Sem vložte zadání Vaší práce.

České vysoké učení technické v Praze
Fakulta informačních technologií
Katedra teoretické informatiky

Bakalářská práce

Detekce phishingových zpráv

Tomáš Duda

Vedoucí práce: doc. RNDr. Ing. Marcel Jiřina, Ph.D.

10. května 2015

Poděkování

Rád bych na tomto místě poděkoval doc. Marcelu Jiřinovi za mnoho užitečných rad a trpělivost při vedení bakalářské práce a dále svým rodičům za podporu v průběhu celého studia.

Prohlášení

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

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 10. května 2015 .....

Odkaz na tuto práci

Duda, Tomáš. *Detekce phishingových zpráv*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2015.

Abstrakt

Tato bakalářská práce se zabývá detekcí phishingových zpráv v českém a anglickém jazyce. Popsány jsou charakteristické znaky phishingových zpráv a existující řešení, které phishing potírají. Na základě zjištěných informací je navržen algoritmus detekce phishingových e-mailů, jenž je implementován v Javě. Uvedené řešení je otestováno na česky i anglicky psaných zprávách.

Klíčová slova detekce a filtrování phishingových zpráv, data mining, text mining, strojové učení, klasifikace

Abstract

This bachelor thesis deals with the detection of phishing messages written in Czech or English language. Common features of phishing e-mails and present countermeasures are described. Based on gained information, an algorithm for phishing detection is proposed and implemented in Java. The algorithm is tested on messages written in Czech and English language.

Keywords detection and filtering of phishing messages, data mining, text mining, machine learning, classification

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Mezi ohromným množstvím nevyžádaných zpráv, které zaplavují e-mailové schránky uživatelů, se setkáváme s jedním specifickým typem, jenž je obzvlášť nebezpečný. Jde o phishing, tedy podvodnou techniku, jejímž cílem je od uživatele získat osobní údaje, které může útočník zpeněžit nebo využít k neoprávněnému přístupu k uživatelově kontu.

Na rozdíl od běžné nevyžádané pošty tedy hrozí přímá ztráta finančních prostředků, což činí tuto techniku mimořádně nebezpečnou. S rostoucí sofistikovaností těchto útoků vzniká potřeba se aktivně bránit a místo pouhé úpravy existujících spamových filtrů vyvíjet metody, které se na phishing zaměřují a cíleně ho potírají.

Tato práce si dává za cíl shrnout aktuální informace o phishingových útocích a metodách obrany a na základě jejich analýzy implementovat vlastní nástroj v programovacím jazyku Java, jenž bude phishingové zprávy filtrovat od běžné pošty. Téma práce je inspirováno zadáním jednoho z projektů sdružení CESNET.

Členění práce

Bakalářskou práci jsem rozdělil do 6 kapitol. První kapitola uvádí čtenáře do prostředí phishingu, vymezuje klíčové pojmy a předkládá příklad phishingového útoku.

Druhá kapitola obsahuje výčet technik, jež jsou v současné době využívané v boji s tímto problémem. Důraz je kladen na strategie filtrování phishingových zpráv, neboť z nich bude vycházet vlastní řešení.

Ve třetí kapitole se zaobírám technickou stránkou řešení. Popisuji způsob abstrahování informací z přijatých zpráv, metody pro výběr měřených vlastností, jež poskytnou maximum informací algoritmu, který bude posuzovat, o jaký typ zprávy se jedná. Rovněž na základě analýzy získaných dat navrhuji

nové příznaky a popisuji metodiku pro detekci česky psaných phishingových zpráv.

Čtvrtá kapitola pojednává o strojovém učení, algoritmech využitelných pro binární klasifikaci a metodikou testování uvedených řešení.

Pátá kapitola popisuje postup implementace vlastního řešení a navrženou architekturu aplikace. Rovněž jsou popsány využité knihovny a rozhraní, které výsledná aplikace poskytuje. Dále se v ní zabývám popisem shromážděných e-mailů, jež slouží jako vstupní data pro strojové učení.

V šesté kapitole popisuji výběr vhodné podmnožiny příznaků pro trénování klasifikačního modelu, ladění parametrů algoritmů strojového učení a úspěšnost detekce jak anglicky, tak i česky psaných phishingových zpráv. Hodnotím dosažené výsledky a v závěru popisuji možnosti dalšího vývoje.

Kapitola 1

Vymezení základních pojmů

V této kapitole pojednávám o klíčovém termínu práce, kterým je phishing a phishingový útok. Speciálně se zaměřím na vymezení phishingového e-mailu, aby bylo jasné, jaké zprávy by měl cílový program filtrovat. Rovněž představím příklad útoku a pro úplnost pohovořím o dalších specifických typech phishingu, které se buď od typického scénáře útoku odlišují, nebo probíhají přes jiné komunikační kanály, než je e-mail.

