek méně než 5 000 nových sezení, zatímco u nově navštívených stránek šlo o 40 000. Celkový počet sezení zaznamenaných ve sledovaném období je 34 310.

Průměrný počet stránek navštívených v průběhu jednoho sezení je 8,5. Detailnější statistiku počtu stránek, které uživatel během sezení navštíví, nabízí obrázek 3.19. Vidíme, že většina sezení je z hlediska počtu navštívených stránek velmi krátká. Je zde na místě zdůraznit, že pokud se stránka, kterou uživatel prohlíží, přenačte, vytvoří se nový záznam, i když z pohledu uživatele k přechodu mezi stránkami nedošlo.

Počty událostí zaznamenaných v průběhu sezení ukazují grafy na obrázku



Obrázek 3.18: Počty sezení po měsících.

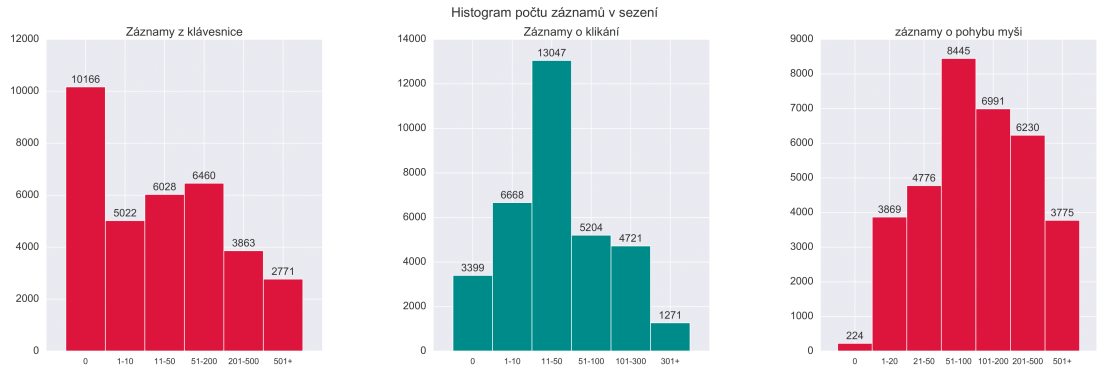


Obrázek 3.19: Počet navštívených stránek za sezení.

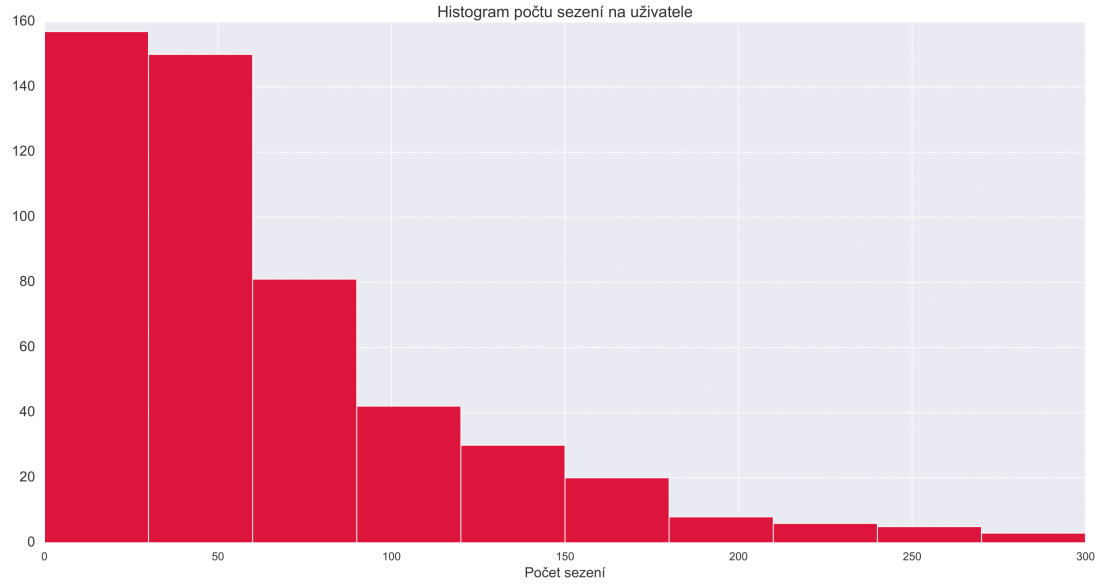
3.20. Tyto počty jsou klíčové z hlediska extrakce příznaků pro sezení. Například pro skoro 30 % sezení nebude možné použít žádné příznaky odvozené od práce s klávesnicí, protože s ní uživatel vůbec nepracoval. Pokud bychom požadovali dlouhý vstup z klávesnice o minimálně 250 použitých klávesách, pak tento požadavek splní pouze 8 % sezení. V případě klikání je zcela bez příslušných záznamů pouze 10 % sezení a nejlépe v tomto ohledu vychází záznamy o pohybu myši, kterých je v naprosté většině sezení dostatek.

Pokud budeme sledovat medián počtu záznamů každé kategorie, pak na jedno sezení připadá 16 vstupů z klávesnice, 24 událostí spojených s klikáním a 98 záznamů polohy myši. (Pro klikání i klávesnicí je opět počet reálných akcí uživatele poloviční oproti počtu záznamů.)

Poslední graf na obrázku 3.21 ukazuje histogram počtu sezení, která máme pro jednotlivé uživatele k dispozici. U 225 z celkového počtu 518 uživatelů máme data z méně než 30 sezení. Rozumný počet sezení mezi 30 a 200 sezeními máme k dispozici pro 267 uživatelů. Zbývajících 26 uživatelů pak má 200 a více sezení.



Obrázek 3.20: Počet záznamů o událostech v sezení.



Obrázek 3.21: Počty sezení uživatelů.

4. Experimentální evaluace příznaků

V této kapitole postupně přiblížíme jednotlivé příznaky, které jsme zvažovali jako diskriminační charakteristiky uživatelů. Pro každý z nich popíšeme vstupní data, ze kterých příznak extrahujeme, a také postup jejich předzpracování (především odstranění odlehlých hodnot). Následně ukážeme vizualizace hodnot příznaku mezi uživateli.

Diskriminační potenciál příznaku pak odhadneme porovnáním toho, jak je daný příznak stabilní v sezeních jednoho uživatele, s tím, jak se liší mezi sezeními různých uživatelů. Na závěr zhodnotíme, zda je v naší situaci výhodné příznak v autentizačním modelu použít.

4.1 Určení diskriminačního potenciálu

Nejdůležitějším krokem při výběru užitečných příznaků je odhad diskriminačního potenciál příznaku, tedy toho jak dobře lze pomocí něj odlišit sezení jednotlivých uživatelů.

Vzdálenost sezení x a y budeme značit $dist(x,y)$ a budeme jí rozumět absolutní hodnotu rozdílu zkoumaného příznaku mezi sezeními. Množinu všech uživatelů, jejichž data máme k dispozici, označíme $users$. Vhodnost příznaku otestujeme dvěma způsoby.

První z nich spočívá v porovnání průměrné vzdálenosti mezi sezeními jednoho uživatele ve srovnání se vzdáleností mezi sezeními různých uživatelů. Nejprve pro každého uživatele náhodně vybereme k dvojic jeho sezení $(o_{1,1},o_{1,2})$,$(o_{2,1},o_{2,2})$,..., $(o_{k,1},o_{k,2})$ a z nich určíme průměrnou vzdálenost mezi vlastními sezeními uživatele jako

$$ownDist_k = \frac{\sum_{i=1}^{k} dist(o_{i,1},o_{i,2})}{k}.$$

Obdobně poté určíme průměrnou vzdálenost uživatelových sezení od sezení ostatních. Do dvojic tentokrát vybereme vždy jedno vlastní a jedno cizí sezení (o_1,f_1),(o_2,f_2),...,(o_k,f_k) a vzdálenost od cizích sezení vypočítáme jako

$$foreignDist_k = \frac{\sum_{i=1}^{k} dist(o_i,f_i)}{k}.$$

Výsledky pro všechny uživatele dohromady můžeme shrnout tak, že spočítáme buď absolutní, nebo relativní průměrný rozdíl vzdálenosti mezi sezeními jednoho uživatele ($ownDist_k$) a vzdálenosti jeho sezení od ostatních ($foreignDist_k$).

Absolutní vzdálenost definujeme jako

$$\delta_{abs,k} = \frac{\sum_{user \in users} foreignDist_k(user) - ownDist_k(user)}{|users|}.$$

Tato veličina má stejnou jednotku jako zkoumaný příznak. Oproti tomu relativní vzdálenost je bezrozměrná a definujeme ji jako

$$\delta_{rel,k} = \frac{\sum\limits_{user \in users} \frac{foreignDist_k(user) - ownDist_k(user)}{ownDist_k(user)}}{|users|}.$$

Můžeme ji interpretovat jako očekávaný procentuální nárůst vzdálenosti, pokud místo dvou uživatelových sezení poměřujeme jeho sezení s cizím. Čím vyšší je tato hodnota, tím větší je variabilita v příznaku mezi uživateli ve srovnání s variabilitou mezi sezeními jednotlivce a tím užitečnější je zkoumaný příznak.

Druhý způsob posouzení diskriminačního potenciálu spočívá v tom, že počítáme uživatelova sezení mezi nejbližšími sousedy daného sezení. Zavedeme proto pro každého uživatele funkci $neigh_{user,n}(x)$, která určuje kolik sezení uživatele $user$ se nachází mezi n nejbližšími sousedy sezení x. Nejbližšími sousedy z hlediska daného příznaku rozumíme sezení, která se v tomto příznaku od sledovaného sezení nejméně liší.

Pro každého uživatele $user$ náhodně zvolíme k' jeho sezení o_1,o_2,...,$o_{k'}$ a k' cizích sezení f_1,f_2,...,$f_{k'}$. Pro každé sezení x z vybraných pak spočítáme počty uživatelových sezení mezi n nejbližšími sousedy $neigh_{user,n}(x)$.

Všechny výsledky uživatele u pro každou z variant (okolí vlastního, nebo cizího sezení) můžeme sledovat zvlášť, nebo je agregovat do jediné průměrné hodnoty

$$ownNeigh_{k',n}(user) = \frac{\sum\limits_{i=1}^{k'} neigh_{user,n}(o_i)}{k'}$$

pro okolí vlastních sezení a

$$foreignNeigh_{k',n}(user) = \frac{\sum\limits_{i=1}^{k'} neigh_{user,n}(f_i)}{k'}$$

pro okolí cizích sezení.

Z těchto hodnot můžeme dále určit, kolikrát více je uživatelových sezení v okolí jeho vlastních sezení oproti okolí cizích sezení, tedy

$$\frac{ownNeigh_{k',n}(user)}{foreignNeigh_{k',n}(user)}.$$

Abychom měli jedinou hodnotu, podle které budeme při tomto postupu diskriminační potenciál posuzovat, použijeme průměrnou hodnotu této charakteristiky přes všechny uživatele

$$\Delta_{k',n} = \frac{\sum\limits_{user \in users} \frac{ownNeigh_{k',n}(user)}{foreignNeigh_{k',n}(user)}}{|users|}.$$

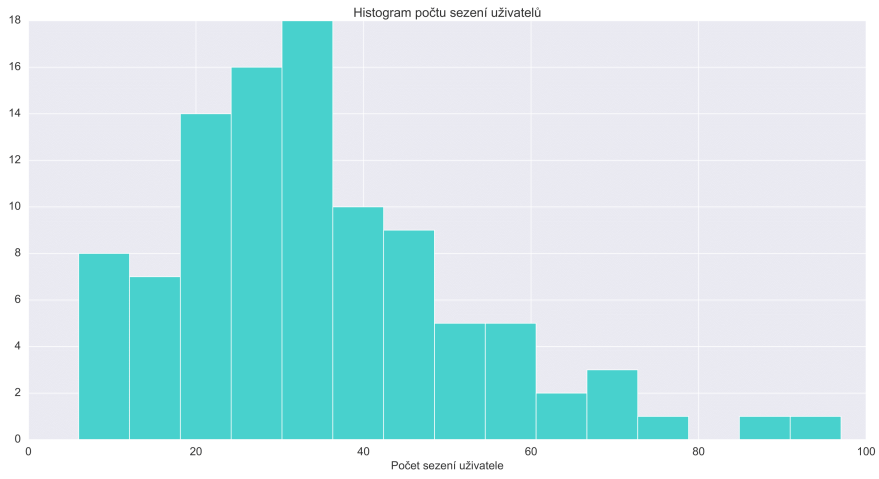
I zde se potenciálně užitečné příznaky vyznačují co nejvyšší hodnotou Δ. Ta odpovídá tomu, že uživatelská sezení se v příznakovém prostoru shlukují k sobě, tedy jsou si navzájem podobnější než dvě náhodná sezení.

4.2 Dynamika klávesnice

Nejprve se budeme zabývat příznaky, které lze extrahovat ze záznamů o interakci uživatele s klávesnicí. Budeme vycházet ze vstupního souboru, který obsahuje 271 606 záznamů o těchto akcích od vzorku 100 různých uživatelů z období od konce října 2019 do února 2020. Jde o část dat obsažených v datasetu práce s klávesnicí popsaném v sekci 3.3.2. Soubor se záznamy akcí obsahuje následující informace:

- identifikátor uživatele,
- identifikátor sezení (odpovídá identifikátoru záznamu z první navštívené stránky),
- identifikátor záznamu o návštěvě stránky,
- čas stisku klávesy (v ms od načtení stránky),
- čas uvolnění klávesy (v ms od načtení stránky),
- označení klávesy,
- počet zaznamenaných stisků klávesy před jejím uvolněním.

Pro další analýzu je důležité, kolik sezení máme pro jednotlivé uživatele v záznamech k dispozici. To můžeme vidět v histogramu na obrázku 4.1. Jako minimální požadovaný počet sezení jsme stanovili 30. Tuto podmínku splňuje 59 z původního 100 uživatelů. Při výběru příznaků budeme dále pracovat jen s daty od nich. Tím se celkový počet záznamů sníží z původních 271 606 na 192 787 (71 % původního počtu).



Obrázek 4.1: Počty sezení uživatelů ve vstupním souboru.

4.2.1 Délka držení klávesy

Prvním uvažovaným příznakem je délka stisku klávesy na klávesnici. Jde o jeden z nejčastěji používaných příznaků pro analýzu práce s klávesnicí. Je zmíněna téměř ve všech článcích na toto téma (blíže viz 2.2.1). Tato charakteristika typicky vykazuje konzistenci v rámci dat jediného uživatele a dostatečnou odlišnost při porovnávání dat různých uživatelů.

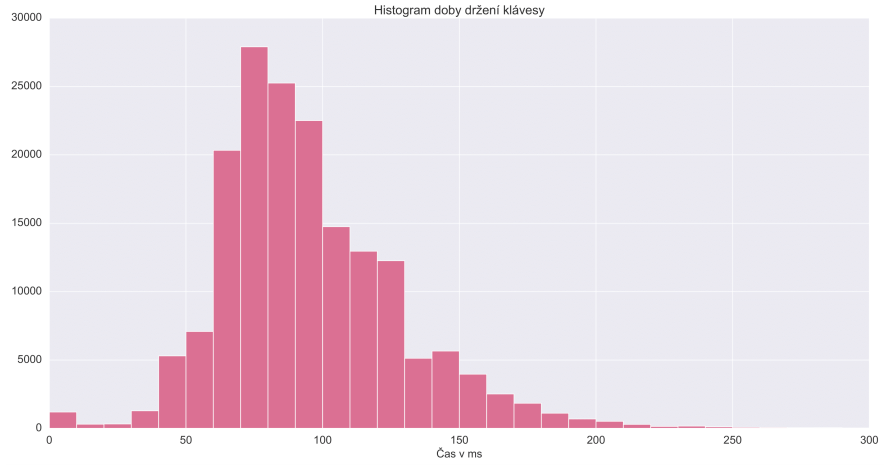
Předzpracování

V rámci předzpracování vypočítáme pro každý záznam dobu držení v milisekundách jako rozdíl mezi časem uvolnění a stisknutí klávesy.

Protože nás zajímá pouze doba, po kterou uživatel drží klávesu, pokud ji chtěl pouze stisknout a hned uvolnit, odstraníme z dat všechny záznamy, kde uživatel drží klávesu cíleně dlouho, aby se zopakoval její efekt, například při mazání více znaků držením kláves Delete a Backspace nebo navigaci pomocí šipek. Po tomto kroku zůstává 190 568 (99 %) záznamů z původních dat vybraných uživatelů.

Další vzorec chování, které je běžné, ale do dat pro extrakci doby držení klávesy ho uvažovat nechceme, je práce s kontrolními klávesami jako je Shift, Alt nebo Control. Ty se používají pouze v kombinaci s další klávesou a uživatelé je proto drží záměrně dlouho. Po jejich vyloučení zbývá 178 796 záznamů (93 % původních).

Nakonec eliminujeme také odlehlé hodnoty. Ty mohly být do dat zaneseny chybou při zaznamenávání událostí v prohlížeči nebo neočekávaným chováním uživatelů. Z histogramu 4.2 vidíme, že naprostá většina zaznamenaných hodnot leží v intervalu od 40 do 200 ms. Tyto hodnoty proto stanovíme jako hranice přípustných hodnot. Po úpravě nám zbývá 169 365 záznamů (88 % původního počtu). Přitom uživatelé přijdou průměrně o 14 % svých záznamů a pouze čtyři z vybraných 59 ztratí více než čtvrtinu svých záznamů.

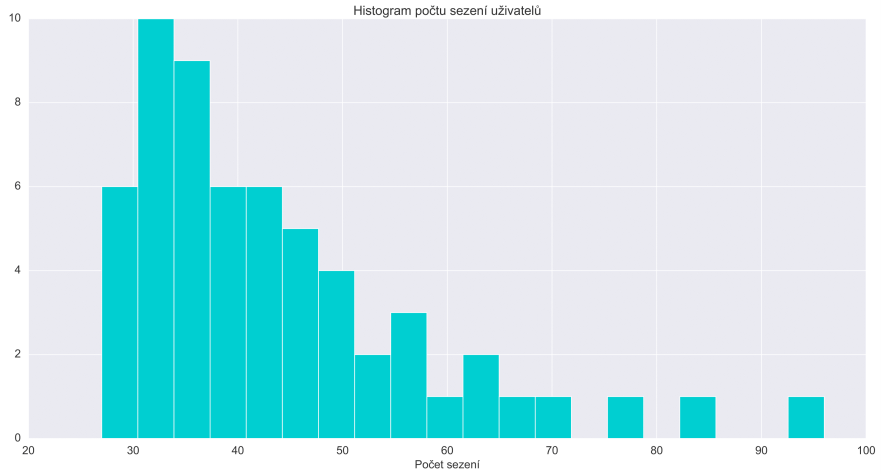


Obrázek 4.2: Doba držení klávesy.

Rozdělení do sezení

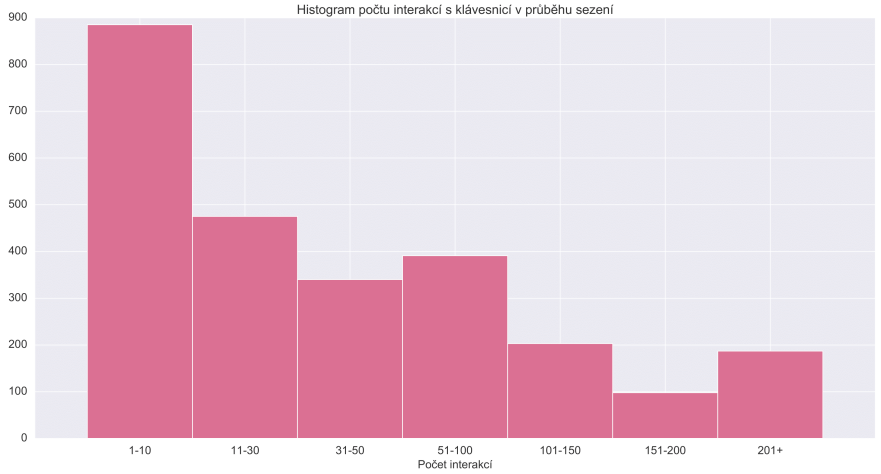
Nyní můžeme zbylé záznamy o jednotlivých akcích na klávesnici rozdělit do uživatelských sezení (pro definici sezení v našem kontextu viz 3.3.1). Četnosti se-

zení jednotlivých uživatelů shrnuje histogram 4.3. Jejich počty se pohybují v intervalu od 27 do 96 s průměrnou hodnotou 44.



Obrázek 4.3: Počty sezení uživatelů.

Důležitým faktorem je také počet záznamů o práci s klávesnicí v jednotlivých sezeních. Jak vidíme v histogramu na obrázku 4.4, ve většině sezení použil uživatel klávesnici méně než 50 krát. Nejčastěji se počet akcí v průběhu sezení nachází v intervalu od 0 do 20. Při bližším zkoumání můžeme zjistit, že v tom případě uživatelé často používali pouze klávesy pro navigaci na stránce nebo zadávali počáteční písmena slova pro rychlejší vyhledávání v rozbalovacím seznamu.



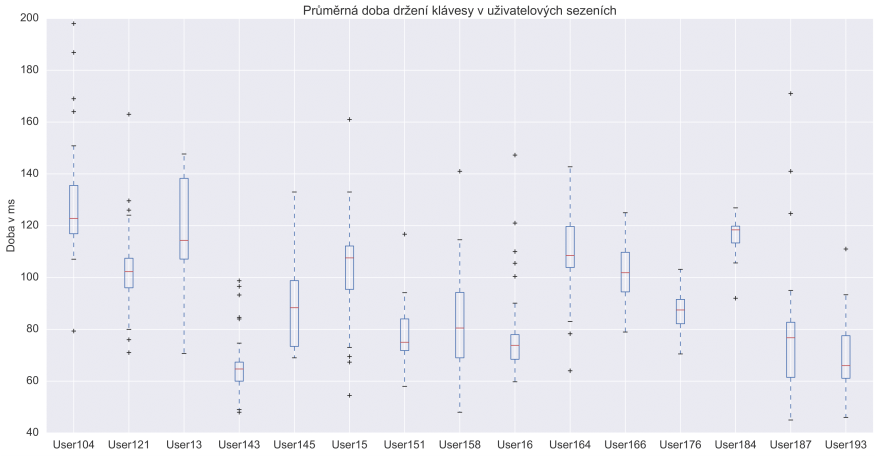
Obrázek 4.4: Počty stisků kláves v průběhu sezení.

Explorace

Pro každé sezení určíme průměrnou dobu držení klávesy ze všech záznamů, které se k sezení vztahují. Následně můžeme všechny tyto průměry pro jednotlivé uživatele shrnout do podoby boxplotu 4.5 (pro přehlednost zobrazujeme pouze data prvních 15 uživatelů).

Už tento graf ukazuje, že doba držení klávesy by mohla být dobrým charakterizujícím příznakem pro rozlišení uživatelů, neboť boxploty jednotlivých uživatelů

se od sebe výrazně liší. Většina z nich je také relativně kompaktní, což ukazuje stabilitu příznaku v rámci uživatelových sezení.

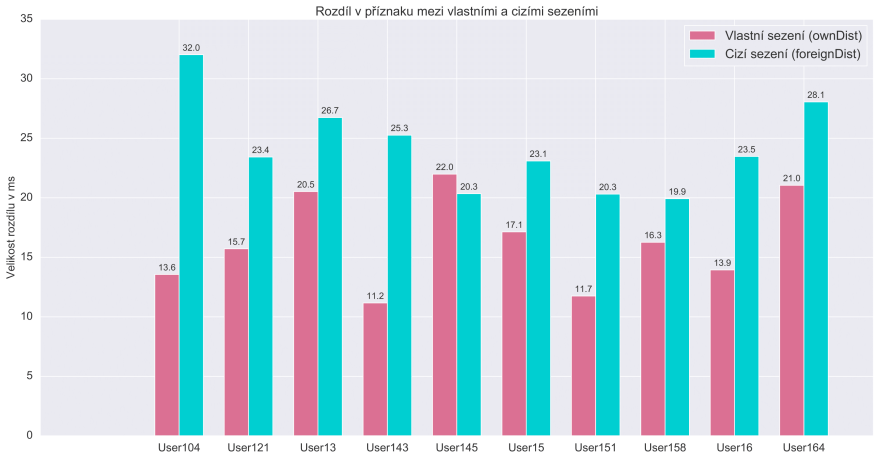


Obrázek 4.5: Průměrná doba držení klávesy u uživatelů.

Diskriminační potenciál

Nejprve budeme sledovat rozdíly ve vzdálenosti mezi vlastním a cizím sezením a vlastními sezeními navzájem. Pro každou z variant pro každého uživatele náhodně vybereme 25 dvojic sezení, mezi kterými budeme rozdíl měřit, a spočítáme hodnoty $foreignDist_{25}$ a $ownDist_{25}$.

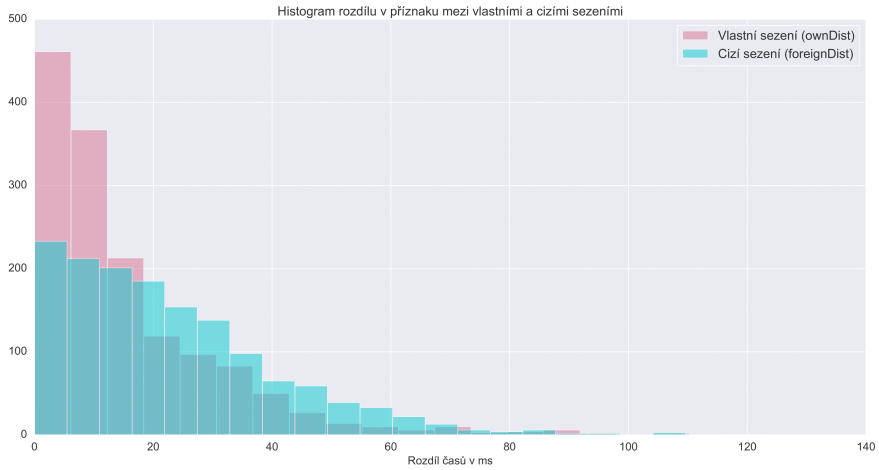
Tyto hodnoty pro vzorek uživatelů ukazuje graf 4.6. Z něj vidíme, že v naprosté většině případů jsou si vlastní sezení uživatele podobnější než jeho sezení porovnána s cizími. Průměrně mezi všemi 59 uživateli je absolutní rozdíl při porovnání vzdáleností cizích a vlastních sezení $\delta_{abs,25} = 7\,\text{ms}$, což odpovídá nárůstu průměrné vzdálenosti o $\delta_{rel,25} = 69\ \%$, pokud místo sezení jednoho uživatele porovnáváme navzájem sezení různých uživatelů.



Obrázek 4.6: Porovnání podobností vlastních a cizích sezení.

Stejný závěr ilustruje i histogram na obrázku 4.7. Zde jsou shrnuty výsledky pro všechny uživatele dohromady a sledujeme četnosti různých velikostí rozdílu.

Rozdíly mezi vlastními sezeními ($ownDist_{25}$), vyznačené růžově, v drtivé většině případů zůstávají v okolí 0. Oproti tomu při vzájemném porovnávání sezení různých uživatelů ($foreignDist_{25}$) jsou časté i výrazně větší rozdíly.

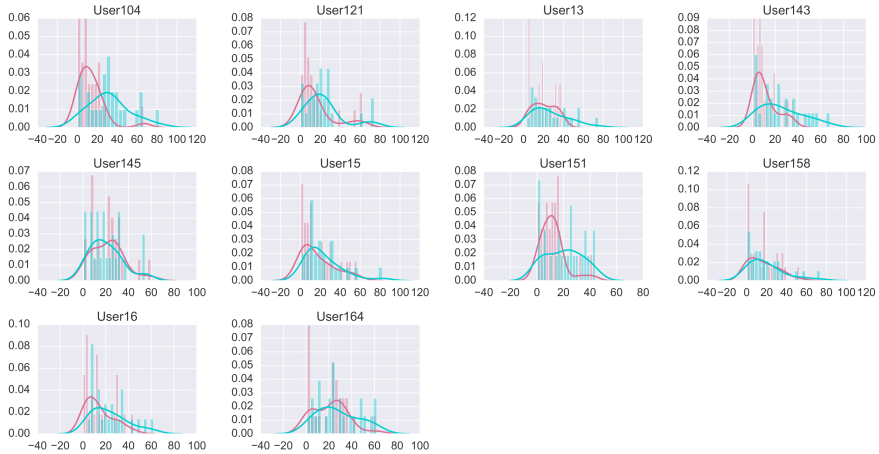


Obrázek 4.7: Rozdíly v době držení klávesy mezi sezeními.

