Univerzita Pardubice

Fakulta ekonomicko-správní

Využití neuronových sítí při autentizaci prostřednictvím dynamiky psaní na klávesnici

Bc. Ctirad Kovář

Diplomová Práce

2014

Univerzita Pardubice
Fakulta ekonomicko-správní
Akademický rok: **2013/2014**

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Ctirad Kovář**
Osobní číslo:	**E100428**
Studijní program:	**N6209 Systémové inženýrství a informatika**
Studijní obor:	**Informatika ve veřejné správě**
Název tématu:	**Využití neuronových sítí při autentizaci prostřednictvím dynamiky psaní na klávesnici**
Zadávající katedra:	**Ústav systémového inženýrství a informatiky**

Zásady pro vypracování:

Cílem práce je aplikovat poznatky neuronových sítí na biometrickou autentizaci prostřednictvím dynamikly psaní na klávesnici.

Obsahem práce bude:

- analýza současného stavu autentizace prostřednictvím dynamiky psaní na klávesnici,
- navržení modelu autentizace využívajícího neuronové sítě,
- porovnání navrženého modelu se stávajícími přístupy.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

BÍLA, Jiří. Umělá inteligence a neuronové sítě v aplikacích. Vyd. 2., přepracované. Praha: ČVUT, 1998, 135 s. ISBN 80-010-1769-9.
DOBDA, Luboš. Ochrana dat v informačních systémech. 1. vyd. Praha: Grada Publishing, 1998, 286 s. ISBN 80-716-9479-7.
DOUCEK, Petr, Luděk NOVÁK a Vlasta SVATÁ. Řízení bezpečnosti informací. 1. vyd. Praha: Professional Publishing, 2008, 239 s. ISBN 978-80-86946-88-7.
DRAHANSKÝ, Martin a Filip ORSÁG. Biometrie. 1. vyd. [Brno: M. Drahanský], 2011, 294 s. ISBN 978-80-254-8979-6.
PATO, Joseph N a Lynette I MILLETT. Biometric recognition: challenges and opportunities. Washington, D.C.: National Academies Press, c2010, xv, 165 p. ISBN 03-091-4207-5.

Vedoucí diplomové práce: **doc. Ing. Miloslav Hub, Ph.D.**
Ústav systémového inženýrství a informatiky

Datum zadání diplomové práce: **1. října 2013**
Termín odevzdání diplomové práce: **30. dubna 2014**

L.S.

doc. Ing. Renáta Myšková, Ph.D.
děkanka

prof. Ing. Jan Čapek, CSc.
vedoucí ústavu

V Pardubicích dne 1. října 2013

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Dolní Dobrouči dne 14. 4. 2014

Ctirad Kovář

Poděkování

Na tomto místě bych rád poděkoval vedoucímu práce doc. Ing. Miloslavu Hubovi, Ph.D. za velice vstřícný přístup při vedení mé práce. Rád bych také poděkoval celé své rodině a všem přátelům za podporu, kterou mi poskytovali v průběhu celého studia a zejména při psaní této práce.

Děkuji vám.

ANOTACE

Práce se zaobírá ověřením pravosti uživatele - autentizací. Současné principy autentizace založené na znalosti (heslo, fráze) jsou často rozšiřovány o biometrickou složku. Jeden z možných biometrických údajů je například dynamika psaní na klávesnici, která byla použita v této práci. Zpracování údajů bylo provedeno s využitím neuronové sítě a několika dalších metod. V závěru práce jsou jednotlivé výsledky porovnány.

KLÍČOVÁ SLOVA

autentizace, dynamika psaní, biometrie, neuronová síť, biometrická autentizce

TITLE

Using of neural networks for authentication via keystroke dynamics

ANNOTATION

This work deals with authentication of user. Today, traditional methods of authentication (like password, passphrase) are more and more often enhanced with biometrics. One of possible biometric measures is keystroke dynamics which was used in this work. Data were processed using neural network and some other methods. In conclusion, all results are compared together.

KEYWORDS

authentication, keystroke dynamics, biometrics, neural network, biometric authentication

Obsah

Seznam zkratek

AAMLP - auto asociativní vícevrstvá síť(Auto-Associative Multi Layer Perceptron)

AANN - auto asociativní neuronová síť (Auto-Associative Neural network)

DD - prodleva mezi stiskem dvou kláves (Down-Down)

DET - kompromis chyb (Detection Error Tradeoff)

DNA - deoxyribonukleová kyselina (Deoxyribonucleic Acid)

EER - shodná míra chybovosti (Equal Error Rate)

FAR - chyba typu II (False Acceptance Rate)

FRR - chyba typu I (False Rejection Rate)

H - doba stisku (Hold)

PIN - osobní identifikační číslo (Personal Identification Number)

PP - prodleva mezi stiskem dvou kláves (Press-Press)

RP - prodleva mezi uvolněním jedné a stiskem druhé klávesy (Release-Press)

RR - prodleva mezi uvolněními dvou kláves (Release-Release)

SIM - účastnická identifikační karta (Subscriber Identity Module)

SMS - krátká textová zpráva (Short Message Service)

UD - prodleva mezi uvolněním jedné a stiskem druhé klávesy (Up-Down)

Seznam symbolů

D_{Euklid} - euklidovská vzdálenost

Di - čas potřebný pro stisk dvou kláves

$D_{Mahalanobis}$ - Mahalanobis vzdálenost

$D_{Manhattan}$ - Manhattan vzdálenost

D_N - míra shody skóre s předlohou

EER - shodná míra chybovosti

$event_i$ - i-tá událost klávesy

f - aktivační funkce neuronu

FAR - chyba typu II

FRR - chyba typu I

H - délka stisku klávesy

h_j - výstup j-tého neuronu

$N(P)_i$ - i-tá složka vektoru generovaného neuronovou sítí

N_{EAA} - počet všech pokusů oprávněných uživatelů o autentizaci

N_{FA} - počet chybných přijetí

N_{FR} - počet chybných odmítnutí

N_{IAA} - počet všech pokusů neoprávněných narušitelů o autentizaci

P' - referenční šablona

P - vstupní vzor

P_i - i-tá složka vektoru autentizačního pokusu

P'_i - i-tá složka vektoru referenční šablony

PP - prodleva mezi stiskem dvou kláves

RP - prodleva mezi uvolněním jedné a stiskem druhé klávesy

RR - prodleva mezi uvolněním dvou kláves

s - míra ztotožnění

Sim - vztah ztotožnění

s_N - skóre generované neuronovou sítí

Th - práh citlivosti

$time_i$ - časová značka i-té události

w_i - vstupní váhy do j-tého neuronu

x_i - výstupní hodnoty neuronů v předchozí vrstvě

σ_i - směrodatná odchylka i-té složky vektoru referenční šablony

σ_i^2 - rozptyl i-té složky vektoru referenční šablony

Seznam obrázků a tabulek

Úvod

Autentizace jako prostředek ověření identity člověka je v dnešní době nedílnou součástí lidské existence. Ať už člověk s pomocí přihlašovacího jména a hesla přistupuje na internetu ke své emailové schránce, potvrzuje platební pokyn v internetovém bankovnictví zadáním ověřovacího kódu zaslaného na mobilní telefon, nebo v obchodě platí debetní kartou a zadává PIN (personal identification number [25]), vždy se s setkává s určitou formou autentizace.

Zjednodušeně lze říci, že s autentizací se člověk setkává vždy, když jakýmkoliv způsobem prokazuje svoji totožnost - dokazuje druhé straně, že se jedná skutečně o něho. Nejedná se tak pouze o elektronickou autentizaci, ale za autentizaci je možné považovat i předložení občanského průkazu na poštovní přepážce.

Tato diplomová práce je zaměřena na využití neuronových sítí při autentizaci prostřednictvím dynamiky psaní na klávesnici. Dynamika psaní na klávesnici patří mezi stále častěji využívané metody biometrické autentizace. V práci jsou rozvedeny a popsány stávající přístupy na poli biometrické autentizace, zejména pak autentizace s využitím dynamiky psaní na klávesnici.

Součástí práce je vytvoření modelu autentizace prostřednictvím dynamiky psaní na klávesnici s využitím neuronových sítí. V další části práce je tento model aplikován na datech veřejně dostupných na internetu a výsledek je porovnán s několika metodami, které neuronové sítě nevyužívají.

1. Autentizace

Původ samotného slova autentizace lze hledat v řeckém authentikos - pravý, spolehlivý. K přejmutí do českého jazyka došlo pravděpodobně z německého Authentifizierung či Authentisierung, francouzského authentification, případně z anglického authentication. Ve všech případech má slovo obdobný význam: proces ověření identity uživatele.

V češtině se pak tyto výrazy uvádějí zpravidla jako autentizace, autentikace nebo autentifikace. Akademický slovník cizích slov zachycuje pouze podstatné jméno autentičnost, autenticita a sloveso autentizovat (ověřovat, ověřit). Od toho odvozené podstatné jméno je autentizace. Anglické authentication se do češtiny překládá jako ověření, legalizace, prokázání pravosti, důkaz pravosti. „Výraz autentifikace jsme našli několikrát zachycený v databázi Českého národního korpusu, např.: mechanismy autentifikace vlastních systémů www - v počítačové praxi se běžně užívá. Domníváme se, že z hlediska slovotvorného jsou v pořádku výrazy autentizace, autentizovat, autentifikace je rovněž přípustná - význam: ověření pravosti.“[2] Nejčastěji používaný výraz autentizace bude používán i v této práci.

Pod pojmem slova autentizace se skrývají dvě části. Tvrzení o své identitě a její potvrzení či ověření. Autentizace je proces, kde osoba nebo počítačový program prokazuje svoji identitu, aby získal přístup k informacím. Lidská identita je prosté tvrzení nebo například konkrétní přihlašovací ID pro některou aplikaci. Nejdůležitější částí autentizace je dokázání a důkazem může být zpravidla něco známého (například heslo), něco vlastněného (čipová karta) nebo něco unikátního ve vzhledu dané osoby (například otisk prstu).[1]

Jak již bylo zmíněno, ověření identity uživatele lze provést na základě tří skutečností:

- Něco, co daný uživatel zná.
- Něco, co daný uživatel má.
- Něco, čím daný uživatel je.

Dané přístupy lze kombinovat a v takovém případě se jedná o vícefaktorovou autentizaci.

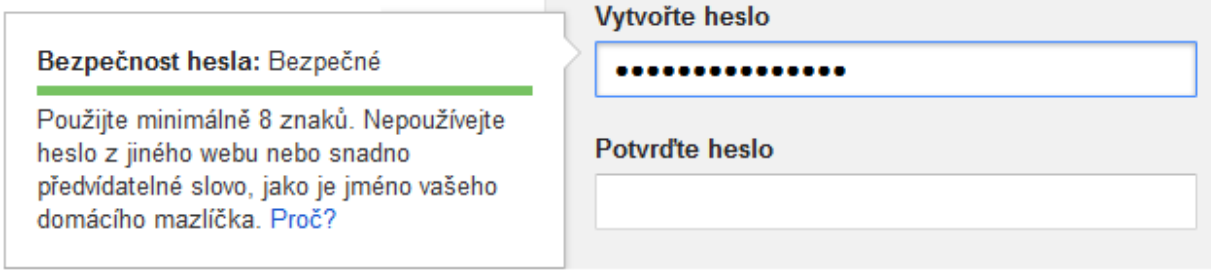
1.1. Něco zná

Autentizace založená na znalosti určitého faktu či skutečnosti (tajemství) je díky své nenáročnosti na zdroje nejrozšířenější formou autentizace. Tato metoda spoléhá na předpoklad, že daná znalost (heslo, PIN, passphrase, ...) je tajemstvím známým pouze autentizované osobě.

Výhodou znalostní autentizace je, že se v jejím případě nejedná o fyzický objekt, ale o abstraktní znalost, kterou lze snadno přenášet nebo zadávat do počítače. Systémy pro tuto metodu autentizace lze snadno ovládat a nevyžadují složitou údržbu. Nevýhodou však je, že tajná informace může být snadno zjištěna, a to dokonce bez vědomí uživatele. Navíc lidská paměť je s ohledem na zapamatování náhodných informací poměrně omezená (složitá hesla si lze jen velmi obtížně zapamatovat), což negativně ovlivňuje celkovou bezpečnost této autentizační metody.[19]

Nejčastější formou autentizace založené na znalosti uživatele jsou hesla a PINy. Pro zjednodušení lze uvést příklad, kdy v případě použití hesel je na straně uživatele (Alice) znalost hesla s identifikačními údaji, systém (Bob) má znalosti o identifikačních údajích a heslech všech svých uživatelů. Alice zasílá systému společně se svými identifikačními údaji i heslo, Bob podle identifikačních údajů vyhledá Alici a ověří, zda se hesla shodují. Jsou-li údaje shodné, systém Alici vyhodnotí jako oprávněného uživatele a Alice tedy úspěšně prošla autentizací. V reálném případě samozřejmě vstupuje do procesu autentizace šifrování. Samotné heslo bývá řetězec znaků dlouhý zpravidla 6-15 znaků, který je v ideálním případě odolný proti slovníkovému útoku (netriviální slova) a útoku hrubou silou (použití rozšířené abecedy, malé a velké znaky, číslice, speciální znaky). Heslo však musí zůstat uživatelem zapamatovatelné.

Běžní uživatelé si většinou nejsou vědomi nebezpečnosti, která se v heslu skrývá. „Dnešní systémy spravující hesla proto umožňují kontrolu bezpečnosti vkládaných hesel (včetně populárních indikátorů vhodnosti), příp. uživateli vygenerují heslo s požadovanými parametry. Požadavky kladené na tato hesla jsou pak součástí bezpečnostních politik systému. Stinnou stránkou tohoto přístupu ale je, že uživatel si heslo bude obtížněji pamatovat a často zapomínat.“[19] Příkladem budiž formulář pro tvorbu hesla na stránkách Google.cz z obrázku 1. Bezpečnost hesla odpovídá politikám daného provozovatele a ovlivňuje ji jak délka, tak složení použitých znaků.



Obrázek 1: Příklad tvorby hesla, zdroj [10]

Heslo „traktorista1“ sice splňuje zde požadovanou podmínku minimálně 8 znaků, ale je vyhodnoceno jako nedostatečně bezpečné, kdežto kratší heslo „bRt-8k_12A“ je vyhodnoceno jako bezpečné. Jako bezpečné heslo lze považovat takové heslo, jehož prolomení obvyklými technikami je časově velmi náročné. Typicky se jedná o řetězec s délkou 8-12 znaků, který obsahuje znaky z více různých abeced (velká a malá písmena, číslice, symboly) a zároveň není dostupný v běžných slovnících (a tedy je odolný slovníkovému útoku). Doporučovaným způsobem pro zvyšování bezpečnosti hesla je zvětšování základní množiny znaků před prodlužováním [19]. Zde se potvrzuje i předchozí příklad, kdy je kratší heslo s více druhy znaků vyhodnoceno jako bezpečnější.

PIN je druhou častou metodou znalostní autentizace. PIN bývá zpravidla kratší než heslo, protože jeho bezpečnost je založena na omezeném množství pokusů k jeho zadání. Pokud v daném počtu pokusů není PIN zadán správně, je zablokován a k jeho odblokování je potřeba využití složitějšího autentizačního mechanismu. Může se jednat například o výrazně delší PIN (zvaný PUK) u mobilních telefonů, či nutnosti osobního kontaktu s předložením identifikačních dokumentů (občanský průkaz) v případě zablokování platební karty. Potřebným předpokladem pro fungování tohoto mechanizmu je však nutnost fyzického vlastnictví autentizačního předmětu, díky tomu se jedná o takzvanou dvoufaktorovou autentizaci. Bez vlastnictví autentizačního předmětu totiž není možné PIN vůbec zadat. Autentizačním předmětem může být mobilní telefon, SIM (subscriber identity module [28]) karta, nebo kreditní karta.[19]

1.2. Něco má

Druhou metodou je autentizace založená na vlastnictví určitého předmětu, často nazývaného token. Tokeny jsou zařízení, která mohou uživatelé nosit neustále s sebou a jejichž vlastnictví je nutné pro to, aby se mohli autentizovat do systému. Tokeny mají buď specifické fyzické vlastnosti (elektrický odpor, tvar, hmotnost, elektrickou kapacitu, ...), nebo obsahují specifické tajné informace (heslo nebo kryptografický klíč), nebo jsou dokonce schopny provádět specifické výpočty.[19]

V současnosti nejčastěji používaný token je karta. Podle jejich vlastností existuje několik typů karet. Nejjednodušší jsou karty s magnetickým proužkem, který obsahuje určitou neměnnou informaci sloužící k autentizaci. Druhým a složitějším typem jsou čipové karty. Mezi čipové karty patří například SIM karty nebo platební karty.

Dalším častým typem tokenu je autentizační kalkulátor. Existují dva základní druhy autentizačních kalkulátorů. Samotné kalkulátory mohou být založeny buď na tajemství, které je současně uloženo v kalkulátoru a v autentizačním serveru, nebo na synchronizovaných hodinách. „Důležitou vlastností kalkulátorů je způsob komunikace s uživatelem - klasické komunikační rozhraní typicky zahrnuje pouze klávesnici a displej, speciální optická rozhraní či infračervený port umožňují navíc kalkulátoru komunikovat přímo s počítačem.“[19] Autentizační kalkulátory bývají zpravidla k vidění v bankovním sektoru jako součást zabezpečení platebních účtů. Příklad autentizačního kalkulátoru je na obrázku 2.

Obrázek 2: Autentizační kalkulátor, zdroj [28]

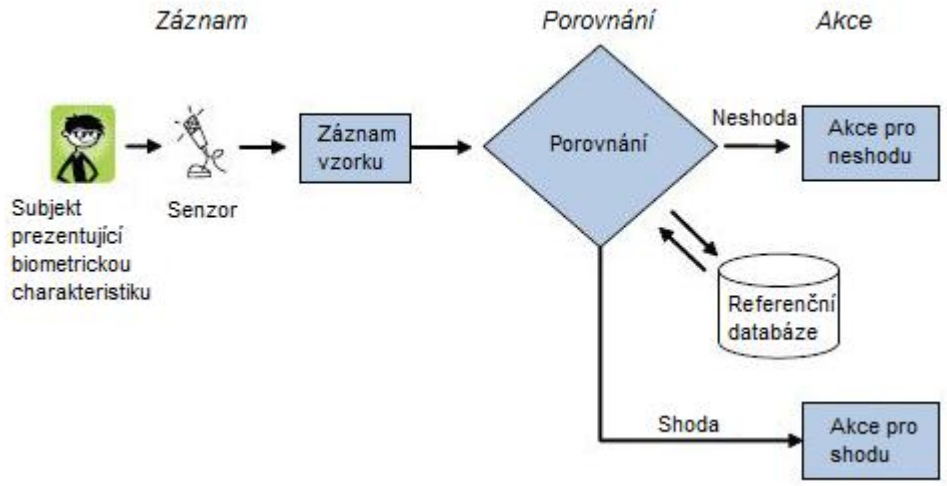
Za autentizační token může být považován i mobilní telefon s jehož pomocí je opsáním bezpečnostního kódu zaslaného prostřednictvím SMS (short message service [28]) potvrzována platební transakce na bankovním účtu či jiné podobné operace.