1.1 Phishing

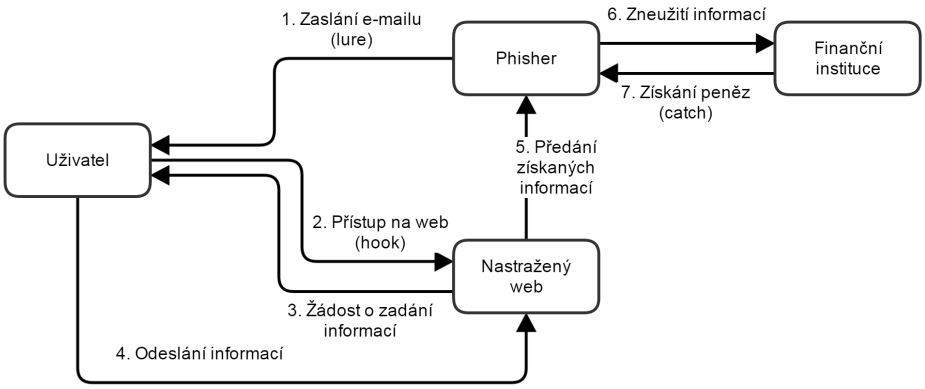
Phishing je forma sociálního inženýrství, při které se útočník, někdy označován jako phisher, pokouší podvodně získat uživatelovy tajné nebo citlivé údaje, [1]. Vydává se přitom za zástupce důvěryhodné společnosti. Typické je využití automatického rozesílání e-mailových zpráv, které odkazují příjemce na podvržené stránky známých společností, na kterých jsou umístěny formuláře, jež jsou využity ke shromažďování citlivých informací, které do nich oběť zadá. Příkladem takových informací jsou například hesla, čísla kreditních karet nebo národní identifikační čísla. V českém prostředí jde třeba o číslo občanského průkazu nebo rodné číslo.

Samotný pojem phishing je potom odvozeninou anglického slova fishing[1], jenž poukazuje na fakt, že útočník se snaží vylákat osobní informace pomocí návnady, která má podobu důvěryhodně vypadající zprávy a webové stránky.

1.2 Fáze phishingového útoku

Běžný phishingový útok má tři klíčové komponenty, které se v citované literatuře označují jako lure, hook a catch[2], [1].

[1] V překladu rybaření. Záměna písmen 'f' za 'ph' vychází z psaného dialektu hackerů.
[2] V překladu návnada, háček a úlovek.



Obrázek 1.1: Grafické znázornění průběhu phishingového útoku, [1].

Lure je typicky rozeslání množství zpráv do schránek uživatelů elektronické pošty. Tyto zprávy svým formátováním kopírují podobu e-mailu společnosti, za kterou se phisher v rámci útoku vydává. Zpráva obvykle obsahuje text, jehož obsah má čtenáře přimět kliknout na uvedený odkaz, jenž směřuje na webovou stránku rovněž kontrolovanou phisherem. S lure se nejvíce pojí dříve zmíněný pojem sociálního inženýrství, což je v daném kontextu označení pro manipulativní techniku, při které útočník uvádí důvody, proč by měl čtenář přejít na uvedený odkaz. Mezi nejběžněji uváděné důvody patří následující:

- Žádost o doplnění osobních údajů v účtu, které by měli sloužit pro zvýšení bezpečnosti skrz lepší autentizaci uživatelů.
- Žádost o doplnění chybějících údajů v osobním profilu, či jejich aktualizaci.
- Nabídka účasti v soutěži nebo možnost získání určité výhody, což je podmíněno vyplněním dotazníku, nebo poskytnutím určitých údajů.
- Falešné oznámení o proběhnuté aktualizaci uživatelova konta, u kterého je uvedeno upozornění, že pokud nebyla daná aktualizace iniciována uživatelem, měl by se na poskytnutém odkazu přihlásit ke svému účtu a ohlásit jeho zneužití.

Samotné rozesílání zpráv je většinou prováděno pomocí botnetů, což jsou sítě počítačů připojených k internetu, jež jsou nakaženy malwarem, který je umožní útočníkům ovládat na dálku a zneužít pro zmíněnou kriminální činnost.

Jako hook je označována samotná webová stránka, na kterou potenciální oběť přechází v druhé fázi útoku. Podobně jako zaslaná zpráva i webová

stránka přesně kopíruje vzhled reálné stránky napadené instituce. Je zde umístěn formulář s políčky určenými pro vyplnění osobních údajů.

Pokud oběť požadované informace vyplní a odešle, přesouváme se k třetí a finální části nazývané catch. V této fázi útočník zneužije získané informace, typicky k neoprávněnému přístupu ke kontu uživatele nebo ke zpeněžení získaných údajů na černém trhu. Popsaný proces je graficky znázorněn na diagramu 1.1.

1.3 Phishingový e-mail

Abych mohl v dalších kapitolách práce přistoupit k popisu metod detekce phishingových zpráv, je potřeba zadefinovat pojem phishingového e-mailu, který není v literatuře zabývající se tématem jednotný.

V knize [1] autor popisuje phishingovou zprávu velmi obšírně, jednak přes využití technik sociálního inženýrství, jež jsou popsány v sekci 1.1 a výčtem následujících typických technických aspektů:

- Využití ochranných značek a log společností, za které se phisher vydává.
- Podvržení e-mailové adresy. To spočívá v úmyslné manipulaci s informacemi v hlavičce zprávy, což vede ke skrytí skutečného odesílatele.
- Manipulace s URL, jejíž cílem je odstínit čtenáře od informace, že odkaz umístěný v těle e-mailu nevede na doménu, kterou využívá společnost ve skutečnosti. Toho je dosaženo například pomocí příslušného formátování odkazů pomocí HTML a CSS nebo umísťováním jména kompromitované společnosti do části URL za TLD.