Pokud data z předchozího grafu rozdělíme podle jednotlivých uživatelů a proložíme je odhadem hustoty (*kernel density estimation* [16]) vznikne obrázek 4.8.

Co nejlepší diskriminační potenciál se vyznačuje výraznými vrcholy růžové v okolí 0, což znamená, že všechna sezení uživatele jsou si z hlediska doby držení kláves hodně podobná. Naopak pro modrou křivku je nejlepší vrchol daleko od 0, což odpovídá tomu, že uživatelova sezení se od sezení ostatních výrazně liší.

Na obrázku vidíme, že tento ideální vzor se projevil asi u poloviny zobrazených uživatelů, u zbytku nejsou rozdíly mezi vlastními a cizími sezeními příliš výrazné.



Obrázek 4.8: Podobnosti vlastních a cizích sezení pro uživatele.

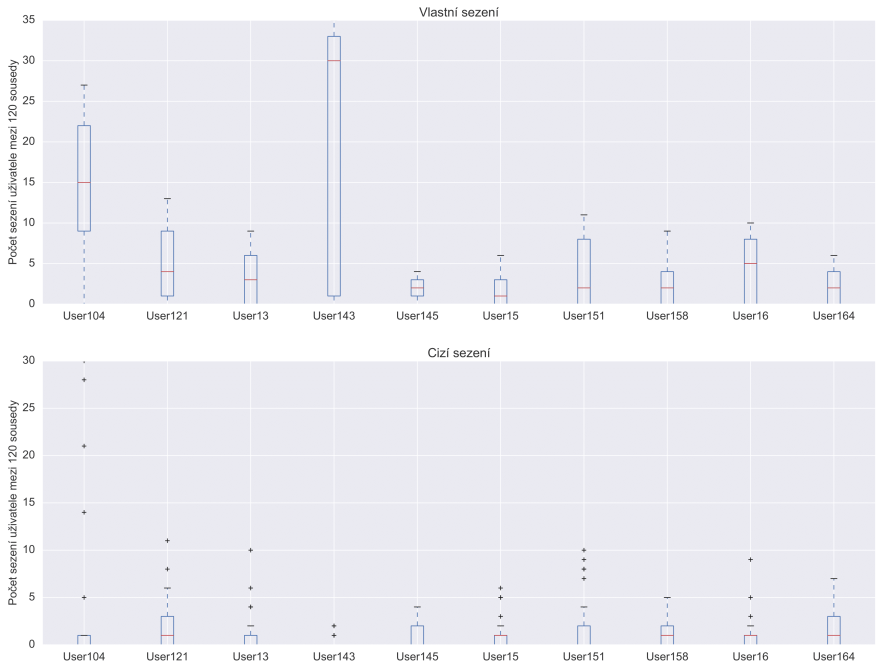
Druhým způsobem, kterým odhadujeme diskriminační potenciál příznaku, je sledování nejbližších sousedů jednotlivých sezení. Blíže je tento postup popsán v sekci 4.1. V tomto případě za nejbližší sousedy považujeme sezení, která se nejméně liší v průměrné době držení klávesy.

Při analýze budeme za okolí považovat 120 nejbližších sousedů a pro každého uživatele budeme sledovat okolí 20 vlastních a 20 cizích sezení. Výsledky tohoto

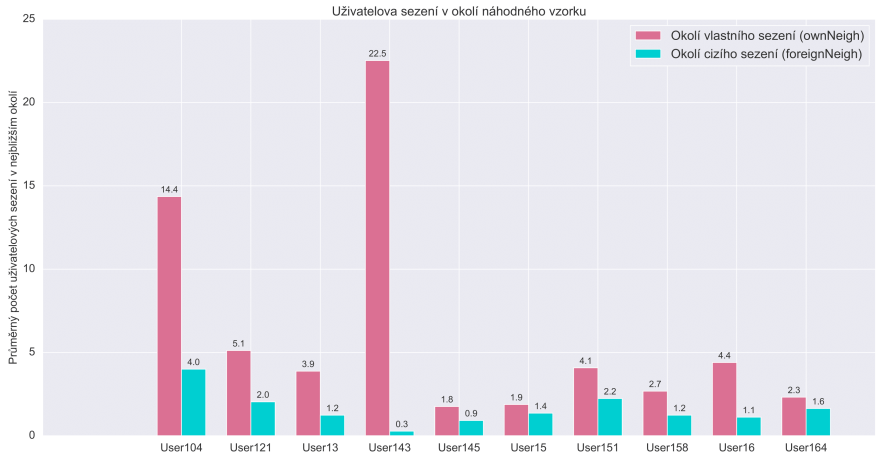
postupu, tedy hodnoty $ownNeigh_{20,120}$ a $foreignNeigh_{20,120}$, ukazují grafy na obrázcích 4.9 a 4.10.

V prvním z nich vidíme boxplot zastoupení dalších uživatelových sezení v okolí výchozího cizího nebo vlastního sezení. Druhý obrázek pak ukazuje průměrné počty uživatelových sezení v okolí pro obě tyto možnosti. Vidíme z něj, že skoro ve všech případech je okolo vlastních sezení alespoň dvakrát více dalších uživatelových sezení než v okolí náhodného cizího sezení.

Mezi všemi 59 testovanými uživateli je potom $\Delta_{20,120} = 3{,}6$, což znamená, že je 3,6 krát více uživatelových sezení v okolí jeho vlastního sezení než v okolí náhodného cizího sezení.



Obrázek 4.9: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.10: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

Závěr

Předchozí analýza ukazuje, že délka doby držení klávesy zůstává v sezeních jednoho uživatele podobná, zatímco mezi sezeními různých uživatelů se většinou výrazně liší. Je tedy vhodné tuto charakteristiku při ověřování identity uživatelů využít.

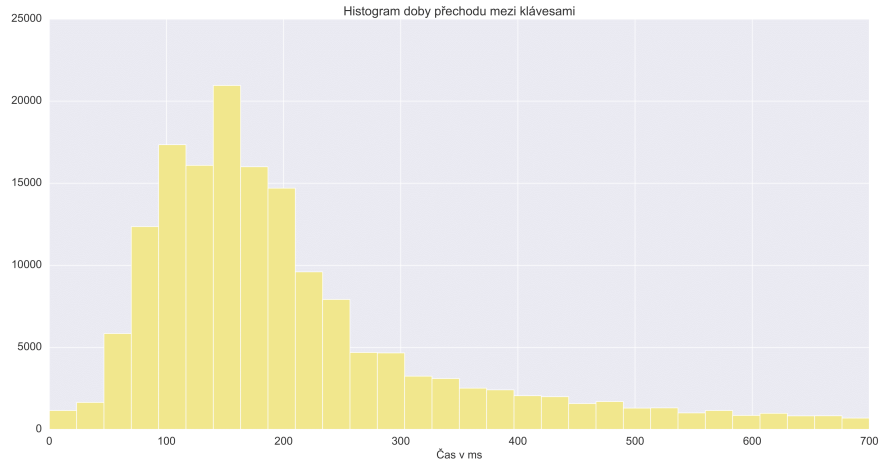
Často se doba držení klávesy určuje také pro každou klávesu zvlášť, což umožňuje detailněji charakterizovat uživatelův styl práce s klávesnicí. V našem případě to ovšem není možné, jelikož, jak ukazuje graf 4.4, ve většině sezení bylo použito minimum kláves a nemáme tak informaci o době držení ostatních.

4.2.2 Doba přechodu mezi klávesami

Doba přechodu mezi klávesami je hned po délce držení kláves dalším z nejčastěji používaných příznaků pro analýzu práce s klávesnicí. Proto se dále budeme zabývat právě jí. Existují čtyři varianty, jak dobu přechodu definovat (viz 2.2.1). V našem případě jsme se rozhodli pro rozdíl časů stisku předchozí a následující klávesy.

Předzpracování

V rámci předzpracování vypočítáme doby přechodů mezi každou po sobě jdoucí dvojicí kláves, které uživatel na jednotlivých stránkách použil. Tím získáme 184 741 záznamů o dobách přechodů. Histogram těchto časů ukazuje obrázek 4.11.

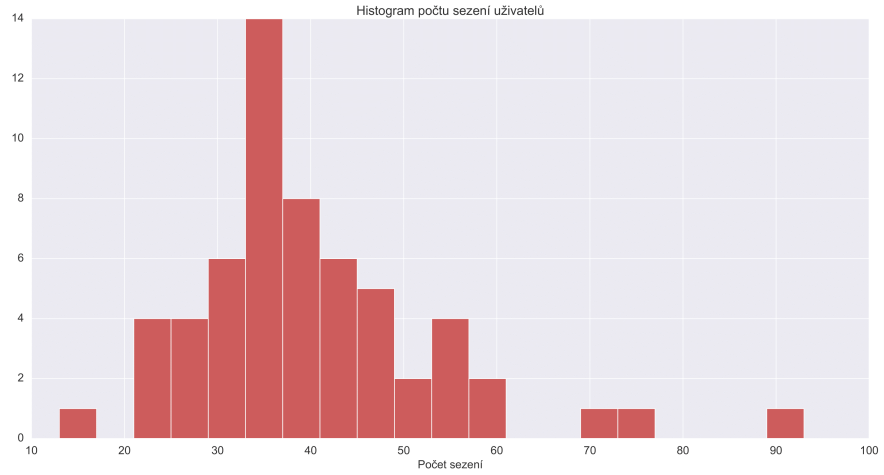


Obrázek 4.11: Doba přechodu mezi klávesami.

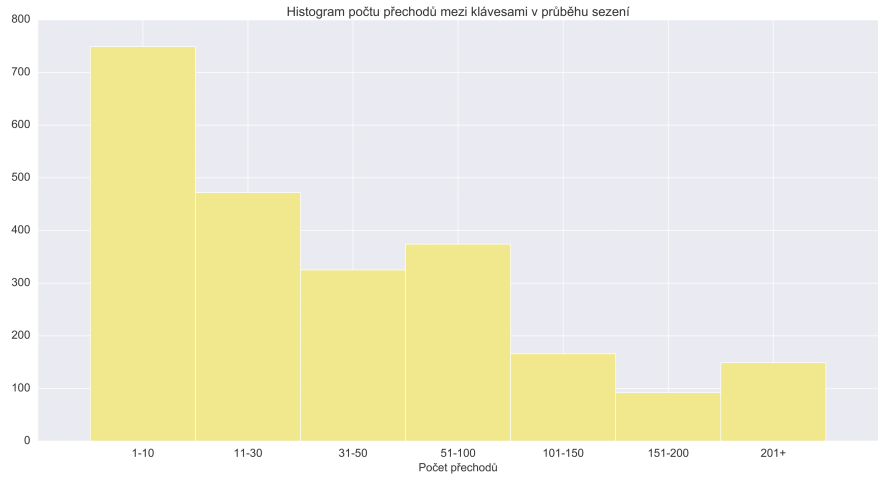
Zásadní je stanovit limit pro maximální čas, který může uplynout mezi dvěma stisky kláves v nepřerušené sekvenci. Jinak budeme uvažovat i časy, kdy uživatel nepsal na klávesnici kontinuálně, ale s přestávkami. To by bylo zkreslující, protože nás zajímá pouze přirozená doba přechodů při souvislém psaní. Naopak příliš krátká doba přechodu by znamenala, že uživatel stiskl klávesy prakticky současně, nikoli postupně. Jako vhodný interval přípustných hodnot jsme proto zvolili 50 až 500 ms. Po přefiltrování nám zůstane 148 151 záznamů, což je 80 % z původního počtu.

Rozdělení do sezení

Zbylé záznamy dále rozdělíme do jednotlivých uživatelských sezení a pro každé určíme průměrnou dobu přechodu mezi klávesami v jeho průběhu. Počty sezení jednotlivých uživatelů ukazuje graf na obrázku 4.12 a počty akcí v sezení pak graf 4.13.



Obrázek 4.12: Počty sezení uživatelů.

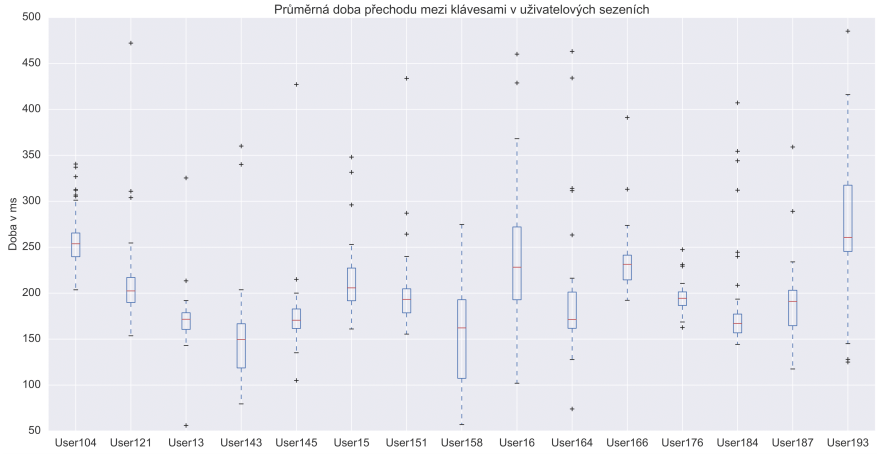


Obrázek 4.13: Počty přechodů mezi klávesami v průběhu sezení.

Průměrně máme k dispozici asi 40 sezení na uživatele. Jediného uživatele, pro kterého je k dispozici méně než 20 sezení, z dat vyřadíme. Stejně jako u doby držení klávesy i zde vidíme, že sezení zřídkakdy obsahují delší interakci s klávesnicí.

Explorace

Boxplot na obrázku 4.14 ukazuje průměrné doby přechodů mezi klávesami v sezeních vzorku uživatelů. Stejně jako u dob držení klávesy i zde si můžeme všimnout výrazných rozdílů mezi jednotlivými uživateli.

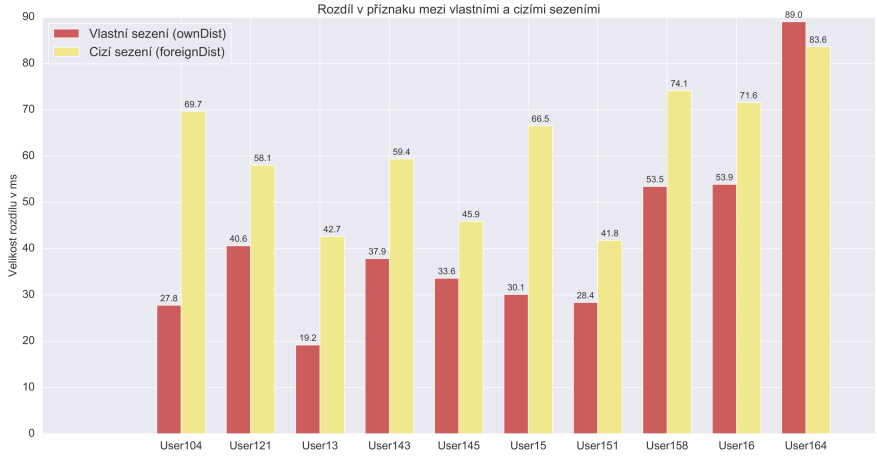


Obrázek 4.14: Průměrná doba přechodu mezi klávesami u uživatelů.

Diskriminační potenciál

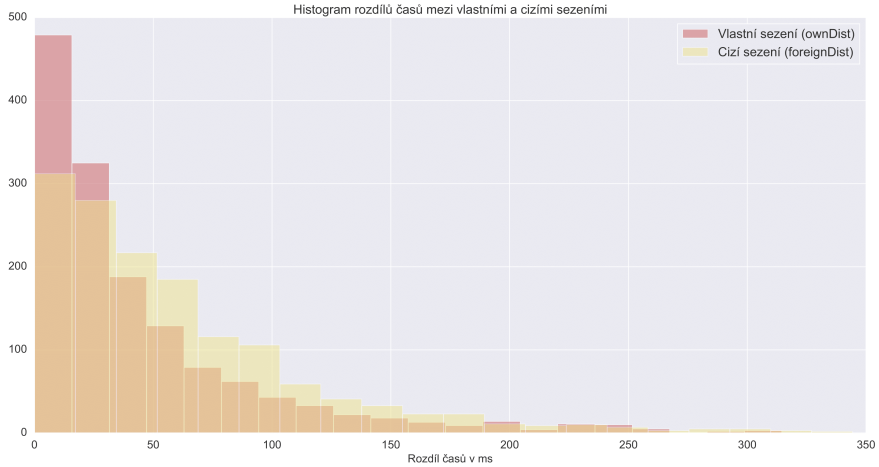
Ze srovnání vzdáleností mezi vlastními a cizími sezeními uživatelů vychází grafy 4.15, 4.16, 4.17. (Jejich význam je stejný jako u délky držení klávesy v sekci 4.2.1.)

Vidíme, že v naprosté většině případů jsou si vlastní sezení výrazně podobnější než cizí. Mezi všemi 58 testovanými uživateli je průměrná velikost rozdílu $\delta_{abs,25} = 15$ ms, což odpovídá nárůstu vzdálenosti o $\delta_{rel,25} = 47$ %, pokud místo vlastního sezení vezmeme cizí.

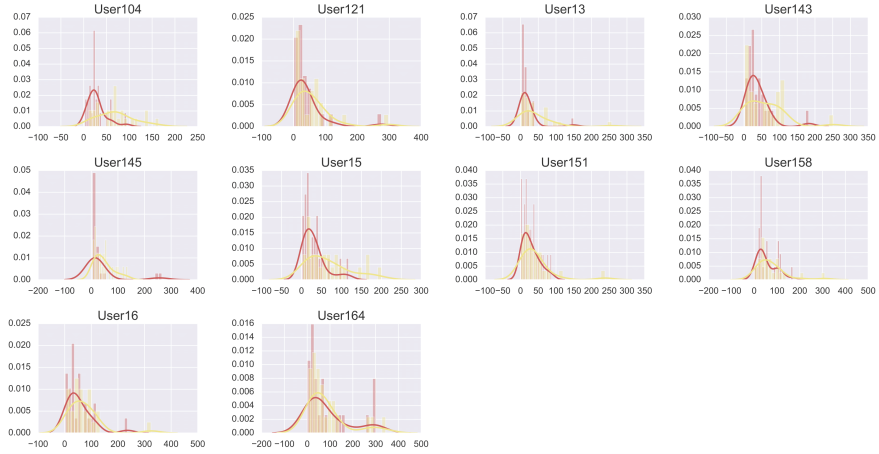


Obrázek 4.15: Porovnání podobností vlastních a cizích sezení.

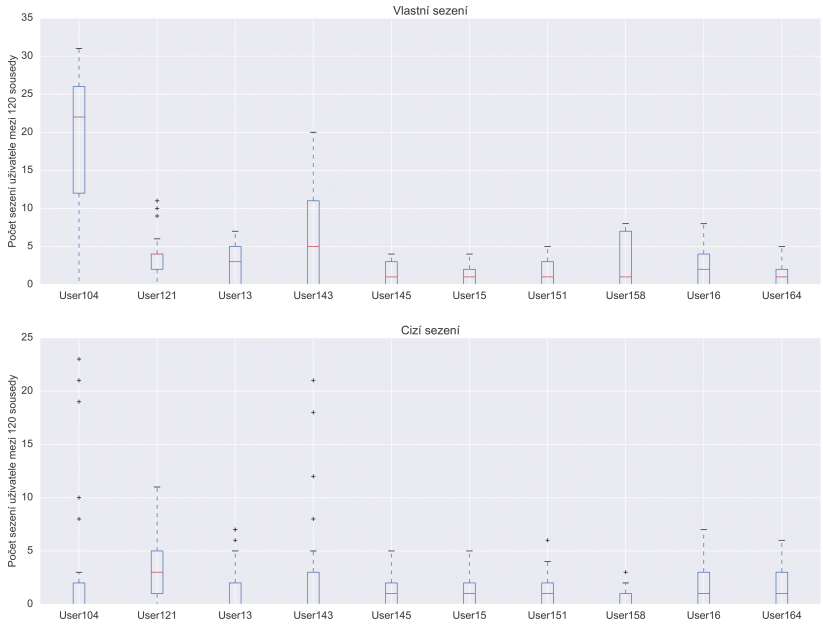
Počty uživatelských sezení v okolí vlastního a cizího sezení ilustrují obrázky 4.18 a 4.19. Zde vidíme, že až na několik výjimek se počty uživatelských sezení v okolí vlastních a cizích bodů výrazně neliší. Průměrně v celém vzorku 58 uživatelů je ale v nejbližším okolí vlastního sezení $\Delta_{20,120} = 2{,}5$ krát více dalších vlastních sezení než v okolí sezení cizího.



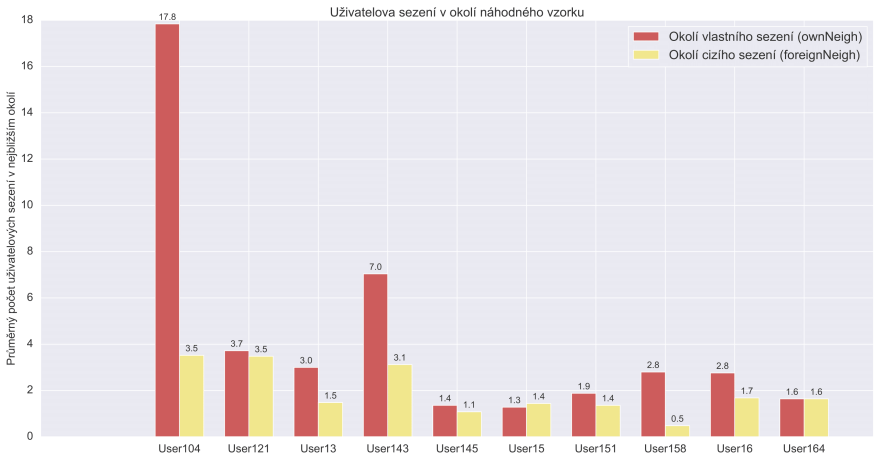
Obrázek 4.16: Rozdíly v době přechodu mezi klávesami v sezeních.



Obrázek 4.17: Podobnosti vlastních a cizích sezení pro uživatele.



Obrázek 4.18: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.19: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

Závěr

Přestože doba přechodu mezi klávesami pravděpodobně nebude pro rozlišování uživatelů tak dobrým příznakem jako doba držení klávesy, předchozí analýza ukazuje, že by i tak mohla být užitečná. Proto ji do autentizačních modelů zahrneme také.

Jak ukazuje graf na obrázku 4.13, nemá v našem případě smysl rozlišovat doby přechodů pro jednotlivé bigramy zvlášť, protože nejčastěji jsou v sezeních interakce s klávesnicí velmi krátké a bigramy mají příliš malé četnosti.

4.2.3 Procento překrytí kláves při psaní

Jednoduchým příznakem, který lze získat z údajů o uživatelově práci s klávesnicí, je procento případů, kdy při psaní dojde k překryvu kláves, tedy k tomu, že uživatel stiskne následující klávesu, aniž předtím uvolnil předchozí. Tato charakteristika je do jisté míry podobná předchozí době přechodu mezi klávesami, proto otestujeme i možnosti jejího využití.

Předzpracování

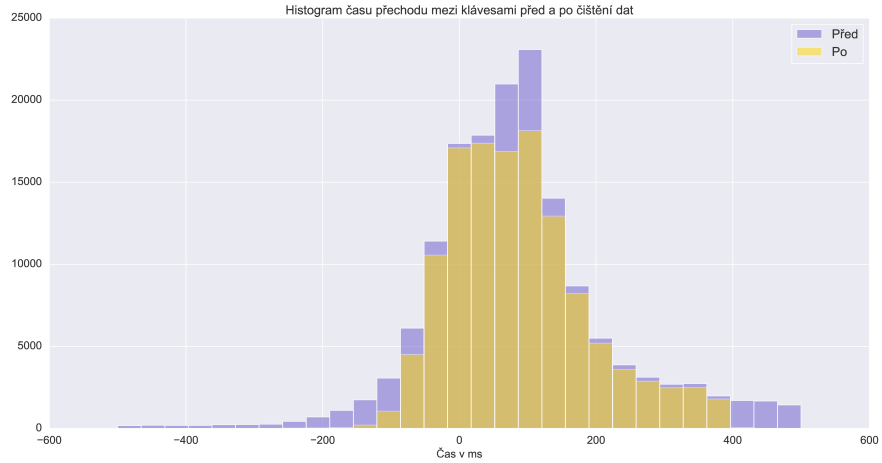
V rámci předzpracování nejprve určíme doby mezi uvolněním předchozí a stiskem následující klávesy pro každou dvojici po sobě jdoucích kláves, které uživatel na stránce použil.

Protože chceme sledovat překryvy při psaní, ke kterým dochází bez uživatelova vědomí, je třeba vyloučit práci s kontrolními klávesami jako Shift, Alt nebo Control, kdy k překryvu stisků kláves dochází cíleně. Dále pak nebudeme uvažovat situace, kdy uživatel dvakrát za sebou zmáčkl tutéž klávesu, protože zde je naopak překryv kláves nutně vyloučen. Mezi 184 741 zaznamenanými přechody bylo 12 524 (6,8 %) záznamů prvního typu a 14 555 (7,9 %) záznamů druhého typu.

Stejně jako u dob přechodů mezi klávesami i zde je zásadní stanovit maximální čas, který může uplynout mezi uvolněním jedné a stiskem následující klávesy v nepřerušené sekvenci. Chceme vyloučit případy, kdy uživatel nepsal na klávesnici

kontinuálně, ale s přestávkami. Jako vhodný interval přípustných hodnot jsme zvolili −250 až 400 ms (záporný čas zde znamená, že došlo k překryvu).

Histogram dob přechodů mezi klávesami před a po přefiltrování vidíme na obrázku 4.20. Po odstranění nevhodných záznamů nám tak zůstává 125 576 (68 %) záznamů.

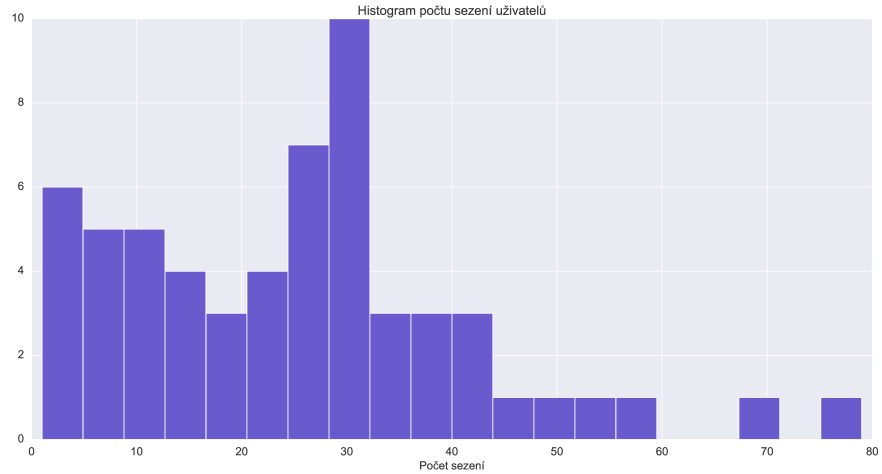


Obrázek 4.20: Časy přechodů před a po čištění dat.

Rozdělení do sezení

Následně záznamy rozdělíme do uživatelských sezení. Histogram počtu sezení uživatelů ukazuje obrázek 4.21. Bohužel téměř třetina uživatelů má méně než 20 sezení. Když jejich záznamy odstraníme, zůstanou nám pro následující analýzu data 39 různých uživatelů.