1.3. Něčím je

Třetí metodou je autentizace založená na fyziologických informacích nebo vlastnostech člověka. Tyto biologické ukazatele se nazývají biometriky. Výhodou biometrik je, že jsou vlastní dané osobě a nelze je zapomenout ani ztratit. Biometriky mají dvě základní využití. Jedním využitím je autentizace, kdy se subjekt identifikuje například pomocí ID či přihlašovacího hesla a biometrický údaj slouží k jeho ověření. Druhá možnost je samotná identifikace, kdy se pomocí biometrického údaje určuje, o kterou osobu z existující databáze těchto údajů se jedná. „Při identifikaci (nebo také vyhledání) člověk identitu sám nepředkládá, systém prochází všechny (relevantní) záznamy v databázi, aby našel patřičnou shodu a identitu člověka sám rozpoznal. Systém odpovídá na otázku: Kdo to je? Je zřejmé, že identifikace je podstatně náročnější proces než verifikace. Se zvyšujícím se rozsahem databáze se přesnost identifikace snižuje a rychlost klesá.“[19] V procesu identifikace tedy neslouží biometrický údaj k ověření, ale k určení o kterou osobu se jedná. „Podle základního principu identit každá osoba je identická jen a pouze sama se sebou. Jestliže vědecky prokážeme (a je prokázáno), že i naše fyzické (a psychické) charakteristiky jsou jedinečné,

pak je lze úspěšně použít pro efektivní identifikaci osoby s velmi vysokým stupněm jedinečnosti a tedy následně i bezpečnosti a prokazatelnosti. Identitu osoby je pak téměř nemožné absolutně napodobit nebo pozměnit. Nelze ji ani odcizit, protože identifikační charakteristiky jsou bezprostředně spojené s identifikovanou osobou. Biometrická identita je pro každého člověka navíc přirozená - vlastní; je s ním spojena již od narození.“[26] Biometrická autentizace i identifikace je tedy založena na využití těchto jedinečných a neopakovatelných fyzikálních (fyziologických) znaků.

Ať se jedná o identifikaci nebo autentizaci, uživatel vždy prezentuje některý ze svých biometrických údajů systému. Základní schéma takovéhoto systému je na obrázku 3.



Obrázek 3: Schéma biometrického systému, zdroj [23]

V tomto schématu vždy vystupuje osoba (subjekt), která předá pomocí snímače (senzor) do systému svůj biometrický údaj, vzor. Systém porovná tento vzorek (porovnání) s databází ostatních vzorů (referenční databáze) a rozhodne, zda se jedná o shodu (shoda) nebo nikoliv (neshoda). V případě autentizace je při shodě uděleno povolení a osoba je autentizována (akce pro shodu), není-li shoda nalezena, osoba není autentizována (akce pro neshodu). V případě nalezení shody při identifikaci je osoba označena jako osoba s pasujícím vzorem v databázi (akce pro shodu), není-li shoda nalezena, není osoba identifikována (akce pro neshodu).

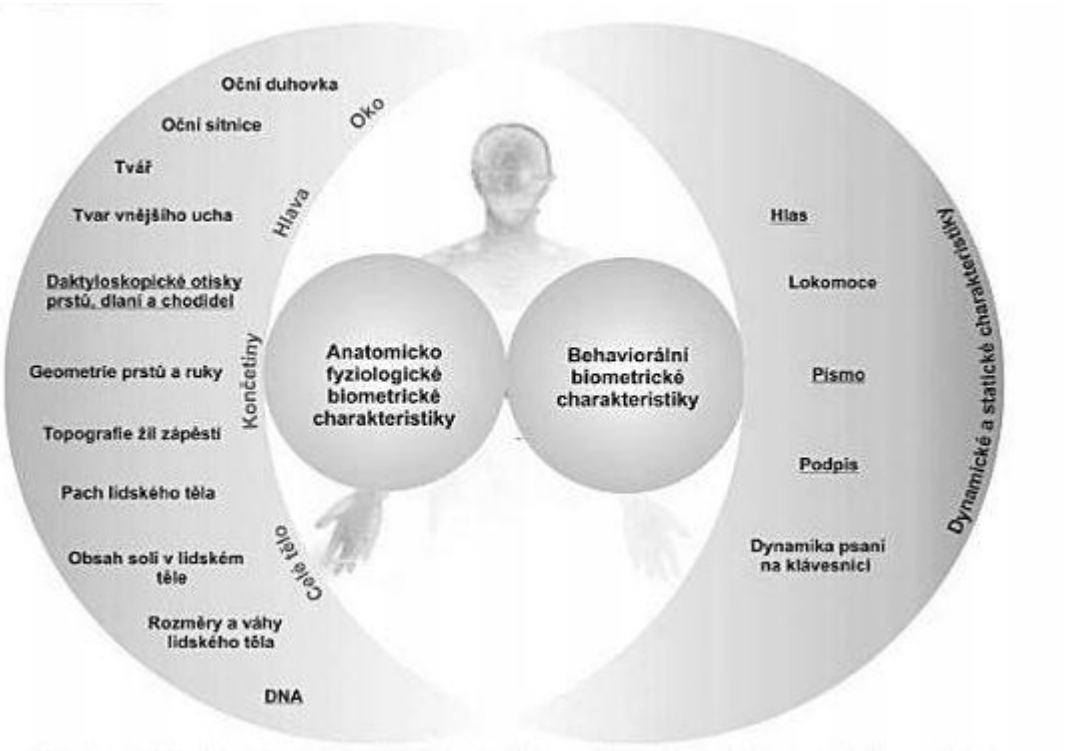
1.3.1. Biometrická autentizace

Biometrická autentizace se od autentizací založených na znalosti nebo vlastnictví nějakého předmětu liší v reprezentaci výsledků. Vlastnictví či znalost lze snadno ověřit a systém na jejich základě uživatele autentizuje či nikoliv. Z pohledu výstupu systému nezáleží zda se jedná o oprávněného uživatele nebo narušitele, který znalost či předmět získal

neoprávněně. Systému jsou prezentována fakta a na jejich základě vznikne rozhodnutí ano/ne. Největší rozdíl mezi biometrickými a tradičními technologiemi autentizace je odpověď systému na autentizační požadavek. Biometrické systémy nedávají jednoduché odpovědi typu ano/ne. „Heslo buďto je *abcd* nebo ne, magnetická karta s číslem účtu *1234* jednoduše je nebo není platná. Podpis člověka však není vždycky naprosto stejný, stejně tak pozice prstu při snímání otisku se může trochu lišit. Biometrický systém proto nemůže určit identitu člověka absolutně, ale místo toho řekne, že s určitou pravděpodobností (vyhovující autentizačním/identifikačním účelům) se jedná o daného jedince.“[19] Reprezentace výstupu biometrických systémů v podobě pravděpodobnosti existuje s přihlédnutím k uživatelům. „Mohli bychom samozřejmě vytvořit systém, který by vyžadoval pokaždé téměř 100% shodu biometrických charakteristik. Takový systém by však nebyl prakticky použitelný, neboť naprostá většina uživatelů by byla téměř vždy odmítnuta, protože výsledky měření by byly vždy alespoň trochu rozdílné. Abychom tedy udělali systém prakticky použitelný, musíme povolit určitou variabilitu biometrických charakteristik.“[19] Žádný biometrický systém není bezchybný, a proto je potřeba s určitou variabilitou počítat. Právě volba kdy ještě shodu povolit a kdy již zamítnout je zásadní. V příliš přísném systému bude mít i oprávněný uživatel problém s autentizací, naopak v příliš uvolněném nebude problém vstupu i pro neoprávněného narušitele.

Ani samotný fakt, že je v databázi nalezena biometrická charakteristika srovnatelná s právě sejmutou, ještě neznamená, že se jedná o přímo tutéž osobu. O biometrických ukazatelích se zpravidla mluví jako o jednoznačném ukazateli, ale tvrzení: „Jedinec je v čase sám sobě podobnější než kdokoliv jiný v jakémkoliv čase [23],“ nelze potvrdit, protože spousta (částečných) znaků je do značné míry podobná mezi lidmi navzájem a některé jiné charakteristiky se mění v čase. Množina všech ostatních lidí je navíc natolik veliká, že neexistuje možnost jak toto tvrzení potvrdit či vyvrátit. Proto bylo dané tvrzení upraveno na: „Jedinec je v čase sám sobě podobnější než kdokoliv, s kým se pravděpodobně může setkat [23].“ Záleží samozřejmě i na dané biometrické charakteristice. Dynamika podpisu se bude měnit pravděpodobně více než otisk prstu, stejně tak bude snadnější ji napadnout narušitelem.

Biometriky využívané v rámci biometrické autentizace lze rozdělit na dvě skupiny. První skupinou jsou anatomicko-fyziologické biometrické charakteristiky a druhou skupinou jsou behaviorální biometrické charakteristiky, rozdělení ilustruje obrázek 4.



Obrázek 4: Biometrické charakteristiky, zdroj [26]

1.3.2. Anatomicko-fyziologické biometrické charakteristiky

Anatomicko-fyziologické biometrické charakteristiky jsou založeny na měření fyziologických či anatomických vlastností osoby. Mezi tyto charakteristiky patří rozšířené snímání profilu tváře, oční duhovky, otisky prstů, geometrie dlaně nebo test skladby DNA (deoxyribonucleic acid [22]). Patří sem i další charakteristiky jako stavba vnějšího ucha, topografie žil zápěstí nebo obsah solí v lidském těle. Anatomicko-fyziologické biometrické charakteristiky jsou unikátní a časově stálé.[26]

Zřejmě nejčastěji využívanou anatomickou biometrikou je snímání otisků prstů. Otisk prstu reprezentuje vzor rýh, takzvaných papilárních linií, které jsou po celý život téměř neměnné. Tato vlastnost je známa a otisky prstů se využívají již více než sto let. Jeden z faktorů úspěšnosti rozpoznávání je, zda je jako vzorek použit otisk jednoho nebo více (případně dokonce všech deseti) prstů. Otisky více prstů logicky poskytují dodatečné informace, které mohou být velmi cenné v rozsáhlých systémech. „Naráží se však na problém, že tyto rozsáhlé systémy jsou velmi náročné na výpočetní výkon, zejména hledají-li shodu mezi miliony záznamů [23].“ Výhodou autentizačních metod pracujících na základě otisku prstu je jejich velké rozšíření, které technologii výrazně zlevňuje. Metoda je navíc zkoumána již velmi dlouho a tak jsou vyvinuty velmi spolehlivé klasifikační techniky a také kvalitní snímače

otisků. Nevýhodou některých snímačů je náchylnost na nečistoty na prstech nebo na porušení papilárních linií (způsobené například náročnou prací), které mnohdy autentizaci znemnožní.[3]

Další z často využívaných anatomických biometrik je snímání tváře. Snímání tváře může být statické nebo dynamické (video). Ať už statický nebo dynamický, snímek vždy slouží k rozpoznání tváře. „Moderní přístupy jsou pouze nepřímo založené na umístění, tvaru a prostorových vztazích částí obličeje jako jsou oči, nos, rty, brada, atd. Rozpoznávání může být velmi úspěšné, pokud je zachováno jednoduché pozadí a základní póza, může ho však narušit osvětlení nebo změna úhlu snímání. Čas mezi pořízením vzorku a rozpoznáváním také hraje roli, protože lidská tvář se v čase mění.“[23] Časová nestálost otisku tváře je netypická pro anatomicko-fyziologické charakteristiky

Snímání duhovky je další z častých anatomických biometrik. Duhovka, kruhová zbarvená membrána okolo oční čočky, je dostatečně komplexní k využití při rozpoznávání [23]. Duhovka na rozdíl od otisku prstů podléhá drobným změnám s přibývajícím věkem.

Geometrie dlaně je založená na tvaru ruky, velikosti dlaně, délce a šířce prstů. „Protože není jasné, jak rozdílná je geometrie dlaně v rámci velkých populací, takovéto systémy jsou zpravidla používány k ověřování spíše než k identifikaci.“[23] Většímu rozšíření využití geometrie dlaně, například u notebooků, brání oproti snímači otisku prstů vyšší náročnost na prostor.

Vzorek DNA lze izolovat ze všech částí lidského těla, tedy například z krve, slin nebo vlasů. V současné době se používá v kriminalistice pro jednoznačnou identifikaci pachatele [3]. Z hlediska autentizace v počítačových systémech není skladba DNA vhodná biometrika, neboť není možné získat vzor bezkontaktní metodou, stejně jako je nevyhovující přílišná náročnost testu.

1.3.3. Behaviorální biometrické charakteristiky

Behaviorální biometrické charakteristiky jsou založeny na využití poznatků o chování člověka. Tradiční behaviorální charakteristikou je podpis, hlasový profil, dynamika pohybu těla (lokomoce) nebo dynamika psaní na klávesnici. Tyto charakteristiky jsou unikátní, existuje u nich však možnost proměny v čase, jsou časově nestálé. Systémy založené na fyziologických vlastnostech (anatomicko-fyziologických biometrických charakteristikách) jsou obvykle spolehlivější a přesnější než systémy založené na chování člověka. Měření

fyziologických vlastností jsou totiž lépe opakovatelná a nejsou v tak velké míře ovlivněna daným psychickým a fyziologickým stavem jako je např. stres nebo nemoc.[19]

Podpis patří mezi typické bahaviorální charakteristiky. Způsob, jakým se člověk podepisuje, se mění s věkem i s podmínkami, ve kterých k podpisu dochází. Aktuální psychické rozpoložení má na podpis také významný vliv.

Hlasový profil se řadí k behaviorálním charakteristikám, přestože je ovlivněn biologickými vlastnostmi člověka. Zvuk, který člověk při mluvě vydává, je založen na fyzických aspektech těla (velikost a tvar úst, nosu, rtů, hlasivek, atd.) a může být ovlivněn věkem, emocemi, zdravotním stavem i dalšími faktory.[23]

Lokomoce, dynamika pohybu těla, je způsob, jakým se člověk pohybuje. „Metoda vychází z předpokladu, že dynamický stereotyp chůze je výrazně rozlišný pro různé osoby. Jsou vyhodnocovány trajektorie pohybu (lokomoční pohyby) přesně daných bodů (klouby dolních končetin, významné body na trupu, hlavě), a také jsou vyhodnocovány úhlové změny ve velkých kloubech dolních končetin. Vybrané zkoumané vlastnosti jsou vždy vybírány tak, aby se výrazně neměnily s věkem.“[3]

Další behaviorální charakteristikou je dynamika psaní na klávesnici. Tato metoda je založena na specifických způsobech psaní autentizované osoby na klávesnici. Většinou je zkoumána rychlost a délka stisku jednotlivých kláves, čas mezi stisky popřípadě histogram doby stisku jednotlivých kláves. Při analýze není důležité pouze to, jaké klávesy jsou tisknuty, ale hlavně je zkoumáno, jakým způsobem jsou tisknuty [3]. Tato metoda je do značné míry závislá na kontextu, v jakém je vzorek pořízen. Dynamika psaní na klávesnici je silně ovlivněna psychickým stavem, pozicí při psaní, typem klávesnice a podobně [23]. Lze rozlišovat dva způsoby autentizace prostřednictvím dynamiky psaní na klávesnici - statický a dynamický. Statická autentizace využívá kontroly dynamiky pouze v určitý moment (například při zadání hesla). Statický přístup poskytuje robustnější uživatelské ověření než pouhé heslo, ale neposkytuje žádné průběžné zabezpečení - není možné odhalit záměnu uživatele po úvodním ověření [21]. Dynamická autentizace monitoruje dynamiku psaní v průběhu celé interakce. Tato metoda je z hlediska reálné aplikace i výpočetně mnohem náročnější.

1.3.4. Pravděpodobnost chybného odmítnutí a chybného přijetí

Ať už se jedná o biometrickou identifikaci nebo autentizaci, má vždy bezpečnostní charakter. „Cílem je, aby oprávněné (autorizované) osobě byly bezchybně umožněny garantovaná práva

(například přístupu do objektu, k vykonávání specifikovaných činností atd.). Naopak osoba, jež tyto práva nemá, musí být stejně bezchybně rozpoznána a odmítnuta. Touto osobou může být v krajním případě i podvodník, vydávající se zcela vědomě za někoho jiného, a to z nejrůznějších důvodů. Zároveň je žádoucí, aby případná záměna identity s jinou osobou byla vyloučena.“[26] Pro každý systém tak musí existovat způsob, jakým lze vyjádřit jeho spolehlivost a kvalitu.

K vyjádření spolehlivosti biometrického systému existuje několik různých metrik. Dvě nejčastěji používané, budou použity i v této práci, jsou pravděpodobnost chybného odmítnutí a pravděpodobnost chybného přijetí. „Pravděpodobnost chybného přijetí nebo odmítnutí biometrických metod nelze teoreticky vypočítat. Biometrické metody identifikace/verifikace jsou založeny na statistickém vyhodnocování podobnosti biometrického vzoru a biometrické šablony. Při každém snímání biometrického vzoru nejsou zaznamenávány absolutně stejné hodnoty, stejné markanty pořizovaných charakteristik. V důsledku se pak i obě porovnávané šablony nepatrně liší. Míra ztotožnění (nazývána rovněž výsledek porovnání, tzv. skóre) je pak pokaždé odlišná a závisí především na každé biometrické aplikaci a jejím technickým řešením.“[26] Systém takto vypočítá skóre pro každého autentizovaného uživatele. Podle [26] můžeme označit vstup autentizovaného vzoru do systému P, referenční šablonu biometrického vzoru P'. Míra ztotožnění referenční šablony P' a načteného vzoru P je popsána rovnicí 1:

$$s = Sim(P, P') \tag{1}$$

kde:

s - míra ztotožnění

Sim - vztah ztotožnění

P - vstupní vzor

P' - referenční šablona

Aplikace má nastavený určitý práh citlivosti Th. O tom, zda je uživatel oprávněn nebo neoprávněn, rozhoduje míra ztotožnění vzorů v závislosti na nastaveném prahu citlivosti. Jestliže míra ztotožnění je vyšší než daná prahová hodnota, pak P je ztotožněno s P' a osoba, která je majitelem vzoru P, je vyhodnocena jako osoba správná, tedy úspěšně autentizovaná. Její vzor tak musí být dostatečně podobný stávajícímu referenčnímu vzoru v databázi. Naopak je-li míra ztotožnění pod prahovou hodnotou, pak P nelze ztotožnit s P' a osoba, která žádá o autentizaci je zamítnuta. Vzor se příliš liší od referenční hodnoty.

Takto mohou vzniknout čtyři situace: oprávněný uživatel se úspěšně autentizuje, oprávněný uživatel je odmítnut, narušitel je odmítnut, narušitel je chybně autentizován. Vše je v pořádku v případě, že se oprávněný uživatel autentizuje nebo v případě, kdy je narušitel odmítnut. Ve zbylých dvou případech se využívá pravděpodobnost chybného odmítnutí a pravděpodobnost chybného přijetí.