V publikaci jsou dále phishingové útoky rozděleny do šesti podkategorií, z nichž se s přítomností zpráv počítá u dvou (zbylé jsou zmíněny na konci kapitoly).

- Deceptive phishing. Jde o nejběžnější typ útoku, který začíná rozesláním e-mailu, jenž technikami sociálního inženýrství nabádá uživatele k určité akci, typicky navštívení poskytnutého odkazu, na němž se má přihlásit do svého účtu.
- Malware-based phishing. U tohoto typu obsahuje zaslaný e-mail přílohu, jenž obsahuje škodlivý kód, který umožní phisherovi získat z počítače citlivé informace. Může jít například o keyloggery, screenloggery nebo trojské koně.

Jiná definice kategorizuje útoky rozdílným způsobem na phishing a scam, [2]. Scam je zde popisován jako podvodná technika, při které útočník žádá přímo odeslání finanční částky na cizí bankovní účet, nebo sdělení osobních údajů přímo v odpovědi na e-mail. Naopak phishing je zde definován skrz přítomnost podvodné webové stránky.

Ve své práci se budu držet relativně přímé definice phishingové zprávy uvedené v publikaci [3], jenž uvádí tři požadavky:

- Při phishingu dochází ke zneužití známé značky. Útočník se tak snaží v uživateli vyvolat pocit, že jedná se zástupcem důvěryhodné společnosti.
- Vždy je vedle podvodného e-mailu přítomna webová stránka, na kterou se snaží útočník potenciální oběť nasměrovat.
- Útočník žádá o citlivé informace, typicky přihlašovací údaje k bankovním účtům nebo čísla platebních karet.

V kontextu prvních dvou definic se tedy budu zaměřovat na deceptive phishing, jehož klíčovou částí je odkaz v těle e-mailu, jenž směruje na podvodnou zprávu. Naopak nebude mým cílem detekovat malware-based phishing ani scam.

1.4 Příklad phishingového útoku

Jako typický případ phishingu uvedu útok, který byl v roce 2014 cílen na klienty České spořitelny, [4]. Potenciálním obětem přišel podvodný e-mail, jehož screenshot je na obrázku 1.2, ve kterém se phisher vydává za zástupkyni banky. Čtenáře informuje o blížícím se vypršení přístupu do internetového bankovnictví a požaduje, aby si uživatel na poskytnutém odkazu svůj účet aktualizoval. V uvedené terminologii tato zpráva představuje lure.

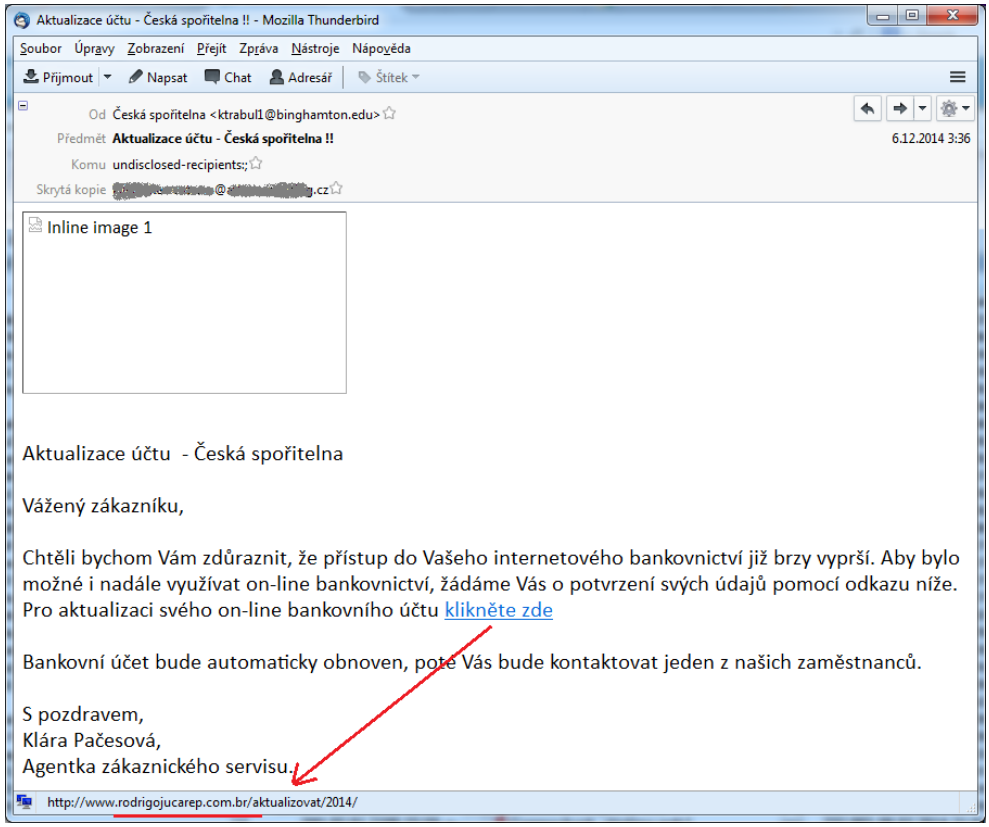
Když uživatel využil poskytnutý odkaz, dostal se na podvodnou webovou stránku, která vzhledem kopíruje skutečné stránky České spořitelny. Po vyplnění a odeslání přítomného přihlašovacího formuláře byla data předána útočníkovi.