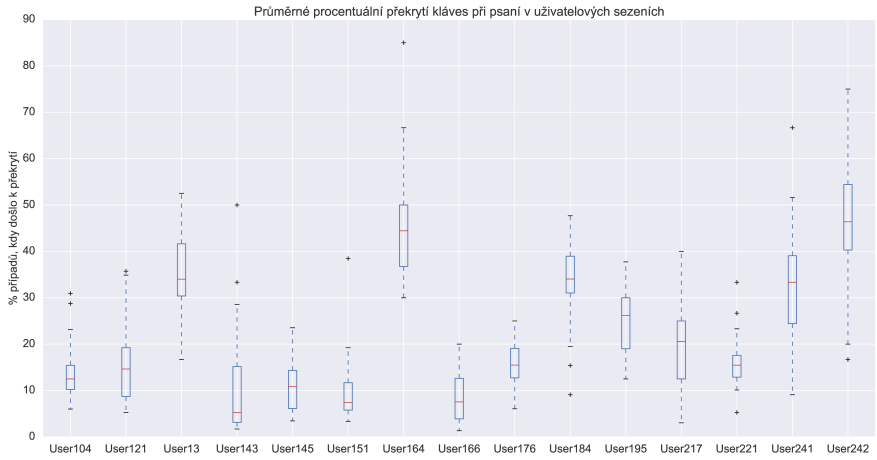
Procento překrytí kláves při psaní pak určíme jako podíl počtu záznamů, kde je čas mezi uvolněním předchozí a stiskem následující klávesy záporný, a celkového počtu záznamů o přechodech mezi klávesami v průběhu sezení.



Obrázek 4.21: Počty sezení uživatelů.

Explorace

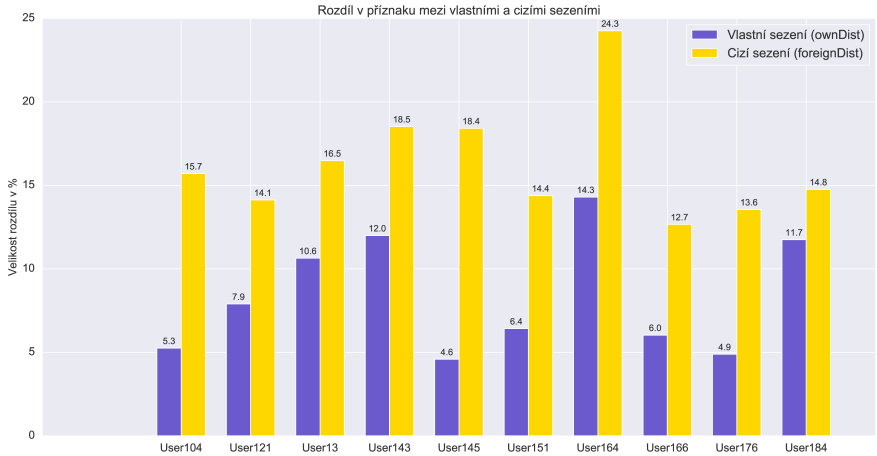
Boxplot na obrázku 4.22 ukazuje procentuální překrytí kláves při psaní v sezeních vzorku uživatelů. To se, jak vidíme, pohybuje téměř vždy v intervalu od 0 do 50 %. U většiny uživatelů pak leží průměr pod hranicí 25 %.



Obrázek 4.22: Průměrné procentuální překrytí kláves při psaní u uživatelů.

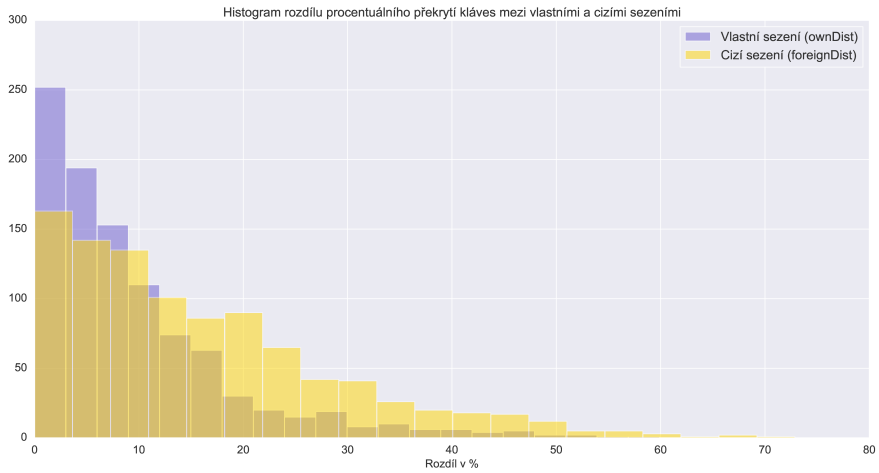
Diskriminační potenciál

Základní porovnání vzdáleností vlastních a cizích sezení ukazují grafy 4.23 a 4.24. Z obou grafů vyplývá, že uživatelova vlastní sezení jsou si výrazně podobnější než sezení různých uživatelů navzájem. Průměrně je mezi všemi testovanými uživateli rozdíl procentuálního překrytí mezi vlastním a cizími sezením o $\delta_{abs,25} = 6$ % vyšší než mezi uživatelovými vlastními sezeními, což odpovídá nárůstu vzdáleností o $\delta_{rel,25} = 101$ %.

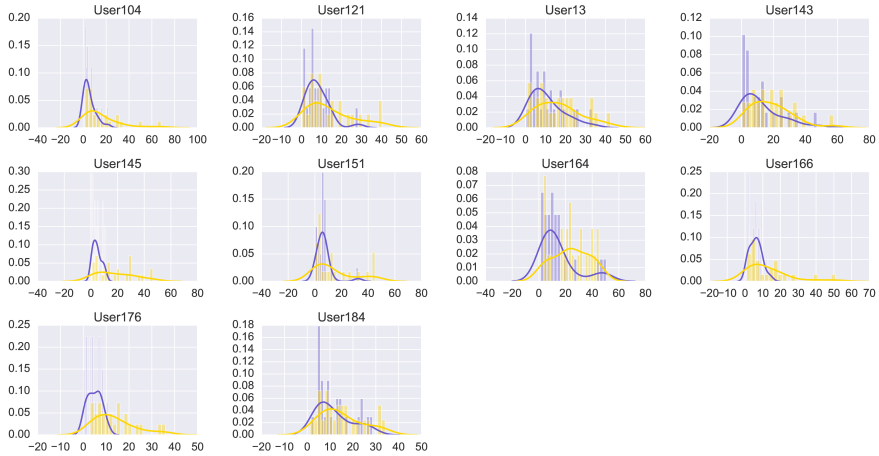


Obrázek 4.23: Porovnání podobností vlastních a cizích sezení.

Na křivkách pravděpodobnostní hustoty na obrázku 4.25 vidíme u všech zobrazených uživatelů výrazný modrý vrchol v okolí nuly oproti méně výraznému žlutému vrcholu více vpravo. To odpovídá ideální situaci, kdy jsou si uživatelova sezení hodnotou příznaku bližší než jeho sezení v porovnání se sezeními ostatních.



Obrázek 4.24: Rozdíly v procentuálním překrytí kláves při psaní mezi sezeními.

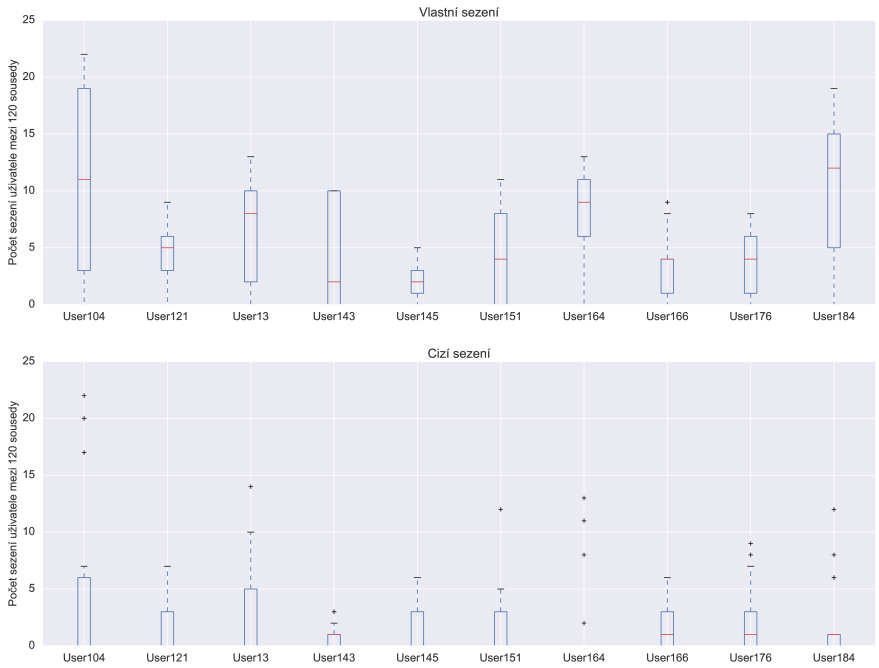


Obrázek 4.25: Podobnosti vlastních a cizích sezení pro uživatele.

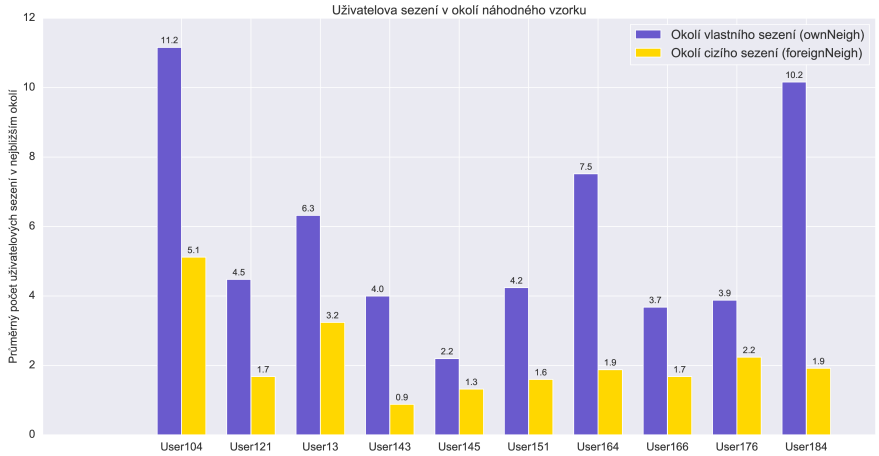
K analýze nejbližších sousedů patří grafy na obrázcích 4.26 a 4.27. Z nich můžeme vidět, že u všech uživatelů ve vzorku je výrazně více uživatelových sezení v okolí jeho vlastních sezení. Celkově mezi všemi 39 zkoumanými uživateli je $\Delta_{20,120} = 3$, tedy v okolí vlastních sezení je průměrně třikrát více dalších uživatelových sezení než v okolí cizích sezení.

Závěr

Z předchozí analýzy vyplývá, že by procentuální překrytí kláves při psaní mělo být užitečným příznakem pro odlišování uživatelů, pravděpodobně s ještě lepšími výsledky než předchozí doba přechodu mezi klávesami. Proto tento příznak do autentizačního modelu zahrneme.



Obrázek 4.26: Počet uživatelových sezení mezi nejbližšími sousedy.



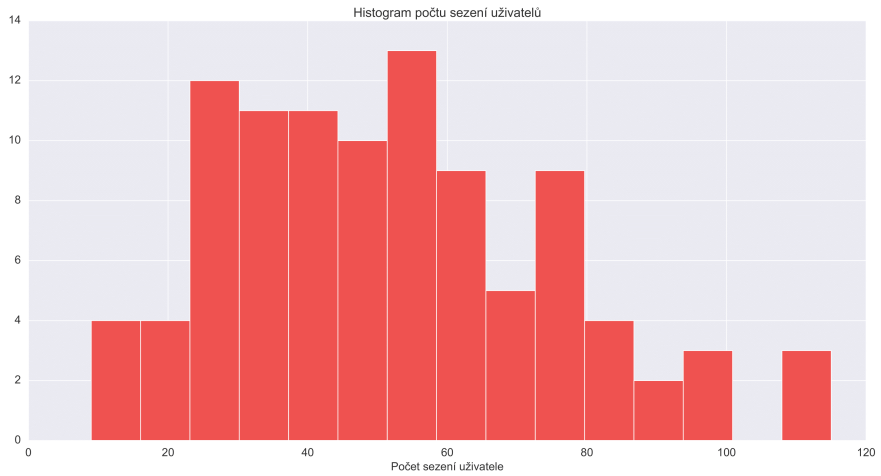
Obrázek 4.27: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

4.3 Styl klikání myší

V této sekci budeme zkoumat příznaky, které lze odvodit ze způsobu, jakým uživatel kliká myší. Vyjdeme ze vstupního souboru, který obsahuje data 100 stejných uživatelů jako v předchozím případě práce s klávesnicí. Tentokrát máme pro období říjen 2019 až únor 2020 k dispozici 162 187 záznamů. Jde o část dat obsažených v datasetu klikání popsaném v sekci 3.3.2. Každý záznam obsahuje následující informace o akcích uživatele:

- identifikátor uživatele,
- identifikátor sezení (odpovídá identifikátoru záznamu z první navštívené stránky),
- identifikátor záznamu o návštěvě stránky,
- čas stisku tlačítka myši (v ms od načtení stránky),
- čas uvolnění tlačítka myši (v ms od načtení stránky),
- xPath elementu stránky, na který uživatel klikl,
- x-ová pozice kliknutí relativně k velikosti elementu (zaokrouhlená na jednotky procent),
- y-ová pozice kliknutí relativně k velikosti elementu (zaokrouhlená na jednotky procent).

Opět aplikujeme požadavek na minimální počet 30 sezení na uživatele. Původní počty uživatelských sezení ukazuje histogram na obrázku 4.28. Podmínku na počet sezení splňuje 82 z výchozího počtu 100 uživatelů. Odstraněním dat ostatních klesá celkový počet použitelných záznamů z 162 187 na 134 333 (83 %).



Obrázek 4.28: Počty sezení uživatelů ve vstupním souboru.

4.3.1 Délka kliknutí

Jedním z nejjednodušších příznaků, které lze ze záznamů o klikání extrahovat, je doba trvání kliknutí. Používá se ve většině studií zabývajících se dynamikou práce s myší (blíže viz sekce 2.2.2), protože typicky vykazuje konzistenci v rámci dat jediného uživatele a dostatečnou odlišnost při porovnávání různých uživatelů. Obvykle se nerozlišuje, zda při klikání uživatel použil levé nebo pravé tlačítko myši. Ani my tento rozdíl nebudeme zohledňovat.

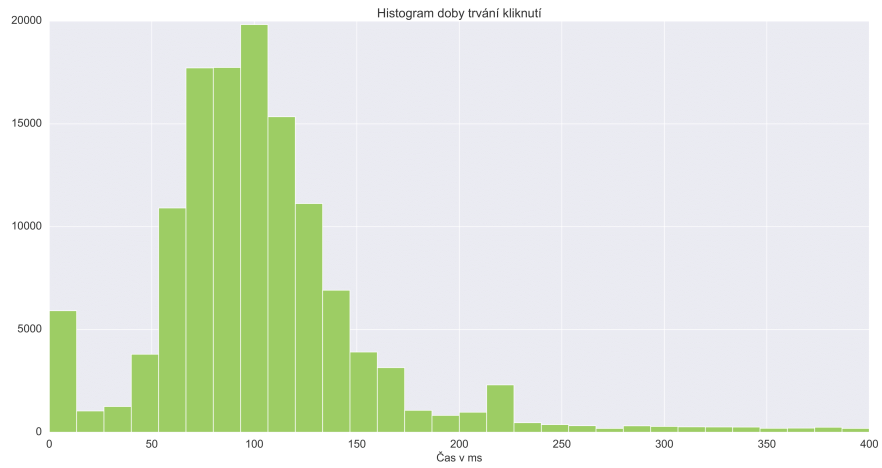
Předzpracování

Dobu trvání kliknutí spočítáme jako rozdíl času uvolnění a stisknutí tlačítka myši.

Stejně jako v případě doby držení klávesy i pro délku kliknutí potřebujeme stanovit interval validních hodnot příznaku. Histogram na obrázku 4.29 ukazuje, že naprostá většina kliknutí trvala méně než 220 ms. Tuto hodnotu proto zvolíme jako horní hranici.

Jako dolní limit použijeme 40 ms. V grafu můžeme vidět, že tím ztrácíme nezanedbatelné množství záznamů, kde byla doba trvání nejkratší. Je však prakticky nemožné, aby uživatel stihl cíleně tak rychle stisknout a pustit tlačítko myši. S největší pravděpodobností jde o chybu při zaznamenávání události v prohlížeči nebo neočekávané chování uživatele.

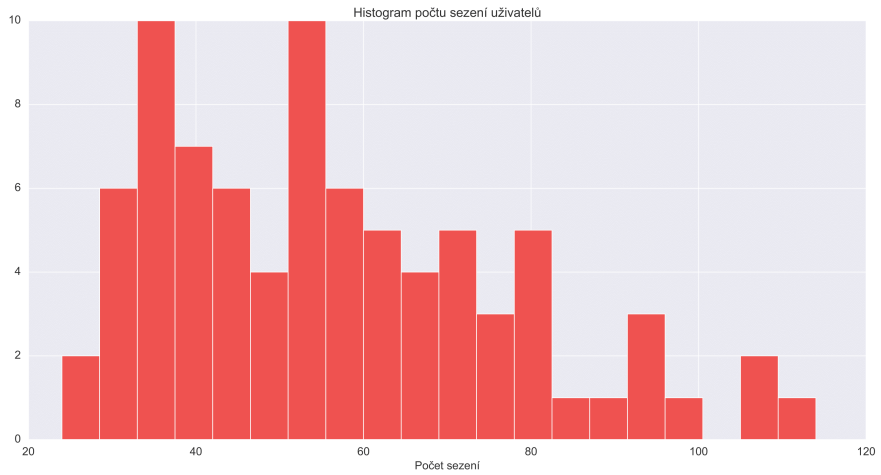
Po eliminaci hodnot za stanovenými hranicemi nám zůstává 115 217 záznamů, což je 86 % původního počtu.



Obrázek 4.29: Doba trvání kliknutí.

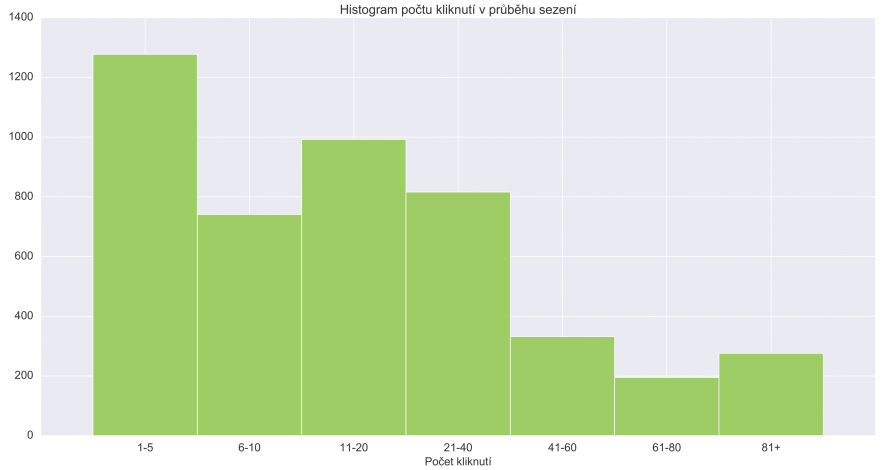
Rozdělení do sezení

Zbylé záznamy nyní rozdělíme podle uživatelských sezení. Počty sezení mezi uživateli ukazuje histogram na obrázku 4.30. Nejmenší počet sezení jednoho uživatele je 24 a největší pak 114 s průměrnou hodnotou 57. Všichni uživatelé tak mají sezení dostatek a data žádného z nich nemusíme před další analýzou vyloučit.



Obrázek 4.30: Počty sezení uživatelů.

Pro zajímavost se můžeme podívat i na počty kliknutí v průběhu sezení na obrázku 4.31. Vidíme, že nejčastější je méně než pět kliknutí za celé sezení. S rostoucím počtem kliknutí četnost odpovídajících sezení rychle klesá a sezení s více než 80 kliknutími už jsou velmi vzácná.



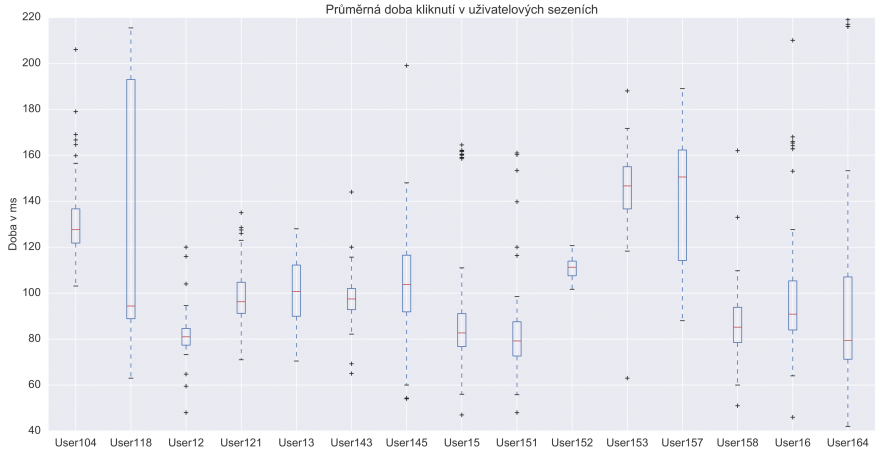
Obrázek 4.31: Počty kliknutí v průběhu sezení.

Explorace

Pro každé sezení určíme průměrnou dobu trvání kliknutí přes všechny záznamy, které k sezení patří. Tyto průměry pro vzorek uživatelů vidíme v podobě boxplotů na obrázku 4.32. Už zde lze pozorovat výrazné rozdíly v chování uživatelů.

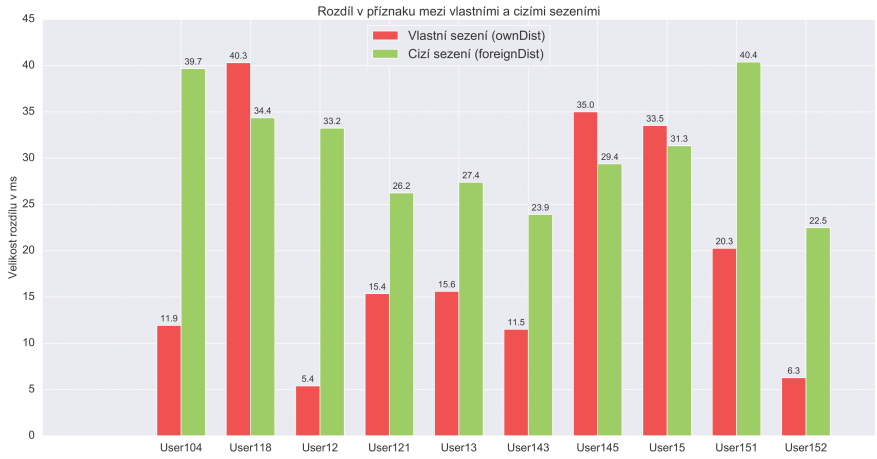
Diskriminační potenciál

Porovnání vzdáleností cizích a vlastních sezení ukazují grafy na obrázcích 4.33, 4.34 a 4.35. Na prvním z nich vidíme, že příznak funguje dobře pro většinu uživatelů ve vzorku. U těch jsou si vlastní sezení výrazně podobnější než cizí ($ownDist_{25} < foreignDist_{25}$), u zbývajících tří pak sledujeme mírně opačný

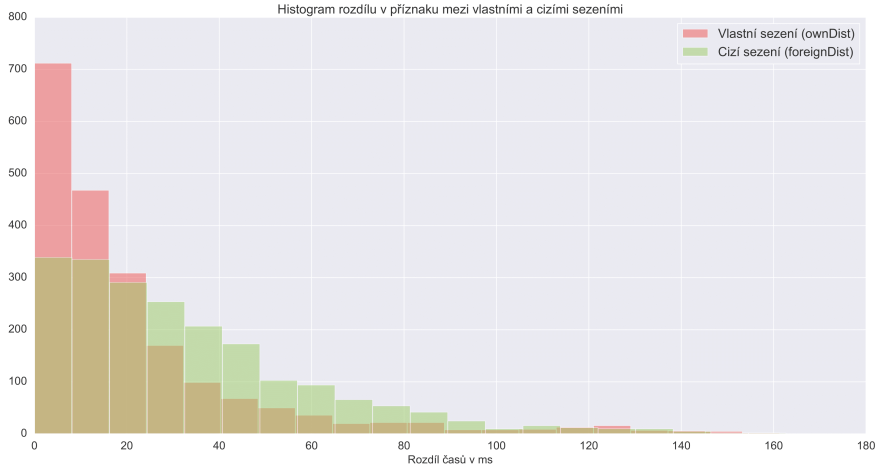


Obrázek 4.32: Průměrná doba trvání kliknutí u uživatelů.

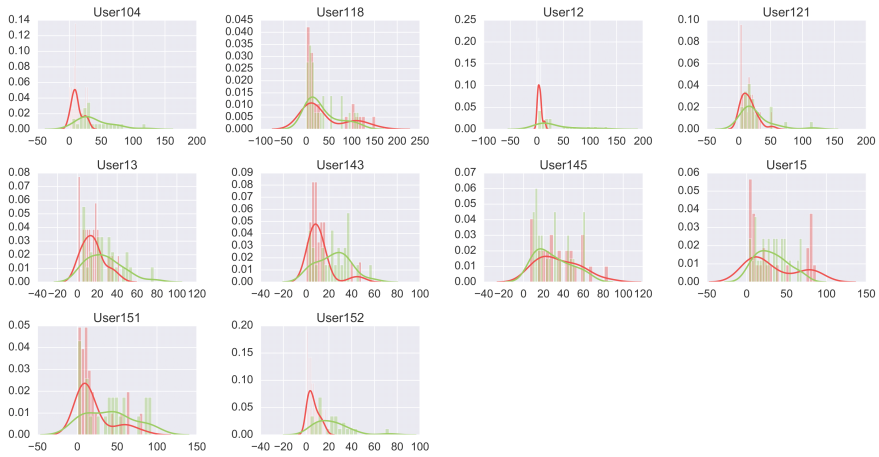
trend. Tomuto pozorování odpovídají i pravděpodobnostní hustoty na posledním z grafů. Pro všechny uživatele celkově vychází $\delta_{abs,25} = 11{,}3$ ms a $\delta_{rel,25} = 84$ %, což lze považovat za velmi slibný výsledek.



Obrázek 4.33: Porovnání podobností vlastních a cizích sezení.

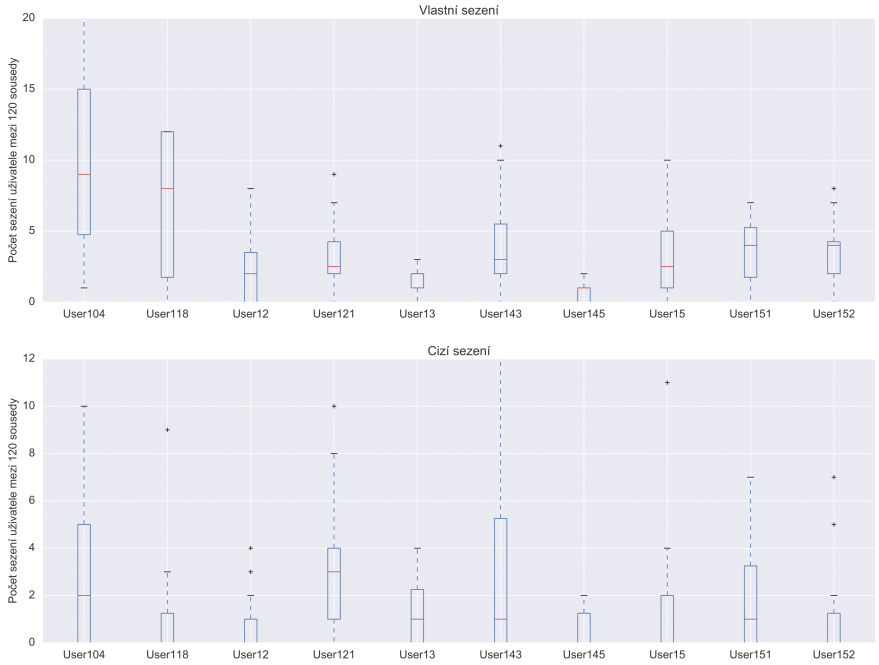


Obrázek 4.34: Rozdíly v době trvání kliknutí mezi sezeními.