Pravděpodobnost chybného odmítnutí, v anglické literatuře značená False rejection rate (FRR), se též nazývá chyba 1. druhu. Jedná se o pravděpodobnost, že systém nesprávně odmítne autentizovat oprávněného uživatele. Jinými slovy jeho biometrický vzor nesprávně vyhodnotí jako odlišný od vzoru uloženého v databázi. Vztah popisuje rovnice 2:

$$FRR = \frac{N_{FR}}{N_{EAA}} \tag{2}$$

kde:

FRR - chyba typu I

N_{FR} - počet chybných odmítnutí

N_{EAA} - počet všech pokusů oprávněných uživatelů o autentizaci

Chybné odmítnutí je nežádoucí z pohledu uživatelské přívětivosti biometrické metody a spolehlivosti daného zařízení. V praxi není žádoucí, aby existoval velký počet oprávněných osob, které jsou biometrickým zařízením odmítnuty. V důsledku toho pak klesá důvěra v dané zařízení, jenž je přehnaně tvrdé i k osobám, které ve skutečnosti mají garantovaný přístup. Při chybném odmítnutí oprávněného uživatele nemůže dojít například k úniku dat, přesto se jedná o narušení dostupnosti, a tedy se jedná o bezpečnostní riziko. FRR se může zvýšit kvůli podmínkám v prostředí nebo díky nesprávnému použití, například přiložení špinavého prstu ke snímači otisků. FRR se většinou snižuje, když uživatelé získají zkušenosti s biometrickým zařízením.[24],[26]

Pravděpodobnost chybného přijetí, anglicky False acceptance rate (FAR), je označovaná jako chyba 2. druhu. Tato chyba vyjadřuje pravděpodobnost, že systém chybně autentizuje (přijme) neoprávněného narušitele. Tedy že vzor narušitele bude chybně označen za dostatečně blízký vzoru oprávněného uživatele uloženého v databázi. Tato pravděpodobnost lze vyjádřit zápisem uvedeným v rovnici 3:

$$FAR = \frac{N_{FA}}{N_{IAA}} \quad (3)$$

kde:

FAR - chyba typu II

N_{FA} - počet chybných přijetí

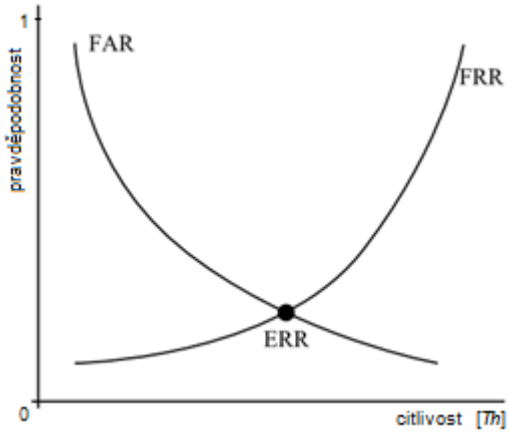
N_{IAA} - počet všech pokusů neoprávněných narušitelů o autentizaci

Druhou stranou mince je požadavek na vysokou bezpečnost biometrického zařízení, které nesmí akceptovat neoprávněné osoby (například ke vstupu do libovolného objektu) [26]. Každý narušitel, který je chybně autentizován jako oprávněný uživatel, představuje pro systém riziko.

Zdroj [9] uvádí čtyři typy uživatelů z hlediska autentizace:

- Ovce (Sheep) - uživatel, který je snadno rozpoznatelný (větší množství uživatelů typu ovce přispívá k nižší hodnotě FRR)
- Jehně (Lamb) - uživatel, kterého je snadné napodobit (takovýto uživatel přispívá ke zvyšování FAR)
- Koza (Goat) - uživatel, který je špatně rozpoznatelný (zvyšuje FRR)
- Vlk (Wolf) - uživatel se schopností napodobit biometrické charakteristiky ostatních uživatelů (díky tomu přispívá ke zvyšování FAR)

Pomocí hodnot FAR a FRR lze hodnotit spolehlivost autentizačního systému a také jednotlivé systémy vzájemně porovnávat. Dle nastavení různých parametrů konkrétní autentizační metody se hodnoty FAR a FRR mění, ale vždy platí, že zmenšení FAR vyvolá zvětšení FRR a naopak. Stav, kdy jsou hodnoty FAR a FRR shodné (průsečík jejich křivek), se nazývá equal error rate (EER), shodná míra chybovosti.[3] Příklad křivek FAR a FRR spolu s hodnotou EER je na obrázku 5.



Obrázek 5: FRR a FAR, zdroj [3]

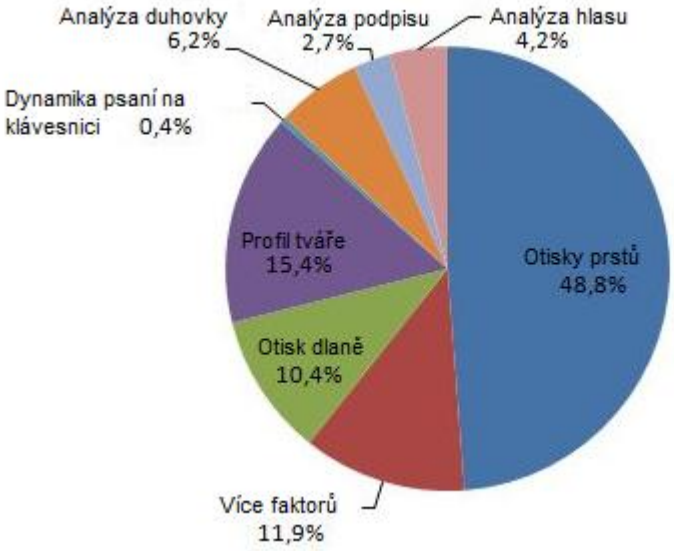
Na vertikální ose je vyznačena pravděpodobnost chybných přijetí a chybných odmítnutí, horizontální osa znázorňuje citlivost, tedy prahovou hodnotu *Th*. Je-li prahová hodnota nastavena příliš vysoko, bude i u oprávněných uživatelů často docházet k zamítnutí - vyšší FRR za cenu nižšího FAR. Naopak je-li prahová hodnota příliš nízká, bude docházet ke ztotožnění vzorů i pro neoprávněné narušitele, výsledkem bude vysoká FAR a nízká FRR.

Hodnota EER se často uvádí jako ideální kompromis mezi FRR a FAR. Vždy ale záleží na konkrétní aplikaci a nastavení systému. Jiné zabezpečení se očekává například od internetového bankovnictví a jiné od přístupu do internetového fóra.

V ideálním případě se FRR a FAR budou stýkat na nulové hodnotě osy y (EER = 0), nicméně tato situace je v praxi nedosažitelná.

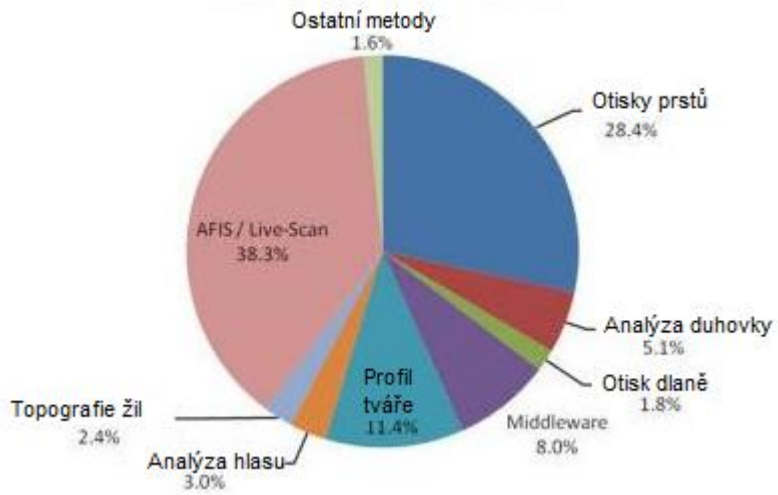
2. Autentizace s využitím dynamiky psaní na klávesnici

Podle [18] nepatří autentizace s využitím dynamiky psaní na klávesnici mezi příliš rozšířené biometrické metody. Podíl na trhu (v roce 2003) znázorňuje obrázek 6. Systémy založené na dynamice psaní na klávesnici pokrývají 0,4 % trhu.



Obrázek 6: Podíl biometrických technologií na trhu 2003, zdroj [18]

I podle dalšího, aktuálnějšího (2009), zdroje [7] patří autentizace pomocí dynamiky psaní na klávesnici mezi okrajové biometriky. V tomto případě je zahrnuta mezi ostatní metody, které mají úhrnnou hodnotu 1,6 % trhu. Samotná dynamika psaní na klávesnici bude mít ještě menší podíl. Podíly jednotlivých technologií na trhu znázorňuje obrázek 7.

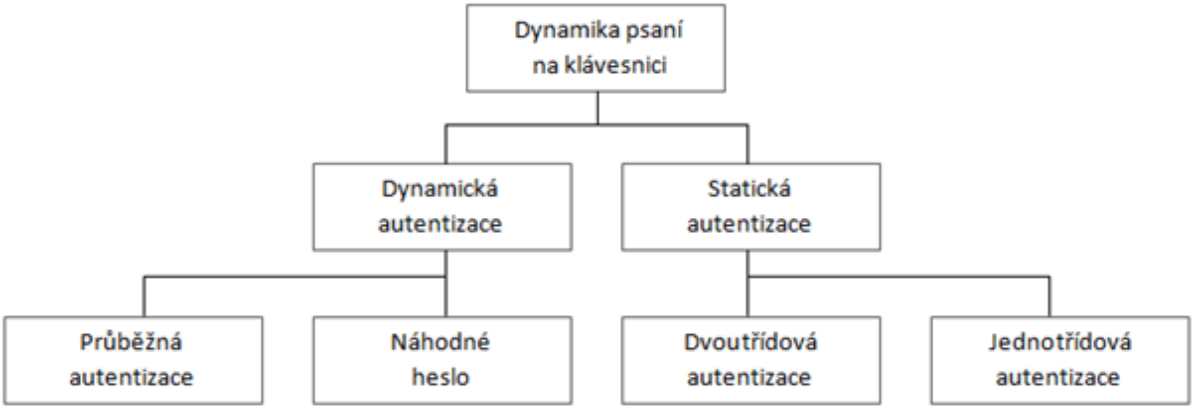


Obrázek 7: Podíl biometrických technologií na trhu 2009, zdroj [7]

Přestože se v případě dynamiky psaní na klávesnici nejedná o příliš rozšířenou biometrickou metodu, i na tomto poli dochází k neustálému vyvíjení a zlepšování metod.

2.1. Statický a dynamický přístup

Jak již bylo v práci dříve zmíněno, autentizaci prostřednictvím dynamiky psaní na klávesnici lze dělit na statickou a dynamickou. Statickou autentizaci je dále možno rozdělit na jednotřídovou autentizaci a dvoutřídovou autentizaci. Dynamická autentizace se pak dělí na průběžnou a autentizaci založenou na náhodném heslu. Rozdělení ilustruje obrázek 8.



Obrázek 8: Dynamická a statická autentizace, zdroj [9]

2.1.1. Statická autentizace

V případě statické autentizace je po uživateli vždy nejprve požadováno zadání stejného vzoru - řetězce znaků - několikrát po sobě. Z takto získaných dat je pro daného uživatele vytvořen záznam v referenční databázi. V průběhu fáze autentizace je pak po uživateli vyžadováno opětovné zadání stejného řetězce. O tom, zda je dynamika obdobná se rozhoduje na základě porovnání autentizovaného vzoru a vzoru referenčního. Takovýmto řetězcem znaků bývá nejčastěji uživatelské heslo, což znamená, že v případě změny hesla je před možnou autentizací potřebné opakovat i fázi vytvoření nového referenčního záznamu.

Rozdíl mezi jednotřídovou a dvoutřídovou autentizací je v použitém uživatelském heslu. V případě jednotřídové autentizace má každý z uživatelů vlastní heslo a metody rozpoznání vzorů, které jsou zde použity, využívají klasifikátory nebo vzdálenostní metriky pouze pro danou třídu. Systém tedy porovnává vzor s referenční databází pouze pro jednoho daného uživatele a dané heslo. V případě dvou třídové autentizace sdílejí všichni uživatelé stejné heslo a je potřeba řešit dvou třídový problém (vzory pro uživatele i narušitele). Tyto systémy

mohou fungovat i v případě, že ne všichni narušitelé byli přítomní ve fázi získávání dat. Díky stejnému heslu pro všechny uživatele je zde zapotřebí ve fázi přípravy systému rozlišit, které vzory patří uživateli a které narušiteli a s touto znalostí musí počítat i následný rozpoznávací mechanismus.[9]

Většina komerčních softwarových aplikací je založena na statické autentizaci úpravou přihlašovací procedury. Autentizační formulář je upraven k zachycení časových informací hesla a kromě ověření samotného hesla je ověřován i způsob, jakým bylo napsáno. Souhlasí-li s uživatelovým profilem, je autentizován. V opačném případě je odmítnut a vyhodnocen jako narušitel. Takovéto systémy využívají vlastně dvou faktorové autentizace. Prvním faktorem je něco, co uživatel zná - ve většině případů heslo. Druhým faktorem je něco, čím uživatel je, tedy jeho rytmus psaní na klávesnici. Vzhledem k tomu, že správné dodržování zásad o heslech (pravidelná změna, využívání komplexních hesel, nezapisování „na lísteček“, apod.) je často příliš restriktivní a málokdy dodržované. Právě zde je zajímavé využití dynamiky psaní na klávesnici., jež dokáže vyloučit narušitele, který byl schopen získat heslo k autentizaci namísto oprávněného uživatele. Pro zlepšení efektu je možné využít monitorování dynamiky psaní nejen pro heslo, ale i pro jméno uživatele, případně využít výrazně delší heslo v podobě passphrase.[9]

2.1.2. Dynamická autentizace

Metody dynamické autentizace umožňují autentizovat uživatele nezávisle na tom, co uživatel na klávesnici píše. Na rozdíl od statických metod tedy nejsou omezeny zadáním předem určené a naučené posloupnosti znaků. Dynamické metody autentizace zpravidla vyžadují, aby uživatel poskytl velké množství psaných dat k vytvoření vlastního modelu (buď přímým dotazem k vepsání dostatečně dlouhého textu nebo nepřímo monitorováním jeho počítače během určeného časového úseku). Průběžná autentizace, jak už název napovídá, poté sleduje chování uživatele (jeho dynamiku psaní) po celý průběh, kdy s daným počítačem pracuje. V modelovém případě by u statické autentizace mohl být oprávněný uživatel po zadání a ověření hesla nahrazen narušitelem, v případě dynamické průběžné autentizace by však byl tento narušitel odhalen. Počítač je v takové situaci schopný ukončit sezení, jestliže zjistí, že aktuální uživatel je rozdílný od uživatele dříve autentizovaného. Takovýto monitoring může být využit i k analýze chování uživatele (namísto vyhodnocení identity) a může detekovat nenormální aktivitu při přístupu k vysoce chráněným dokumentům nebo při vykonávání úkolů v prostředí, kde musí být uživatel vždy v pozoru.[9]

Druhý přístup dynamické autentizace je založen na zadání náhodně vygenerovaného hesla, fráze, věty, atp. Jestliže je možné modelovat chování uživatele, ať už píše cokoliv, je také možné ho autentizovat pomocí výzvy během běžného přihlašovacího procesu. Uživateli je zadán požadavek vepsání náhodné fráze nebo sdíleného tajemství (jako je například jednorázové heslo).[9] Tento přístup pracuje s celkovou dynamikou psaní uživatele, avšak neumožňuje odhalení záměny uživatele během sezení (to umožní pouze v případě průběžně opakované autentizační žádosti).

2.2. Způsob sběru dat

Fáze sběru dat je v případě biometrické autentizace považována za velice důležitý proces. Sběr dat probíhá ve dvou rozdílných momentech - při registraci a při autentizaci.

V průběhu registrace dochází ke sběru více vstupních vzorů každého uživatele a vytvoření jeho referenční báze. V závislosti na typu systému dynamiky psaní na klávesnici může být registrační fáze značně rozdílná (opakované zapisování stejného řetězce znaků, monitorování využití počítače, ...) a množství požadovaných dat se může výrazně lišit od pěti vstupů po více než sto [9]. Záleží tedy na nastavení a požadavcích systému, co vše a jakým způsobem bude ukládat.

Ve fázi ověření zadává uživatel jeden vzor, pro který jsou zaznamenávány stejné charakteristiky jako v případě registrace. Takto získaná hodnota je následně porovnána s modelem z referenční báze.

2.2.1. Klávesnice

Stejně jako kterákoliv jiná biometrická metoda, i metoda využívající dynamiku psaní na klávesnici vyžaduje určitý hardware, který je schopen zaznamenat požadovaná data. V případě této metody je dostačujícím hardwarem klávesnice, která je přítomná u každého počítače, ať již stolního nebo notebooku.

Ne všechny klávesnice jsou však stejné. Je potřeba rozlišit několik věcí, které mohou výrazně ovlivnit dynamiku psaní uživatelů:

- tvar klávesnice,
- odpor kláves,
- rozložení kláves.

Tvar klávesnic se může výrazně lišit. Existují klávesnice standardní rovné, mírně prohnuté, velmi prohnuté, či ergonomické. Porovnání několika rozdílných tvarů externích klávesnic ukazuje obrázek 9.

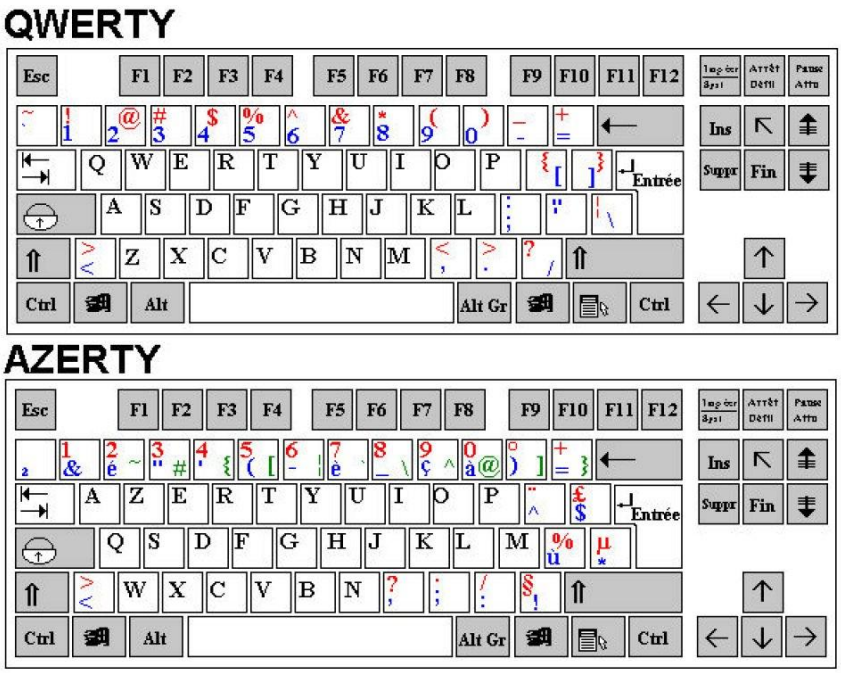
Obrázek 9: Různé tvary klávesnic, zdroj [17]

Je logické, že dynamika psaní se změní uživateli postavenému před klávesnici s výrazně odlišným tvarem, než na který byl doposud zvyklý.

Stejně tak rozdílný odpor stisku kláves představuje pro uživatele značnou změnu a může ovlivnit jeho dynamiku psaní. Tento rozdíl je zpravidla patrný při přechodu mezi tradiční stolní klávesnicí a klávesnicí laptopu, která má nízký zdvih a klade menší odpor při stisku kláves. Rozdíl mezi klávesnicí laptopu s nízkým zdvihem a klasickou stolní klávesnicí s výrazným zdvihem ilustruje obrázek 10.

Obrázek 10: Klávesnice laptopu a stolní klávesnice, zdroj [4]

Dalším faktorem výrazně ovlivňujícím dynamiku psaní je v neposlední řadě rozložení kláves. V zemích používajících latinku existují tři základní rozložení. Označují se šesticí písmen vpravo od klávesy Tab - QWERTY, QWERTZ a AZERTY. Na obrázku 11 je porovnání rozložení kláves QWERTY a AZERTY.



Obrázek 11: Rozdílné rozložení kláves, zdroj [20]

Nejrozšířenější rozložení je QWERTY, které se používá ve většině světa. Rozložení QWERTZ se používá převážně v Německu a okolních státech (střední Evropa). Třetí z nejčastěji používaných rozložení - AZERTY - se používá například ve Francii, Irsku nebo Belgii.