1.4.1 Základní vlastnosti phishingového e-mailu

Z uvedeného příkladu lze odvodit základní rysy phishingové zprávy, [5].

- V zobrazené části hlavičky je vidět, že zpráva pochází z domény, která nemá s Českou spořitelnou nic společného[3]. Dochází zde tedy ke zneužití známé značky.
- Taktéž odkaz uvedený v těle e-mailu nevede na webovou stránku České spořitelny. Jde o kopii portálu internetového bankovnictví.
- Je využito sociálního inženýrství. Autor zprávy se snaží obsahem zprávy vzbudit ve čtenáři důvěru, jenž by ho vedla k poskytnutí citlivých údajů.

[3] Česká spořitelna využívá webovou stránku http://www.csas.cz, respektive http://www.servis24.cz pro internetové bankovnictví.



Obrázek 1.2: Příklad phishingového e-mailu v podobě, jak je zobrazen uživateli e-mailu. Převzato z [5].

1.5 Další typy phishingu

Pro úplnost zde uvádím výčet dalších typů méně obvyklých útoků, jenž se někdy řadí mezi phishing. V dalších kapitolách se jimi nebudu dále zabývat, jelikož jsou diametrálně odlišné od běžného phishingu popsaného výše a nebylo by možné vyvinout jednotný software, jenž by je dokázal postihnout.

- Spear phishing je označení pro phishingový útok, jenž je adresován na klienty konkrétní společnosti, někdy i na konkrétní osoby. Takové zprávy je náročné rozpoznat, jelikož rozdíl mezi nimi a běžnou korespondencí bývá ještě menší než u běžného phishingu, kde jsou zprávy psané generickým způsobem, aby zacílily na co největší počet příjemců, [6].

- Vishing, IM phishing a SMS phishing jsou názvy pro phishing provozovaný přes jiné komunikační kanály, jmenovitě hlasové služby, messengery nebo SMS zprávy, [3].

- DNS-Based phishing je založen na útocích na DNS služby, které slouží pro překlady mezi IP adresami a doménovými jmény. Jednou z možných forem je například napadení lokální DNS cache na počítači oběti, které může způsobit, že uživatel bude při zadání určité domény do prohlížeče odkázán na falešnou webovou stránku, [1].
- Content-injection phishing je jméno pro útok, při kterém útočník vloží podvodný obsah přímo do webové stránky společnosti skrz mezeru v zabezpečení. Mohou být využity například techniky cross-site scripting nebo SQL injection, [1].
- Man-in-the-middle phishing je založený na odposlouchávání komunikace třetí osobou a následném sbírání citlivých údajů. K tomuto jevu může docházet na nezabezpečených Wi-Fi hotspotech, kde se útočník snadno dostane k nešifrované komunikaci, [1].
- Při Search engine phishingu útočník vytvoří falešný e-shop, jenž optimalizuje tak, aby se po zaindexování dostal na přední místa ve vyhledávačích. Posléze sbírá osobní údaje uživatelů, kteří se na tomto webu pokusili odeslat objednávku, [1].

Kapitola 2

Současná řešení

Tato kapitola se zabývá popisem současných metod, které se nasazují v boji proti phishingovým útokům. Jednotlivé strategie jsou řazeny podle toho, v jaké části útoku bývají nasazovány. Toto řazení nemůže být dokonalé, jelikož některé techniky, například blacklisting, je možné nasadit ve více různých fázích. Největší důraz je vzhledem k cíli práce kladen na strategie aplikovatelné pro detekci phishingových e-mailů. Metody cílené na podvodné stránky nebo na analýzu komunikace mezi webovými servery uvádím, jelikož může jít o cenný zdroj technik, jež mohou mít potenciální využití i při detekci nebezpečných zpráv.

2.1 Systematické vzdělávání uživatelů

Systematické vzdělávání uživatelů, zejména zvyšování jejich povědomí o existenci phishingových útoků a pravidelné testování jejich znalostí je často zmiňovanou metodou, která sice nedokáže aktivně odstínit phishingové útoky, ale může snížit počet případných obětí. Toho je využíváno jednak některými společnostmi, jež vzdělávají své zaměstnance, aby zabránily škodám na svém majetku a dále banky, či e-shopy, které na danou problematiku upozorňují své klienty, [1][7].

Jako příklad komplexnějšího řešení uvádím dva projekty s názvy Anti-Phishing Phil a PhishGuru popsané v práci [8], které byly v roce 2008 sjednoceny do komerčního produktu Anti-Phishing Training Suite.

Aplikace tohoto produktu probíhá ve čtyřech krocích. Nejdříve jsou zaměstnancům zákazníka rozeslány simulované phishingové e-maily. Zaměstnanci, kteří na útok zareagují, jsou poté zařazeni do vzdělávacího programu, jenž je pomocí interaktivní aplikace informuje o nebezpečných přílohách, podvržených odkazech a dalších aspektech phishingu. Výsledky simulovaného útoku a systematického vzdělání jsou poté sumarizovány a případně dochází k dalšímu cyklu, [9].

Jak je zmíněno v publikaci [8], vzdělávání uživatelů může zabránit některým uživatelům padnout za oběť phishingovým útokům, jeho provádění je však komplikované, a proto je nutné zabývat se technikami, jenž útokům aktivně zabraňují.