Obrázek 4.35: Podobnosti vlastních a cizích sezení pro uživatele.

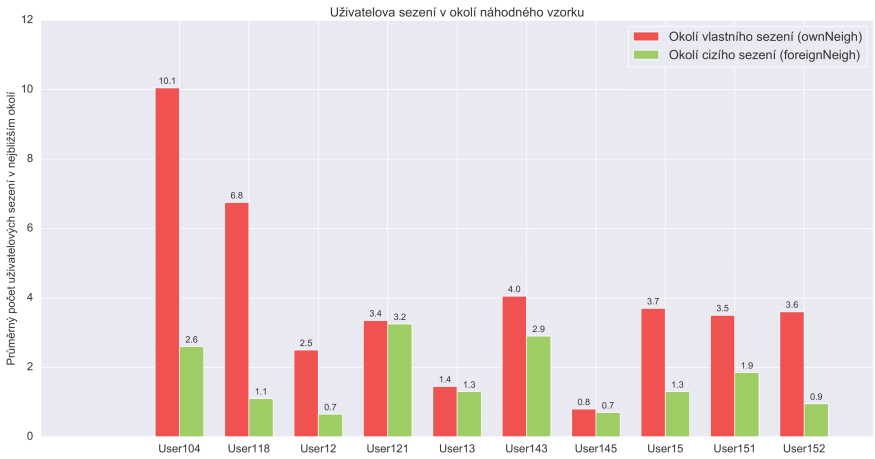
Výsledky analýzy nejbližších sousedů zobrazují grafy na obrázcích 4.36 a 4.37. Z nich vidíme, že většina uživatelů ve vzorku má výrazně více svých sezení v okolí vlastních sezení než v okolí sezení ostatních. Pouze u tří z nich je tento rozdíl jen nepatrný. Celkově je pro dobu trvání kliknutí $\Delta_{20,120} = 3{,}4$.



Obrázek 4.36: Počet uživatelových sezení mezi nejbližšími sousedy.

Závěr

Předchozí analýza potvrdila, že průměrná délka trvání kliknutí zůstává mezi jednotlivými sezeními uživatele podobná, zatímco u sezení různých uživatelů se typicky výrazně liší. Je tedy vhodné tuto charakteristiku při ověřování identity uživatele využít.



Obrázek 4.37: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

4.3.2 Poloha kurzoru při kliknutí na tlačítko

Dalším příznakem, který z dat o klikání uživatele můžeme odvodit, je poloha kurzoru při kliknutí na konkrétní tlačítko. Tuto polohu budeme vztahovat k levému hornímu rohu tlačítka a budeme ji brát relativně vzhledem k jeho velikosti.

Podobný příznak se ve studiích o dynamice myši obvykle neobjevuje, protože ve vstupních datech typicky není informace o prvcích na obrazovce, se kterými uživatel interaguje. My ji ovšem k dispozici máme a můžeme tak otestovat, zda by mohlo jít o užitečný příznak.

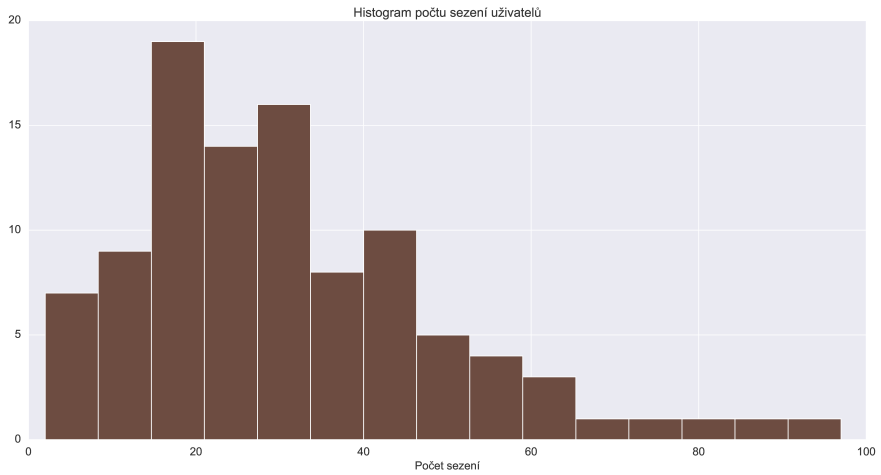
Při analýze jsme se zaměřili pouze na klikání na tlačítko Uložit na stránce pro vytváření záznamu o nové činnosti (bližší popis stránky lze najít v sekci 3.1.1), protože jde pravděpodobně o nejčastěji používané tlačítko pro většinu uživatelů. Pokud by se tento příznak osvědčil, lze jej snadno využít i v jiném kontextu, jelikož tlačítka jsou běžné prvky téměř všech webových aplikací.

Předzpracování

Ze všech záznamů, které máme k dispozici, chceme vybrat pouze ty, kde uživatel použil tlačítko Uložit. Díky tomu, že v datech máme k dispozici xPath prvku, na který uživatel klikl, můžeme snadno vyhledat odpovídající záznamy. Poté, co ostatní odstraníme, zůstane nám 13 213 záznamů, což je 8 % původního počtu.

Rozdělení do sezení

Dále všechny zbylé záznamy rozdělíme podle sezení, ke kterým patří. Počty sezení, které máme pro jednotlivé uživatele k dispozici, shrnuje histogram na obrázku 4.38. Příslušné hodnoty se pohybují v rozmezí od 2 do 97. Pro další analýzu požadujeme od každého uživatele minimálně 20 sezení. Po odstranění těch, kteří tuto podmínku nesplňují, nám zůstanou data od 69 uživatelů.

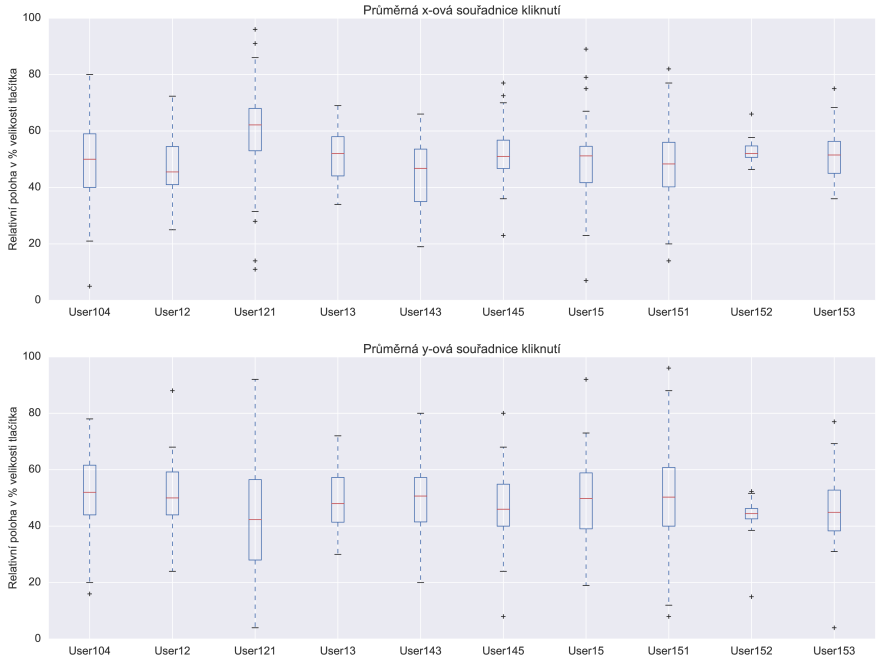


Obrázek 4.38: Počty sezení uživatelů.

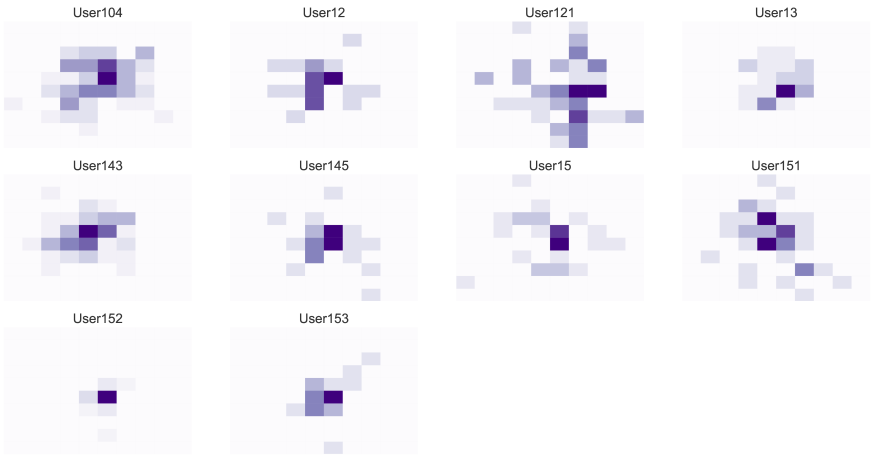
Explorace

Následně pro každé sezení určíme průměrnou vertikální a horizontální polohu, kam uživatel kliká. Příslušné hodnoty pro vzorek uživatelů vidíme v boxplotu na obrázku 4.39. Totéž můžeme sledovat i na obrázku 4.40 v podobě 2D histogramu. Čím sytější je odstín barvy, tím častější je klikání do daného místa.

Především z boxplotu je vidět, že chování uživatelů ve vzorku se vzájemně nijak výrazně neliší a všichni většinou klikají doprostřed tlačítka.



Obrázek 4.39: Průměrné souřadnice polohy klikání na tlačítko v sezeních uživatelů.

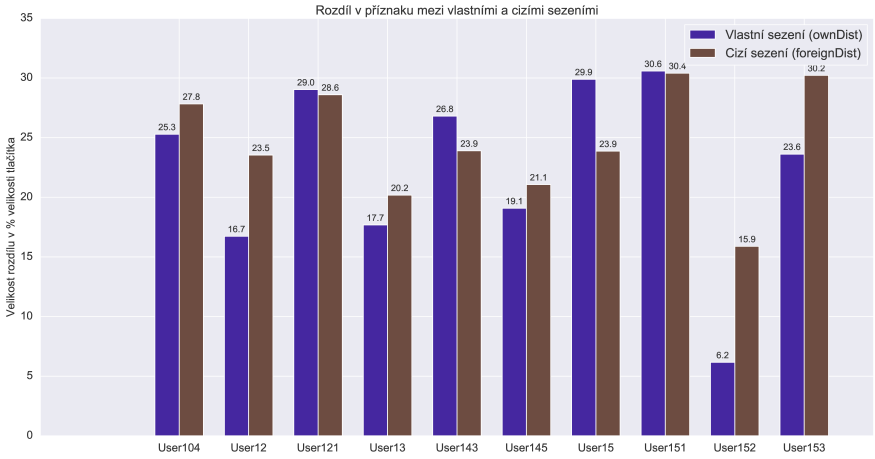


Obrázek 4.40: Frekvence polohy klikání na tlačítko v sezeních uživatelů.

Diskriminační potenciál

Z porovnání vzdáleností cizích a vlastních sezení vycházejí grafy 4.41, 4.42 a 4.43. Na všech z nich vidíme, že rozdíly mezi oběma variantami jsou minimální. Tomu odpovídají i nízké hodnoty rozdílu mezi všemi uživateli dohromady $\delta_{abs,25} = 2{,}3$ % a $\delta_{rel,25} = 16$ %.

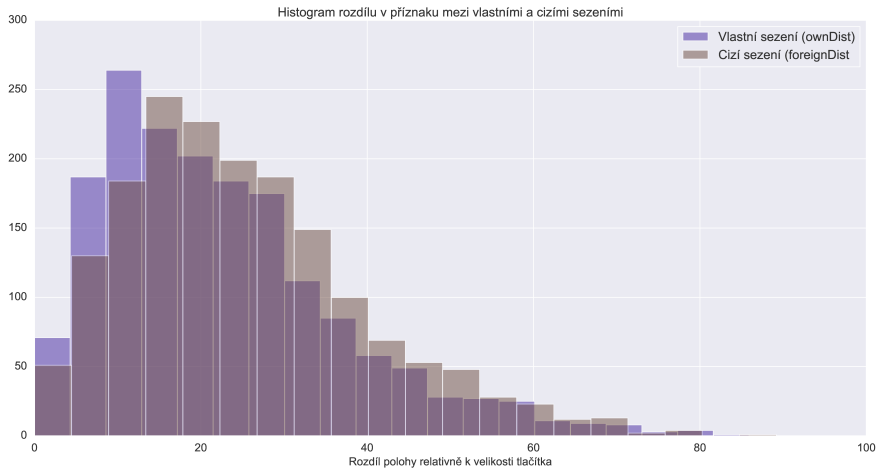
Analýzu uživatelských sezení v okolí vlastních a cizích sezení pro vzorek uživatelů ukazují grafy na obrázcích 4.44 a 4.45. Pouze u dvou uživatelů, jejichž data jsou zobrazena, můžeme pozorovat výrazný nárůst počtu vlastních sezení, pokud sledujeme okolí jejich sezení namísto cizího. Průměrně pro všechny testované uživatele je tento nárůst $\Delta_{20,120} = 1{,}4$, což je opět relativně málo ve srovnání s odpovídající hodnotou u předchozích příznaků.



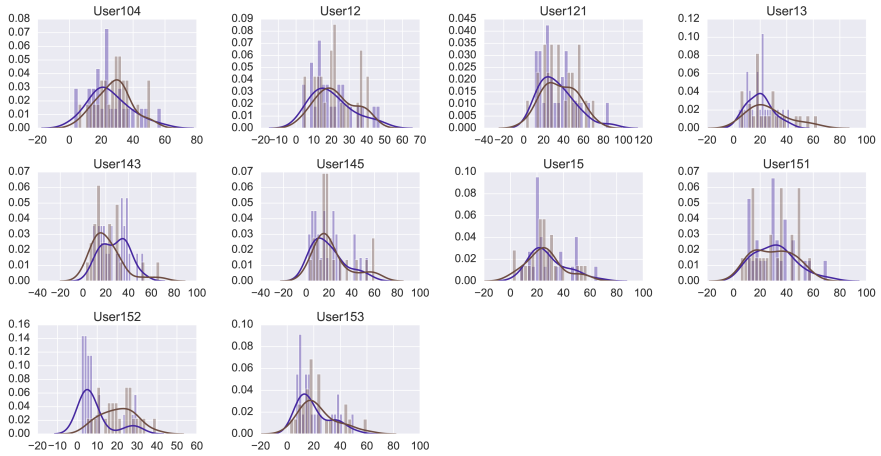
Obrázek 4.41: Porovnání podobností vlastních a cizích sezení.

Závěr

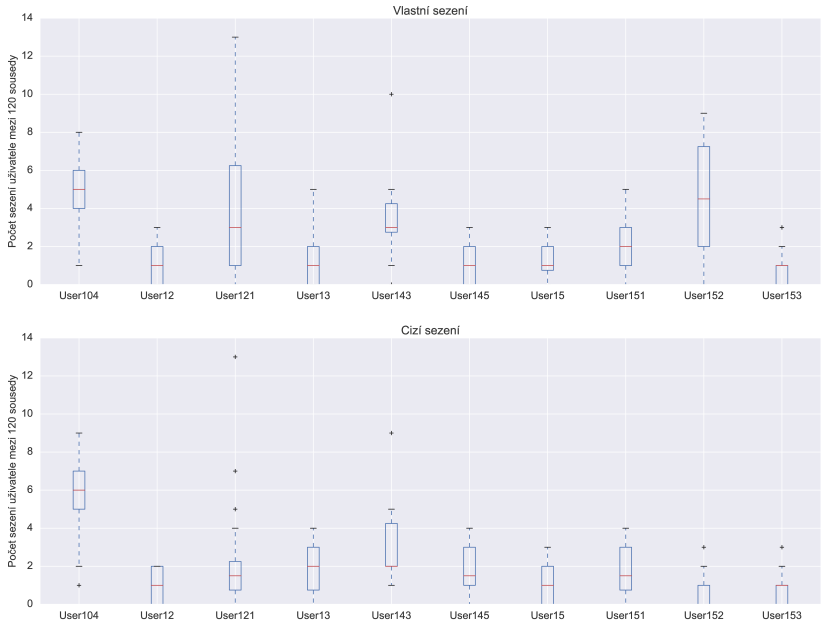
Výsledky předchozích analýz nenasvědčují tomu, že by poloha kurzoru při kliknutí na zvolené tlačítko dokázala dostatečně spolehlivě charakterizovat uživatele, protože většina z nich se v tomto ohledu chová velmi podobně. Tento příznak proto dále využívat nebudeme.



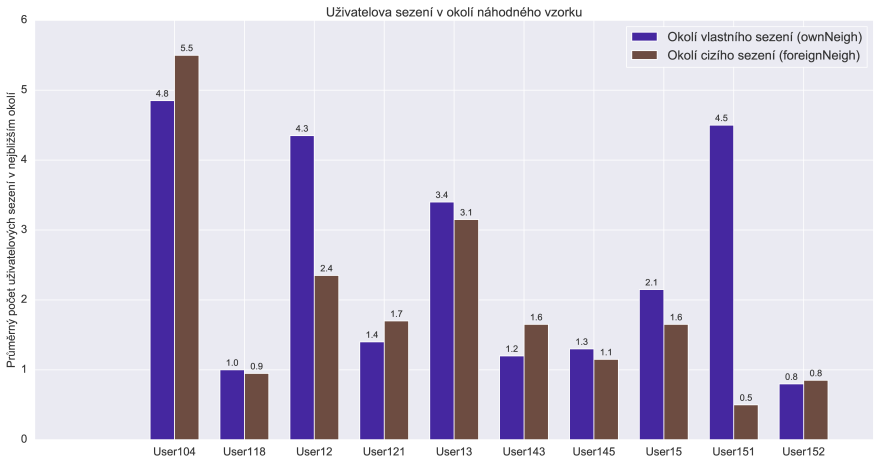
Obrázek 4.42: Rozdíly v poloze kliknutí mezi sezeními.



Obrázek 4.43: Podobnosti vlastních a cizích sezení pro uživatele.



Obrázek 4.44: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.45: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

4.3.3 Poloha kurzoru při kliknutí na položku menu

Dále se budeme zabývat podobným příznakem jako v předchozím případě. Opět budeme sledovat polohu kurzoru při klikání, ale tentokrát se zaměříme na klikání na položku menu v liště vlevo na libovolné stránce. Polohu budeme opět vztahovat k levému hornímu rohu prvku a relativně k jeho velikosti.

I tento příznak je pro studie dynamiky myši netypický, protože obvykle ve vstupních datech nejsou zaznamenávány informace o tom, na který prvek na obrazovce uživatel klikal. Může být proto zajímavé studovat, jestli se v našich datech, kde potřebné údaje máme, ukáže příznak jako stabilní v chování jednotlivce a současně dostatečně odlišný mezi různými uživateli navzájem.

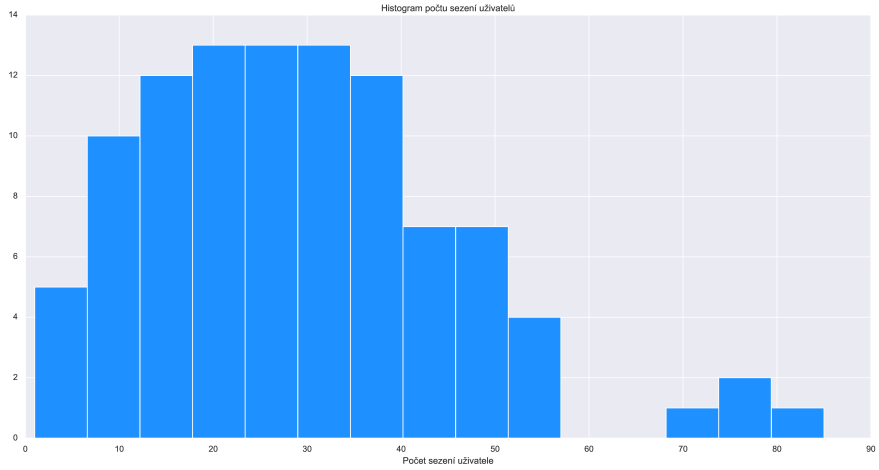
Je ovšem možné, že zjistíme, že všichni uživatelé se z hlediska klikání na položku menu chovají velmi podobně, stejně jako tomu bylo v případě klikání na tlačítko Uložit.

Předzpracování

Ze všech záznamů o klikání uživatelů chceme vybrat pouze ty, kde uživatel klikal na položky menu. Ty můžeme snadno určit podle jejich xPath identifikátoru. Pro další analýzu pak máme k dispozici 8 642 záznamů, což je 5 % všech.

Rozdělení do sezení

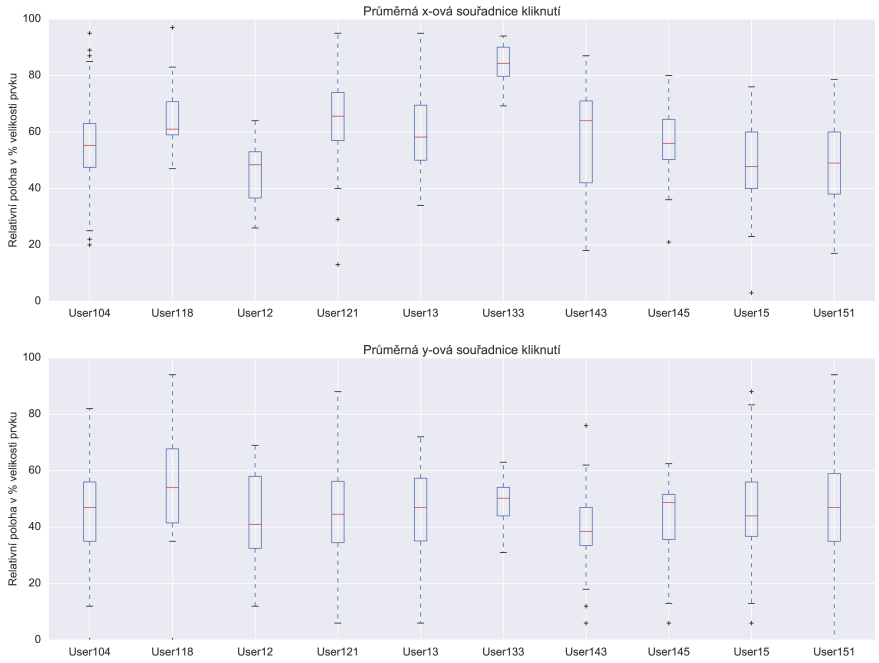
Následně rozdělíme zbylé záznamy podle toho, ke kterému sezení patří. Počty sezení, které máme pro jednotlivé uživatele k dispozici, ukazuje histogram na obrázku 4.46. Stejně jako v předchozích případech požadujeme alespoň 20 sezení od uživatele. Tuto podmínku splní 69 z původních 100 uživatelů.



Obrázek 4.46: Počty sezení uživatelů.

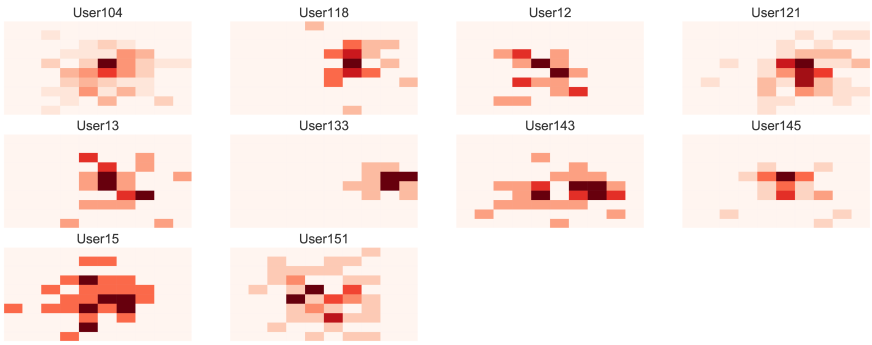
Explorace

Dále pro každé sezení určíme průměrnou hodnotu horizontální a vertikální souřadnice místa na položce menu, kam uživatel klikal. Tyto hodnoty pro vzorek uživatelů vidíme v boxplotu na obrázku 4.47 pro každou složku zvlášť a společně ve 2D histogramu na obrázku 4.48.



Obrázek 4.47: Průměrné souřadnice polohy klikání na položku menu v sezeních uživatelů.

Z grafů vidíme, že poloha kurzoru při klikání uživatelů se liší hlavně v horizontální složce, zatímco vertikálně míří všichni převážně do středu. Oproti tomu na horizontále kliká většina uživatelů spíše vpravo. To odpovídá skutečnosti, že menu se nachází na levém okraji stránky, takže pravá strana odkazu je uživateli, který předtím pracoval ve středu obrazovky, nejblíže.

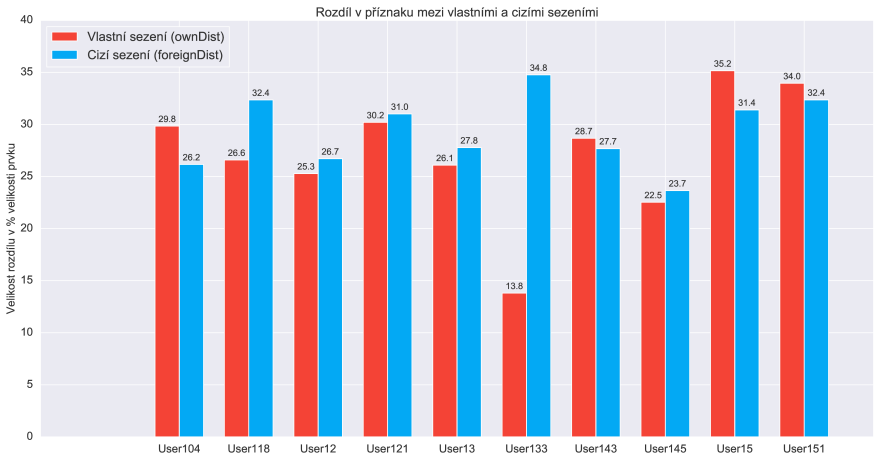


Obrázek 4.48: Frekvence polohy klikání na položku menu v sezeních uživatelů.

Diskriminační potenciál

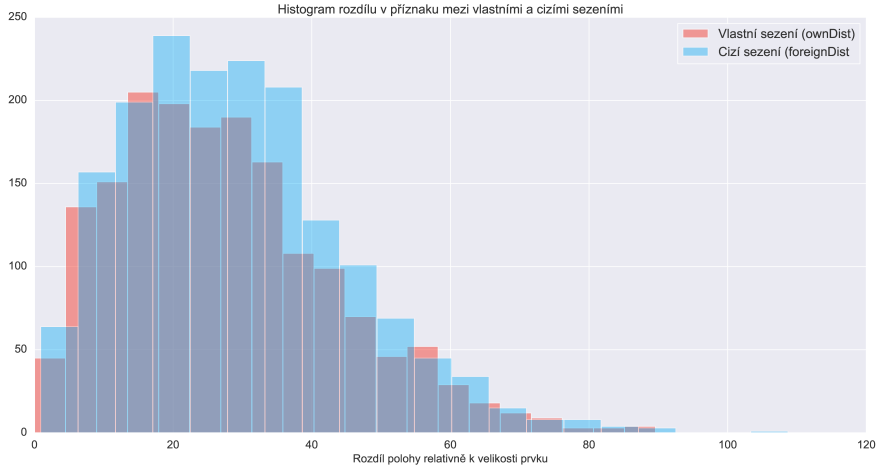
Opět nejprve porovnáváme vzdálenosti mezi sezeními jednoho uživatele a mezi sezeními různých uživatelů. Výsledky ukazují grafy na obrázcích 4.49, 4.50 a 4.35.

Situace je zde podobná jako u předchozí polohy kurzoru při klikání na tlačítko. S výjimkou jediného uživatele nikde nevidíme výrazné rozdíly při porovnávání vzdáleností k vlastním a k cizím sezením. Celkově pro všech 69 uživatelů vychází absolutní rozdíl obou variant $\delta_{abs,20} = 1{,}9$ %, což odpovídá relativnímu rozdílu $\delta_{rel,25} = 11$ %.