Existují i další způsoby rozložení kláves (například Dvorak, Colemak, Neo, BÉPO, atd.), které však mají většinou specifické využití a jsou rozšířeny spíše lokálně.

2.2.2. Měřené údaje

Samotné vlastnictví klávesnice však ke sběru dat nestačí a je potřeba ho rozšířit o mechanismus k měření času stisku jednotlivých kláves.

Události na klávesnici jsou způsobené uživatelem a jeho interakcí s klávesami. Surová biometrická data pro dynamiku psaní na klávesnici jsou prostým chronologickým seznamem těchto událostí. Seznam začíná prázdný, stene-li se nějaká událost, je doplněna na konec seznamu [9]. Ukládají se tři druhy informací: událost (event), kód klávesy (key code) a časová značka (time stamp).

Událost znamená nějakou akci klávesy, přičemž jsou dvě různé události:

- press - stisknutí klávesy,
- release - uvolnění klávesy.

Časová značka zaznamenává čas, ve kterém došlo k dané události.

Kód klávesy značí, ze které klávesy přichází událost. Z tohoto kódu lze získat znak klávesy (například za účelem ověření, zda řetězec znaků odpovídá heslu). Kód klávesy je zajímavější údaj, neboť poskytuje určitou informaci o umístění klávesy na klávesnici (což může být využito některými dalšími metodami) a dovoluje rozlišit různé klávesy píšící stejný znak. Tento kód může být závislý na platformě a použitém jazyku.[9]

Příklad surových dat sebraných dle výše popsaného postupu je v tabulce 1. Událost značená P znamená stisk klávesy (Press), událost značená R je uvolnění klávesy (Release). Kód klávesy označuje o kterou klávesu se jedná. Tabulka zachycuje modelový případ surových dat pro dva uživatele, kteří zapisují dvě písmena „TE“.

Tabulka 1: Příklad surových dat, zdroj vlastní

ID	událost	kód klávesy	časová značka	událost	kód klávesy	časová značka	událost	kód klávesy	časová značka	událost	kód klávesy	časová značka
u001	P	T	0:01	R	T	0:03	P	E	0:04	R	E	0:07
u002	P	T	0:01	P	E	0:02	R	T	0:04	R	E	0:06

Z takto zaznamenaných surových dat lze úpravami získat další znaky. Nejčastěji používané znaky vznikají odečtením časových hodnot. Následující vzorce převzaty z [9].

Délka stisku (Duration, Hold) - určuje ji doba, po kterou je klávesa stisknuta. Lze zapsat rovnicí 4:

$$H = time_i\{event_i = Release\} - time_i\{event_i = Press\} \quad (4)$$

kde:

H - délka stisku klávesy

$time_i$ - časová značka i-té události

$event_i$ - značí příslušnou akci klávesy i-té události

Prodleva (Latency) - určuje ji doba mezi jednotlivými stisky kláves, existuje několik druhů prodlev. Prodlevu mezi stisknutím jedné a stisknutím následující klávesy označíme PP (Press - Press), případně lze značit i DD (key_Down - key_Down). Příklad značení PP ilustruje rovnice 5:

$$PP = time_{i+1}\{event_{i+1} = Press\} - time_i\{event_i = Press\} \quad (5)$$

kde:

PP - prodleva mezi stisky dvou kláves

$time_i$ - časová značka i-té události

$time_{i+1}$ - časová značka události s pořadovým číslem i+1

$event_i$ - značí příslušnou akci klávesy i-té události

Prodlevu mezi uvolněním první klávesy a uvolněním klávesy následující označíme RR (Release - Release), případně UU (key_Up - key_Up). Příklad značení RR zobrazuje následující rovnice:

$$RR = time_{i+1}\{event_{i+1} = Release\} - time_i\{event_i = Release\} \quad (6)$$

kde:

RR - prodleva mezi uvolněními dvou kláves

$time_i$ - časová značka i-té události

$time_{i+1}$ - časová značka události s pořadovým číslem i+1

$event_i$ - značí příslušnou akci klávesy i-té události

Prodlevu mezi uvolněním první a stiskem následující klávesy označíme RP (Release - Press), jinak UD (key_Up - key_Down). Následující rovnice označuje tuto prodlevu:

$$RP = time_{i+1}\{event_{i+1} = Press\} - time_i\{event_i = Release\} \quad (7)$$

kde:

RP - prodleva mezi uvolněními jedné a stiskem druhé klávesy

$time_i$ - časová značka i-té události

$time_{i+1}$ - časová značka události s pořadovým číslem i+1

$event_i$ - značí příslušnou akci klávesy i-té události

Další znaky, které je možné ze surových dat získat, mají spíše globální charakter. Může to být například celkový čas potřebný k napsání daného textu (řetězce znaků, hesla) nebo střední čas značící rozdíl mezi stiskem úvodního a stiskem středního znaku z řetězce.

Jinou metodou naložení se surovými daty je možnost vytvořit digrafy. Digraf představuje čas potřebný pro stisk dvou kláves, tedy čas od stisku první klávesy k uvolnění druhé klávesy. Tato metoda vychází z konceptu, že některé kombinace jsou těžší na zapsání než jiné. Obtížnost zápisu je založena na vzdálenosti (na klávesnici) mezi dvěma následujícími znaky a faktu, že některé znaky vyžadují současný stisk více kláves (například použití klávesy Shift) [9]. Digraf může být vyjádřen rovnicí:

$$Di = time_{i+1}\{event_{i+1} = Release\} - time_i\{event_i = Press\} \quad (8)$$

kde:

Di - čas potřebný pro stisk dvou kláves

$time_i$ - časová značka i-té události

$time_{i+1}$ - časová značka události s pořadovým číslem i+1

$event_i$ - značí příslušnou akci klávesy i-té události

Digraf je možné rozšířit na trigraf až n-graf, kdy je zkoumán čas potřebný pro stisk tří nebo více (n) kláves.

Příklad vlastností získaných výpočtem ze surových dat je v tabulce 2. Hodnoty H.T a H.E označují délku stisku klávesy H a klávesy E. Hodnota PP.T.E je prodleva mezi stiskem

klávesy H a stiskem klávesy E, zatímco RP.T.E je prodleva mezi uvolněním klávesy T a stiskem klávesy E.

Tabulka 2: Vlastnosti získané ze surových dat, zdroj vlastní

ID	H.T	PP.T.E	RP.T.E	H.E
u001	0,033	0,050	0,017	0,050
u002	0,050	0,017	-0,033	0,067

Hodnota RP.T.E je u druhého uživatele záporná, protože ke stisku následující klávesy (E) došlo ještě před uvolněním klávesy první (T). Všechny vypočítané vlastnosti pro každého uživatele je možno vyjádřit jako vektor hodnot odpovídající danému uživateli. Hodnoty jednotlivých vlastností odpovídají složkám vektoru. Pro prvního uživatele by tak tento vektor vypadal následovně:

$$u_1 = (0{,}033;\ 0{,}050;\ 0{,}017;\ 0{,}050)$$

Dynamika psaní na klávesnici funguje se standardní klávesnicí a počítačem a není třeba pořizovat další specifické zařízení. Existují však i jiné měřitelné údaje, o které je možno (s patřičným vybavením) rozšířit zachycované informace. Podle [6] je například možné využít klávesnici rozšířenou o snímač tlaku (biokeyboard). Toto zařízení zaznamenává spolu se stiskem klávesy i tlak, který je při jejím stisku vyvíjen uživatelem. Většina algoritmů dynamiky psaní na klávesnici využívá pouze časovou informaci (latenci) o stisku kláves a zanedbává sílu aplikovanou při stisku. Pressure-Based Typing Biometric úspěšně schraňuje obě informace, časovou i aplikovanou sílu. Aplikace síly v čase znamená významnou informaci ve formě signálu (tlakový vzor). Tento přístup je dynamičtější a charakterističtější pro uživatele.

Další možností jak rozšířit množinu snímaných informací je využití senzoru náhodných pohybů (sudden motion sensor). Takový (nebo podobný) senzor bývá často přítomný v moderních noteboocích a bývá využit k detekci náhlých pohybů počítače, aby pomohl ochránit pevný disk v případě hrozícího nebezpečí. Snímaný pohyb ve vertikální ose může být použit jako doplňující informace.[9]

Je možné využít i zvukový signál produkovaný stiskem kláves. Analýzou záznamu zvukového signálu produkovaného klávesnicí je možné nepřímo získat dobu stisku klávesy,

dobu uvolnění klávesy a sílu stisku [9]. Takto získané informace je možné využít samostatně nebo v kombinaci s časovými údaji získanými běžnou metodou.

2.3. Metody vyhodnocení

Jakmile jsou získána data pro každého uživatele (z registrační i autentizační fáze), je potřeba vyhodnotit, zda se jedná o oprávněného uživatele nebo neoprávněného narušitele. Model dynamiky psaní pro každého uživatele je vytvořen, jakmile byla obdržena biometrická data ve fázi registrace. Způsob zpracování závisí na použité ověřovací metodě. Během autentizace tato ověřovací metoda porovnává uživatelův vzorek (biometrická data zachycená při autentizačním pokusu) s referenčním modelem. Na základě výsledku tohoto porovnání (kterým bývá nejčastěji míra nepodobnosti) je rozhodnuto o přijetí či zamítnutí uživatele.[9] Vyhodnocení je závislé na hodnotách poskytnutých v registrační fázi stejně tak jako na hodnotách dodaných v autentizační fázi. Fáze registrace by měla ke správnému vytvoření modelu zaznamenat minimálně 20 vzorků [9]. V případě rozsáhlejšího sběru dat se může pro uživatele jednat o činnost zdlouhavou a nepřívětivou, je však možné ji rozdělit do několika sezení, čímž se báze získaných dat posílí nejen v rozsahu, ale mohou se takto odstínit i negativní vlivy emocí, či vyčerpání uživatele.

2.3.1. Euklidovská metrika

V případě využití euklidovské metriky se každé heslo modeluje jako bod v n-rozměrném prostoru, kde n je počet vlastností každého vektoru. V registrační fázi je vypočítána střední hodnota z časových vektorů - referenční šablona. V autentizační fázi je vypočtena anomální hodnota jako čtverec euklidovské vzdálenosti mezi testovaným vektorem a vektorem střední hodnoty referenční množiny [15]. Vzdálenost mezi vektorem autentizačního pokusu P a vektorem P', který reprezentuje střední hodnotu z dat referenční množiny, lze značit dle následující rovnice:

$$D_{Euklid}(P, P') = \sqrt{\sum_{i=1}^{n} (P_i - P'_i)^2} \qquad (9)$$

kde:

D_{Euklid} - euklidovská vzdálenost

P_i - i-tá složka vektoru autentizačního pokusu

$P'i$ - i-tá složka vektoru středních hodnot z referenčních dat

P - vektor autentizačního pokusu

P' - vektor referenční šablony

Vychází se z předpokladu, že vzdálenost bude nižší pro oprávněného uživatele než pro neoprávněného narušitele. Vzdálenost se tak poměřuje vzhledem k určité prahové hodnotě *Th*. Změnou prahové hodnoty se změní i množství autentizovaných uživatelů (případně narušitelů). Na následujících řádcích bude uvedeno vyhodnocení několika prací využívajících euklidovskou metriku, omezíme-li se, pro zjednodušení, na vyhodnocení výstupu v podobě střední míry chybovosti (EER), tedy takový výstup, kdy jsou v rovnováze chyby 1. a 2. druhu (FAR a FRR).

Zdroj [15] uvádí EER při použití euklidovské metriky 0,171. Délka hesla v tomto případě je 10 znaků, což po extrakci surových dat dává 31 vlastností dostupných pro každého uživatele, tedy dimenze vektorů n = 31. Počet uživatelů v dané práci je 51, přičemž každý opakoval zápis 400×. Prvních dvě stě opakování bylo použito v registrační fázi k vytvoření vektoru P' (referenční šablony). Zbývajících dvě stě vektorů bylo použito k autentizaci daného uživatele. K autentizaci v podobě narušitele bylo použito vždy prvních pět pokusů každého uživatele s cílem autentizovat se za někoho jiného.

Další práce [27] využívá heslo o délce 10 znaků a v tomto případě bylo ze surových dat extrahováno 21 vlastností, dimenze n = 21. Stejně tak uživatelů, kteří se na výzkumu podíleli, bylo 21. V tomto případě ne každý účastník poskytl shodné množství dat. Průměrný počet opakování na účastníka byl 95. Vektor referenční šablony byl vytvořen vždy z prvních dvanácti hodnot. Zbylé hodnoty byly použity k autentizaci. V takto nastaveném pokusu bylo dosaženo EER = 0,43. Pro zkvalitnění výstupu byla použitá metrika upravena o zahrnutí směrodatné odchylky σ_i jednotlivých vlastností podle následujícího vztahu:

$$D_{Euklid/\sigma}(P, P') = \sqrt{\sum_{i=1}^{n} \frac{(P_i - P'_i)^2}{\sigma_i}} \qquad (10)$$

kde:

$D_{Euklid/\sigma}$ - rozšířená euklidovská vzdálenost

P_i - i-tá složka vektoru autentizačního pokusu

P'_i - i-tá složka vektoru středních hodnot z natrénovaných dat

P - vektor autentizačního pokusu

P' - vektor referenční šablony

σ_i - směrodatná odchylka i-té složky vektoru z natrénovaných dat

Využitím podílu směrodatné odchylky došlo ke snížení střední chyby EER na 0,336. Touto úpravou se metoda přiblížila Mahalanobisově metrice.

2.3.2. Manhattanská metrika

Postup vyhodnocení je v případě Manhattanské metriky, někdy též nazývané city-block, obdobný jako v případě využití euklidovské vzdálenosti. Stejně jako v prvním případě je v registrační fázi vytvořen střední vektor reprezentující dynamiku uživatele a následně je porovnán s autentizovaným vektorem. K výpočtu vzdálenosti se využívá Manhattanské vzdálenosti dle rovnice:

$$D_{Manhattan}(P, P') = \sum_{i=1}^{n} |P_i - P'_i| \qquad (11)$$

kde:

$D_{Manhattan}$ - vzdálenost Manhattan

P_i - i-tá složka vektoru autentizačního pokusu

P'_i - i-tá složka vektoru středních hodnot z natrénovaných dat

P - vektor autentizačního pokusu

P' - vektor referenční šablony

S použitím této metody bylo v případě [15] (při zachování stejných podmínek jako v případě euklidovské metriky) dosaženo výsledku, kdy EER = 0,153.

V případě [27] byla také využita Manhattanská metrika (opět bylo zachováno stejné nastavení pokusu jako v případě euklidovské metriky). V tomto případě bylo dosaženo EER = 0,415. Stejně jako v předchozím případě se tak jedná o mírně lepší výsledek než s využitím euklidovské metriky. Tato práce opět udává možnost zpřesnění pomocí využití směrodatné odchylky σ_i, kdy je rovnice upravena na:

$$D_{Manhattan/\sigma}(P, P') = \sum_{i=1}^{n} \frac{|P_i - P'_i|}{\sigma_i} \qquad (12)$$

kde:

$D_{Manhattan/\sigma}$ - rozšířená Manhattan vzdálenost

P_i - i-tá složka vektoru autentizačního pokusu

P'_i - i-tá složka vektoru středních hodnot z natrénovaných dat

P - vektor autentizačního pokusu

P' - vektor referenční šablony

σ_i - směrodatná odchylka i-té složky vektoru z natrénovaných dat

Touto úpravou bylo dosaženo zlepšení EER na hodnotu 0,27, což představuje proti původní Manhattan metrice výrazné zlepšení.

S takto upravenou rovnicí Manhattan metriky pracuje i [15], kde se tato označuje jako Manhattan (scaled). Oproti klasické Manhattan metrice s EER hodnotou 0,153 bylo v tomto případě s upravenou rovnicí dosaženo hodnoty EER = 0,096.

2.3.3. Mahalanobisova metrika

Tato metoda je, stejně jako v případě euklidovské či Manhattanské metriky, založena na vytvoření středního vektoru z registračních dat a následného výpočtu jeho vzdálenosti od autentizačnho vektoru. Na Mahalanobisovu metriku lze nahlížet jako na rozšíření euklidovské metriky, která zohledňuje vztahy v jednotlivých vlastnostech [15]. Vzdálenost v tomto případě zohledňuje velikost rozptylu σ_i^2 jednotlivých vlastností a vypočítá se podle vztahu z následující rovnice:

$$D_{Mahalanobis}(P, P') = \sqrt{\sum_{i=1}^{n} \frac{(P_i - P'_i)^2}{\sigma_i^2}} \qquad (13)$$

kde:

$D_{Mahalanobis}$ - Mahalanobis vzdálenost

P_i - i-tá složka vektoru autentizačního pokusu

P'_i - i-tá složka vektoru středních hodnot z natrénovaných dat

σ_i^2 - rozptyl i-té složky vektoru z natrénovaných dat

Využití Mahalanobisovy metriky přineslo v případě [15] - všechny parametry pokusu (n = 31, 51 uživatelů, 400 opakování, atd.) zůstaly zachovány - zlepšení EER na 0,11. Tento výsledek je za daných podmínek lepší než Manhattan i euklidovská metrika.

Použití Mahalanobis metriky v [27] (opět při zachování ostatních parametrů jako v případě euklidovské a Manhattan metriky) přineslo zlepšení EER na 0,31. Opět se jedná o výsledek lepší než v případě euklidovské či Manhattan metriky (s výjimkou upravené Manhattanské).

2.3.4. Metoda nejbližšího souseda

Metoda nejbližšího souseda (nearest neighbour) na rozdíl od předchozích metrik nevyužívá měření vzdálenosti k vypočtené střední hodnotě, ale porovnává autentizované vektory

s každým z vektorů v referenční bázi a vybírá tu, která je nejnižší. Tato metoda v registrační části ukládá seznam vektorů použitých při trénování a vypočítává kovarianční matici. V autentizační části se vypočítává vzdálenost mezi každým vektorem z registrační fáze a testovacím vektorem (pomocí Mahalanobisovy metriky, která díky využití σ_i^2 ve jmenovateli přiřazuje vyšší váhu těm hodnotám, které mají menší odchylky). Vzdálenost testovacího vektoru od nejbližšího trénovacího vektoru je označena za výsledné skóre porovnání. Je-li skóre nižší než určená prahová hodnota (*Th*), je vektor vyhodnocen jako dostatečně blízký a uživatel je autentizován. [15]

Využití metody nejbližšího souseda přineslo v případě [15] výsledek EER v hodnotě 0,10. Opět byly zachovány podmínky jako v předešlých případech (dimenze vektoru, rozdělení na registrační a autentizační fázi, atd).

Tuto metodu využívá mimo jiné i [14]. V tomto případě se pokusu zúčastnilo 21 osob (uživatelů), každá zapsala heslo dlouhé 7 znaků. Ze surových dat pak bylo získáno 15 vlastností, n = 15. Každý účastník své heslo zapsal 150 - 400× v průběhu několika dní, přičemž posledních 75 bylo vybráno pro účely autentizace, zatímco zbytek posloužil k registraci. V další části byla 15 narušitelům prozrazena hesla a každý z narušitelů se 5× pokusil o autentizaci za každého z uživatelů. Tímto způsobem bylo pro autentizační fázi k 75 vektorům každého oprávněného uživatele získáno 75 vektorů narušitelů. V tomto případě není chyba udávána v podobě EER, ale je požadována nulová tolerance narušitelů a chyba je udávána v podobě FRR = 0,3 za dané podmínky (FAR = 0).