2.2 Network level protection

Prvním okamžikem, při kterém můžeme zastavit phishingový útok, je komunikace cílového poštovního serveru s poštovním serverem, jenž provozuje phisher. Cílový server v rámci komunikace získá IP adresu a doménové jméno svého protějšku, které může ověřit proti databázím phisherů, [2]. Problematika blacklistingu a whitelistingu je blíže specifikovaná v sekci 2.4.2.

2.3 Techniky založené na zabezpečené komunikaci

Jde o metody, které se buď snaží o ustanovení bezpečné komunikace mezi čtenářem a odesílatelem zprávy, případně přímo mezi klientem a finanční institucí, u které hrozí vysoké nebezpečí, že se její klienti stanou obětí phisherů. Rovněž do této kategorie řadíme ochranu proti malware-based phishingu, která spočívá v instalaci antiviru, jenž může detekovat škodlivý software v přílohách e-mailů.

Jedním z příkladů je zavedení dvoucestné autentizace při provádění transakcí u velkého množství bank. Ta zaručuje, že i když jsou uživateli útočníkem zcizeny přihlašovací údaje od internetového bankovnictví, nejde o postačující množství informací potřebných k provedení finanční operace. Ta musí být stvrzena například pomocí TAN zaslaným SMS zprávou na číslo klienta, [10].

Dalším příkladem mohou být technologie Sender ID a DKIM, které umožňují příjemci ověřit, zda byla zpráva odeslána z domény, která je uvedena v hlavičce a zda nebyl obsah zprávy pozměněn v průběhu přenosu. V případě Sender ID k tomuto účelu slouží systém SPF, jenž umožňuje v rámci DNS záznamu uvést seznam hostů, kteří mají právo odeslat zprávu z dané domény. Příjemce zprávy poté ověří, zda je IP adresa odesílajícího MTA nacházající se v hlavičce e-mailu přítomná mezi adresami zapsanými v příslušném DNS záznamu. DKIM autorizuje zprávy obdobným způsobem, avšak odesílatele validuje pomocí digitálního podpisu, [1].

Uvedené techniky mohou sloužit k ověření identity odesílatele a tudíž k potlačení podvodných zpráv. Problémem uvedených technologií je však nutnost jejich implementace jak na straně odesílatele, tak i na straně příjemce, [11].

2.4 Client side tools

Mezi nástroje pracující na straně klienta řadíme prohlížeče, případně rozšíření prohlížečů, jež jsou schopné detekovat phishingové stránky. V současnosti im-

plementují určitou úroveň ochrany proti phishingu všechny populární prohlížeče, [12]. Typicky k detekci využívají metodu blacklistů a whitelistů. Existují i další rozšíření prohlížečů[4], které využívají pokročilejších metod, například analýzu URL nebo detekci phishingu na základě obsahu webové stránky. Výše zmíněné metody jsou podrobněji popsány v následujících třech sekcích.

Největší výhoda těchto řešení spočívá ve vysoké dostupnosti. Naopak nevýhodou je, že uživatelé nejsou od útoku odstíněni úplně, a jak prokázaly výzkumy, přes 10 % z nich varování prohlížečů ignoruje, [1]. Z tohoto důvodu novější verze prohlížečů místo drobných upozornění v adresním řádku zobrazí uživateli varování přes celou obrazovku, [14]. To zase může znepříjemňovat práci při falešných detekcích a potenciálně vést k vypnutí této služby uživatelem.

2.4.1 Detekce phishingových webů na základě URL

Jedním z článků, které se zabývají detekcí phishingových webů na základě URL, je [15]. Autoři rozdělují podvodné URL do čtyřech kategorií. Do první kategorie spadají odkazy ve formě IP adresy. Druhá je charakteristická tím, že obsahuje skutečnou doménu společnosti, za kterou se útočník vydává, uvedenou za TLD[5]. Třetí je podobná jako předchozí s tím rozdílem, že doména obětní společnosti je obsažena na začátku odkazu a jsou za ní připojena další jména[6]. U posledního typu URL buď jméno společnosti neobsahují vůbec, nebo ve zkomolené formě.

Na základě předchozích typů jsou následně modelovány příznaky, které jednak indikují kvalitu cílové domény a její stáří, důvěryhodnost domény, měřenou podle přítomnosti domény mezi nejnavštěvovanějšími adresami pro jednotlivá TLD, podobnost s jednotlivými kategoriemi podvodných linků z předchozí sekce a nakonec přítomnost klíčových slov, například confirm, account nebo secure, jež mají tendenci se v podvodných URL vyskytovat. Celkový počet extrahovaných příznaků v tomto řešení je 18.

Autoři následně zvolili kvůli požadavku na vysoký výkon klasifikaci pomocí logistické regrese. Vstupní množinu 2508 URL rozdělili náhodnou selekcí v poměru 66:34 na trénovací, respektive testovací množinu. Při měření dosáhli přesnosti 97,31 % s 1,2 % false-positive rate.

2.4.2 Blacklisting a whitelisting

Technika blacklistingu se opírá o kontrolu webových adres proti databázím phishingových webů. Tyto databáze mohou být plněny pomocí manuálního

[4] Jedním z nejznámějších je Netcraft, [13].
[5] Příkladem může být www.badsite.com/paypal.com/.
[6] Například www.paypal.com.badsite.com/.

nahlašování podvodných webů, honeypots[7] nebo webcrawlerů. Blacklisty jsou následně využity v prohlížečích a toolbarech, kde slouží k upozornění uživatele na nebezpečný obsah, [8]. Příkladem databáze phishingových webů je PhishTank, [16].