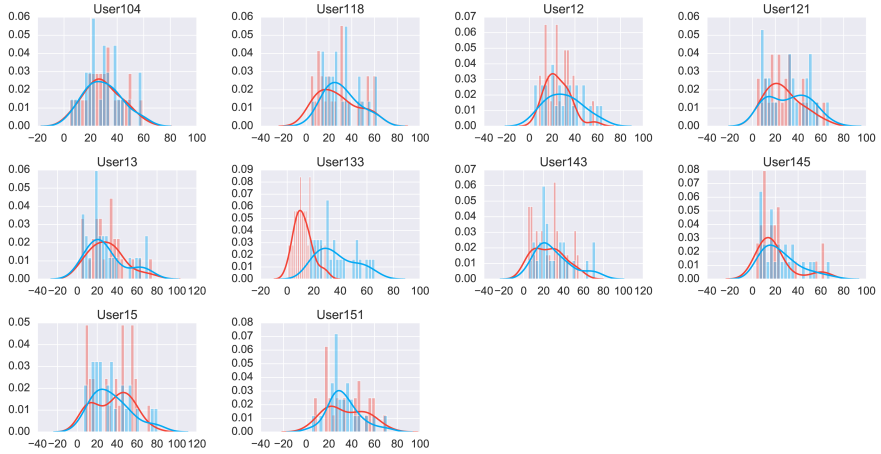


Obrázek 4.49: Porovnání podobností vlastních a cizích sezení.

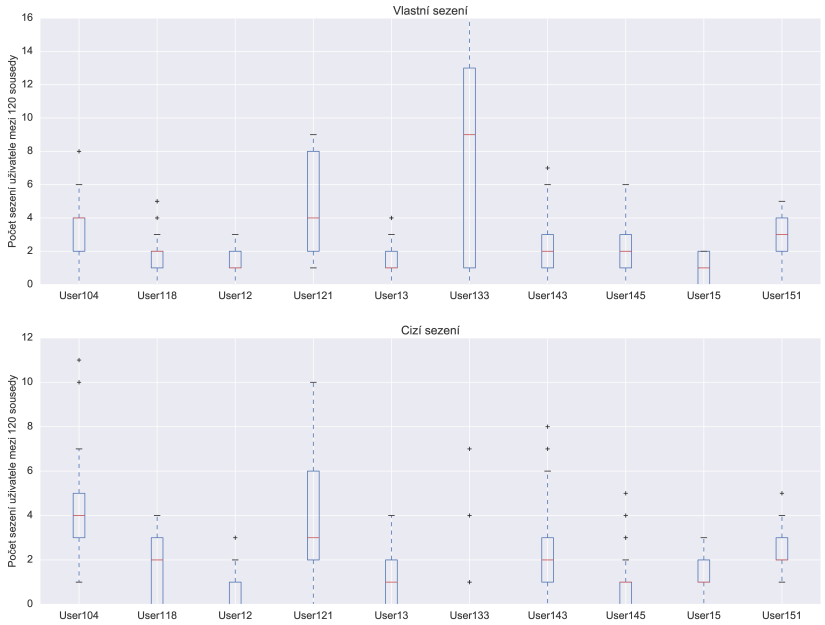
Počty uživatelských sezení mezi nejbližšími sousedy můžeme vidět na obrázcích 4.52 a 4.53. Ani zde s výjimkou jediného uživatele nepozorujeme ve vzorku velké rozdíly mezi okolím vlastních a cizích sezení. Mezi všemi testovanými uživateli je pak průměrný nárůst počtu uživatelských sezení v okolí vlastního sezení oproti cizímu $\Delta_{20,120} = 1{,}5$ násobný.



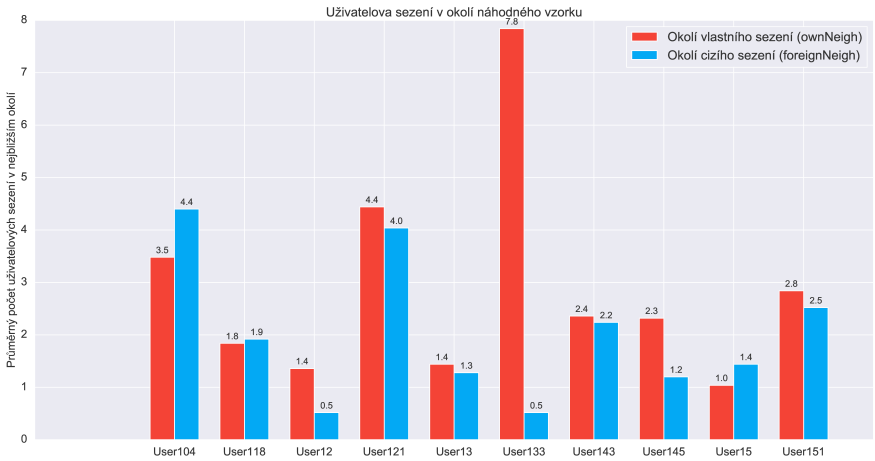
Obrázek 4.50: Rozdíly v poloze kliknutí mezi sezeními.



Obrázek 4.51: Podobnosti vlastních a cizích sezení pro uživatele.



Obrázek 4.52: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.53: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

Závěr

Výsledky analýzy jsou podobné jako v předchozím případě polohy kurzoru při klikání na tlačítko. Ukazují, že pravděpodobně ani v případě klikání na odkaz není možné dobře odlišovat jednotlivé uživatele. Proto ani tento příznak ve finálním autentizačním modelu nepoužijeme.

4.4 Dynamika pohybu myši

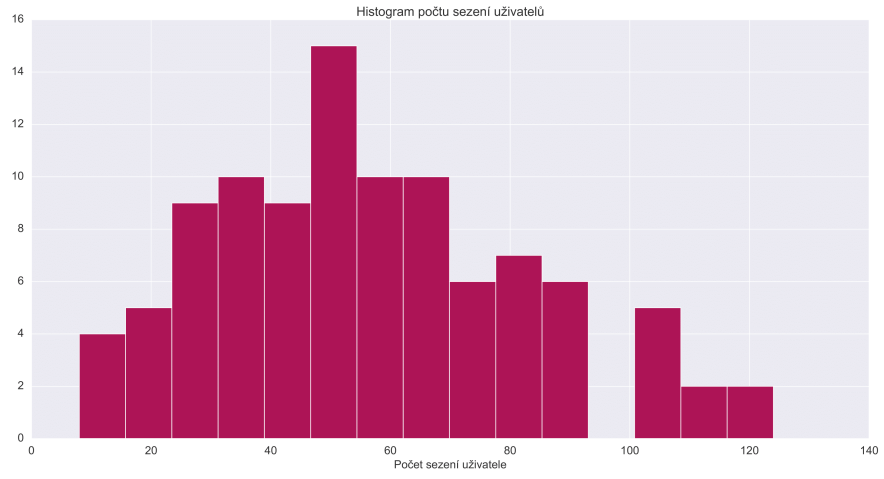
Poslední skupinou příznaků, jimž se budeme věnovat, jsou ty, které lze odvodit od způsobu, jakým uživatel při práci v aplikaci pohybuje myší. Společný vstupní soubor pro extrakci těchto příznaků zahrnuje týchž 100 uživatelů jako v předchozích případech. Z období od října 2019 do února 2020 pro ně máme k dispozici celkově 1 116 599 záznamů. (Jde o část datasetu pohybu myši popsaném v sekci 3.3.2.)

Soubor se záznamy o pohybech myši obsahuje následující informace:

- identifikátor uživatele,
- identifikátor sezení (odpovídá identifikátoru záznamu z první navštívené stránky),
- identifikátor záznamu o návštěvě stránky,
- URL navštívené stránky,
- čas záznamu (v ms od načtení stránky),
- x-ová poloha myši (v pixelech),
- y-ová poloha myši (v pixelech).

Je otázkou, jak se příznaky extrahované z těchto dat osvědčí vzhledem k tomu, že nezaznamenáváme každý detekovaný pohyb, ale pohyby vzorkujeme s periodou sedm (pro bližší informace o zaznamenávaných akcích viz 3.1.2).

Histogram na obrázku 4.54 shrnuje počty sezení, které máme pro jednotlivé uživatele k dispozici. Průměr je 57 sezení na uživatele.



Obrázek 4.54: Počty sezení uživatelů ve vstupním souboru.

4.4.1 Rychlost pohybu myši

Rychlost pohybu myši je nejtypičtějším příznakem, který se zkoumá v rámci studia dynamiky práce s myší. Je zmíněn v téměř všech článcích na toto téma (blíže viz 2.2.2). Proto jsme se i my rozhodli ověřit, zda by mohlo jít o užitečný příznak pro naši situaci.

Problémem by však mohlo být, že naše data jsou příliš hrubá. Polohu myši totiž ukládáme jen při každé sedmé zaznamenané události pohybu. Hrozí tak, že nebudeme moci určit rychlost myši dostatečně přesně, aby s její pomocí bylo možné rozlišovat jednotlivé uživatele.

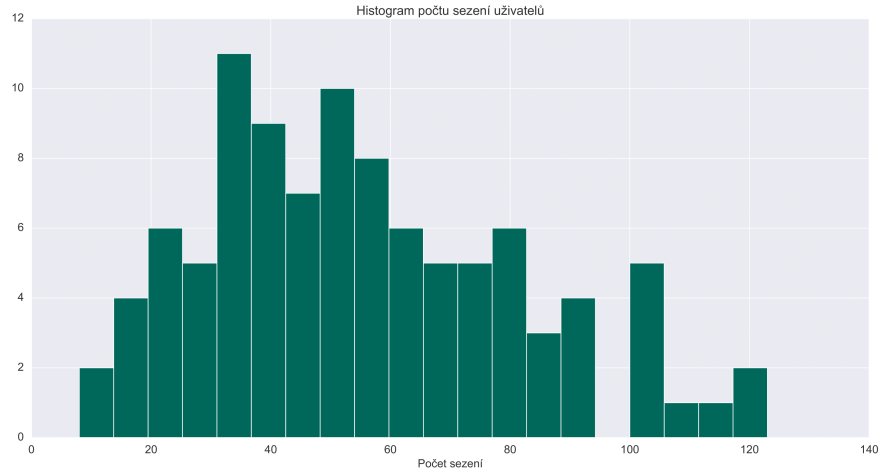
Předzpracování

Abychom mohli ze vstupních dat spočítat průměrnou rychlost pohybu myši, stanovíme nejprve uraženou vzdálenost a časovou prodlevu pro každou dvojici po sobě jdoucích záznamů o poloze myši na stránce. Rychlost pohybu pak odpovídá podílu těchto dvou hodnot.

Zásadní je stanovit limit pro maximální čas, který může uplynout mezi zaznamenanými pohyby myši, abychom je považovali za souvislé. Jinak budeme uvažovat i časy, kdy uživatel nepohyboval myší kontinuálně, ale s přestávkami, což by vypočtenou rychlost snižovalo a zkreslovalo. Maximální dobu mezi záznamy jsme zvolili jako 200 ms, minimální a maximální vzdálenost pak jako 5 a 400 pixelů. Do stanoveného rozmezí náleží 710 241 záznamů, což je 64 % původního počtu.

Rozdělení do sezení

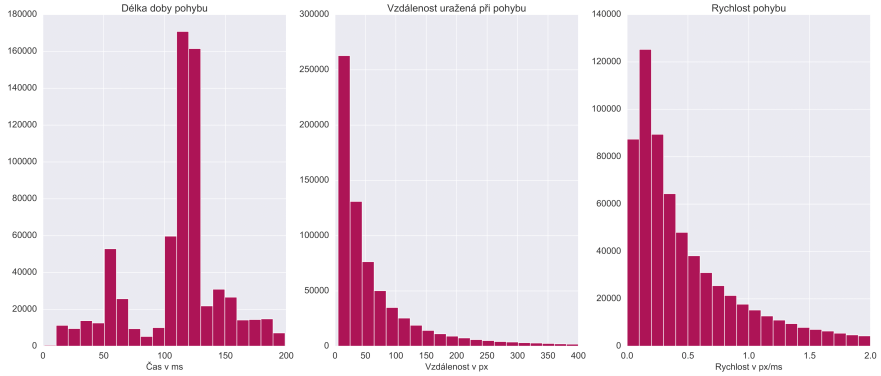
Dalším krokem je rozdělení záznamů podle příslušných sezení. V histogramu na obrázku 4.55 vidíme, kolik máme k dispozici sezení pro jednotlivé uživatele. Pro další analýzu požadujeme minimálně 20 sezení na uživatele. Tuto podmínku splní 94 z původních 100 uživatelů.



Obrázek 4.55: Počty sezení uživatelů.

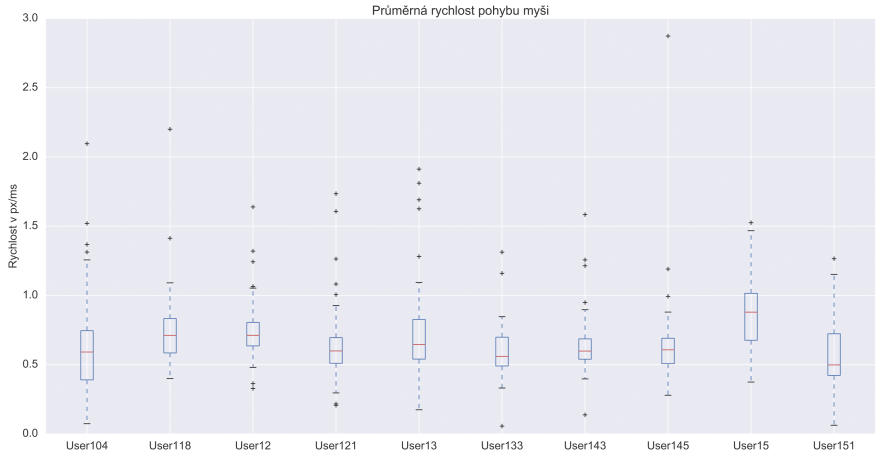
Explorace

Histogramy na obrázku 4.56 ukazují rozložení doby trvání pohybu, uražené vzdáleností a rychlosti pohybu myši ve všech záznamech uživatelů.



Obrázek 4.56: Doba trvání pohybu, uražená vzdálenost a rychlost pohybu myši.

Poté, co záznamy rozdělíme podle sezení, určíme pro každé z nich průměrnou rychlost pohybu myši v jeho průběhu. Výsledné hodnoty pro vzorek uživatelů ukazuje boxplot na obrázku 4.57. Z něj můžeme vidět, že u všech je charakteristika velmi podobná s průměrem mezi 0,5 a 1 px/ms.

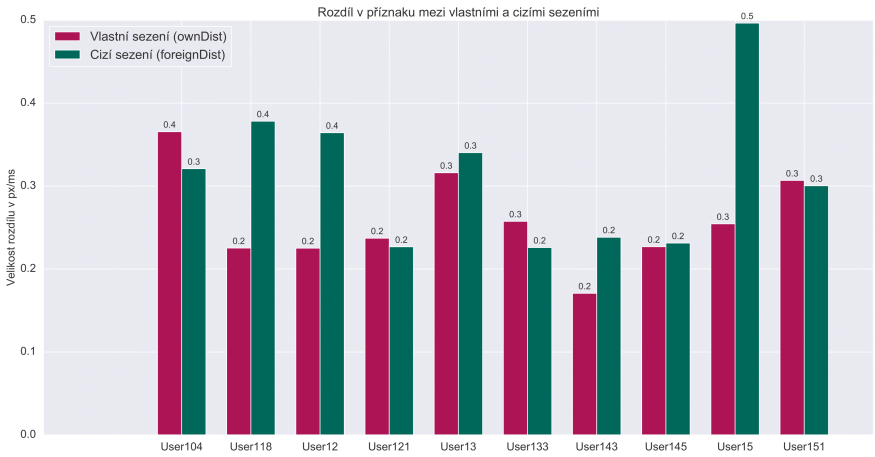


Obrázek 4.57: Průměrná rychlost pohybu myši u uživatelů.

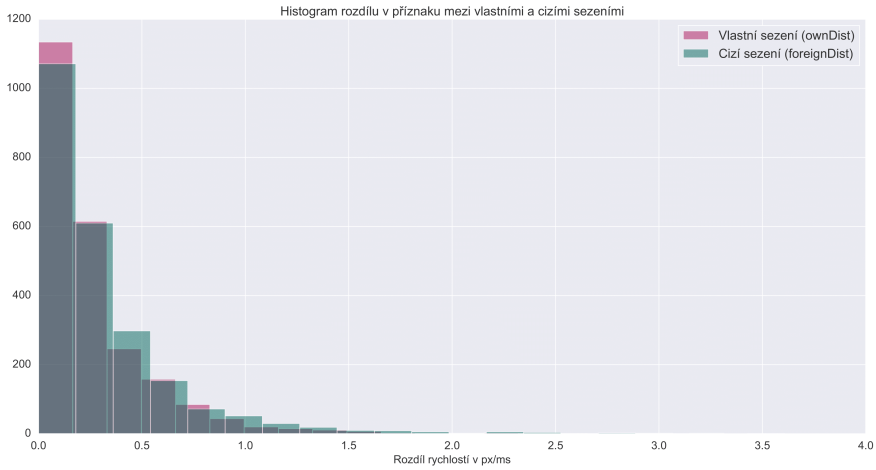
Diskriminační potenciál

Porovnání vzdáleností vlastních a cizích sezení ukazují grafy na obrázcích 4.58, 4.59 a 4.60. Na prvním a třetím z nich můžeme vidět, že u většiny uživatelů ve vzorku je rozdíl při porovnání obou variant zanedbatelný. Totéž potvrzuje i histogram vzdáleností pro všechny uživatele dohromady v obrázku 4.59. Průměrně je pro ně rozdíl vzdáleností mezi vlastními a cizími sezení pouze $\delta_{abs,25} = 0{,}05$ px/ms, což odpovídá relativnímu rozdílu $\delta_{rel,25} = 27$ %.

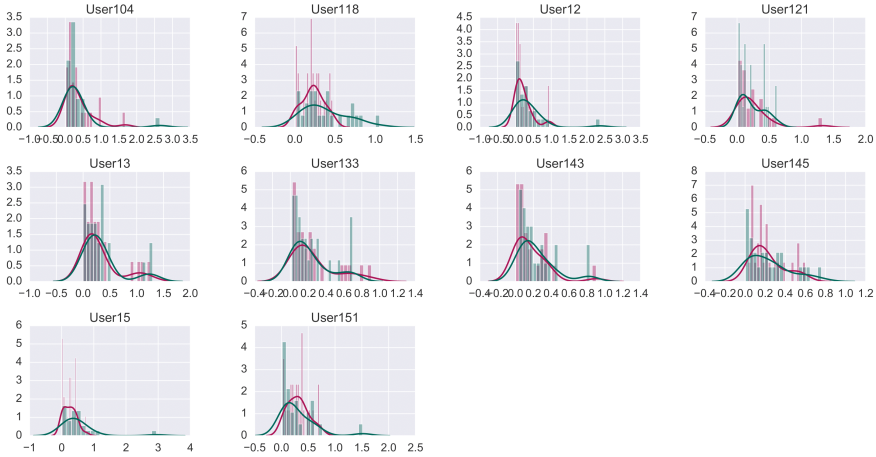
Výsledky zkoumání nejbližších sousedů pro vzorek uživatelů zobrazují grafy 4.61 a 4.62. Protože tentokrát máme v analyzovaných datech výrazně větší množství sezení než u předchozích příznaků, do okolí bodu zahrneme 250 nejbližších sousedů.



Obrázek 4.58: Porovnání podobností vlastních a cizích sezení.

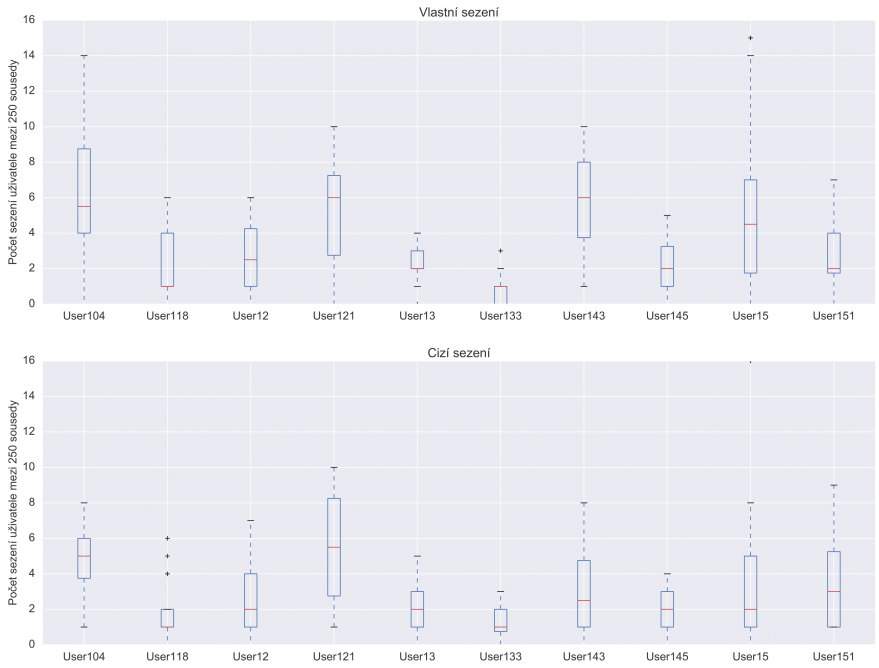


Obrázek 4.59: Rozdíly v rychlosti pohybu myši mezi sezeními.

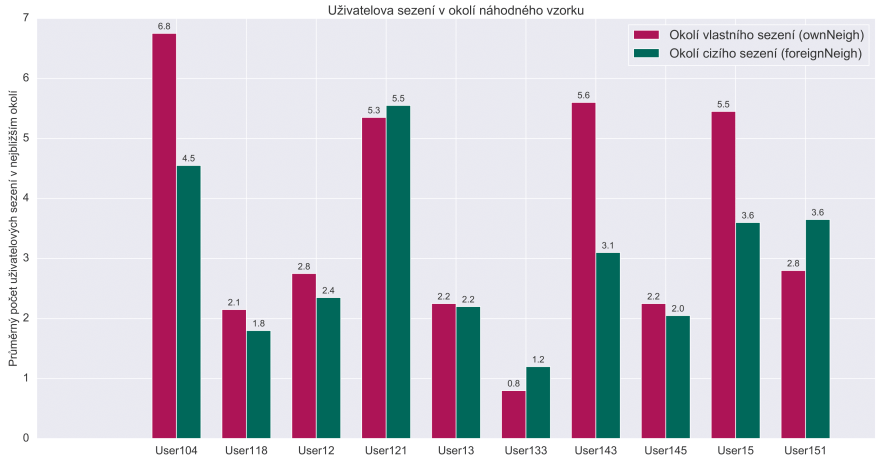


Obrázek 4.60: Podobnosti vlastních a cizích sezení pro uživatele.

I zde je situace obdobná jako v předchozím případě a okolí vlastních a cizích sezení se u většiny uživatelů příliš neliší. Celkově je mezi všemi analyzovanými uživateli průměrně $\Delta_{20,250} = 1{,}5$ krát více uživatelských sezení, pokud sledujeme nejbližší sousedy vlastního, a ne cizího sezení.



Obrázek 4.61: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.62: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

Závěr

Z výsledků předchozí analýzy vidíme, že průměrnou rychlost pohybu myši není možné použít pro spolehlivou charakterizaci uživatele, protože všichni uživatelé se v tomto ohledu chovají velmi podobně.

Otázkou zůstává, zda mohla být nevhodnost tohoto příznaku v našem kontextu zapříčiněna příliš hrubým zaznamenáváním pohybu myši, nebo by se prokázala, i kdybychom měli k dispozici detailní informace o každém detekovaném pohybu.

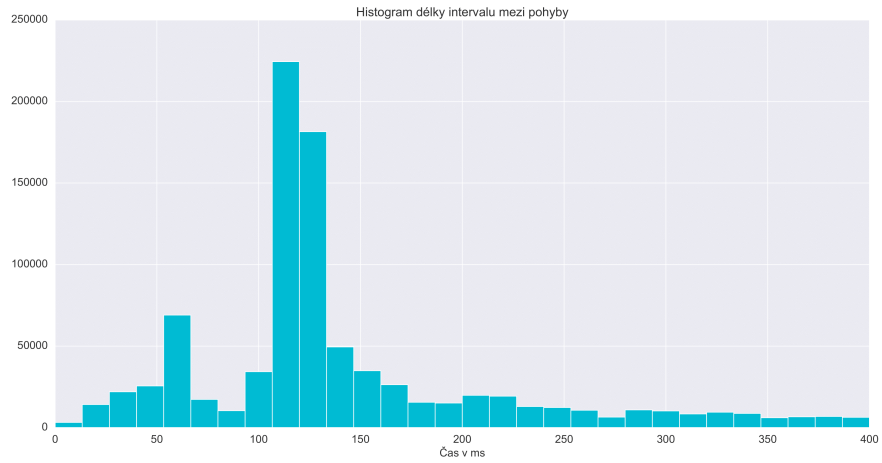
4.4.2 Pauza mezi pohyby myši

Délka prodlevy mezi dvěma záznamy o pohybu myši je jednoduchým příznakem, který můžeme z dat o práci uživatele s myší extrahovat. Tato charakteristika je v rámci autentizace uživatelů často zkoumána (blíže viz 2.2.2). Má tedy nepochybně potenciál ukázat se jako užitečná i v našem případě.

Otázkou je, jak její důvěryhodnost ovlivní skutečnost, že zaznamenáváme pouze omezený počet údajů o pohybu a jen každou sedmou takovou událost.

Předzpracování

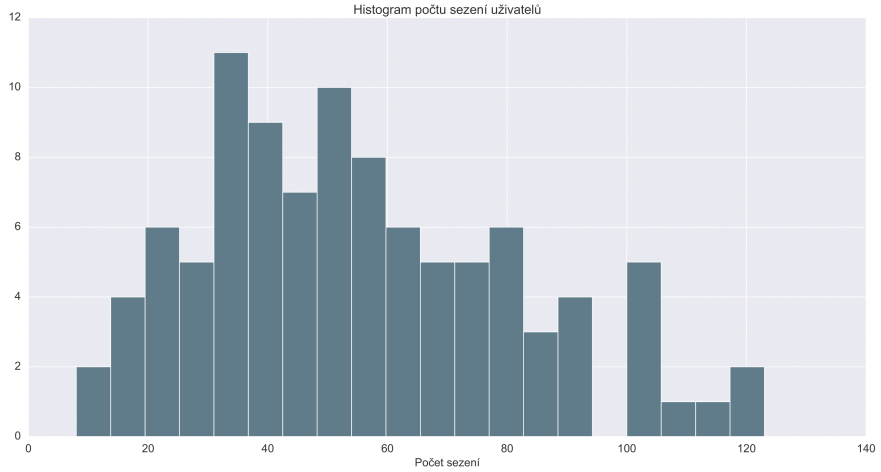
Pro každou dvojici záznamů pořízených při návštěvě stránky určíme dobu intervalu mezi nimi. Rozložení příslušných hodnot ukazuje histogram na obrázku 4.63. Stejně jako u průměrné rychlosti chceme i nyní uvažovat pouze pauzy mezi záznamy při souvislém pohybu. Proto i zde použijeme hranici 200 ms. Po odstranění nevyhovujících záznamů jich máme k dispozici 743 953 (69 % původního počtu).



Obrázek 4.63: Délka pauzy mezi pohyby myši.

Rozdělení do sezení

Zbylé záznamy rozdělíme podle toho, do kterých sezení patří. Počty sezení, které máme pro jednotlivé uživatele k dispozici, shrnuje histogram na obrázku 4.64. Naprostá většina uživatelů, 94 z nich, splňuje podmínku minimálního počtu 20 sezení. Data zbývajících šesti v další analýze nepoužijeme.



Obrázek 4.64: Počty sezení uživatelů.