2.3.5. Neuronové sítě

Existují dva základní přístupy využití neuronových sítí při autentizaci prostřednictvím dynamiky psaní na klávesnici, standardní a auto-asociativní.

U standardní metody je v trénovací (registrační) části vytvořena dopředná neuronová síť, kde počet neuronů ve vstupní vrstvě odpovídá dimenzi n (počet vlastností vstupních vektorů) a ve výstupní vrstvě je jeden neuron. Tato metoda vyžaduje určitou znalost vstupních vektorů narušitelů - lze využít například uvnitř jedné firmy, kde se pohybují stále stejní zaměstnanci (tedy fixní počet uživatelů, resp. případných narušitelů). Při učení neuronové sítě jsou do vstupní vrstvy dodány trénovací vektory a na výstupním uzlu je požadována hodnota 1,00 pro každý z vektorů uživatele a hodnota 0,00 pro vektory narušitelů. V testovací fázi jsou na vstupní vrstvu přivedeny autentizační vektory a podle hodnoty výstupu se určí, zda

se jedná o uživatele či narušitele. Výstup generovaný neuronovou sítí pro každý z vektorů je skóre s_N. Toto skóre by mělo být v případě oprávněného uživatele blízké hodnotě 1,00 = neuronová síť má z trénovací fáze naučenou jeho dynamiku psaní (při které generuje výstup 1,00) a při zavedení jeho vlastního vektoru ve fázi testování by měla vygenerovat číslo blízké hodnotě 1,00. Pokud je na vstup přiveden vektor narušitele, měla by se generovaná hodnota blížit hodnotě 0,00. K určení, zda se jedná o uživatele či narušitele, tak slouží míra:

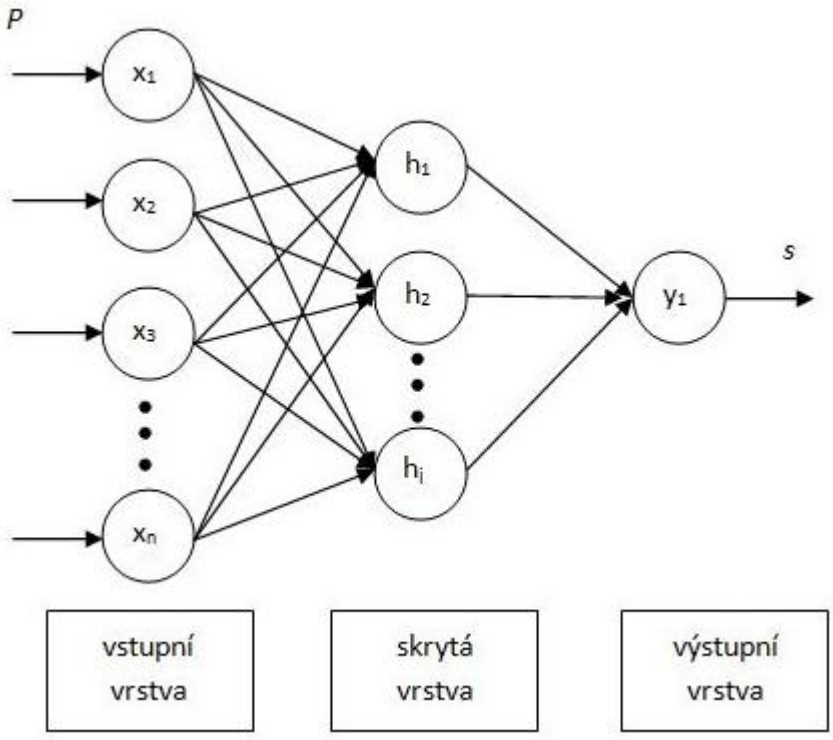
$$D_N = |1 - s_N| \tag{14}$$

kde:

D_N - míra nepodobnosti

s_N - skóre generované neuronovou sítí

Protože hodnota *s* je v případě oprávněného uživatele blízká hodnotě 1,00, bude hodnota D_N minimální. Je-li však D_N vyšší než nastavený práh, pak je uživatel vyhodnocen jako narušitel. Míra podobnosti vstupního vzoru je příliš malá. Neuronová síť tohoto typu je zobrazena na obrázku 12.

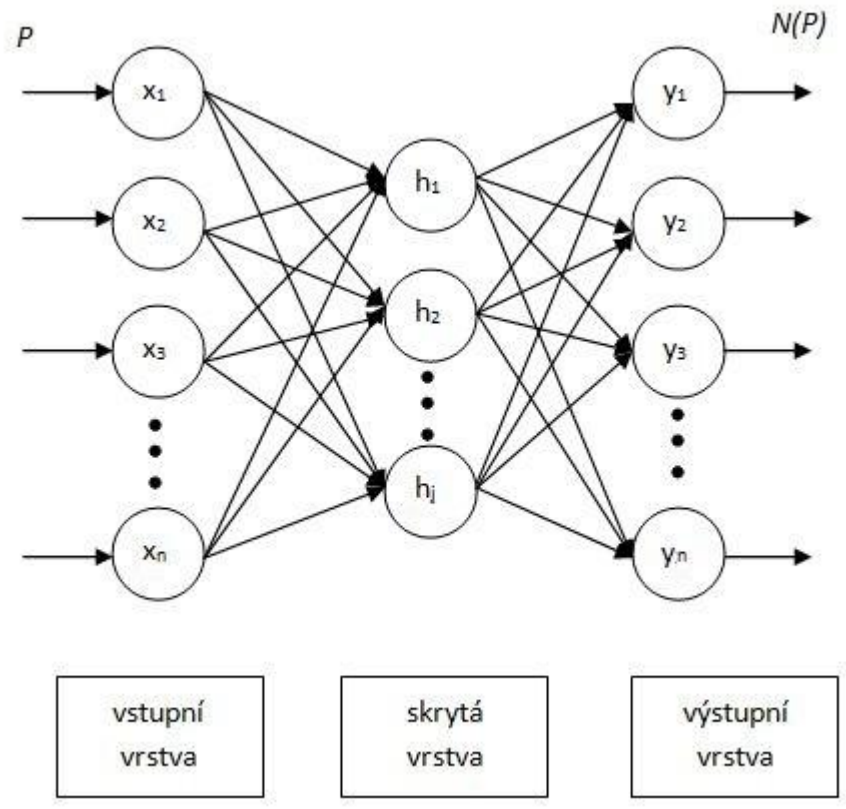


Obrázek 12: Neuronová síť, dle zdroje [31]

Tato metoda byla využita v [15]. Jsou zachovány všechny parametry pokusu jako v předešlých případech (počet uživatelů, dimenze n, ...). Neuronová síť má 31 neuronů

ve vstupní vrstvě, ve skryté vrstvě bylo použito 20 neuronů, 1 neuron na výstupu. S takto nastavenou neuronovou sítí bylo dosaženo hodnoty EER = 0,828.

Druhý způsob využití neuronové sítě při autentizaci prostřednictvím dynamiky psaní na klávesnici představuje auto-asociační přístup. Takováto neuronová síť bývá označována jako auto-associative multi layer perceptron (AAMLP) nebo auto-associative neural network (AANN). Jedná se opět o dopřednou neuronovou síť, ale na rozdíl od předchozího případu je počet neuronů ve výstupní vrstvě shodný s počtem neuronů na vstupní vrstvě. Vstupní hodnoty jsou požadovány jako výstup, neuronová síť tak produkuje pro každý n-rozměrný vektor vstupu (označíme *P*) podobný n-rozměrný vektor na výstupu (označíme *N(P)*). Rozhodnutí o tom, zda vyprodukovaný vektor patří autentizovanému uživateli nebo narušiteli pak probíhá na základě euklidovské vzdálenosti vektorů *P* a *N(P)* a porovnání této vzdálenosti s prahovou hodnotou. Vzdálenost menší než prahová hodnota je přípustná, naopak vzdálenost větší než prahová hodnota je nepřípustná a značí narušitele. Schéma takovéto neuronové sítě je na obrázku 13.



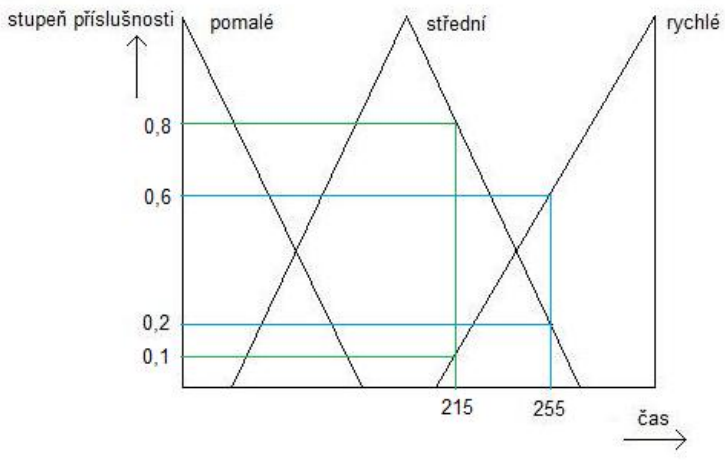
Obrázek 13: Auto asociativní neuronová síť, dle zdroje [8]

Neuronová síť s použitím této metody v [15] obsahovala 31 neuronů na každé z vrstev (odpovídá 31 dimenzím vektorů). Ostatní parametry pokusu zůstaly shodné jako v případě standardní neuronové sítě. Výsledkem bylo EER s hodnotou 0,161.

Stejná metoda byla použita i v [14]. Na vstup neuronové sítě jsou zde přiváděny vektory s 15 vlastnostmi. Chyba zde opět není uváděna v podobě EER, ale jako FRR = 0,1 za předpokladu FAR = 0.

2.3.6. Fuzzy množiny

Základní myšlenka tohoto přístupu je, že rozsahům časových vektorů jsou přiřazeny fuzzy množiny (například čas v rozmezí 210 - 290 milisekund je částí množiny pojmenované „rychlý“). Množiny jsou fuzzy, protože jednotlivé prvky patří do množiny s určitým stupněm příslušnosti (například čas 255 ms je silně součástí „rychlý“ zatímco čas 290 ms sem patří jen slabě). [15] Ve fázi trénování se určí jak silně jednotlivé složky vektoru (časové ukazatele, vlastnosti) patří různým fuzzy množinám a každá složka je přiřazena fuzzy množině, ke které má nejvyšší míru příslušnosti. Například délka stisku klávesy Y bude přiřazena množině „rychlý“ za předpokladu, že většina časových údajů o délce stisku klávesy Y bude v požadovaném čase (například oněch 255ms). Ve fázi autentizace je u každé složky zkoumáno, zda patří do stejné množiny (jako trénovací data) s nejvyšším stupněm příslušnosti. Z předchozího příkladu by bylo testováno, zda má délka stisku klávesy Y nejvyšší stupeň příslušnosti k množině „rychlý“. Výsledné skóre je vypočteno jako průměrná chyba příslušnosti napříč všemi testovacími vektory [15]. Z toho vyplývá, že čím častěji bude každá část vstupního vektoru přiřazena správné fuzzy množině, tím menší bude chybovost. Příklad příslušnosti k fuzzy množině je na obrázku 13.



Obrázek 14: Fuzzy množina, dle zdroje [12]

Zdroj [15] uvádí při využití fuzzy množin střední chybu EER s hodnotou 0,221. Počet uživatelů, opakování a další vlastnosti jsou shodné jako u předešlých metod využitých v [15].

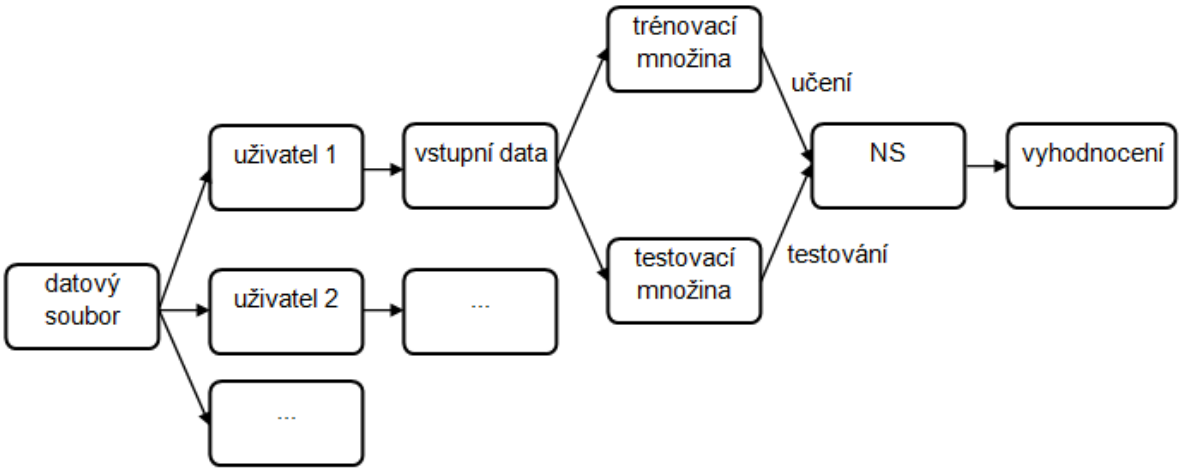
2.3.7. K-Means

Tato metoda využívá shlukovací algoritmus K-Means k identifikaci shluků mezi trénovacími vektory (vektory přivedenými v registrační fázi). Po vytvoření shluků na trénovacích datech se zjišťuje, zda vektor z testovací fáze (autentizační pokus) je blízko nějakého shluku. Vzdálenost vektoru se uvažuje k těžišti vzniklého shluku. Ke zjištění vzdálenosti se využívá euklidovská metrika. [14]

Při zachování stejných podmínek jako u předešlých metod bylo v [14] s použitím této metody dosaženo hodnoty EER = 0,372.

3. Návrh vlastního modelu

Model vychází z metody založené na využití dopředné auto-asociativní neuronové sítě [14] popsané v kapitole 2.3.5 této práce. Schéma modelu je zobrazeno na obrázku 15.



Obrázek 15: Schéma modelu, zdroj vlastní

V práci budou využita volně dostupná data uživatelské dynamiky psaní na klávesnici. Každý z uživatelů bude mít vlastní neuronovou síť (NS), proto je nejprve nutno rozdělit datový set na data pro jednotlivé uživatele. Tyto data je pro vytvoření neuronové sítě dále potřeba rozdělit na trénovací a testovací množinu. Pomocí trénovací množiny se neuronová síť naučí rozpoznávat vektory jednotlivých uživatelů. Poté neuronová síť zpracuje (otestuje) data z testovací množiny a přiřadí jim odpovídající výstupy. U těchto výstupů je potřeba vyhodnotit, zda odpovídají oprávněným uživatelům nebo narušitelům. Jako prostředek vyhodnocení bude sloužit Euklidovská metrika dle vzorce 9 z druhé kapitoly této práce.

3.1. Rozdělení dat

Jak již bylo zmíněno, neuronová síť vyžaduje data rozdělená do dvou kategorií - trénovací a testovací. K tomu v jakém poměru data rozdělit lze dohledat několik teorií.

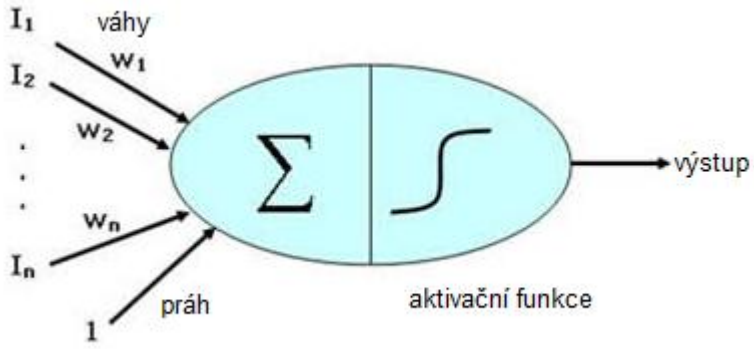
„Velmi často se volí rozdělení 70 % pro trénovací a 30 % pro testovací data. Myšlenka je taková, že více trénovacích dat zpřesní model [13].“ Případně další zdroj uvádí: „Běžně užívaná strategie použití 2/3 vzorků na trénink je téměř optimální pro rozsáhlejší datasety (více než 100 záznamů) [5].“

S přihlédnutím k výše zmíněným radám bylo zvoleno rozdělení dat, kdy 2/3 budou použity jako trénovací a 1/3 jako testovací. Testovací data budou obsahovat 50 % vzorků

oprávněných uživatelů a 50 % vzorků narušitelů tak, aby bylo množství uživatelů a narušitelů v testovacích datech souměrné.

3.2. Neuronová síť

V práci bude použita auto-asociativní neuronová síť, která má klasickou topologii acyklického síťového grafu. Síť obsahuje tři vrstvy neuronů - vstupní, skrytou a výstupní. Grafické znázornění takovéto sítě je na obrázku 13 v předchozí kapitole. Každý neuron má na vstup přivedeny výstupy všech neuronů z předchozí vrstvy a pomocí aktivační funkce počítá výstup, který posílá do další vrstvy. Ohodnocené spojnice mezi neurony se nazývají váhy. Výstup neuronu znamená vstup pro každý z neuronů v následující vrstvě. Schéma neuronu je na obrázku 16.



Obrázek 16: Schéma neuronu, zdroj [30]

Každý neuron se skládá ze dvou částí - agregační a aktivační. V agregační části neuronu dochází k transformaci více-rozměrného vektoru vstupu na skalár (vstupní potenciál). Aktivační část převádí pomocí aktivační funkce hodnotu vstupního potenciálu na výstupní hodnotu neuronu. Výstup j-tého neuronu skryté vrstvy h_j lze zapsat ve tvaru rovnice:

$$h_j = f\left(\sum_{i=1}^{n} w_i * x_i\right) \qquad (15)$$

kde:

h_j - výstup j-tého neuronu

w_i - vstupní váhy do j-tého neuronu skryté vrstvy

x_i - výstupní hodnoty neuronů v předchozí vrstvě

f - aktivační funkce neuronu

Zmíněná aktivační funkce neuronu má sigmoidální charakter, který lze zapsat rovnicí:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{16}$$

Protože se jedná o auto-asociační neuronovou síť, vstupní hodnoty se modelují na výstupu, a tedy počet neuronů ve vstupní vrstvě odpovídá počtu neuronů ve vrstvě výstupní. „Auto-asociativní neuronová síť je neuronová síť, která v trénovací fázi používá vstupní vektor jako výstupní. Síť je pak donucena nějakým způsobem zakódovat vstupní vektor do skryté vrstvy a následně ho dekódovat zpět do vrstvy výstupní.“[14] Je-li pak na vstupu neuronové sítě n-rozměrný vektor *P*, výstupem sítě bude taktéž n-rozměrný vektor *N(P)* vygenerovaný neuronovou sítí.

Počet neuronů ve skryté vrstvě byl zvolen stejný jako u vrstvy vstupní i výstupní, tedy n. K této volbě lze dohledat několik doporučení. Vše podle [11].

- Počet neuronů ve skryté vrstvě by se měl pohybovat mezi velikostí vstupní vrstvy a velikostí výstupní vrstvy.
- Počet neuronů ve skryté vrstvě by měl být 2/3 velikosti vstupní vrstvy plus velikost výstupní vrstvy.
- Počet neuronů ve skryté vrstvě by měl být menší než dvojnásobek velikosti vstupní vrstvy.

Velikost vstupní i výstupní vrstvy je shodně n. Pro zjednodušení a urychlení výpočtu byla zvolena stejná hodnota pro počet neuronů ve skryté vrstvě, která splňuje dvě ze tří doporučení (první a třetí).