Výše zmíněná strategie je někdy zesílena použitím whitelistu, který opačně obsahuje seznam bezpečných stránek. Ten lze sestavovat například ze stránek, na které uživatel v průběhu používání prohlížeče úmyslně přistupuje. Cílem je snížit počet falešných detekcí nebezpečných webů.

Nevýhodou těchto metod je omezená účinnost vůči zero-day attacks, což je označení pro útoky využívající webové adresy, které ještě nejsou v databázích phishingových webů přítomny, [8].

2.4.3 Analýza obsahu phishingových webů

Další způsob, kterým lze získat informace použitelné ke klasifikaci phishingových webů, je analýza obsahu stránek. Zdroj [8] poskytuje popis dvou strategií při vytěžování informací z webů.

První metodou je systém GoldPhish prezentovaný v článku [17]. V něm probíhá analýza webu ve třech krocích. Nejdříve je pomocí interního prohlížeče vykreslena stránka ze zadané URL a uložen její snímek. V další fázi je pomocí technologie OCR rozeznán text obsažený v pořízeném snímku, který je ve finálním kroku vložen do vyhledávače Google. Pokud se mezi prvními čtyřmi výsledky nachází doména shodná s tou ve vstupní URL, není stránka označena jako phishingová. Na testovací množině 200 stránek, ze kterých bylo 100 phishingových, dosáhli autoři zdárné identifikace 98 % phishingových webů. Hlavní výhoda této metody tkví ve schopnosti pracovat s textem uloženým v logu společnosti, které je obvykle v horní části webové stránky[8]. Je navíc odolná vůči technice, kdy phisher celou stránku vyjma přihlašovacího formuláře zkopíruje v podobě obrázku. Nevýhodou je potom rychlost zpracování.

Druhá metoda využívá vlastností DOM[9] pro systematický průchod webové stránky a extrakci množiny slov, jež danou stránku identifikují. Řešení tedy sestává ze dvou částí. Nejdříve Identity extractor získá klíčová slova, jež extrahuje například z titulku stránky nebo meta-tagů. Posléze určí slova, jejichž frekvence vykazuje největší statisticky signifikantní odchylku od ostatních. Při testování potom byla porovnávána očekávaná identita[10] se zjištěnou. Celková přesnost řešení byla 84 % měřeno na 279 phishingových a 100 normálních stránkách, s false-positive rate 29 %. Vyšší přesnosti 95 % autoři dosáhli

[7] Název pro techniku, která spočívá ve vytvoření programu bežícím na serveru, jenž se snaží reagovat na příchozí komunikaci a ze získaných dat extrahovat informace o možných útocích. V kontextu phishingu jde o adresy podvodných stránek, [1].

[8] Autoři pořizují snímek fixní velikosti zahrnující pouze horní část webu.

[9] Document object model je rozhraní nezávislé na jazyku a platformě, jež umožňuje programům dynamicky přistupovat a obměňovat obsah a strukturu dokumentů, [18].

[10] Například pro www.paypal.com byla očekávána identita PayPal, stejně jako pro phishingový web, jenž se za oficiální doménu PayPalu vydává.

při druhém navrženém přístupu, kdy z DOM modelu extrahovali 10 příznaků a následně klasifikovali stránky pomocí klasifikátoru SVM, [19].

Existují podobná řešení, jež pro zjištění klíčových slov používají TF-IDF, což je statistická metoda pro výpočet důležitosti jednotlivých slov v dokumentu. Pro zautomatizování klasifikace navrhují podobně jako u analýzy odkazů využít Google Search a porovnat URL testované stránky s nejvýše umístěnými výsledky vyhledávače při dotazu na získaná klíčová slova, [8].

2.5 Techniky založené na analýze obsahu e-mailu

Tyto techniky pracující na straně poštovního serveru obvykle fungují na principu analýzy příchozích zpráv a jejich filtrování do kategorií bezpečné pošty a phishingových e-mailů. V případě dostatečné přesnosti je potom možné phishingové zprávy blokovat a ochránit tak uživatele před potenciálním nebezpečím, [8].

Zmíněná analýza obvykle probíhá ve dvou krocích. Nejdříve je z e-mailu extrahována množina příznaků, které nesou informaci, zda zpráva vykazuje typické prvky phishingové zprávy. Z této množiny příznaků je vytvořen charakteristický vektor, jenž je vstupem pro klasifikační model, který byl natrénován pomocí trénovací množiny složené z běžných i phishingových zpráv. Výstupem je poté klasifikační třída, která nás informuje, zda je příchozí e-mail bezpečný či nikoliv, [11].

Jelikož se většina publikovaných řešení liší zejména ve složení množiny extrahovaných příznaků, jejich počtu a použitém klasifikačním algoritmu, budu v následujících příkladech řešení dávat důraz zejména na tyto informace. Komplexnímu soupisu navržených příznaků, využitých klasifikátorů a jejich logiky jsou dále věnovány kapitoly 3 a 4.