Explorace

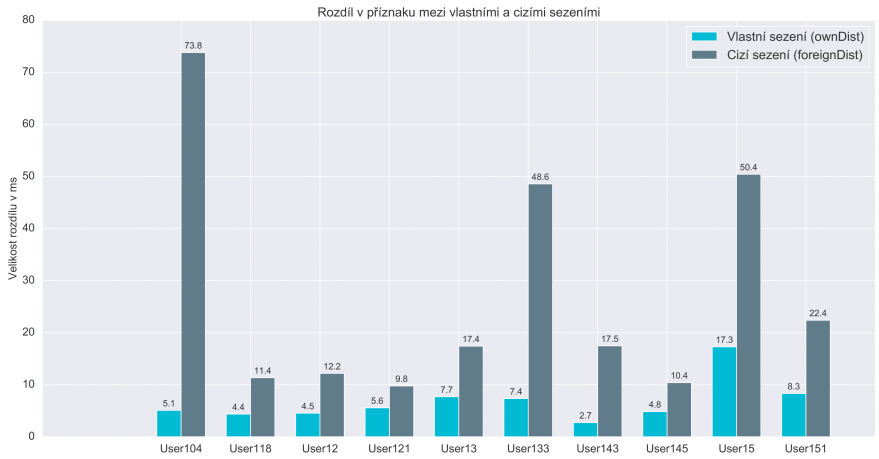
Pro každé sezení určíme průměrnou dobu, která uplyne mezi dvěma zaznamenanými pohyby. Výsledné hodnoty pro část uživatelů vidíme na obrázku 4.65. Na první pohled si v něm můžeme všimnout trojice uživatelů, jejichž data se od zbytku výrazně liší. Hodnoty ostatních se pak nejčastěji pohybují v intervalu od 120 do 140 ms.

Obrázek 4.65: Průměrná délka pauzy mezi pohyby myši u uživatelů.

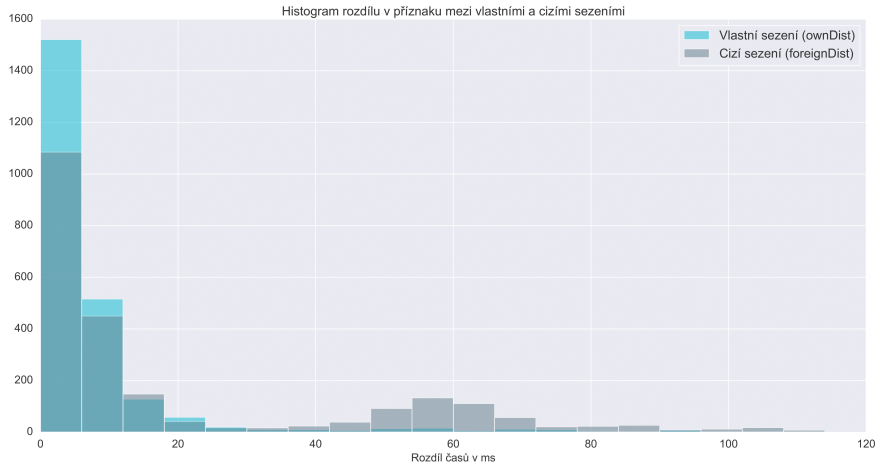
Diskriminační potenciál

Porovnání vzdáleností od vlastních a cizích sezení pro vzorek uživatelů ukazuje graf na obrázku 4.66. Z něj se zkoumaný příznak jeví jako velmi užitečný, neboť u téměř všech uživatelů je minimálně dvakrát větší vzdálenost k cizím než k vlastním sezením.

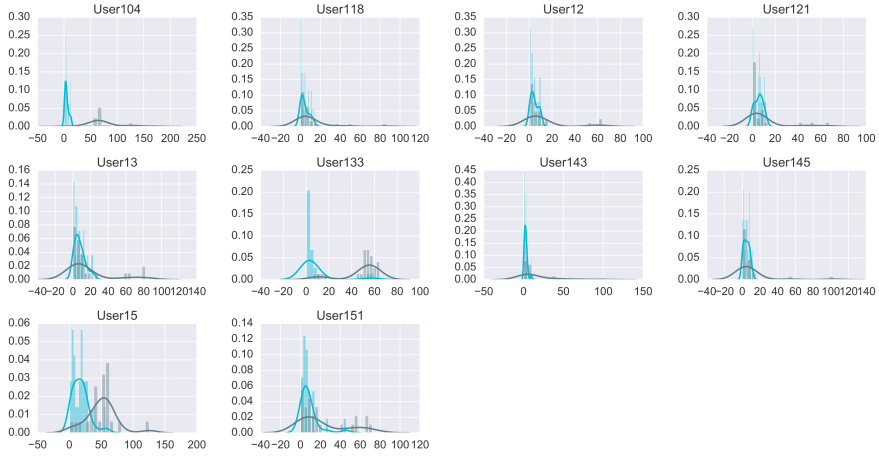
Histogram na obrázku 4.67 ukazuje rozložení těchto vzdáleností pro všechny uživatele. I zde se potvrzuje, že vzdálenosti mezi vlastními sezeními uživatele jsou zpravidla menší než mezi sezeními různých uživatelů navzájem. Průměrně mezi všemi uživateli je rozdíl vzdáleností vlastních a cizích sezení $\delta_{abs,25} = 13{,}1$ ms, což odpovídá relativnímu rozdílu $\delta_{rel,25} = 225$ %.



Obrázek 4.66: Porovnání podobností vlastních a cizích sezení.



Obrázek 4.67: Rozdíly v délce pauzy mezi pohyby myši v sezeních.



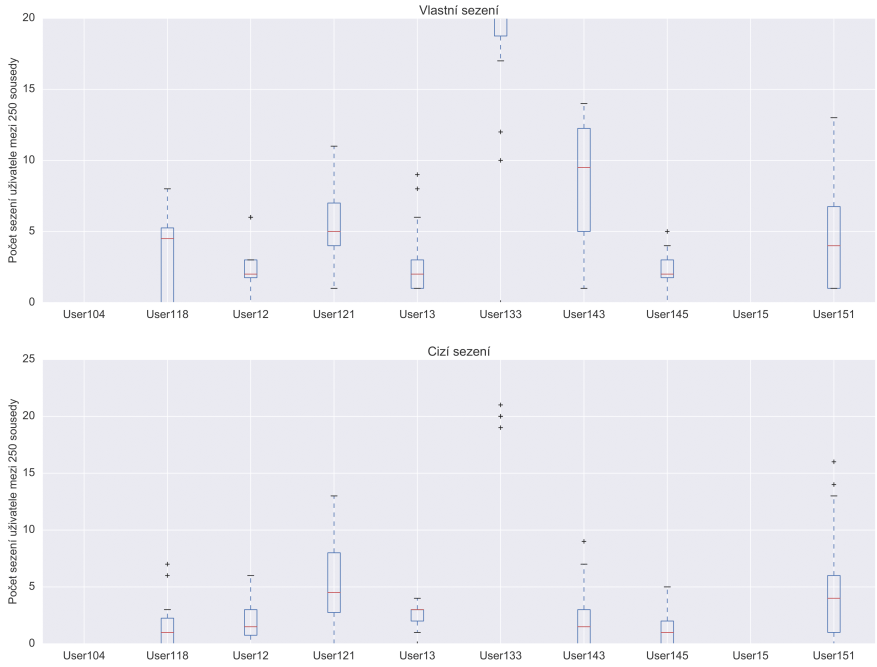
Obrázek 4.68: Podobnosti vlastních a cizích sezení pro uživatele.

Pravděpodobnostní hustoty pro vzorek uživatelů na obrázku 4.68 vypadají také ideálně. Výrazné vrcholy modré v okolí 0 znamenají, že všechna sezení uživatele jsou si z hlediska délky intervalů mezi pohyby hodně podobná. Naopak vrcholy šedé křivky leží od 0 dále nebo jsou nižší, což odpovídá tomu, že uživa-

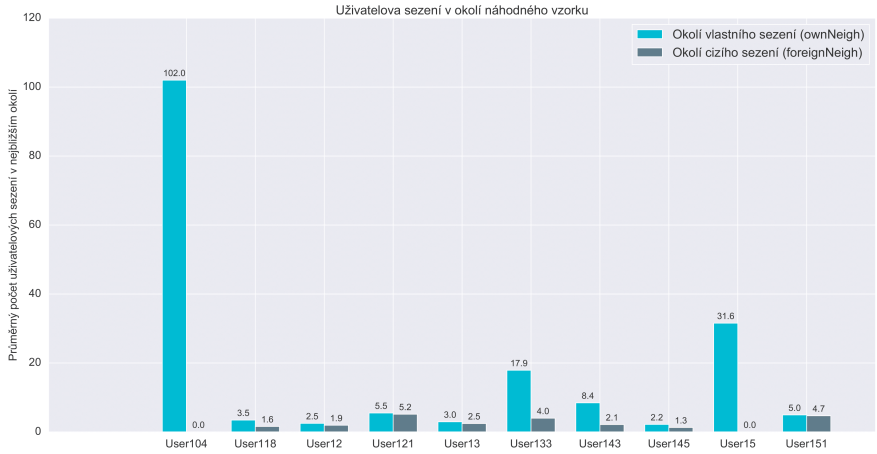
telova sezení se od sezení ostatních výrazně liší.

Výsledky analýzy uživatelských sezení mezi nejbližšími sousedy ukazují grafy na obrázcích 4.69 a 4.70. Vidíme, že pro část uživatelů je rozdíl v počtu jejich sezení v okolí vlastních a cizích sezení velmi výrazný. U většiny uživatelů je v okolí vlastních sezení minimálně dvakrát více dalších jejich sezení.

U několika uživatelů došlo k tomu, že v okolí žádného z náhodně vybraných cizích sezení nejsou mezi nejbližšími sousedy žádná jejich sezení. Aby i u nich bylo možné určit poměr počtu uživatelských sezení okolo vlastního oproti okolí cizího sezení, uměle zvýšíme tento počet pro okolí cizích sezení na 1. Takto upravený výpočet vede k $\Delta_{20,250} = 11{,}6$, což odpovídá tomu, že průměrně je mezi všemi testovanými uživateli 11,6 násobný nárůst počtu uživatelských sezení pro okolí vlastních sezení.



Obrázek 4.69: Počet uživatelových sezení mezi nejbližšími sousedy.



Obrázek 4.70: Srovnání počtu dalších uživatelových sezení v nejbližším okolí vlastního a cizího sezení.

Závěr

Z výsledků předchozí analýzy vidíme, že délka časového intervalu mezi záznamy o pohybu myši je charakterizujícím rysem pro většinu uživatelů. Hodnoty při testování jejího diskriminačního potenciálu vyšly nejlépe ze všech zkoumaných příznaků. Určitě má proto smysl zahrnout ji mezi příznaky pro ověřování identity uživatele.

Výhodou také je, že pohyb myši, ze kterého tuto charakteristiku odvozujeme, je nejčastějším typem interakce uživatele se stránkou. Obvykle tak máme k dispozici velký objem dat s informacemi o něm.

4.5 Shrnutí výsledků měření

Výsledky měření diskriminačního potenciálu všech uvažovaných příznaků shrnuje tabulka 4.1.

Název příznaku	Značení	δ_{abs}	δ_{rel}	Δ
Doba držení klávesy	KeyHold (H)	7 ms	69 %	3,6
Doba přechodu mezi klávesami	FlightTime (F)	15 ms	47 %	2,5
Procento překrytí kláves při psaní	OverlapsPerc (O)	6 %	101 %	3,0
Délka kliknutí	ClickDuration (C)	11 ms	84 %	3,4
Poloha kliknutí na tlačítko	SaveButtonClick	2 %	16 %	1,4
Poloha kliknutí na položku menu	MenuItemClick	2 %	11 %	1,5
Rychlost pohybu myši	MouseMoveSpeed	0,05 px/ms	27 %	1,5
Pauza mezi pohyby myši	SilencePeriod (S)	13 ms	225 %	11,6

Tabulka 4.1: Diskriminační potenciál uvažovaných příznaků.

Rozdíl vzdáleností vlastních a cizích sezení u jednotlivých příznaků můžeme porovnávat pomocí příslušného relativního rozdílu δ_{rel}. Pro potenciálně užitečné příznaky požadujeme minimálně 30% nárůst vzdálenosti, když místo sezení jednoho uživatele navzájem porovnáváme sezení různých uživatelů, neboli aby platilo $\delta_{rel} \geq 30$ %.

Z hlediska analýzy nejbližších sousedů sezení jsme zvolili jako hraniční hodnotu $\Delta = 2$. To znamená, že v okolí uživatelových sezení musí být průměrně alespoň dvojnásobně více dalších jeho sezení než v okolí cizího sezení.

Obě stanovené podmínky splňuje pět z osmi uvažovaných příznaků. Jsou jimi všechny tři příznaky odvozené od práce s klávesnicí, délka kliknutí a délka intervalu mezi pohyby myši. Poslední jmenovaná dosáhla zdaleka nejlepších výsledků z hlediska obou kritérií.

Naopak rychlost pohybu myši, ani žádný z příznaků týkajících se polohy kurzoru při klikání nedosahují ani jedné ze stanovených hranic. V autentizačním modelu je proto nepoužijeme.

5. Modely a jejich úspěšnost

V této kapitole nejprve představíme základní modely strojového učení, jež pro ověřování identity uživatelů budeme používat. Poté popíšeme jejich vstupní data, která jsme vytvořili extrakcí vybraných příznaků z datasetu popsaného v sekci 3.3.2.

V části 5.3 se budeme zabývat výběrem nejlepších kombinací příznaků pomocí analýzy klastrů a jejich vizualizace s použitím algoritmu t-SNE. Konečně v závěru kapitoly otestujeme úspěšnost jednotlivých modelů při autentizaci i identifikaci uživatelů.

5.1 Použité modely

Pro autentizaci a identifikaci uživatelů jsme uvažovali pětici modelů strojového učení – klasifikátor používající algoritmus k nejbližších sousedů (k *nearest neighbors*, KNN), naivní Bayesovský klasifikátor (*naive Bayes classifier*, NB), rozhodovací stromy (*decision trees*, DT), support vector machines (SVM) a random forest (RF). Jsou zde tak zastoupeny modely obvykle používané ve studiích zabývajících se dynamikou práce s myší a klávesnicí, včetně SVM, které zpravidla dosahují mezi ostatními nejlepších výsledků (blíže viz 2.3).

Modely budou použity pro klasifikaci. To znamená, že jejich vstupem bude vektor hodnot jednotlivých příznaků, také označovaný jako vzor, a výstupem modelu bude třída, kam klasifikátor předložený vzor zařadil. Při autentizaci jsou možné výstupní třídy pouze dvě - sezení je buď označeno za oprávněné, nebo za podvodné. V kontextu identifikace odpovídají výstupní třídy jednotlivým uživatelům systému.

Při popisu základního fungování modelů vycházíme z [16] a [7].

5.1.1 Algoritmus k nejbližších sousedů

Klasifikátor používající algoritmus k nejbližších sousedů patří mezi neparametrické modely. Neprobíhá zde žádná fáze učení, model si pouze zapamatuje všechny předložené trénovací vzory. Při klasifikaci nového vzoru se určí k jemu nejbližších uložených bodů v příznakovém prostoru a vzoru je přiřazena třída, která je mezi nejbližšími sousedy nejčastější.

Pro dosažení dobrých výsledku je klíčová volba vhodného k, tedy počtu uvažovaných nejbližších sousedů. Dále je třeba zvolit vhodnou metriku, ve které bude měřena vzdálenost vzorů v příznakovém prostoru. Nejčastěji používanou metrikou je Eukleidovská.

KNN lze s výhodou použít v situacích, kdy má jedna třída více různých základních vzorů, díky tomu, že dokáže vytvářet nepravidelné rozhodovací hranice mezi třídami v příznakovém prostoru. Na druhou stranu je kvůli tomu tento druh klasifikátoru citlivý na lokální strukturu dat.

Pokud mají vstupní příznakové vektory příliš vysokou dimenzi, projevuje se zde tzv. „prokletí dimenzionality“ (*curse of dimensionality*), kdy jsou všichni nejbližší sousedé velmi daleko od klasifikovaného vzoru. V takovém případě je vhodné nejprve na data aplikovat některou z metod pro redukci dimenzionality.

5.1.2 Naivní Bayesovský klasifikátor

Naivní Bayesovský klasifikátor je model založený na Bayesově větě pro výpočet podmíněné pravděpodobnosti. Označení naivní má proto, že používá předpoklad vzájemné podmíněné nezávislosti jednotlivých příznaků, pokud je známa příslušná výstupní třída.

Při klasifikaci jsou pro nově předložený příznakový vektor (x_1,x_2,...,x_n) určeny pravděpodobnosti toho, že náleží do jednotlivých výstupních tříd. Pro třídu C_k je pravděpodobnost p, že k ní daný vzor patří, dána vztahem

$$p(C_k|x_1,x_2,...x_n) \propto p(C_k) \cdot \prod_{i=1}^{n} p(x_i|C_k)$$

Vzoru je nakonec přiřazena třída, pro kterou vyšla tato pravděpodobnost nejvyšší.

Trénovací data slouží k tomu, aby bylo možné určit relativní četnosti výstupních tříd a podmíněné pravděpodobnosti hodnot příznaků pro každou z nich. Pokud jsou příznaky kategoriální, používají se k výpočtu příslušných pravděpodobností relativní četnosti jednotlivých kategorií. Pokud jsou příznaky spojité, je vhodné použít Gaussovský naivní Bayesovský klasifikátor, kde jsou pravděpodobnosti hodnot příznaků modelovány normálním rozdělením, jehož parametry jsou odvozeny na základě trénovacích dat.

Výhodou použití NB je, že tento přístup lze bez problému aplikovat i v případě, kdy v předloženém vzoru některý z příznaků chybí. Díky předpokládané podmíněné nezávislosti je také možné pracovat i s vysoce dimenzionálními daty, pro která by bylo obtížné určovat sdružené pravděpodobnosti všech příznaků.

Problematická je situace, kdy máme pro některou kombinaci třídy a hodnoty příznaku příliš málo vzorků, protože pak může být příslušná podmíněná pravděpodobnost podhodnocena. Je také potřeba ošetřit případy, kdy je pravděpodobnost některé kombinace podle trénovacích dat nulová.

5.1.3 Rozhodovací stromy

Rozhodovací stromy fungují na principu postupného rozdělování příznakového prostoru do pravoúhlých oblastí. Každá z oblastí má přiřazenou třídu, do níž jsou vektory v ní klasifikovány.

Způsob dělení prostoru je popsán pomocí struktury stromu. Vnitřní uzly mají přiřazeny příznaky, podle kterých se příznakový prostor rozděluje. Hrany jsou ohodnoceny predikáty vztahujícími se k danému příznaku a vedou do uzlů, jež odpovídají oblastem nově vzniklým po dělení podle jeho hodnoty. Každému z listů stromu a současně oblasti prostoru, kterou list reprezentuje, je přiřazena některá z výstupních tříd modelu.

Při klasifikaci nového příznakového vektoru se prochází vytvořený rozhodovací strom od kořene a v každém uzlu se pokračuje po hraně odpovídající hodnotě příslušného dělicího příznaku. Nakonec se nový vzor ohodnotí třídou, která patří k listu, kde průchod stromu skončil.

Existuje mnoho různých postupů, jak rozhodovací strom na základě trénovacích dat vybudovat. Je třeba rozhodnout o velikosti a hloubce stromu a zvolit strategii pro výběr dělicích atributů. Nejčastěji se jako kritérium výběru příznaku používá buď Giniho index, nebo entropie, které jsou popsány například v [7].

Slabinou modelu je, že při volbě příznaků pro dělení nelze zohlednit jejich vzájemnou korelaci. Problematická je také často velká variabilita vytvářených modelů v závislosti na předložené trénovací množině. Dále může být obtížné zvolit optimální hloubku a tvar tak, aby byl vzniklý strom dostatečně spolehlivý, ale současně se příliš nepřizpůsobil pouze trénovacím datům (problém *overfittingu*).

Naopak jednou z hlavních předností DT je snadná interpretovatelnost. Z výsledného modelu vidíme způsob použití jednotlivých atributů, a lze jej tak využít při výběru nejdůležitějších příznaků. Další výhoda spočívá v tom, že model dokáže pracovat s kategoriálními i spojitými příznaky současně a nevyžaduje normalizaci vstupních dat. Díky jednoduchosti a efektivitě algoritmu je použití DT vhodné i na velkých databázích.

5.1.4 Support vector machines

SVM model je založen na hledání optimální dělicí nadroviny, která co nejlépe oddělí vektory ze dvou výstupních tříd. Nejlepším oddělením je myšleno takové, kde jsou vektory z různých tříd dokonale separovány a současně je maximální vzdálenost dělicí nadroviny a jí nejbližšího trénovacího vzoru. Jde tedy o optimalizační úlohu, kterou lze vyřešit pomocí kvadratického programování s využitím Langrangeových multiplikátorů (blíže viz [16]).

Pokud nejsou vstupní data lineárně separovatelná, může být řešením jejich transformace z příznakového prostoru do prostoru vyšší dimenze. Nejčastěji se k tomuto účelu používají polynomy různých stupňů nebo radiální bázové funkce (RBF). Příslušné zobrazení se obecně označuje jako kernel. Tímto způsobem lze docílit flexibilní nelineární rozhodovací hranice mezi vzory z různých tříd.

Na konci tréninku je dělicí nadrovina popsána pomocí jí nejbližších trénovacích vzorů. Ty se označují jako podpůrné vektory (*support vectors*). Při klasifikaci nového vzoru je mu přiřazena třída podle jeho polohy vzhledem k dělicí nadrovině v transformovaném příznakovém prostoru. Základní verze SVM funguje pro dělení vzorů do dvou tříd. Pokud je úkolem klasifikace do více tříd, je třeba algoritmus dále upravit.

SVM dosahují velmi dobrých výsledků v mnoha různých úlohách. Jejich nevýhodou může být to, že vyžadují příznaky pouze s číselnými hodnotami. Pokud jsou hodnoty příznaku kategorie, je třeba jej transformovat. Na rozdíl od rozhodovacích stromů nelze naučené parametry výsledného modelu, tedy popis dělicí nadroviny, nijak snadno interpretovat.

5.1.5 Random forest

Random forest je model složený z většího množství rozhodovacích stromů. Vstupní příznakový vektor zpracovává každý ze stromů zvlášť a výstupem celého modelu je pak výsledek většinového hlasování mezi všemi obsaženými stromy.

Aby kombinace jednoduchých modelů dosáhla jako celek výrazně lepší přesnosti, je třeba, aby jednotlivé modely byly co nejméně korelované. Toho se u RF dosahuje dvěma způsoby. Zaprvé se každý z vytvářených stromů učí na vlastní trénovací množině. Ta se z původních trénovacích dat vytváří uniformním vzorkováním s opakováním (technika označovaná jako *bootstrap aggregating*). Druhou úpravou vedoucí k odstranění korelace je, že při každém dělení stromu se zvažuje

pouze část ze všech příznaků.

Při použití RF k řešení konkrétního problému musíme kromě parametrů vytvářených stromů určit také, kolik jich model bude obsahovat. Důležitým faktorem je i počet příznaků uvažovaných při dělení stromu.

Využití v ansámblu se pro DT obzvlášť hodí, protože jejich kombinací se snižuje variabilita samostatných modelů. Řeší se zde i problém s přílišným přizpůsobením trénovacím datům, protože každý ze stromů trénuje na jiné množině vzorů. Naopak většina výhod spojených s použitím DT zůstává v random forest modelu zachována. Cenou za lepší výsledky modelu je pouze jeho větší komplexita a výpočetní náročnost.

5.2 Vstupní data

Při tvorbě autentizačního modelu použijeme všechna uživatelská data, která jsme shromáždili. Jde o záznamy o interakci s myší a klávesnicí od více než pěti set uživatelů interní firemní aplikace sbírané po dobu téměř osmi měsíců. Bližším popisem získaných dat se zabývají sekce 3.3 a 3.4.

5.2.1 Příznaky

Z těchto dat extrahujeme příznaky, které jsme v kapitole 4 vyhodnotili jako potenciálně užitečné pro odlišení jednotlivých uživatelů. Použijeme všechny uvažované příznaky z klávesnice, tedy dobu držení klávesy, dobu přechodu mezi klávesami a procento překrytí kláves při psaní. Z příznaků odvozených od dynamiky myši budeme pracovat pouze s délkou kliknutí a pauzou mezi pohyby myši.

K příznakům, které pracují přímo s časy, tedy u všech kromě procentuálního překrytí, přidáme ještě informace o směrodatné odchylce veličin. (Tu budeme značit přidáním písmene *s* ke zkratce příslušného příznaku). Pro každé sezení tak budeme mít celkem devět příznaků, které shrnuje tabulka 5.1.

Název příznaku	Značení
Doba držení klávesy	KeyHold (H)
Směrodatná odchylka doby držení klávesy	KeyHoldStd (Hs)
Doba přechodu mezi klávesami	FlightTime (F)
Směrodatná odchylka doby přechodu mezi klávesami	FlightTimeStd (Fs)
Procento překrytí kláves při psaní	OverlapsPerc (O)
Délka kliknutí	ClickDuration (C)
Směrodatná odchylka délky kliknutí	ClickDurationStd (Cs)
Pauza mezi pohyby myši	SilencePeriod (S)
Směrodatná odchylka délky pauzy mezi pohyby myši	SilencePeriodStd (Ss)

Tabulka 5.1: Použité příznaky.

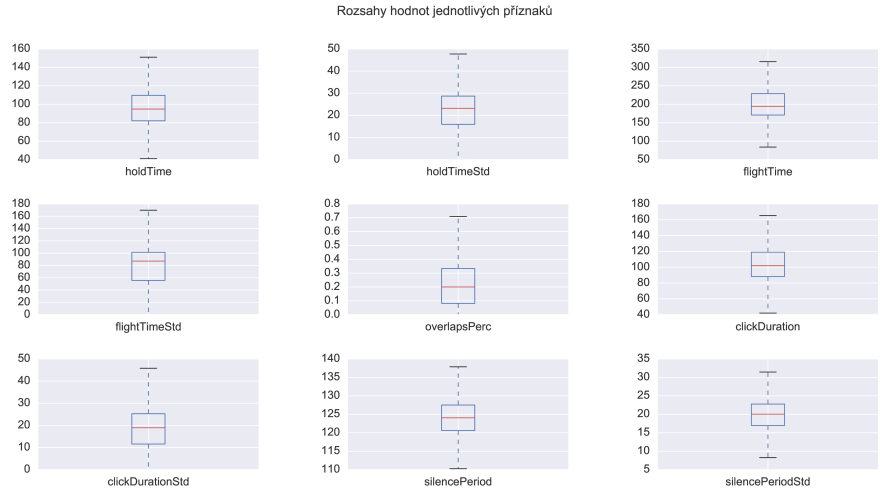
V rámci předzpracování při extrakci příznaků uplatníme podmínky na záznamy, které jsme pro jednotlivé příznaky stanovili v příslušných sekcích v předchozí kapitole:

- Pro dobu držení klávesy uvažujeme jen záznamy, kde nebyla použita kombinační klávesa (Shift, Alt nebo Control), nebyl zaznamenán vícenásobný stisk

klávesy a kde doba jejího držení náleží do intervalu 40 až 200 ms.

- Pro dobu přechodu mezi klávesami je interval platných hodnot záznamů 50 až 500 ms.
- U procenta překrytí kláves při psaní podobně jako u doby držení klávesy neuvažujeme kombinační klávesy ani opakované použití téže klávesy. Je zde také podmínka rozdílu časů uvolnění předchozí klávesy a stisku následující v intervalu od −250 do 400 ms.
- Validní délka kliknutí je od 40 do 220 ms.
- Pauzy mezi pohyby myši omezujeme shora hodnotou 200 ms.

V grafech na obrázku 5.1 můžeme vidět obvyklé rozsahy hodnot jednotlivých příznaků v sezeních uživatelů. Pro další použití budeme všechny příznaky normalizovat pomocí min-max normalizace do intervalu [0,1].



Obrázek 5.1: Rozsahy hodnot příznaků.