Pro každého uživatele je v tomto modelu nutno vytvořit a natrénovat vlastní neuronovou síť. Počet neuronových sítí bude tedy odpovídat počtu uživatelů.

3.3. Metoda vyhodnocení

Výše popsaná neuronová síť poskytne každému vstupnímu vektoru *P* výstup v podobě vygenerovaného vektoru *N(P)*:

$$D_{Euklid}[P, N(P)] = \sqrt{\sum_{i=1}^{n} [P_i - N(P)_i]^2} \tag{17}$$

kde:

D_{Euklid} - Euklidovská vzdálenost vektorů

P - vstupní vektor autentizačního pokusu

$N(P)$ - generovaný výstupní vektor

P_i - i-tá složka vektoru autentizačního pokusu P

$N(P)_i$ - i-tá složka odpovídajícího výstupního vektoru N(P)

Po vypočtení vzdálenosti vektorů bude následovat porovnání s prahovou hodnotou. V případě, že platí vztah:

$$D_{Euklid}[P, N(P)] < Th \tag{18}$$

kde:

D_{Euklid} - Euklidovská vzdálenost vektorů

P - vstupní vektor autentizačního pokusu

$N(P)$ - generovaný výstupní vektor

Th - prahová hodnota

bude vstupní vektor vyhodnocen jako vektor oprávněného uživatele. V případě, že daný vztah neplatí, vektor bude odmítnut jako pokus narušitele o neoprávněnou autentizaci.

Změna velikosti prahové hodnoty *Th* určuje „přísnost“ systému a promítne se do různých velikostí hodnot FAR a FRR.

Vyšší *Th* znamená systém, který připouští větší rozdíly mezi vstupním vektorem *P* a vektorem *N(P)* vygenerovaným neuronovou sítí. Takový systém je otevřenější a povede k nižší FRR, ale na druhou stranu FAR bude vyšší.

Nižší hodnota *Th* naopak značí „přísnější“ systém, který bude mít vyšší FRR a nižší FAR.

4. Aplikace modelu na datech

Navržený model byl aplikován na datech (dataset DSL-StrongPasswordData) volně dostupných ze zdroje [16].

4.1. Data

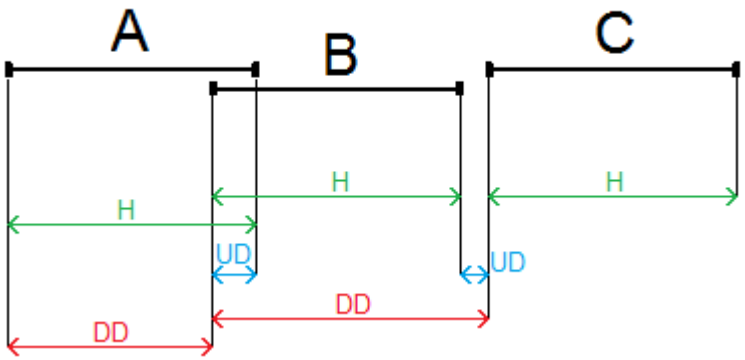
Data byla sesbírána na osobním počítači s operačním systémem Windows XP, vybaveném aplikací pro zaznamenávání časových údajů dynamiky psaní. Software uživateli zobrazil heslo a požadoval jeho opsání. Chybně zadaná hesla nebyla přijata a uživatel byl vyzván k jejich opětovnému zadání. Kdykoliv uživatel stiskl nebo uvolnil klávesu, aplikace zaznamenala tuto událost (event), jméno klávesy (key code) a časovou značku (time stamp).

Účastníků, uživatelů, podílejících se na sběru dat, bylo 51 a jednalo se o studenty a personál Carnegie Mellon University. Všichni uživatelé zapisovali stejné heslo 400×, avšak celý objem nebyl sesbírán najednou. Sběr proběhl v osmi sezeních, každé o 50 opakováních, kdy mezi jednotlivými sezeními byla minimálně jednodenní pauza. Tento přístup byl zvolen k zachycení určité proměnlivosti dynamiky psaní, která se může projevit například z citového nebo tělesného vyčerpání. Rozložením sběru dat do více dní by tento vliv měl být do značné míry odstraněn.

Použité heslo bylo zvoleno s ohledem na typické požadavky moderních systémů. Bylo náhodně vygenerováno pomocí [29] takové, aby bylo 10 znaků dlouhé, obsahovalo malá i velká písmena, číslice a speciální znaky. Výsledkem bylo heslo:

$$.tie5Roanl$$

Jak již bylo zmíněno, každý uživatel heslo zadal 400×. Takto získaná data zaznamenaná softwarovou aplikací byla následně analyzována a byla z nich vytvořena tabulka časových údajů. Ze surových dat byly do tabulky zaznamenány 3 druhy časových informací, které schematicky ilustruje obrázek 17.

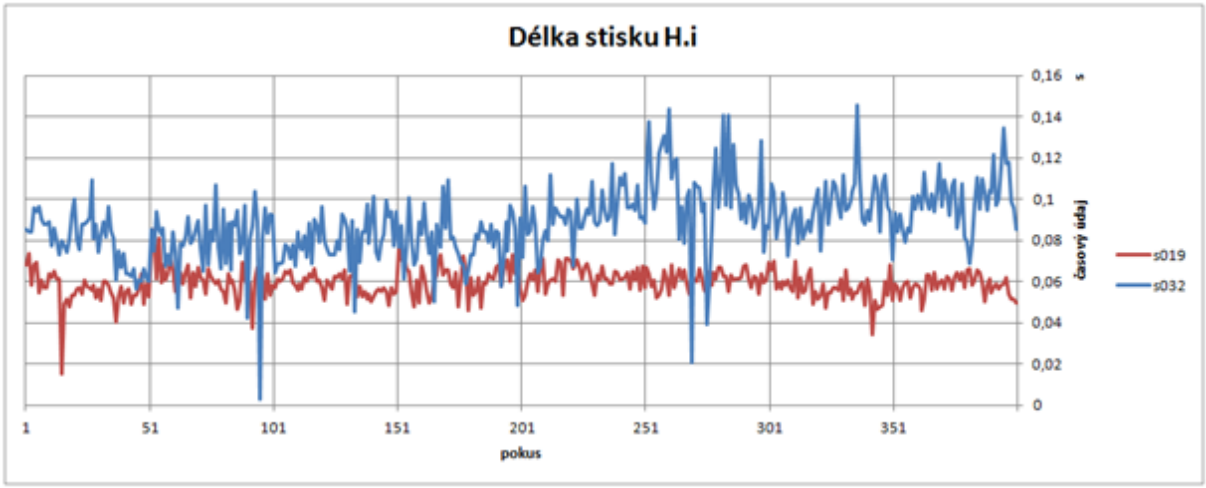


Obrázek 17: Časové údaje, zdroj vlastní

- H - délka stisku, označeno zeleně
- DD - latence stisk / stisk, označeno červeně
- UD - latence uvolnění / stisk, označeno modře

V této tabulce jsou zaznamenány časové informace o stiscích kláves pro všechny pokusy všech uživatelů doplněné o identifikační údaje. Každému účastníkovi pokusu tak odpovídá 400 řádků s údaji ve 34 sloupcích. Z toho 31 sloupců obsahuje časová data a tři sloupce slouží k identifikaci (udávají o kterého uživatele a kolikátý pokus v jakém sezení se jedná).

Pro představu jak data vypadají názorně poslouží graf. Průběh jedné z hodnot - délky stisku náhodně vybrané klávesy „i“ („Hold i“ - H.i) - pro dva náhodně vybrané uživatele (s019 a s032) zobrazuje graf na obrázku 18.



Obrázek 18: Průběh délky stisku vybrané klávesy, zdroj vlastní

V datech existují značné výkyvy, jak je z obrázku na první pohled patrné, a to jak v rámci jednotlivých sezení (každé o padesáti pokusech) tak napříč sezeními. Uživatel s019 má

hodnoty vyrovnanější, což potvrzuje menší směrodatná odchylka odpovídající danému údaji H.i (pro uživatele s019 činí 0,00674 sekund oproti 0,01702 sekund uživatele s032). Stejně tak je na první pohled patrná nižší průměrná hodnota délky stisku (0,05948 sekund oproti 0,08861 sekund).

4.2. Rozdělení dat

Rozdělení na trénovací a testovací data bylo navrženo v poměru 2/3 trénovací a 1/3 testovací. V případě dat ze zdroje [16] je dostupných 400 hodnot pro každého uživatele. Dříve byl však vznesen i požadavek na složení testovacích dat, kdy polovina z nich má být vzory uživatele a druhá polovina narušitele. Pro přibližné zachování těchto podmínek lze data uživatele rozdělit na 300 trénovacích vektorů, zbylých 100 vektorů použít jako vektory testovací a doplnit je o dalších 100 testovacích vektorů od jiných uživatelů (narušitelů).

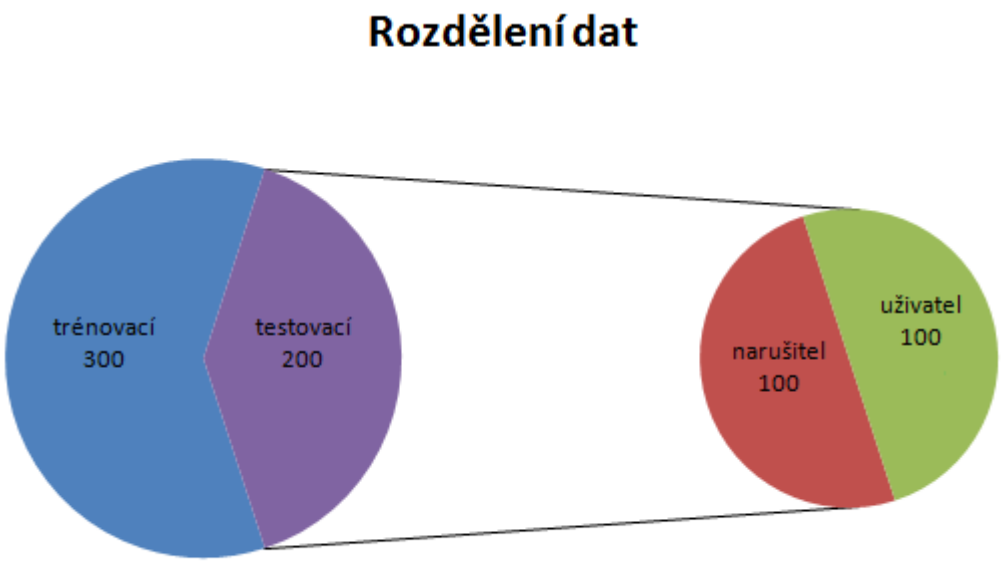
S přihlédnutím k výše zobrazenému průběhu hodnot z obrázku 18 a tomu odpovídajícímu teoretickému poznatku, že vstup závisí na aktuálním psychickém i fyzickém stavu, bylo rozhodnuto, že vstupy (řádky) uživatele budou popořadě rozděleny do čtveřic a do trénovací části budou vybrány vždy první tři řádky z každé čtveřice. Tímto způsobem budou v trénovací i v testovací fázi rovnoměrně obsažena data ze všech osmi sběrných sezení, a tedy by měl být odstíněn vliv fyzického či psychického vyčerpání (budeme-li uvažovat, že se vyčerpání mění pouze mezi sezeními a nikoliv v průběhu sezení samotného). Rozdělení na trénovací a testovací data je schematicky znázorněno v následující tabulce.

Tabulka 3: Rozdělení na trénovací a testovací data, zdroj vlastní

uživatel	sezení	číslo pokusu	trénovací / testovací
s019	1	1	trénovací
s019	1	2	trénovací
s019	1	3	trénovací
s019	1	4	testovací
s019	1	5	trénovací
s019	1	6	trénovací
s019	1	7	trénovací
s019	1	8	testovací
s019	1	9	trénovací
…	…	…	…

Data uživatelů jsou rozdělena v poměru 3:1, čímž je pro každého autentizovaného uživatele získáno 300 vstupních vektorů pro trénink a 100 testovacích vektorů. Zbývá tak doplnit

dalších 100 testovacích vektorů z dat narušitelů. Zde byly vždy pro každého z 50 zbývajících uživatelů náhodně vybrány 2 vektory z množiny jeho testovacích vektorů (celkem 100 vektorů) a těchto 100 vektorů bylo doplněno k testovacím vektorům autentizovaného uživatele. Rozdělení dat a složení trénovací a testovací skupiny každého uživatele ilustruje obrázek 20.



Obrázek 19: rozdělení dat, zdroj vlastní

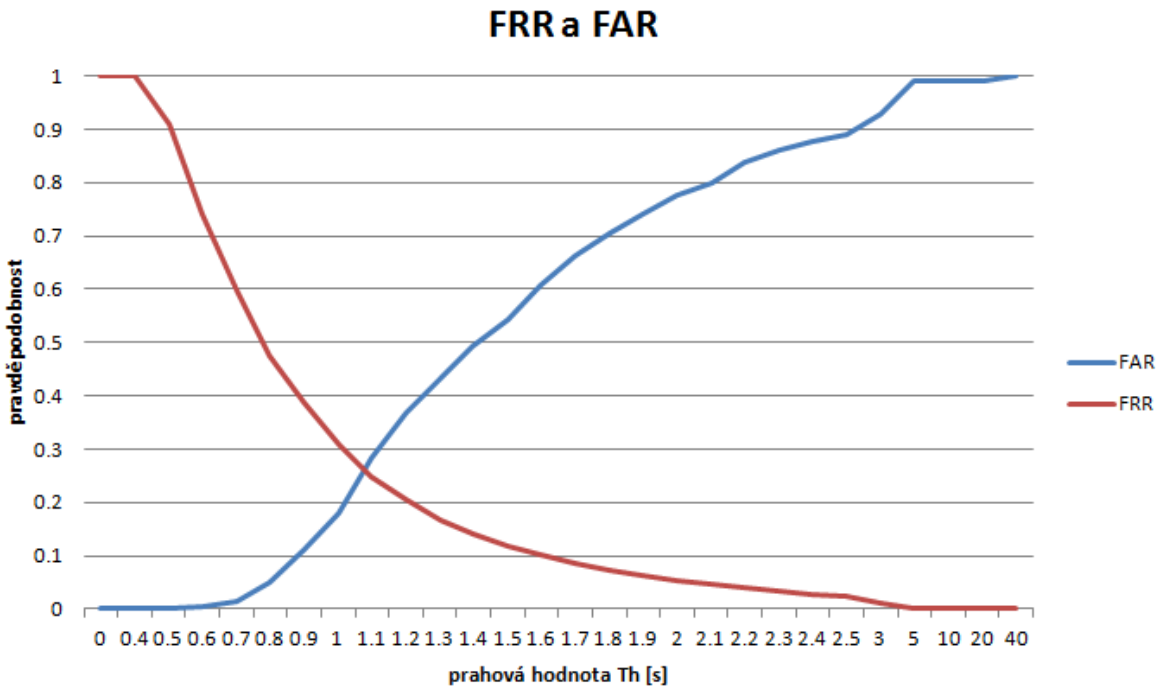
Trénovací množina má velikost 300 a testovací 200 vektorů, kde 100 vektorů patří uživateli a 100 vektorů narušitelům. Ve výsledku je tak dodržena druhá podmínka (složení testovací množiny), přičemž u první podmínky bylo pro snazší výpočet ustoupeno ze 2/3 dat a množství bylo sníženo na 60 %.

4.3. Dosažené výsledky

K vyhodnocení metody bylo využito chyby prvního druhu (FRR) a chyby druhého druhu (FAR). Podle rovnice 17 byla měřena euklidovská vzdálenost vektorů vygenerovaných neuronovou sítí od vektorů vstupních a následně porovnávaná s prahovou hodnotou *Th* podle rovnice 18. Změny prahové hodnoty vedly ke změnám FAR a FRR. Další možností je uvedení shodné míry chybovosti (EER) - míry vyrovnání FAR a FRR, případně graf kompromisu chyb (DET - detection error tradeoff).

4.3.1. Souhrnné hodnoty

Celkovou úspěšnost autentizace systému při různých nastaveních prahové hodnoty *Th* lze vyjádřit pomocí souhrnné FAR a FRR, která zahrnuje chyby všech uživatelů při všech pokusech. Souhrnné FRR a FAR je možno vidět na obrázku 20.



Obrázek 20: Souhrnná FRR a FAR, zdroj vlastní

Jak je z grafu patrné, s rostoucí velikostí prahové hodnoty *Th* se zvyšuje míra chybných přijetí a naopak snižuje se míra chybných odmítnutí. Systém při takových nastaveních upřednostňuje snadnou autentizaci i za cenu toho, že mezi autentizovanými uživateli pronikne více narušitelů.

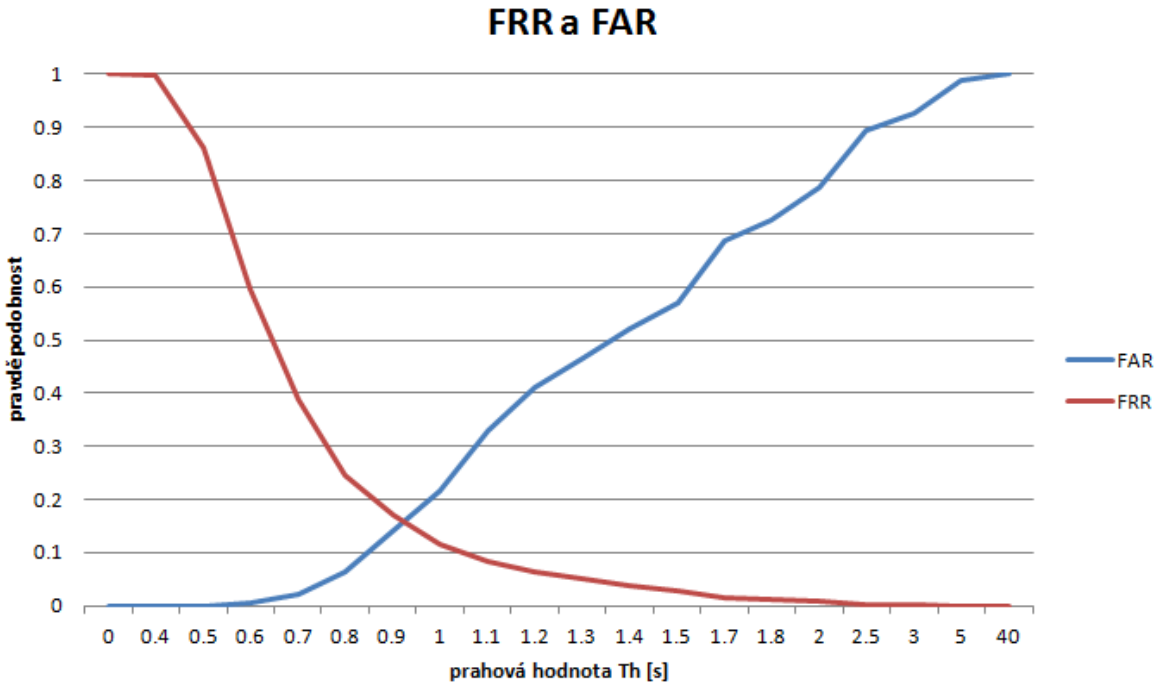
Průsečík křivek FRR a FAR, tedy Equal Error Rate, má hodnotu 0,2618 a bylo ho dosaženo při nastavení prahové hodnoty *Th* = 1,076 s.

Křivky FRR a FAR jsou na sobě nepřímo závislé skrz prahovou hodnotu *Th* a každá její změna vyvolá změny obou křivek v opačných směrech. Není proto možné minimalizovat změnou prahové hodnoty obě chyby zároveň.