2.5.1 Rozdíly mezi jednotlivými řešeními

Learning to Detect Phishing Emails byl jedním z prvních článků, který se zabýval aplikací strojového učení na detekci phishingových e-mailů, [20]. Navrhované řešení se nazývá PILFER. Extrahuje 10 příznaků, z nichž se většina týká odkazů v e-mailu. Měřen je například jejich počet, počet různých domén, na které odkazují, přítomnost URL ve formátu IP adresy[11] nebo přítomnost zfalšovaných odkazů[12]

Nejatypičtější extrahovanou informací je poté binární příznak týkající se stáří domény. Vychází z pozorování, které ukázalo, že domény využívané phishery jsou aktivní pouze v řádu dnů od jejich spuštění. Příznak je tedy pozitivní v případě, že e-mail obsahuje odkaz na doménu, jenž byla registrována

[11] Například http://192.168.0.1/paypal.cgi?fix_account.

[12] Jde o odkazy, které jsou pomocí HTML naformátované tak, aby budily dojem, že odkazují na skutečný web organizace.

v posledních 60 dnech, což je zjišťováno pomocí WHOIS dotazu. Za zmínění stojí i binární příznak, jenž využívá výstupu netrénovaného spam filtru SpamAssasin se základním nastavením. Pro klasifikaci je aplikován random forest klasifikátor. Na množině 6950 běžných a 860 phishingových e-mailů byla dosažena přesnost 99,5 %, s přesností detekce phishingu 96 %. Pro měření byla využita 10-fold cross validace.

Jiným příkladem řešení je technologie SmartScreen vyvinutá Microsoftem, jenž jako zdroj dat pro učící algoritmy používá zpětnou vazbu od uživatelů služby Hotmail a techniku honeypots. Technologie extrahuje přes 100000 příznaků, které reprezentují přítomnost specifických slov, podobu hlavičky nebo informace o reputaci odesílatele a využívá klasifikaci založenou na bayesovské statistice, [1]. Bližší informace o řešení, například jeho přesnost, nejsou dostupné.

Řešení popsané v článku [21] extrahuje ze zprávy 25 příznaků, které dělí do dvou kategorií. Příznaky z první kategorie mají za úkol charakterizovat stylistiku e-mailu. Jedná se tedy například o počet znaků, počet použitých slov, bohatost slovníku[13] nebo přítomnost některého z 18 klíčových slov. Druhá kategorie obsahuje dva binární strukturální příznaky. Ty spočívají v extrahování předmětu a oslovení ze zprávy, a zjištění, zda odpovídají vzoru těchto částí z trénovací množiny zpráv. Pro klasifikaci byl využit algoritmus SVM a na omezené testovací množině 200 zpráv autoři dosáhli přesnosti 100 %.

Velice robustní řešení je navrženo v publikaci [10]. Prezentován je přístup, který jako vstup pro strojové učení využívá 27 základních příznaků a 5 pokročilých příznaků. Základní příznaky jsou podobné jako v předchozích řešeních, za zmínku stojí skupina strukturálních příznaků, které zaznamenávají meta informace o podobě informací přenášených v e-mailu, jmenovitě počet MIME částí v těle zprávy a dále počet diskrétních, kompozitních a alternativních MIME částí. Hodnoty pokročilých příznaků jsou výstupem autonomních modelů:

- Prvním je statistický model, jenž analyzuje sémantiku zprávy pomocí hledání shluků slov, u kterých je pravděpodobné, že se budou nacházet častěji ve třídě phishingových, respektive běžných e-mailů. Výstupem je číselná hodnota, jenž charakterizuje sémantiku příchozí zprávy.

- Ve druhém případě jde o skupinu modelů, jež jsou založeny na Markovových dynamických řetězcích. Pro třídu phishingových a běžných e-mailů jsou natrénovány modely, které s příchozím e-mailem pracují jako s řetězcem bitů z neznámého zdroje. Pro každý e-mail určují pravděpodobnost, se kterou pochází ze zdroje phishingových nebo běžných zpráv. Výstupem těchto modelů jsou celkem 4 příznaky, z nichž 2 popisují pravděpodobnost původu zprávy z jednotlivých zdrojů a další 2 určují, do které třídy byla zpráva modely zařazena.

[13] Tu autoři počítají jako celkový počet slov děleno počtem znaků.

Pro výsledné vytvoření modelu využívají autoři klasifikátor SVM. Na shodné množině dat, na které byl testován PILFER bylo dosáhnuto f-measure 99,46 %. Nicméně v článku [11] je uveden názor, že cenou za vyšší přesnost jsou vysoké výpočetní a paměťové nároky.

V práci [12] autor analyzuje výkonnost různých klasifikátorů za účelem implementace filtrování phishingu v distribuovaném prostředí. Využívá přitom množinu 70 příznaků, z nichž 60 představuje frekvenci různých klíčových slov charakteristických pro phishing, jenž byly zjištěny analýzou několika tisíc podvodných e-mailů. Tyto hodnoty jsou vyjádřeny jako TF-IDF. Zbylých 10 příznaků se týká podoby odkazů, formátování e-mailu pomocí JavaScriptu, přítomnosti HTML tagu form a přítomnosti obrázků, jejichž zdrojem je jiný server, než ze kterého byla zpráva odeslána. Příznaky týkající se odkazů jsou prakticky shodné jako u systému PILFER. Mezi testovanými klasifikátory jsou SVM, random forest, neuronové sítě, naivní bayesovský, logistická regrese, klasifikace pomocí regresních stromů a klasifikace metodou BART[14], u nichž autor prezentuje potřebné modifikace, která umožní využít regresní metodu ke klasifikaci[15]. Na množině 6561 e-mailů využitých k trénování a testování dosáhl největší přesnosti model vytvořený klasifikátorem CBART, měřeno metodou AUC (99,19 %). Je však nutno dodat, že výsledky ostatních klasifikátorů se lišily v řádu jednotek procent.