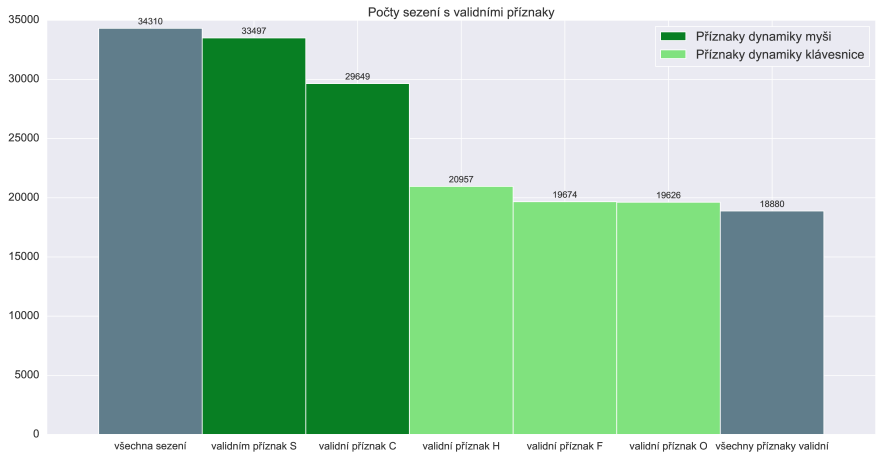
5.2.2 Sezení

Celkem máme k dispozici 34 310 záznamů o sezeních uživatelů. Ne pro všechna sezení je ovšem možné extrahovat všechny požadované příznaky. Častá jsou například sezení, kdy uživatel vůbec nepracoval s klávesnicí, a není proto možné určit hodnotu žádného z příznaků, které jsou od práce s ní odvozeny.

Graf na obrázku 5.2 shrnuje počty sezení, pro něž jsou jednotlivé příznaky k dispozici. Z něj vidíme, že problematické jsou především příznaky spojené s klávesnicí, zatímco příznaky odvozené od práce s myší je možné extrahovat z velké většiny sezení.

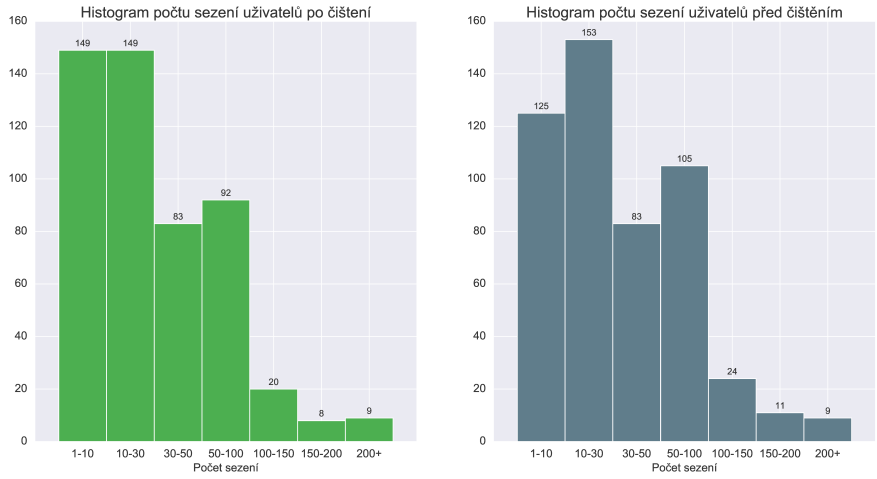
Nadále budeme pracovat pouze se záznamy o sezeních, kde je možné určit všechny požadované příznaky. Takových máme celkem 18 880, což představuje 55 % původního počtu.

Pro učení modelů potřebujeme dostatečně velký vzorek sezení každého z uživatelů. Histogram na obrázku 5.3 ukazuje rozdělení počtů sezení jednotlivých uživatelů. První z dvojice grafů na obrázku ukazuje situaci, se kterou budeme



Obrázek 5.2: Počty sezení, z nichž lze extrahovat jednotlivé příznaky.

pracovat, kdy jsou odstraněna sezení s nekompletními informacemi o příznacích. Druhý graf pak umožňuje porovnat, jak se toto rozdělení změní, pokud ponecháme v datech všechna původní sezení.



Obrázek 5.3: Počty sezení uživatelů.

Jako požadovanou hranici minimálního počtu sezení na uživatele jsme zvolili 30. Tuto podmínku splní 212 z více než 500 uživatelů, jejichž data máme k dispozici.

5.3 Volba kombinace příznaků

V této části vybereme vhodné kombinace příznaků, které budeme předkládat autentizačním modelům. Pro jejich selekci použijeme analýzu klastrů a vybrané kombinace poté vizualizujeme pomocí algoritmu t-SNE.

5.3.1 Analýza klastrů

Hodnoty příznaků vztahující se k danému sezení můžeme interpretovat jako jeho souřadnice v příznakovém prostoru s dimenzí odpovídající počtu uvažova-

ných příznaků. Jako klastr pak budeme označovat množinu bodů příslušných sezením konkrétního uživatele. Takto můžeme v příznakovém prostoru odlišit 212 klastrů vztahujících se k jednotlivým uživatelům.

Ideálně zvolené příznaky definují v příznakovém prostoru kompaktní klastry, které jsou vzájemně co nejvíce odděleny. Kompaktnost klastru odpovídá stabilitě v chování uživatele a vzdálenost od ostatních znamená, že ho pomocí jeho chování dokážeme dobře odlišit od jiných uživatelů.

Pro výběr nejlepších kombinací příznaků budeme porovnávat průměrné hodnoty poměru velikosti klastru a jeho průměrné vzdálenosti od ostatních klastrů v prostoru daném zvolenou kombinací příznaků.

Do klastru uživatele budeme uvažovat 80 % nejkompaktnějších bodů odpovídajících jeho sezením. Střed klastru je pak aritmetickým průměrem jejich souřadnic a velikost klastru je vzdálenost středu od nejvzdálenějšího z bodů, které do klastru zahrnujeme.

Pokud máme tedy uživatelská sezení určená příznakovými vektory x_1,x_2,...,x_n, střed příslušného klastru k určíme jako

$$c_k = \frac{x_1 + x_2 + ... + x_n}{n}$$

a odpovídající velikost klastru je

$$s_k = \max_{1 \leq i \leq n} |c_k - x_i|.$$

Vzdáleností klastrů je myšlena vzdálenost jejich středů. Pro klastr k je tedy průměrná vzdálenost od všech ostatních 211 klastrů definována jako

$$dist_k = \frac{\sum_{i=1, i \neq k}^{212} |c_k - c_i|}{211}.$$

Porovnávaný průměrný poměr velikosti klastru a jeho vzdálenosti od ostatních definujeme jako

$$\chi = \sum_{i=1}^{212} \frac{s_i}{dist_i}.$$

Tuto charakteristiku budeme srovnávat pro všechny možné neprázdné kombinace příznaků.

Výsledné poměry byly v rozmezí od 0,55, pokud uvažujeme pouze délku pauzy mezi pohyby myši, do 2,48, kdy používáme pouze směrodatnou odchylku doby přechodu mezi klávesami.

Nejlepší kombinace příznaků každé z možných velikostí shrnuje tabulka 5.2. V každém řádku je uvedena i příslušná hodnota χ nejlepší kombinace. Tabulka nezahrnuje jedinou kombinaci o velikosti 9, tedy použití všech příznaků, kterému odpovídá charakteristika $\chi = 1{,}1$.

Můžeme si všimnout, že hodnota χ nejlepší kombinace s její velikostí roste. Z tabulky také můžeme odvodit pořadí v kombinacích nejužitečnějších příznaků, kterými jsou postupně délka pauzy mezi pohyby myši (S), délka kliknutí (C) a doba držení klávesy (H). Naopak jako nejméně užitečná informace se jeví směrodatná odchylka doby přechodu mezi klávesami (Fs).

Nadále budeme pracovat pouze s kombinacemi příznaků obsaženými v tabulce a variantou, která používá všechny příznaky. To nám umožní využít nejlepší kombinace a současně sledovat vliv postupného přidávání dalších příznaků.

#	χ	Nejlepší kombinace	Druhá nejlepší kombinace	Třetí nejlepší kombinace
1	0,55	S	H	C
2	0,78	H+S	C+S	F+S
3	0,83	H+C+S	H+S+Ss	H+F+S
4	0,87	H+C+S+Ss	H+Hs+C+S	H+C+Cs+S
5	0,92	H+Hs+C+S+Ss	H+F+C+S+Ss	H+C+Cs+S+Ss
6	0,96	H+Hs+F+C+S+Ss	H+Hs+C+Cs+S+Ss	H+F+C+Cs+S+Ss
7	1,00	H+Hs+F+C+Cs+S+Ss	H+Hs+F+O+C+S+Ss	H+Hs+O+C+Cs+S+Ss
8	1,04	H+Hs+F+O+C+Cs+S+Ss	H+Hs+F+Fs+O+C+S+Ss	H+Hs+F+Fs+C+Cs+S+Ss

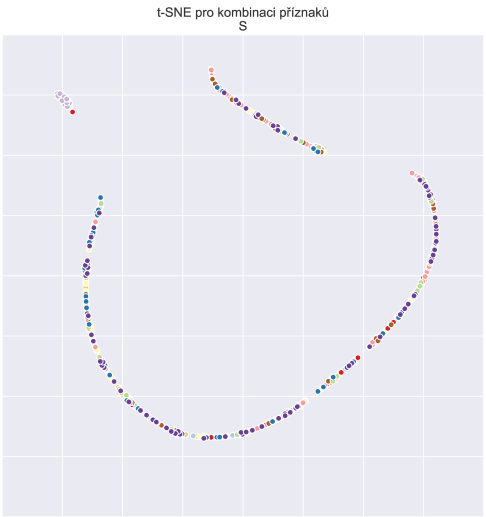
Tabulka 5.2: Nejlepší kombinace příznaků dané velikosti.

5.3.2 Vizualizace pomocí t-SNE

t-SNE neboli *t-distributed stochastic neighbour embedding* je algoritmus popsaný v článku [35], který slouží k redukci dimenzionality dat, přičemž se snaží o co nejlepší zachování charakteru jejich sousednosti. To znamená, že obrazy podobných objektů se po aplikaci t-SNE zobrazí blízko u sebe, zatímco obrazy hodně odlišných objektů daleko od sebe. Tato technika se používá především k vizualizaci vysoce dimenzionálních dat ve dvojrozměrném nebo trojrozměrném prostoru.

V našem případě použijeme t-SNE k tomu, abychom body z příznakového prostoru zobrazili ve 2D. Pro větší přehlednost nebudeme pracovat s daty všech uživatelů, ale pouze malého vzorku deseti náhodně vybraných.

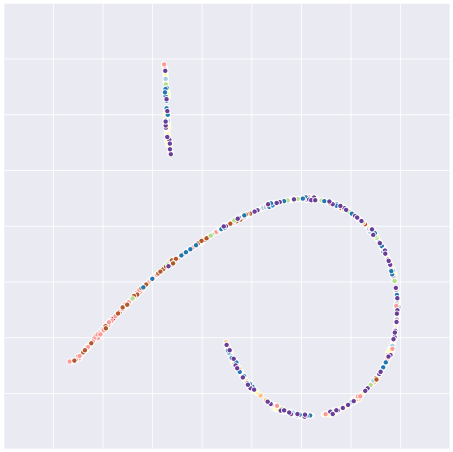
Následujících 25 obrázků ukazuje výsledky aplikace t-SNE na sezení zvolených uživatelů pro každou z uvažovaných kombinací příznaků. Sezení jednotlivých uživatelů jsou zde barevně odlišena.



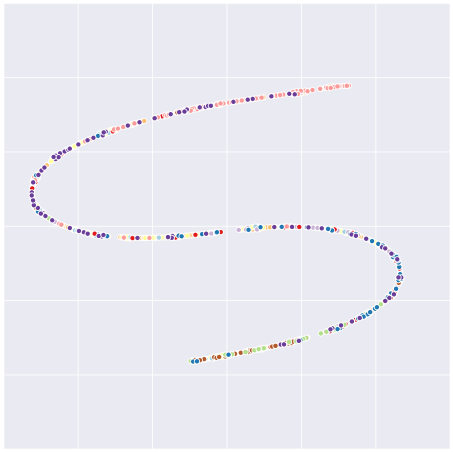
Můžeme si všimnout, že již při použití jediného příznaku S jsme schopni velmi dobře odlišit od zbytku sezení vyznačená světle fialovou barvou. Oddělení klastrů ostatních uživatelů je bohužel již problematičtější. Často sice tvoří body jedné barvy relativně kompaktní shluk, ale tyto shluky se vzájemně překrývají.

Slibné rozložení bodů můžeme vidět například na obrázcích příslušících kombinacím H+C+Cs+S+Ss, H+Hs+C+Cs+S+Ss a H+Hs+O+C+Cs+S+Ss. Zde je relativně dobře oddělen i klastr růžové barvy, od zbytku se odlišují body označené červenou a hnědou a kompaktní se zdají i sezení vyznačená světle zelenou.

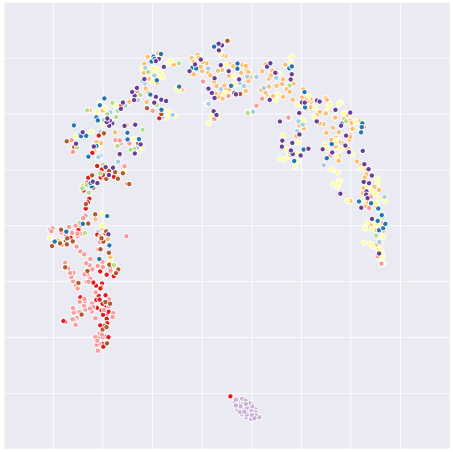
t-SNE pro kombinaci příznaků
H



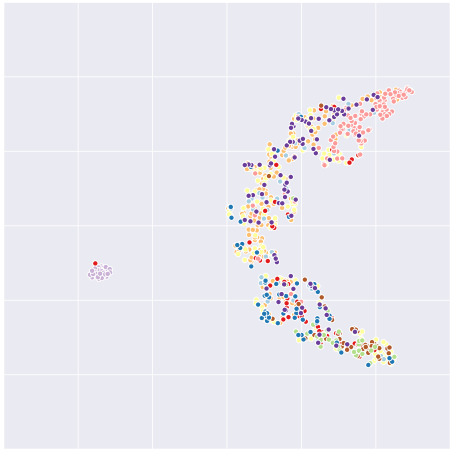
t-SNE pro kombinaci příznaků
C



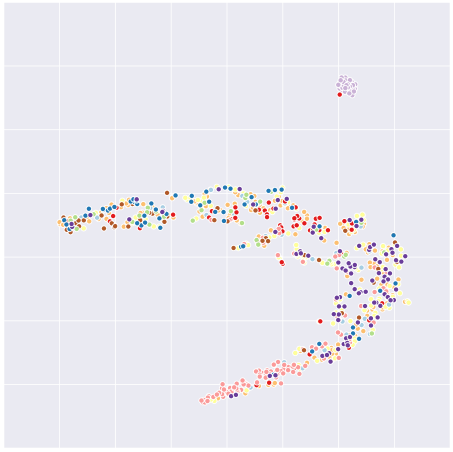
t-SNE pro kombinaci příznaků
H+S



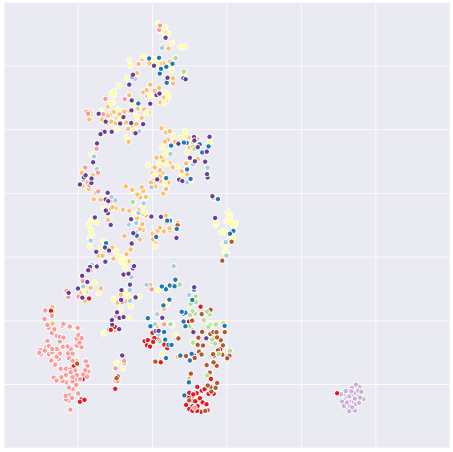
t-SNE pro kombinaci příznaků
C+S



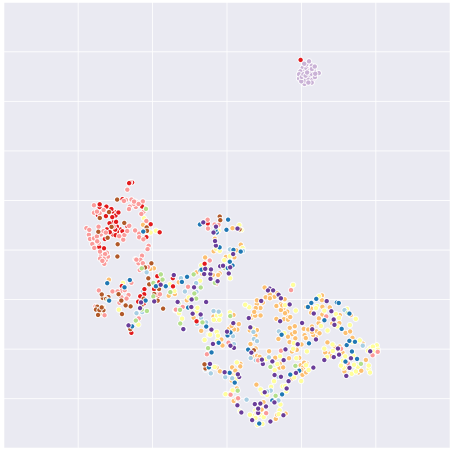
t-SNE pro kombinaci příznaků
F+S



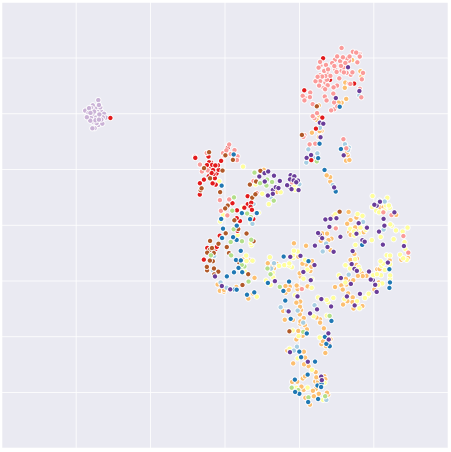
t-SNE pro kombinaci příznaků
H+C+S



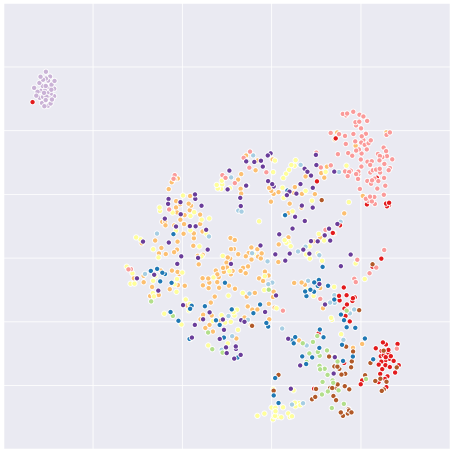
t-SNE pro kombinaci příznaků
H+S+Ss



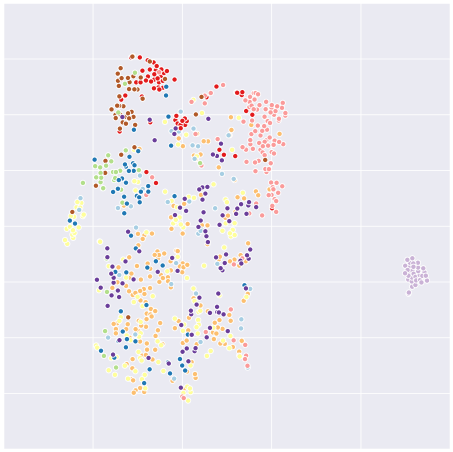
t-SNE pro kombinaci příznaků
H+F+S



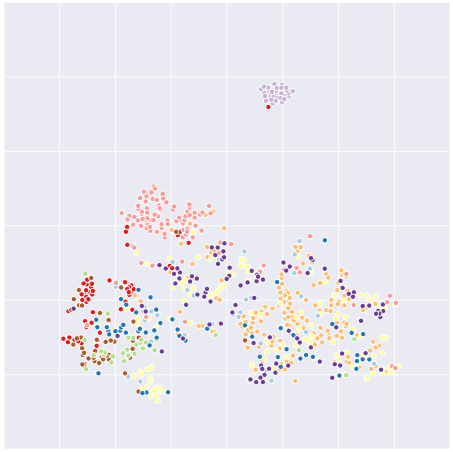
t-SNE pro kombinaci příznaků
H+C+S+Ss



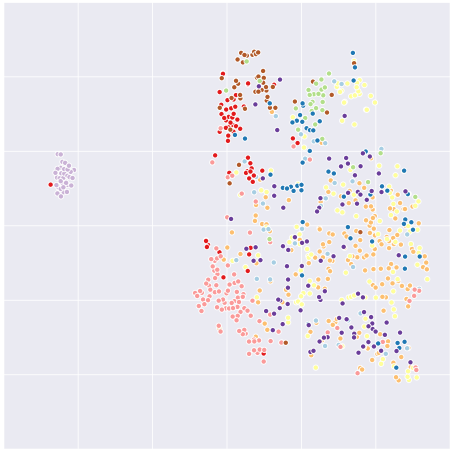
t-SNE pro kombinaci příznaků
H+Hs+C+S



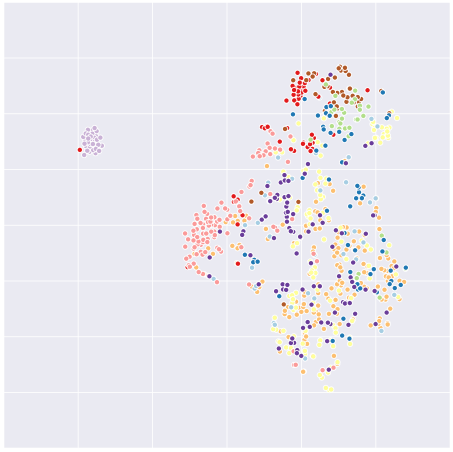
t-SNE pro kombinaci příznaků
H+C+Cs+S



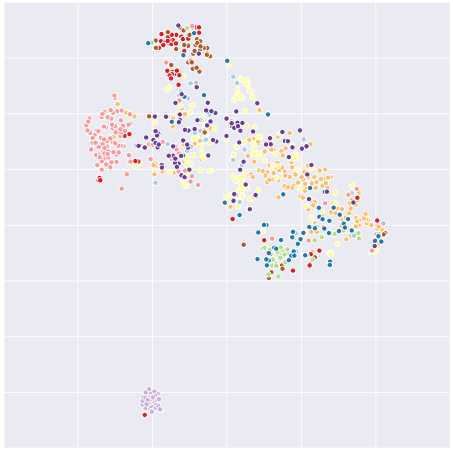
t-SNE pro kombinaci příznaků
H+Hs+C+S+Ss



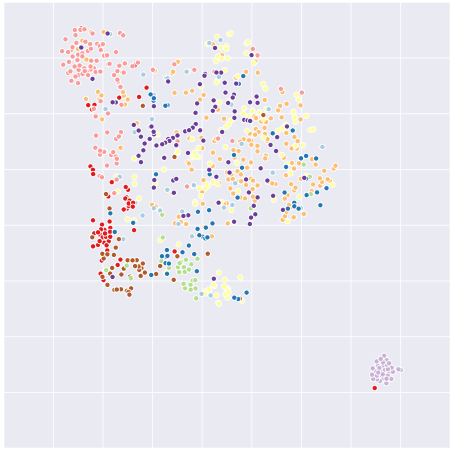
t-SNE pro kombinaci příznaků
H+F+C+S+Ss



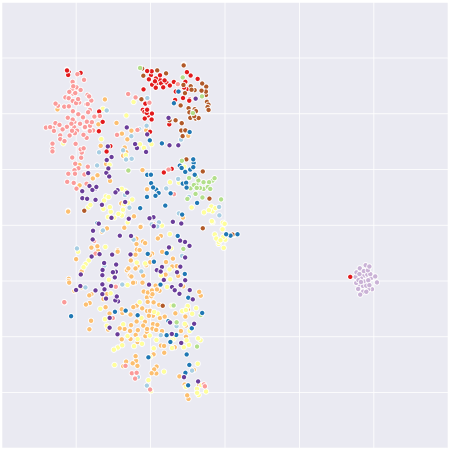
t-SNE pro kombinaci příznaků
H+C+Cs+S+Ss

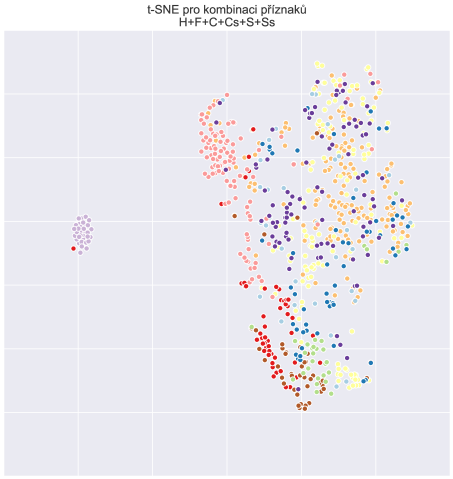


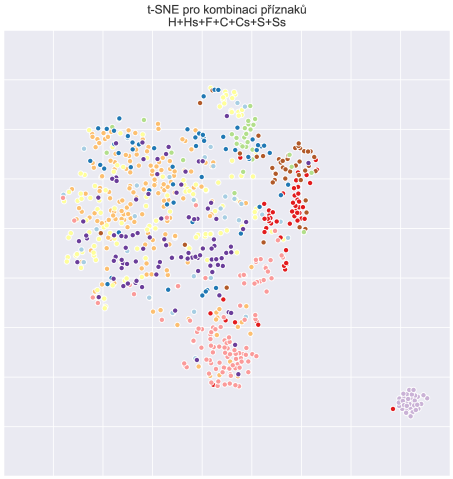
t-SNE pro kombinaci příznaků
H+Hs+F+C+S+Ss

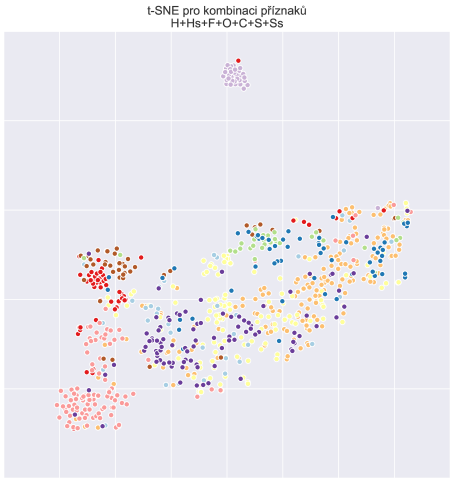


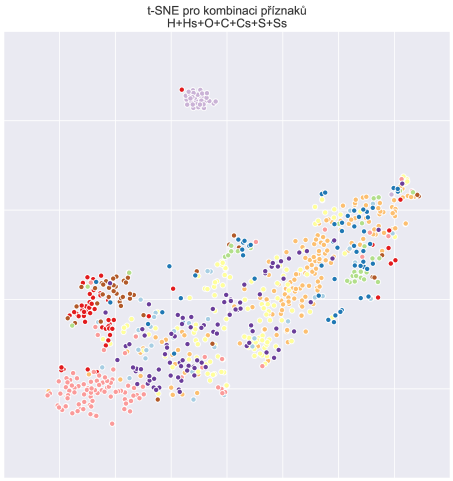
t-SNE pro kombinaci příznaků
H+Hs+C+Cs+S+Ss

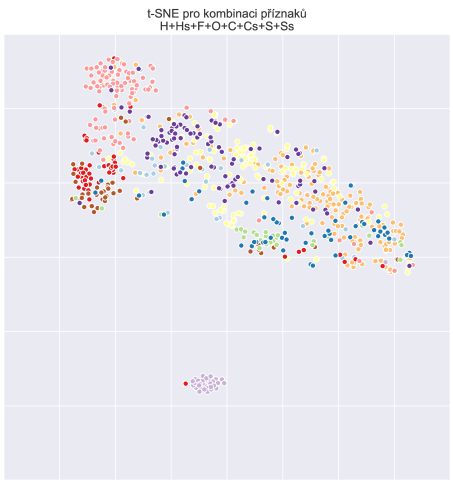


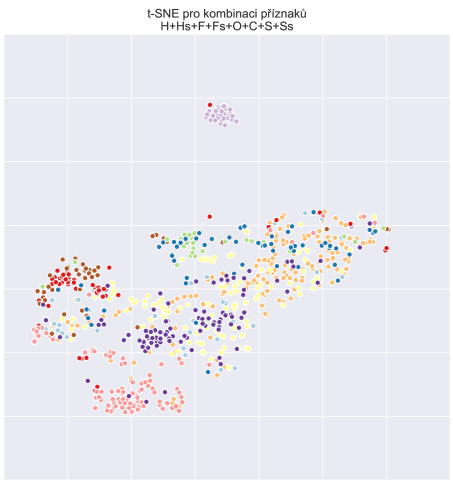

t-SNE pro kombinaci příznaků
H+F+C+Cs+S+Ss

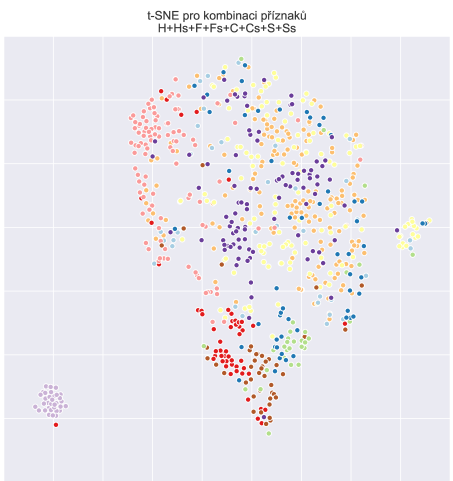

t-SNE pro kombinaci příznaků
H+Hs+F+C+Cs+S+Ss

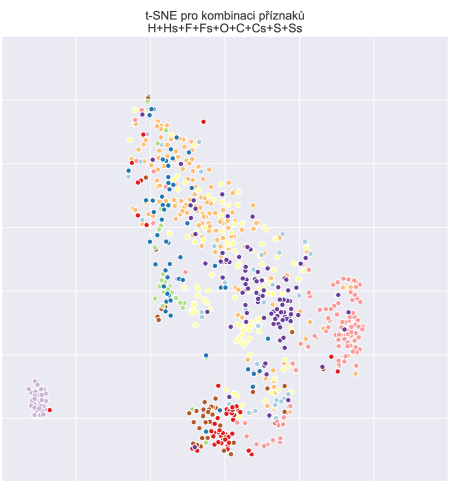

t-SNE pro kombinaci příznaků
H+Hs+F+O+C+S+Ss


t-SNE pro kombinaci příznaků
H+Hs+O+C+Cs+S+Ss


t-SNE pro kombinaci příznaků
H+Hs+F+O+C+Cs+S+Ss


t-SNE pro kombinaci příznaků
H+Hs+F+Fs+O+C+S+Ss


t-SNE pro kombinaci příznaků
H+Hs+F+Fs+C+Cs+S+Ss


t-SNE pro kombinaci příznaků
H+Hs+F+Fs+O+C+Cs+S+Ss

5.4 Úspěšnosti jednotlivých modelů

V této části konečně vyzkoušíme úspěšnosti různých autentizačních modelů. Budeme testovat všechny základní modely strojového učení představené v části 5.1. Použijeme implementace z knihovny *scikit-learn* v Pythonu ([26]). Všechny modely ponecháváme v jejich výchozím nastavení.