Protože hodnota EER = 0,2618 nepatří mezi nejpřesnější, bylo v modelu provedeno několik změn pro zvýšení přesnosti výstupu. V první řadě byl snížen počet neuronů ve skryté vrstvě z původních 31 na 15. Dále byla o 20 % snížena doba učení neuronové sítě, aby nemohlo

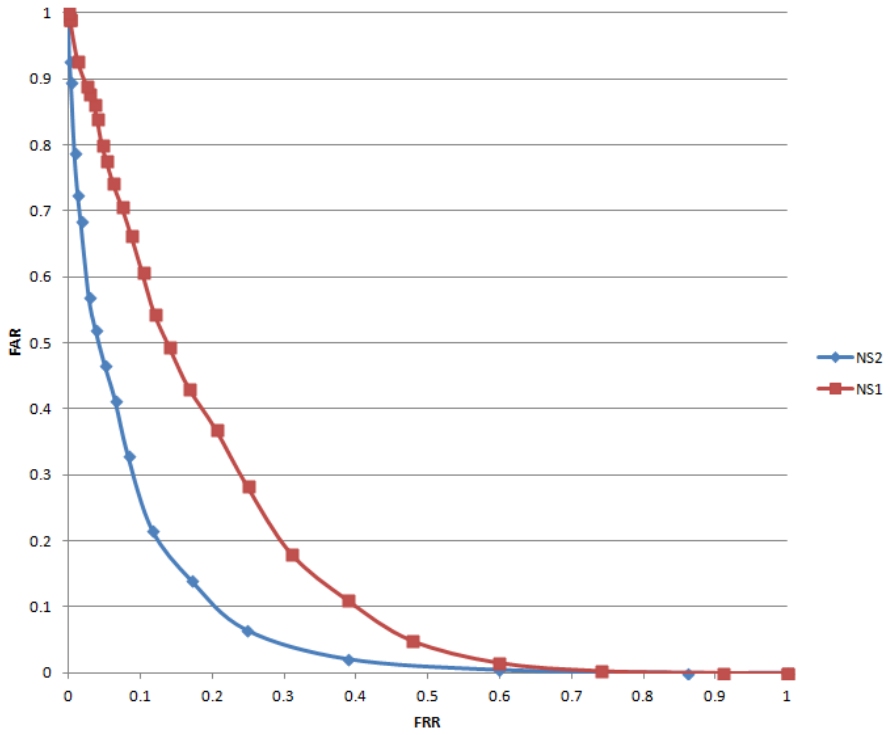
dojít k naučení dat zpaměti a stejně tak byla o 20 % zvýšena rychlost učení (rychlost s jakou se mění váhy ve fázi trénování sítě).

S takto upravenou neuronovou sítí se podařilo dosáhnout hodnoty EER = 0,1641 při prahové hodnotě *Th* = 0,9294 s. Těmto hodnotám odpovídají křivky FAR a FRR na obrázku 21.



Obrázek 21: Souhrnná FRR a FAR po úpravě, zdroj vlastní

Vztah křivek FRR a FAR lze vyjádřit DET grafem na obrázku 22. V DET grafu jsou obsaženy jak původní síť vytvořená podle předem navrženého modelu (NS1), tak síť upravená (označená NS2).



Obrázek 22: DET graf, zdroj vlastní

DET graf vyjadřuje úrovně kompromisu mezi FRR a FAR. Pohybem po křivce grafu směrem vlevo lze sice snížit FRR, ale pouze na úkor zvýšení FAR a opačně.

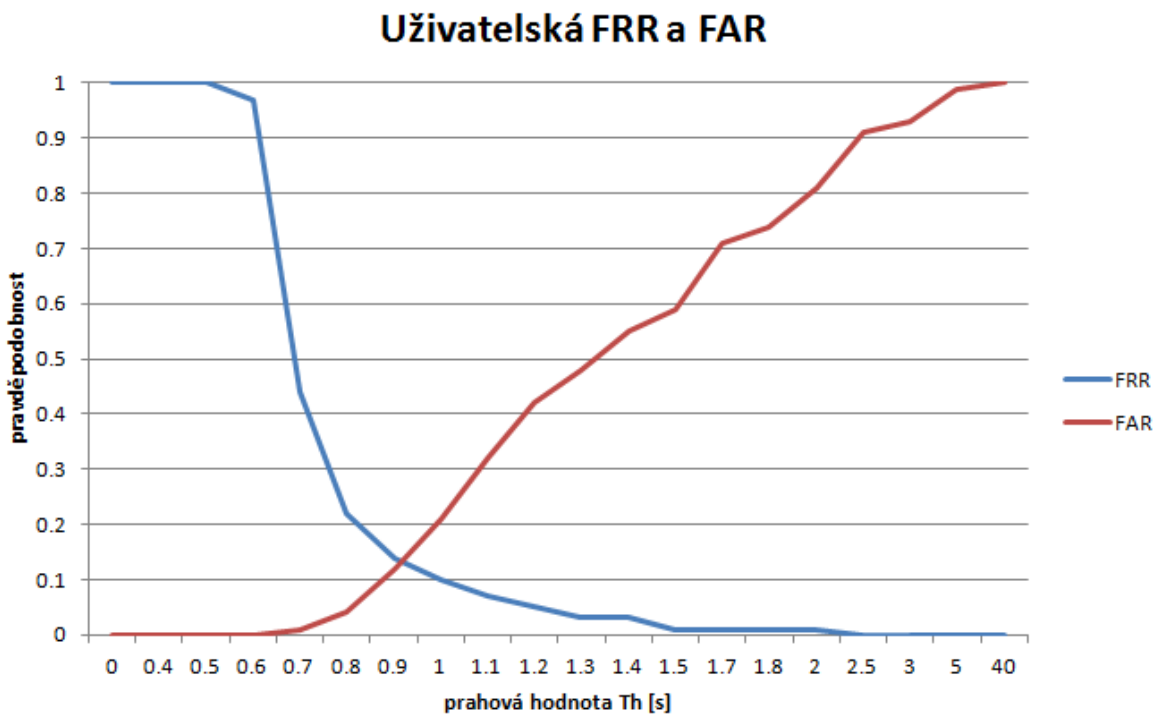
Protože aktuální model nevyžaduje minimalizaci ani jedné z chyb, lze brát za ideální výstup EER = 0,2618 v případě původní neuronové sítě, resp. EER = 0,1641 v případě sítě upravené, kdy jsou obě chyby v rovnováze. DET graf by však své využití našel zejména v případě ladění modelu pro konkrétní situaci.

Měl-li by se daným modelem autentizovat přístup k přísně utajované složce, bude rozumné volit nižší hodnotu FAR i za cenu vyšší FRR. Takovýto postup lze předpokládat v případě, kdy bude mít utajení vyšší prioritu než narušení dostupnosti, tedy odmítnutí oprávněného uživatele.

Naopak pokud by měl být model využit například k autentizaci pro přístup do veřejné internetové chatovací místnosti (kde lze za primární požadavek brát spíše fakt, že se autentizuje skutečný člověk a ne softwarový robot využívaný k šíření reklamy, a kde vpuštěním neoprávněného narušitele nevzniká žádná škoda), bude možné povolit mnohem vyšší FAR.

4.3.2. Uživatelské hodnoty

Existuje samozřejmě i možnost nepočítat hodnoty souhrnné, ale zobrazit výstup pro jednotlivé uživatele. Příklad uživatelské FRR a FAR je na obrázku 23.



Obrázek 23: Uživatelská FRR a FAR, zdroj vlastní

Jedná se o hodnoty FRR a FAR pouze pro jednoho náhodně vybraného uživatele.

EER je v tomto případě 0,13 a je ho dosaženo při *Th* = 0,922 s. Protože v testovací fázi bylo přivedeno na vstup neuronové sítě 100 vektorů uživatele a 100 vektorů narušitelů, tento výsledek značí, že 13 pokusů narušitelů bylo chybně autentizováno a 13 autentizačních pokusů uživatele bylo chybně odmítnuto.

4.4. Návrhy ke zlepšení

Omezíme-li se na hodnotu EER jako ukazatel úspěšnosti modelu, dosažená hodnota 0,2618 nepatří mezi nejlepší i přesto, že výsledky napříč internetem nelze díky často zcela odlišným vstupním hodnotám a rozsahům souborů porovnat. Hodnota EER upravené sítě 0,1641 je již výsledek lepší, přesto i zde lze předpokládat možné zlepšení.

V následujících odstavcích je navrženo několik možností, které by mohly vést ke zkvalitnění výstupu a snížení EER.

4.4.1. Změna výpočtu vzdálenosti

Jednou z možnost zpřesnění výstupu může představovat použití jiné než euklidovské metody pro výpočet vzdálenosti mezi vektory *P* a *N(P)*.

Na základě modelu bylo zkušebně použito i výpočtu Manhattanské vzdálenosti dle rovnice 11 namísto vzdálenosti Euklidovské, avšak souhrnný výsledek v podobě EER = 0,2596 (při *Th* = 4,3507 s) pro první neuronovou síť, resp. EER = 0,1664 v případě upravené neuronové sítě (při *Th* = 3,789 s), jsou hodnoty velmi podobné jako v případě Euklidovské vzdálenosti, a Manhattanská metrika tak nepřináší výrazné zlepšení.

Použití Mahalanobisovy vzdálenosti (rovnice 13), která uvažuje rozptyl v datech uživatelů, také nepřineslo zlepšení. Mahalanobisova metrika se pro tuto aplikaci ukázala jako nejméně vhodná, když s jejím využitím bylo dosaženo nejvyšší chybové hodnoty EER = 0,3066 (při prahové hodnotě *Th* = 6,5489 s^2) pro první neuronovou síť a chyby EER = 0,3094 (při prahové hodnotě *Th* = 41,1219 s^2) pro upravenou neuronovou síť.

4.4.2. Změna v datech

Další možností, jak se pokusit zlepšit výstup systému, může být změna v rozdělení na trénovací a testovací množinu. Současné rozdělení 60:40 by se mohlo změnit například směrem k 70:30 s cílem preferovat více dat pro učení neuronové sítě za cenu méně testovacích pokusů.

Druhou možností jak by se s daty mohlo naložit je zanedbání některých časových ukazatelů a tím i snížení dimenze u vektorů vstupujících do neuronové sítě, případně použití kratšího hesla, které by taktéž vedlo k nižší dimenzi vstupních vektorů. Menší počet atributů nemusí být nutně na škodu, neboť i některé další práce [27], [14] úspěšně operují s vektory o menším rozsahu.

V rámci práce bylo heslo pokusně zkráceno o dva znaky, což vedlo ke snížení vstupní dimenze na 27 časových údajů. Tato změna se projevila snížením chyby EER na hodnotu 0,2553 (při prahové hodnotě *Th* = 0,9981 s) v případě první neuronové sítě. U upravené neuronové sítě bylo dosaženo hodnoty EER = 0,1617 při prahové hodnotě *Th* = 0,8772 s. Tento údaj je však potřeba brát s rezervou, protože takto uměle zkrácené heslo nemusí dynamikou odpovídat skutečně kratšího hesla tak, jak by ho ve skutečnosti uživatelé zapisovali.

4.4.3. Optimalizace pro uživatele

Každý uživatel píše jinak a s jinou proměnlivostí. Někteří mají dynamiku stálejší, jiní naopak proměnlivější. A právě na tomto poznatku by se mohlo stavět.

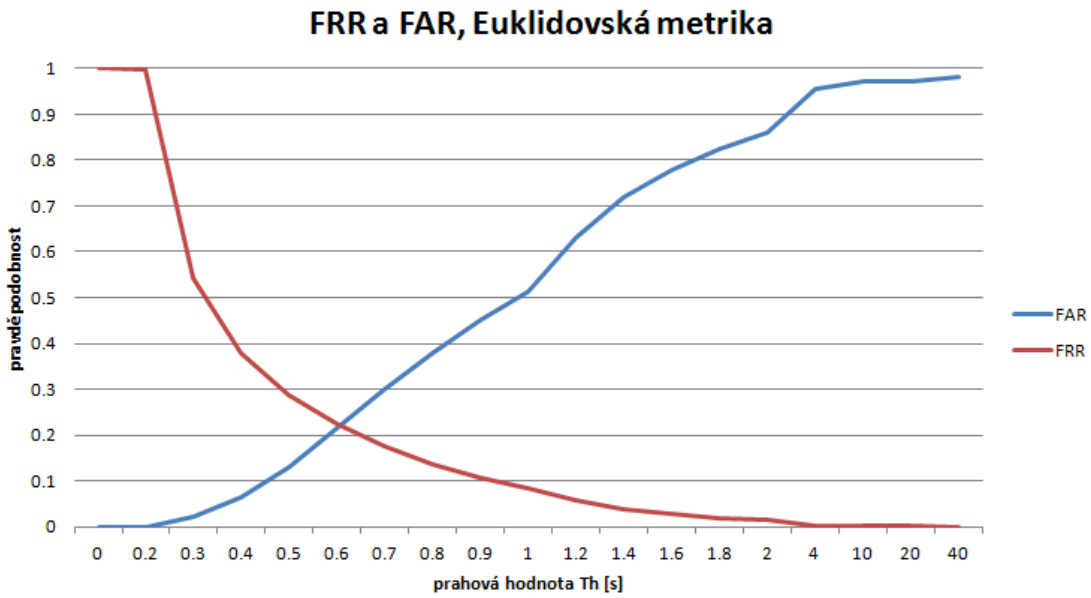
Jak bylo zmíněno v části 4.3.2, pomocí této metody je možné počítat FAR, FRR a tedy i EER zvlášť pro každého z uživatelů. S dostatečnou znalostí dat a odpovídajících výstupů by mohlo ke zpřesnění vést i nastavení prahové hodnoty *Th* specifické pro každého uživatele. Je-li při globálním nastavení hranice souhrnné EER dosaženo například při *Th* = 1,1, některý z uživatelů může být již velmi blízko maxima FAR či naopak. Takovému uživateli individuální nastavení *Th* může výrazně prospět.

V praxi by tento přístup vyžadoval větší pozornost a nutnost neustálé kontroly každého uživatele zejména v případě změny hesla, kdy by bylo nutné po analýze výstupů znovu volit i potřebné prahové hodnoty. Je také nutno brát v potaz i fakt, že sebestálejší (z hlediska dynamiky psaní) uživatel může například vlivem nemoci změnit svůj projev více než obvykle.

4.5. Porovnání s dalšími metodami

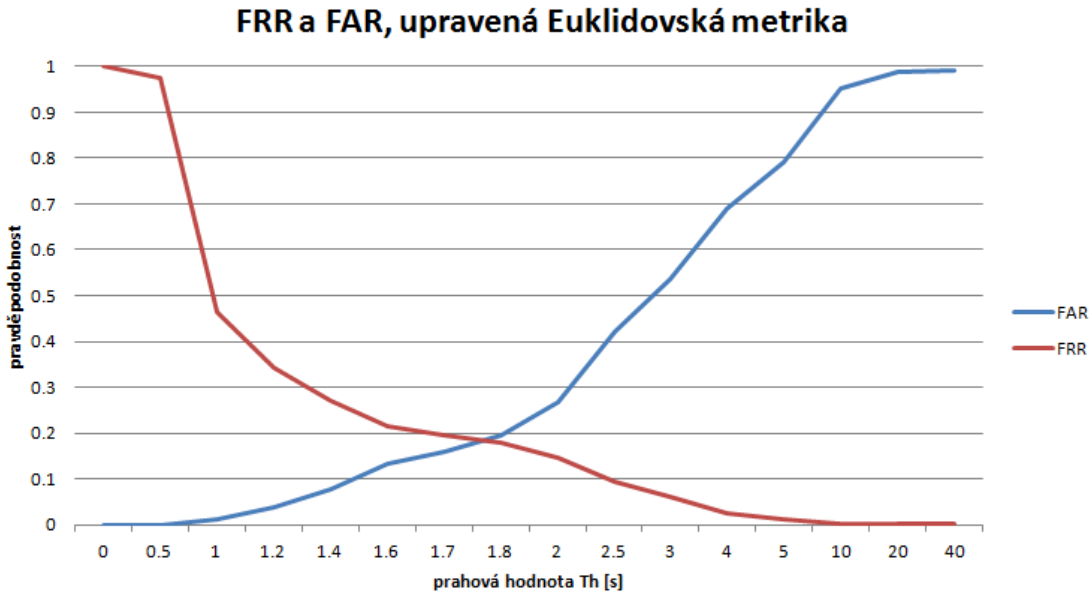
K porovnání výstupu poslouží pět metod popsaných v kapitolách 2.3.1 až 2.3.3 této práce.

Nejprve byla jako metoda k porovnání vybrána euklidovská metrika. Z dat v trénovací fázi je vypočítána střední hodnota z časových vektorů. V testovací fázi je vypočtena hodnota čtverce euklidovské vzdálenosti mezi testovaným vektorem a vektorem střední hodnoty trénovací množiny podle rovnice 9. Použitím této metody bylo dosaženo křivek FAR a FRR zobrazených na obrázku 24. Průsečík křivek odpovídá hodnotě EER = 0,2213 při prahové hodnotě *Th* = 0,6064 s.



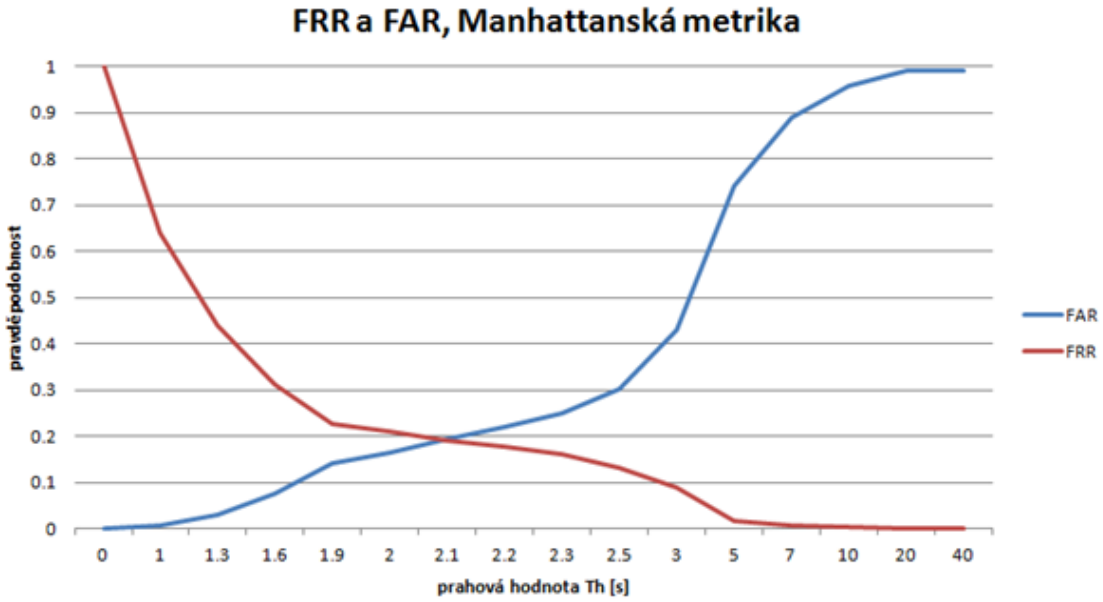
Obrázek 24: FRR a FAR, Euklidovská metrika, zdroj vlastní

Pro zlepšení přesnosti byla Euklidovská metrika stejně jako v [27] rozšířena o zahrnutí vlivu směrodatné odchylky. Výpočtem dle rovnice 10 tak bylo dosaženo EER = 0,1832 při prahové hodnotě *Th* = 1,7675 s. Průběh křivek FRR a FAR takto upravené Euklidovské metriky zobrazuje následující obrázek.



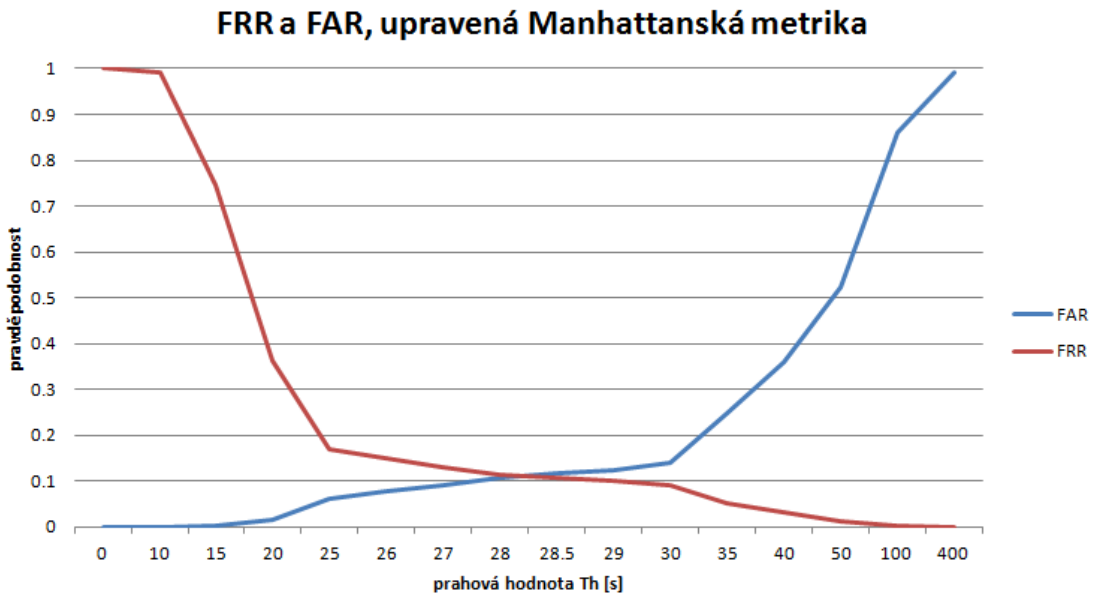
Obrázek 25: FRR a FAR, upravená Euklidovská metrika, zdroj vlastní

Třetí použitou metodou byla Manhattanská metrika. Tato metrika vychází ze vztahu 11 a jejím využitím bylo dosaženo hodnoty EER = 0,1923 při prahové hodnotě *Th* = 2,0991 s. Průběh křivek FAR a FRR je zobrazen na obrázku 26.



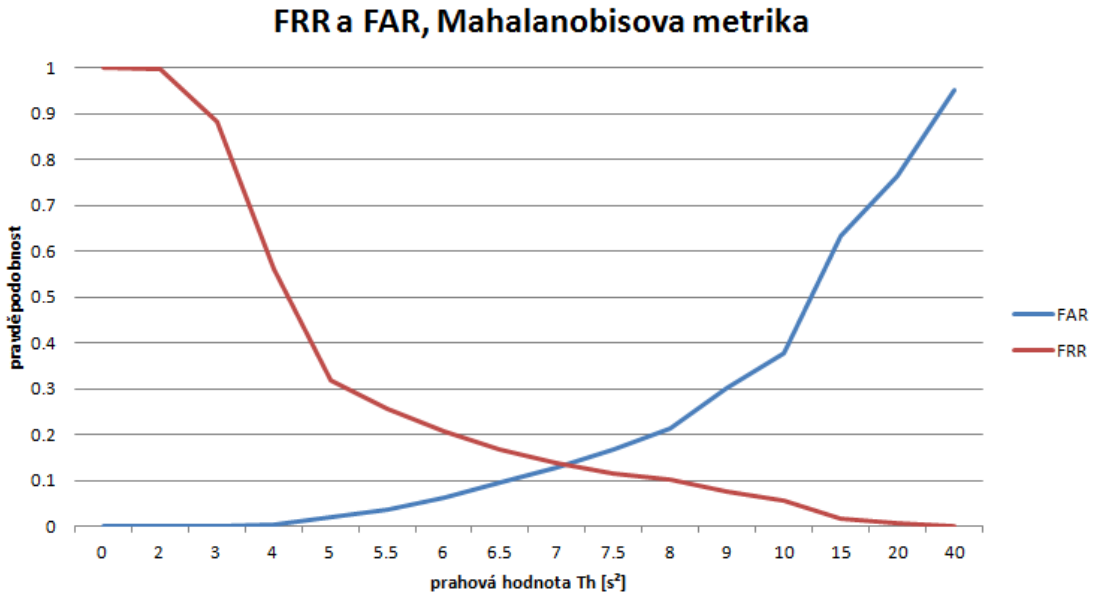
Obrázek 26: FRR a FAR, Manhattanská metrika, zdroj vlastní

Obdobně jako v případě Euklidovské metriky byl i u Manhattan metriky proveden výpočet zahrnující směrodatnou odchylku podle [27] a dle rovnice 12. Použitím takto rozšířené Manhattan metriky bylo dosaženo průběhu křivek zobrazených na obrázku 27. Jak je již z grafu patrné, tento výsledek je dosavadní nejlepší hodnotou EER. Hodnota 0,1106 byla dosažena při *Th* = 28,2351 s.



Obrázek 27: FRR a FAR, upravená Manhattanská metrika, zdroj vlastní

Jako poslední bylo k porovnání využito Mahalanobisovy metriky, která využívá rozptyl v hodnotách. Podle vztahu 13 bylo dosaženo hodnot EER = 0,1339 při Th = 7,1125 s^2. Průběh křivek zobrazuje následující obrázek.



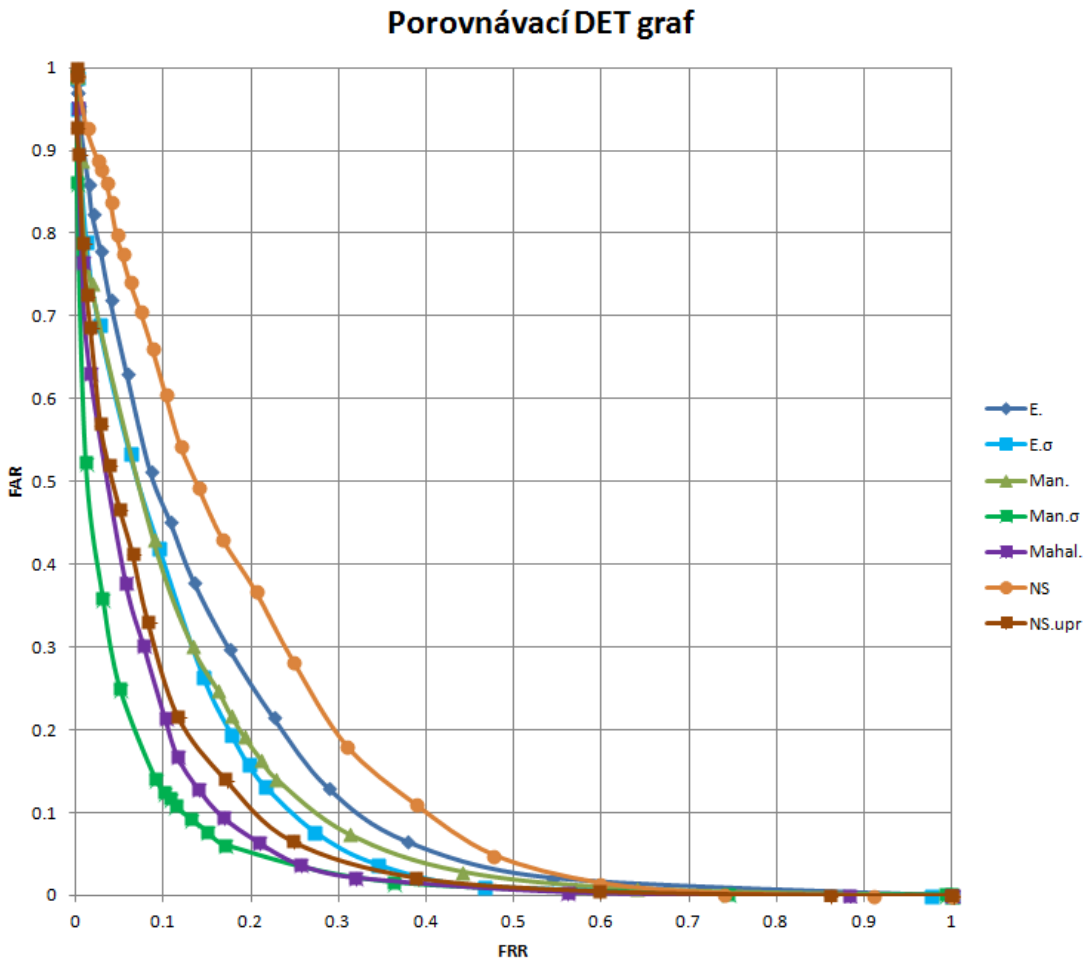
Obrázek 28: FRR a FAR, Mahalanobisova metrika, zdroj vlastní

Přehlednější porovnání jednotlivých výsledků je v tabulce 4. Nejnižší EER bylo dosaženo s využitím rozšířené Manhattan metriky, nejvyšší EER s pomocí původní neuronové sítě.

Tabulka 4: Porovnání výsledků, zdroj vlastní

metoda	hodnota EER
Euklidovská metrika	0,2213
Rozšířená Euklidovská metrika	0,1832
Manhattan metrika	0,1923
Rozšířená Manhattan metrika	0,1106
Mahalanobis metrika	0,1339
Neuronová síť	0,2616
Upravená NS	0,1641

K nejnázornějšímu porovnání všech provedených metod poslouží souhrnný DET graf. Tento graf zobrazuje poměry FRR a FAR pro všechny testované metody. Jak již bylo zmíněno, nejlepších hodnot bylo dosaženo výpočtem rozšířené Manhattan metriky, čemuž odpovídá DET graf nejvíce se přibližující ideální hodnotě FRR = FAR = 0. Tento DET graf je zobrazen na obrázku 29.



Obrázek 29: Souhrnný DET graf, zdroj vlastní

Euklidovská metrika je v grafu označena E. a tmavě modrou barvou, její upravená podoba je světle modrá křivka označená E.σ. Metrika Manhattan je značená zelenou křivkou

s popisem Man. Upravená verze Manhattan metriky je značená Man.σ a její křivka je tmavěji zelená. Mahalanobisova metrika je fialová křivka s označením Mahal. Původní neuronová síť je oranžová křivka označená NS, hnědá křivka označená NS.upr značí upravenou neuronovou síť.

Závěr

Cílem této práce bylo aplikovat metodu využívající neuronové sítě při autentizaci prostřednictvím dynamiky psaní na klávesnici.

V úvodu práce byly rozvedeny a popsány aktuální přístupy autentizace zejména se zaměřením na autentizaci biometrickou s využitím dynamiky psaní na klávesnici spolu s prezentací výsledků některých dalších studií.

V rámci práce byl vytvořen model autentizace prostřednictvím dynamiky psaní na klávesnici, který využívá dopřednou auto-asociativní neuronovou síť a euklidovskou metriku k měření vzdálenosti mezi vstupními a výstupními veličinami neuronové sítě.

Následně byl tento model aplikován na rozsáhlém souboru dat, který je dostupný v [16]. Aplikací modelu na datech bylo dosaženo hodnoty shodné míry chybovosti (EER, průsečík křivek FRR a FAR) v hodnotě 0,2618. Zpřesněný výsledek s upravenou neuronovou sítí dosáhl hodnoty EER = 0,1641.

Tyto hodnoty lze porovnat s hodnotami dosaženými běžnými metodami nevyužívajícími neuronové sítě. Nejpřesnější z použitých srovnávacích metod byla upravená Manhattan metrika, jejíž hodnota EER byla 0,1106. Porovnání přesnosti jednotlivých použitých metod přehledně zobrazuje obrázek 29.

Cíle práce se podařilo naplnit.

Použitá literatura

[1] Authentication. EMC CORPORATION. *Information Security Glossary* [online]. 2013 [cit. 2013-07-11]. Dostupné z: http://www.rsa.com/glossary/?id=1006

[2] BEHÚN, Dalibor a Jan CHROMÝ. Autentizace, autentikace nebo autentifikace?. *Interval.cz* [online]. 2004 [cit. 2013-07-07]. Dostupné z: http://interval.cz/clanky/hrichy-pro-sileneho-korektora-autentizace-autentikace-nebo-autentifikace/

[3] BENEŠ, R. Autentizační metody založené na biometrických informacích. *Access Server*, 2010, roč. 8, č. 1, s. 1-9. ISSN: 1214- 9675.

[4] Dell Home Computers. *Dell Official Site* [online]. 2014 [cit. 2014-04-01]. Dostupné z: http://www.dell.com/us/p/?~ck=mn

[5] DOBBIN, Kevin K. a Richard M. SIMON. Optimally splitting cases for training and testing high dimensional classifiers. *BMC Medical Genomics*. 2011, vol. 4, issue 1, s. 31-. DOI: 10.1186/1755-8794-4-31. Dostupné z: http://www.biomedcentral.com/1755-8794/4/31

[6] ELTAHIR, Wasil Elsadig, M. J. E. SALAMI, Ahmad Faris ISMAIL a Weng Kin LAI. Design and Evaluation of a Pressure-Based Typing Biometric Authentication System. *EURASIP Journal on Information Security*. 2008, vol. 2008, č. 1, s. 1-14. DOI: 10.1155/2008/345047. Dostupné z: http://jis.eurasipjournals.com/content/2008/1/345047

[7] Expected Growth. *AXIS* [online]. 2009 [cit. 2013-07-09]. Dostupné z: http://www.axistech.com/WebPages/biometricexactedgrowth.aspx

[8] GERSHTEYN, Yevgeniy a Larisa PERMAN. Autoassociative Neural Network. *Neural Network & Machine Learning* [online]. 2003, č. 1 [cit. 2013-07-11]. Dostupné z: http://csrit.gershteyn.net/courses/Presentations/2-AutoAssciativeNN.pdf

[9] GIOT, Romain, Mohamad EL-ABED a Christophe ROSENBERGER. *Biometrics: Keystroke Dynamics Authentication* [online]. InTech, 2011 [cit. 2013-07-09]. ISBN 978-953-307-618-8. Dostupné z: http://www.intechopen.com/books/biometrics/keystroke-dynamics-overview

[10] *Google.cz* [online]. 2013 [cit. 2013-07-07]. Dostupné z: http://www.google.cz

[11] HEATON, Jeff. Introduction *to neural networks with Java*. 2nd ed. St. Louis: Heaton Research, 2008, xxxviii, s. 39-439. ISBN 16-043-9008-5.

[12] HONG, Tzung-Pei a Chai-Ying LEE. Induction of fuzzy rules and membership functions from training examples. *Fuzzy Sets and Systems* [online]. Netherlands: Elsevier BV, 1996, č. 84 [cit. 2014-04-01]. Dostupné z: http://comp.eng.ankara.edu.tr/files/2013/03/BLM436Lecture8.pdf

[13] CHISHOLM, Andrew William. Why split data in the ratio 70:30?. *Information Gain Ltd* [online]. 2013 [cit. 2013-07-29]. Dostupné z: http://information-gain.blogspot.cz/2012/07/why-split-data-in-ratio-7030.html

[14] CHO, Sungzuun, Chigeun HAN, Dae Hee HAN a Hyung-Il KIM. *Web based Keystroke Dynamics Identity Verification using Neural Network*. 2000. Dostupné z: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.4863&rep=rep1&type=pdf

[15] KILLHOURY, Kevin a Roy MAXION. *Comparing Anomaly-Detection Algorithms for Keystroke Dynamics*. Computer Science Department of Carnegie Mellon University, 2009. Dostupné z: http://www.cs.cmu.edu/~maxion/pubs/KillourhyMaxion09.pdf

[16] KILLOURHY, Kevin a Roy MAXION. Keystroke Dynamics - Benchmark Data Set. *Carnegie Mellon University: School of Computer Science* [online]. 2009 [cit. 2013-06-04]. Dostupné z: http://www.cs.cmu.edu/~keystroke/

[17] Klávesnice pro počítače Mac, PC a tablety. *Logitech* [online]. 2014 [cit. 2014-04-01]. Dostupné z: http://www.logitech.com/cs-cz/keyboards

[18] KOTHAVALE, Mamta, Robert MARKWORTH a Parmajit SANDHU. Computer Security SS3: Biometric Authentication. *University of Birmingham* [online]. 2004 [cit. 2013-07-09]. Dostupné z: http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS3/

[19] KRHOVJÁK, Jan a Václav MATYÁŠ. Autentizace a identifikace uživatelů. *Zpravodaj ÚVT MU*. 2007, XVIII, č. 1.

[20] MIGNARD, David. Qwerty en azerty. *Informatique astuces* [online]. 2013 [cit. 2013-07-09]. Dostupné z: http://www.informatique-astuces.com/qwerty-en-azerty/

[21] MONROSE, Fabian a Aviel D. RUBIN. Keystroke dynamics as a biometric for authentication. *Future Generation Computer Systems*. 2000, č. 16.

[22] *National Human Genome Research Institute: Deoxyribonucleic Acid (DNA)* [online]. 2012 [cit. 2013-07-08]. Dostupné z: http://www.genome.gov/25520880

[23] PATO, Joseph N a Lynette I MILLETT. *Biometric recognition: challenges and opportunities*. Washington, D.C.: National Academies Press, c2010, xv, 165 p. ISBN 978-0-309-14207-6.

[24] Performance of biometrics. *Biometric-Solutions.com* [online]. 2010 [cit. 2013-06-04]. Dostupné z: http://www.biometric-solutions.com/index.php?story=performance_biometrics

[25] Personal Identification Number - PIN. *Investopedia* [online]. 2013 [cit. 2013-06-05]. Dostupné z: http://www.investopedia.com/terms/p/personal-identification-number.asp

[26] RAK, Roman. *Biometrie a identita člověka ve forenzních a komerčních aplikacích*. 1. vyd. Praha: Grada, 2008, 631 s., 32 s. barev. obr. příl. ISBN 978-80-247-2365-5.

[27] RUNDHAUG, Fred Erlend. *Keystroke dynamics: Can attackers learn someone's typing characteristics*. Gjøvik University College, 2007. Dostupné z: http://brage.bibsys.no/hig/bitstream/URN:NBN:no-bibsys_brage_4240/1/Rundhaug%20-%20Keystroke%20dynamics%20-%20Can%20attackers%20learn%20someone's%20typing%20characteristics.pdf

[28] ŘÍHA, Petr a Luboš KLAŠKA. *Slovník počítačové informatiky a sítí* [online]. 2013 [cit. 2013-07-07]. Dostupné z: http://www.svetsiti.cz/

[29] Secure Password Generator. *Security Guide for Windows - Random Password Generator* [online]. 2013 [cit. 2013-07-11]. Dostupné z: http://www.pctools.com/guides/password

[30] STEFANOV, Emil. Neural Networks. *EmilStefanov.net* [online]. 2013 [cit. 2013-07-10]. Dostupné z: http://www.emilstefanov.net/Projects/NeuralNetworks.aspx

[31] STERGIOU, Christos a Dimitrios SIGANOS. Neural Networks. *Imperial College London: Department of Computing* [online]. 1996 [cit. 2014-07-10]. Dostupné z: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html