Pracujeme tedy s rozhodovacími stromy bez omezení hloubky, kde jako kritérium dělení používáme Giniho index. U algoritmu k nejbližších sousedů jsme navýšili hodnotu k z výchozích 5 na 11. Dále používáme Gaussovský naivní Bayesovský klasifikátor, SVM s výchozím RBF kernelem a random forest velikosti 100.

Při měření úspěšnosti všech modelů použijeme pětinásobnou křížovou validaci.

5.4.1 Autentizace

Systém pro ověřování identity uživatelů funguje tak, že pro každého uživatele má uložený natrénovaný model, který při předložení informací o jeho novém sezení rozhodne, zda se jedná o oprávněný přístup, nebo jde o podvodné přihlášení.

Pro učení model potřebuje mít k dispozici ukázky oprávněných i podvodných sezení. Protože v našem případě žádná skutečná neoprávněná přihlášení v datech nemáme, použijeme pro každého uživatele jako příklady falešných sezení náhodně vybraná sezení ostatních. Doplníme tedy k jeho sezením stejný počet sezení ostatních označených jako negativní příklady.

V tabulce 5.3 vidíme výsledné úspěšnosti jednotlivých modelů pro všechny uvažované kombinace příznaků při pětinásobné křížové validaci. Můžeme zde tak mimo jiné sledovat vliv postupného přidávání dalších příznaků na úspěšnost každého z modelů.

Z vypočtených hodnot vidíme, že za předpokladu, že máme k dispozici alespoň pět příznaků, dosahuje nejlepších výsledků RF. Srovnatelných výsledků lze docílit i s použitím SVM. Ostatní modely fungují ve většině případů o poznání hůře.

Celkově nejvyšší úspěšnosti 84,4 % bylo dosaženo s modelem random forest při použití kombinace H+Hs+F+O+C+Cs+S+Ss, tedy všech uvažovaných příznaků s výjimkou směrodatné odchylky času přechodu mezi klávesami.

Na obrázku 5.7 vidíme rozložení úspěšností autentizace jednotlivých uživatelů pro nejlepší variantu každého z modelů. Celkově nejúspěšnější random forest autentizuje uživatele s přesností od 53,3 do 100 %, s tím, že téměř čtvrtinu (50) z nich autentizuje s alespoň 90% přesností.

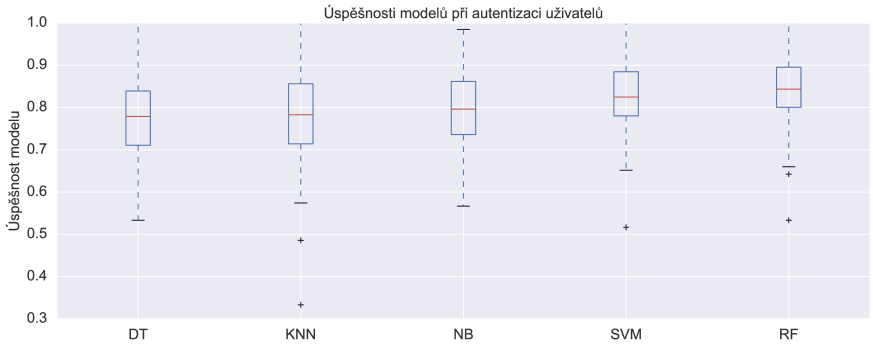
Tabulka 5.4 ukazuje úspěšnost modelů z hlediska jimi dosažených hodnot FAR a FRR (pro vysvětlení těchto metrik viz 2.4). Uvádíme zde výsledky pouze pro dvě nejúspěšnější kombinace příznaků, kterými jsou H+Hs+F+O+C+Cs+S+Ss (varianta 1) a H+Hs+F+Fs+O+C+Cs+S+Ss (varianta 2).

Můžeme si všimnout, že u modelů DT a RF se od sebe hodnoty FAR a FRR nijak výrazně neliší, zatímco pro zbytek modelů je FRR výrazně nižší než FAR. To znamená, že pro tyto modely je častější správné přijetí oprávněného uživatele než zamítnutí neoprávněného přístupu.

Celkově nejlepší výsledné FAR 14,7 % bylo dosaženo za použití modelu random forest a nejlepší FRR 11,1 % pak s algoritmem KNN.

Kombinace	DT	KNN	NB	SVM	RF
H+Hs+F+Fs+O+C+Cs+S+Ss	0,781406	0,778732	0,79717	0,830762	0,840493
H+Hs+F+O+C+Cs+S+Ss	0,781931	0,784398	0,797002	0,830605	**0,844013**
H+Hs+F+Fs+O+C+S+Ss	0,779156	0,776164	0,790956	0,824227	0,838168
H+Hs+F+Fs+C+Cs+S+Ss	0,774928	0,782475	0,790926	0,828606	0,834931
H+Hs+F+C+Cs+S+Ss	0,772983	0,785024	0,791213	0,824907	0,829553
H+Hs+F+O+C+S+Ss	0,780504	0,783258	0,790908	0,825973	0,837432
H+Hs+O+C+Cs+S+Ss	0,777791	0,780281	0,786653	0,821994	0,832064
H+Hs+F+C+S+Ss	0,769483	0,785281	0,782554	0,819225	0,823909
H+Hs+C+Cs+S+Ss	0,764938	0,780935	0,777536	0,81204	0,815369
H+F+C+Cs+S+Ss	0,767645	0,783471	0,781059	0,817448	0,818647
H+Hs+C+S+Ss	0,760817	0,778838	0,769139	0,806025	0,808954
H+F+C+S+Ss	0,760723	0,780255	0,773354	0,81046	0,814026
H+C+Cs+S+Ss	0,773738	0,776322	0,780711	0,814091	0,82451
H+C+S+Ss	0,747445	0,775896	0,757604	0,795767	0,791283
H+Hs+C+S	0,759886	0,779541	0,771983	0,800838	0,803978
H+C+Cs+S	0,755594	0,776777	0,767201	0,795702	0,80393
H+C+S	0,744298	0,772547	0,759638	0,792026	0,788322
H+S+Ss	0,690585	0,721031	0,708041	0,733066	0,731688
H+F+S	0,720861	0,747253	0,731026	0,760323	0,758692
H+S	0,683207	0,716292	0,705506	0,722405	0,718018
C+S	0,708982	0,744071	0,711175	0,74864	0,743382
F+S	0,66526	0,705984	0,669086	0,70392	0,699064
S	0,61537	0,616168	0,628066	0,651403	0,615412
H	0,618433	0,660581	0,672102	0,674728	0,618851
C	0,645352	0,69217	0,678616	0,704638	0,645968

Tabulka 5.3: Úspěšnosti jednotlivých modelů při dané kombinaci příznaků.



Obrázek 5.7: Rozdělení úspěšností modelů při autentizaci jednotlivých uživatelů.

Metrika	Varianta příznaků	DT	KNN	NB	SVM	RF
FAR	1	0,216	0,320	0,267	0,203	**0,147**
	2	0,218	0,325	0,263	0,196	0,152
FRR	1	0,217	**0,111**	0,138	0,135	0,165
	2	0,222	0,117	0,142	0,142	0,167

Tabulka 5.4: Dosažená FAR a FRR jednotlivých modelů.

5.4.2 Identifikace

Pro zajímavost se zaměříme i na to, jak dobře by si jednotlivé modely vedly v úloze identifikace uživatelů. Ta je mnohem obtížnější než pouhé ověřování deklarované identity. V tomto případě je úkolem modelu na základě informací o sezení určit uživatele, kterému patří. Na rozdíl od autentizace zde hraje důležitou roli počet uživatelů v systému. S jejich narůstajícím počtem je stále těžší všechny od sebe navzájem odlišit.

Tentokrát budeme uvažovat pouze dvě kombinace příznaků, které se v předchozí části ukázaly jako nejúspěšnější, což jsou H+Hs+F+O+C+Cs+S+Ss (varianta 1) a H+Hs+F+Fs+O+C+Cs+S+Ss (varianta 2). V tabulce 5.5 vidíme úspěšnosti každého z modelů při různých počtech uživatelů v systému.

Počet uživatelů	Varianta	DT	KNN	NB	SVM	RF
20	1	0,512	0,620	0,558	0,609	0,655
20	2	0,514	0,620	0,558	0,615	0,660
40	1	0,386	0,497	0,426	0,479	0,538
40	2	0,391	0,501	0,430	0,486	0,545
80	1	0,295	0,411	0,337	0,389	0,451
80	2	0,297	0,416	0,338	0,398	0,459
120	1	0,261	0,372	0,299	0,349	0,412
120	2	0,261	0,381	0,301	0,360	0,422
160	1	0,227	0,335	0,265	0,309	0,373
160	2	0,229	0,342	0,267	0,319	0,383
212	1	0,199	0,307	0,239	0,280	0,346
212	2	0,204	0,312	0,241	0,292	0,355

Tabulka 5.5: Úspěšnosti jednotlivých modelů při dané kombinaci příznaků a počtu uživatelů v systému.

Z tabulky vyplývá, že i v tomto případě je nejúspěšnějším modelem random forest. Druhým nejúspěšnějším však už nejsou SVM, ale algoritmus k nejbližších sousedů. Obě testované kombinace příznaků dávají u všech modelů velmi podobné výsledky s tím, že vždy vychází mírně lépe varianta 2, kdy používáme všechny příznaky.

V tabulce můžeme také sledovat vliv počtu uživatelů v systému na jeho úspěšnost. Ta i u nejlepšího modelu výrazně klesá z 65,5 % pro systém s 20 uživateli na pouhých 35,5 % v situaci, kdy je třeba odlišit všech 212 uživatelů.

6. Závěr

V rámci práce byl vytvořen dataset pro studium dynamiky práce s klávesnicí a myší. Tento dataset se skládá ze tří částí. První obsahuje informace o uživatelských interakcích s klávesnicí, druhá údaje o klikání uživatelů a třetí záznamy o pohybech myši. Data byla zaznamenána při práci zaměstnanců firmy Profinit v prostředí její webové aplikace.

Sběr dat probíhal v období od konce října 2019 do konce května 2020. Dataset zahrnuje interakce 512 různých uživatelů a obsahuje 292 568 záznamů o jimi navštívených stránkách v průběhu celkem 34 310 uživatelských sezení. Jde tak o unikátní dataset z hlediska počtu zapojených uživatelů, délky období sběru dat i toho, že data pocházejí z reálně používané aplikace. Dataset je v upravené verzi zveřejněn jako příloha této práce.

Vybrali jsme a experimentálně vyhodnotili diskriminační schopnosti devíti příznaků, které je možné ze získaných dat pro každé sezení extrahovat. Zkoumali jsme tři příznaky odvozené od práce s klávesnicí (dobu držení klávesy, dobu přechodu mezi klávesami a procentuální překryv kláves při psaní), tři příznaky vycházející ze záznamů o klikání (délku trvání kliknutí a polohy kurzoru při klikání na tlačítko a odkaz v menu) a dva příznaky vztahující se k pohybu myši (rychlost pohybu myši a délku pauz mezi zaznamenanými pohyby).

Také jsme navrhli dva možné obecné přístupy, jak diskriminační potenciál příznaků hodnotit. První z nich spočívá v porovnávání podobností vlastních a cizích sezení. Druhá metoda pak sleduje nárůst počtu dalších uživatelových sezení v okolí vlastních sezení. Z devíti uvažovaných příznaků jsme tímto postupem vyhodnotili pět jako potenciálně přínosné.

Nakonec jsme změřili úspěšnost, které při ověřování identity uživatelů dosahují základní modely strojového učení, využívající vybrané příznaky. Uvažovali jsme pětici modelů, konkrétně rozhodovací stromy, algoritmus k nejbližších sousedů, Gaussovský naivní Bayesovský klasifikátor, SVM a Random Forest. Testovací data zahrnovala 212 uživatelů, z nichž pro každého bylo k dispozici minimálně 30 sezení.

Nejvyšší úspěšnosti 84,4 % bylo dosaženo při použití modelu Random Forest. Tomuto výsledku odpovídají hodnoty FAR 14,7 % a FRR 16,5 %. S tímto modelem byla u téměř čtvrtiny všech uživatelů dosažená přesnost autentizace minimálně 90 %.

Dosažené hodnoty FRR a FAR jsou v porovnání s výsledky v citovaných studiích (viz 2.4) výrazně vyšší. Tento rozdíl mohl být způsoben mnoha faktory. Pravděpodobně jde zejména o důsledek malého počtu použitých příznaků. Ostatní studie typicky využívají desítky různých příznaků, zatímco my jsme pracovali pouze s devíti.

Další příčinou je samotný kontext, ve kterém se snažíme uživatele autentizovat. Při práci s jednoduchými stránkami aplikace je většina uživatelských sezení velmi krátká a neposkytuje mnoho údajů o dynamice práce s klávesnicí.

Ambicí této práce nebylo překonání nejlepších výsledků z jiných studií, ale spíše prozkoumání a vyhodnocení jednoduchých příznaků odvozených od dynamiky klávesnice a myši nad unikátním nově vytvořeným datasetem. V tomto světle můžeme výsledek, kterého jsme dosáhli s velmi základní sadou příznaků a

modelů hodnotit jako uspokojivý.

Možnosti dalšího rozšíření

Na dosavadní práci lze navázat například v některém z těchto směrů:

- Jednou z možností, jak vylepšit dosaženou úspěšnost autentizace, je zavedení nových příznaků extrahovaných ze sezení. V rámci práce byla posuzována pouze základní sada příznaků. Větší pozornost by si například jistě zasloužila samotná dynamika práce s myší. Mohli bychom se také zaměřit na obvyklé sekvence akcí uživatelů na konkrétních stránkách aplikace.

- Diskriminačním schopnostem příznaků odvozených od dynamiky myši by mohlo pomoci efektivnější zaznamenávání uživatelských interakcí. Například bychom mohli již v prohlížeči agregovat data o událostech spojených s pohybem, výsledné záznamy by pak byly přesnější, aniž by zabíraly příliš mnoho místa v databázi.

- Alternativně by ke zlepšení dosažených výsledků mohla vést optimalizace autentizačních modelů, které byly v práci použity pouze ve výchozím nastavení knihovny. Je také možné zkoumat další modely, které byly mimo rámec práce. Velký potenciál mají například v dnešní době velmi populární neuronové sítě.

- Vytvořený dataset je možné použít nejen pro studium možností statické autentizace na základě celého sezení jako v našem případě, ale i pro průběžnou autentizaci. Díky tomu, že je u všech zaznamenaných akcí časová značka, je možné pro každé sezení zrekonstruovat jeho průběh. Průběžně autentizující model by pak dostával na vstup jednotlivé akce seřazené chronologicky a jeho úkolem by bylo po co nejmenším počtu předložených akcí rozeznat chování neoprávněných uživatelů.

- Zajímavé by také bylo zaměřit se na vliv časového odstupu mezi sezeními uživatele. Například natrénovat model na sezeních ze začátku sledovaného období a poté jej otestovat na sezeních z doby po několika měsících.

- Dalším faktorem ke studiu by mohl být i počet sezení uživatele. Můžeme například sledovat vztah mezi tím, kolik sezení je pro uživatele při tréninku k dispozici, a přesností, s jakou jej dokáže naučený model autentizovat.

- Problémem, který se v průběhu práce objevil, je situace, kdy z některých sezení není možné extrahovat všechny požadované příznaky (například pokud v průběhu sezení uživatel vůbec nepracuje s klávesnicí). Pro účely testování modelů jsme tuto potíž vyřešili odstraněním nevhodných sezení. V reálné aplikaci by ovšem bylo lepší navrhnout autentizační model tak, aby dokázal pracovat i se sezeními, kde je k dispozici jen část příznaků.

- Další otázkou mimo rozsah této práce je problematika vytvoření referenčního profilu uživatelů. Tedy toho, jak zajistit dostatek sezení, kde je uživatelova identita ověřena, na jejichž základě je možné trénovat nový model pro

jeho budoucí autentizaci. Pro využití v praxi by tedy bylo nejprve třeba navrhnout metodiku získávání registračního vzorku uživatelova chování a stejně tak i způsoby, jak vytvořený uživatelský profil průběžně aktualizovat.

Seznam použité literatury

[1] Ahmed, A. A. E. a Traore, I. (2007). A new biometric technology based on mouse dynamics. *IEEE Transactions on Dependable and Secure Computing*, **4**, 165–179. doi: 10.1109/TDSC.2007.70207. URL http://ieeexplore.ieee.org/document/4288179/.

[2] Ahmed, A. A. E., Ahmed, E. a Traore, I. (2006). A statistical model for biometric verification. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.571.1933&rep=rep1&type=pdf.

[3] Awad, A. a Traore, I. (2010). Mouse dynamics biometric technology. In Liang Wang, X. G., editor, *Behavioral Biometrics for Human Identification: Intelligent Applications*, chapter 10, pages 207–223. IGI Global, Hershey PA.

[4] Baesens, B., Vlasselaer, V. V. a Verbeke, W. (2015). *Fraud Analytics Using Descriptive, Predictive, and Social Network Technique*. John Wiley & Sons. ISBN 1119146828.

[5] Banerjee, S. a Woodard, D. (2012). Biometric authentication and identification using keystroke dynamics: A survey. *Journal of Pattern Recognition Research*, **7**, 116–139. doi: 10.13176/11.427.

[6] Bartlow, N. a Cukic, B. (2006). Evaluating the reliability of credential hardening through keystroke dynamics. In *Proceedings - International Symposium on Software Reliability Engineering, ISSRE*, pages 117 – 126. doi: 10.1109/ISSRE.2006.25. URL http://ieeexplore.ieee.org/document/4021977/.

[7] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer. ISBN 0387310738. URL https://www.microsoft.com/en-us/research/publication/pattern-recognition-machine-learning/.

[8] Bours, P. a Ellingsen, J. (2018). Cross keyboard keystroke dynamics. In *Proceedings - 2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*, pages 1–6. doi: 10.1109/CAIS.2018.8441945.

[9] Cherifi, F., Hemery, B., Giot, R., Pasquet, M. a Rosenberger, C. (2010). Performance evaluation of behavioral biometric systems. In Liang Wang, X. G., editor, *Behavioral Biometrics for Human Identification: Intelligent Applications*, chapter 3, pages 57–74. IGI Global, Hershey PA.

[10] Curtin, M., Tappert, C., Villani, M., Ngo, G., Simone, J. a Cha, S.-H. (2006). Keystroke biometric recognition on long-text input: A feasibility study. *Proceeding International Workshop Scientific Computing and Computational Statistics (IWSCCS 2006)*.

[11] Everitt, R. a McOwan, P. (2003). Java-based internet biometric authentication system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**, 1166 – 1172. doi: 10.1109/TPAMI.2003.1227991.

[12] Feher, C., Elovici, Y., Moskovitch, R., Rokach, L. a Schclar, A. (2012). User identity verification via mouse dynamics. *Information Sciences*, **201**, 19–36. doi: 10.1016/j.ins.2012.02.066.

[13] Fridman, L., Stolerman, A., Acharya, S., Brennan, P., Juola, P., Greenstadt, R. a Kam, M. (2014). Multi-modal decision fusion for continuous authentication. *Computers & Electrical Engineering*, **41**. doi: 10.1016/j.compeleceng.2014.10.018.

[14] Gunetti, D. a Picardi, C. (2005). Keystroke analysis of free text. *ACM Transactions on Information and System Security (TISSEC)*, **8**, 312–347. doi: 10.1145/1085126.1085129.

[15] Handa, J., Singh, S. a Saraswat, S. (2019). A comparative study of mouse and keystroke based authentication. In *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 670–674. ISBN 978-1-5386-5933-5. doi: 10.1109/CONFLUENCE.2019.8776953. URL `https://ieeexplore.ieee.org/document/8776953/`.

[16] Hastie, T., Tibshirani, R. a Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition. ISBN 0387848843. URL `http://ww,w-stat.stanford.edu/~tibs/ElemStatLearn/`.

[17] Henniger, O. (2010). Security evaluation of behavioral biometric systems. In Liang Wang, X. G., editor, *Behavioral Biometrics for Human Identification: Intelligent Applications*, chapter 2, pages 44–56. IGI Global, Hershey PA.

[18] Jain, A., Ross, A. a Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, **14**, 4 – 20. doi: 10.1109/TCSVT.2003.818349.

[19] Jorgensen, Z. a Yu, T. (2011). On mouse dynamics as a behavioral biometric for authentication. In *Proceedings of the 6th International Symposium on Information, Computer and Communications Security, ASIACCS 2011*, pages 476–482, New York, New York, USA, 01 2011. ACM Press. ISBN 9781450305648. doi: 10.1145/1966913.1966983.

[20] Lin, C.-C., Chang, C.-C. a Liang, D. (2012). A new non-intrusive authentication approach for data protection based on mouse dynamics. *2012 International Symposium on Biometrics and Security Technologies, ISBAST 2012*. doi: 10.1109/ISBAST.2012.11.

[21] Mondal, S. a Bours, P. (2013). Continuous authentication using mouse dynamics. In *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI)*.

[22] Mondal, S. a Bours, P. (2015). Context independent continuous authentication using behavioural biometrics. In *Proceeding IEEE International Conference on Identity, Security and Behavior Analysis (ISBA'15)*.

[23] Mondal, S. a Bours, P. (2016). Combining keystroke and mouse dynamics for continuous user authentication and identification. In *2016 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, pages 1–8. IEEE. doi: 10.1109/ISBA.2016.7477228. URL http://ieeexplore.ieee.org/document/7477228/.

[24] Mondal, S. a Bours, P. (2017). Person identification by keystroke dynamics using pairwise user coupling. *IEEE Transactions on Information Forensics and Security*, **PP**. doi: 10.1109/TIFS.2017.2658539.

[25] Nakkabi, Y., Traore, I. a Awad, A. (2010). Improving mouse dynamics biometric performance using variance reduction via extractors with separate features. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, **40**, 1345 – 1353. doi: 10.1109/TSMCA.2010.2052602.

[26] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. a Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

[27] Pusara, M. a Brodley, C. (2004). User re-authentication via mouse movements. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security - VizSEC/DMSEC '04*, pages 1–8. doi: 10.1145/1029208.1029210.

[28] Revett, K., Jahankhani, H., Tenreiro de Magalhaes, S. a Santos, H. (2008). *A Survey of User Authentication Based on Mouse Dynamics*, pages 210–219. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-540-69402-1. doi: 10.1007/978-3-540-69403-8_25.

[29] Seeger, M. a Bours, P. (2011). How to comprehensively describe a biometric update mechanisms for keystroke dynamics. *2011 Third International Workshop on Security and Communication Networks (IWSCN)*. doi: 10.1109/IWSCN.2011.6827718.

[30] Shen, C., Cai, Z. a Guan, X. (2012). Continuous authentication for mouse dynamics: A pattern-growth approach. In *Proceedings of the International Conference on Dependable Systems and Networks*, pages 1–12. IEEE. ISBN 978-1-4673-1624-8. doi: 10.1109/DSN.2012.6263955. URL http://ieeexplore.ieee.org/document/6263955/.

[31] Spillane, R. (1975). Keyboard apparatus for personal identification. *IBM Technical Disclosure Bulletin*, **17**.

[32] Tappert, C. C., Villani, M. a Cha, S.-H. (2010). Keystroke biometric identification and authentication on long-text input. In Liang Wang, X. G., editor, *Behavioral Biometrics for Human Identification: Intelligent Applications*, chapter 16, pages 342–368. IGI Global, Hershey PA.

[33] The jQuery Foundation (2020). jQuery.ajax(). [online]. [cit. 15. 07. 2020]. Dostupné z https://api.jquery.com/jquery.ajax/.

[34] Traore, I., Woungang, I., Obaidat, M., Nakkabi, Y. a Lai, I. (2012). Combining mouse and keystroke dynamics biometrics for risk-based authentication in web environments. In *2012 Fourth International Conference on Digital Home*, pages 138–145. IEEE. ISBN 978-1-4673-1348-3. URL http://ieeexplore.ieee.org/document/6376399/.

[35] van der Maaten, L. a Hinton, G. (2008). Viualizing data using t-sne. *Journal of Machine Learning Research*, **9**, 2579–2605.

[36] World Wide Web Consortium (2017). Xml path language (xpath). [online]. [cit. 15. 07. 2020]. Dostupné z http://https://www.w3.org/TR/xpath/.

[37] Yampolskiy, R. a Govindaraju, V. (2008). Behavioural biometrics: A survey and classification. *International Journal of Biometrics*, **1**. doi: 10.1504/IJBM.2008.018665.

[38] Zheng, N., Paloski, A. a Wang, H. (2011). An efficient user verification system via mouse movements. In *Proceedings of the ACM Conference on Computer and Communications Security*, volume 139–150, pages 139–150, New York, New York, USA, 10 2011. ACM Press. ISBN 9781450309486. doi: 10.1145/2046707.2046725.

[39] Zhong, Y., Deng, Y. a Jain, A. (2012). Keystroke dynamics for user authentication. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 117–123. IEEE. ISBN 978-1-4673-1611-8. doi: 10.1109/CVPRW.2012.6239225. URL http://ieeexplore.ieee.org/document/6239225/.

A. Přílohy

Obsahem elektronické přílohy jsou následující soubory a adresáře:

- `\Reporty` - adresář s analýzami jednotlivých příznaků a modelů exportovanými z Jupyter Notebooku ve formátu html.
  - `KeyHold.html`,
  - `FlightTime.html`,
  - `OverlapsPerc.html`,
  - `ClickDuration.html`,
  - `SaveButtonClick.html`,
  - `MenuItemClick.html`,
  - `MouseMoveSpeed.html`,
  - `SilencePeriod.html`,
  - `Models.html`.
- `\ProfisDataset` - adresář obsahující tři části vytvořeného datasetu ve formátu csv. Data jsou dodatečně anonymizována. V údajích o práci s klávesnicí jsou konkrétní napsaná písmena a číslice nahrazeny souhrnnými popisy „*letter*“ a „*number*“. V záznamech o pohybu myši jsou anonymizovány URL jednotlivých stránek.
  - `ClickData.zip`,
  - `KeyboardData.zip`,
  - `MouseMoveData.zip`.
- `\prace.pdf` - elektronická verze této práce ve formátu PDF/A.
- `\souhlas.pdf` - souhlas firmy Profinit se zpracováním a zveřejněním anonymizované verze vytvořeného datasetu.