S návrhem nových příznaků přichází řešení navržené v publikaci [22]. Navrhované příznaky dělí do dvou kategorií na off-line a online. Off-line příznaky mohou být extrahovány přímo z e-mailu bez nutnosti volání vzdálených služeb. Patří mezi ně například binární příznak, který je pozitivní v případě, že je ve zprávě link obsahující jiné než ASCII znaky. Další navržený příznak vychází z pozorování, že přes 60 % phishingových zpráv odkazuje na servery umístěné ve dvou zemích. Autoři proto do řešení zanáší celkem 51 příznaků, které charakterizují počet IP adres přítomných v e-mailu, které patří mezi rozsahy adres přidělených každé z 50 zemí. Poslední z příznaků slouží pro počet IP adres patřící k zemi, která není mezi 50 sledovanými. V článku není uvedeno, kterých 50 zemí bylo zahrnuto mezi sledované. Extrahování online příznaků zahrnuje stahování dokumentů ze zdrojů uvedených ve zprávě, případně spolupráci s vyhledávači. Autoři navrhují například zaznamenávat úroveň zabezpečení webových stránek, na něž je ve zprávě uvedena adresa nebo analyzovat výsledky vyhledávačů při zadání jednotlivých domén uvedených v e-mailu. Na základě měření informačního zisku pro jednotlivé příznaky autoři vybrali výslednou množinu 30 příznaků a aplikovali klasifikátor SVM. V rámci testování dosáhli přesnosti 99,5 % s 0,2 % false-positive rate. Testovací množina obsahovala 2000 zpráv, z nichž 1000 bylo phishingových.

[14] Bayesian Additive Regression Trees.
[15] Tuto metodu dále označuje jako CBART.

2.5.2 Výhody a nevýhody

Jednou z největších výhod detekce phishingu pomocí filtrování e-mailů je přítomnost největšího množství informací na jednom místě. Z hlavičky e-mailu máme možnost získat informace o směrování zprávy[16], text phishingového e-mailu i odkaz na podvodnou stránku, [20]. Další výhodou je flexibilita řešení. Při změně strategie phisherů je možné promptně změnit množinu extrahovaných příznaků nebo pomocí množiny nových trénovacích zpráv vytvořit nový klasifikační model. Filtrování e-mailů je navíc považováno za nejlepší volbu v boji proti zero-day phishingovým útokům, [11].

Nevýhodou může být požadavek na větší rychlost klasifikace než u nástrojů na detekci phishingových webů, jež mohou využít výpočetní výkon na straně uživatele a není u nich hrozba nutnosti zpracovávat velký počet požadavků v omezeném čase jako na poštovním serveru. Problém tudíž může vyvstat v případě využití příznaků, jež jsou závislé na informaci, kterou je nutné získat ze vzdáleného serveru nebo je nutné spouštět výpočetně náročnější algoritmus.

[16] Avšak tyto informace je nutné brát s rezervou, neboť mohou být zfalšované, [2].

Kapitola 3

Předzpracování dat a vektor příznaků

Jelikož ve většině případů není možné v algoritmech strojového učení pracovat přímo se vstupními daty, je potřeba tato vstupní data převést do více abstraktní formy, kterou nazýváme vektor příznaků. Vektor příznaků obsahuje konečný počet hodnot, které vstupní objekt popisují. V případě phishingových e-mailů půjde například o výskyt některých klíčových slov nebo způsob formátování příchozí zprávy, [23].

Tato kapitola popisuje různé kategorie příznaků a problematiku předzpracování dat, jmenovitě standardizaci příznaků a výběr jejich podmnožiny. V druhé části kapitoly je uveden přehledný výčet příznaků, jež byly extrahovány z phishingových zpráv autory dosavadních řešení. Rovněž popisuji návrh nových příznaků a jejich případnou modifikaci pro účel detekce česky psaných zpráv.

3.1 Typy příznaků

Příznaky dělíme podle množiny hodnot, které mohou nabývat, do čtyřech kategorií, [23]:

- Nominální příznaky nabývají hodnoty konečné množiny konstant, jež pojmenovávají určitou kategorii. V praxi jde většinou o výčet několika číselných hodnot, jež jednotlivé kategorie reprezentují. Při analýze vstupních dat nás často zajímá modus tohoto příznaku, jenž reprezentuje kategorii, do které vstupní objekty nejčastěji spadají.

- Binární příznak je prakticky ekvivalent nominálního s omezením, které spočívá ve skutečnosti, že může nabývat pouze dvou rozdílných hodnot. Někdy se dělí do dvou podkategorií na symetrický a nesymetrický.

  - Nesymetrický binární příznak je takový, kde jedna z hodnot má mnohem vyšší důležitost než druhá.
  - U symetrického je význam obou hodnot rovnocenný.
- Ordinální příznak může nabývat hodnot, mezi kterými existuje uspořádání a zároveň neznáme rozsah těchto hodnot.
- Numerický příznak reprezentuje měřitelnou veličinu, která nabývá diskrétních nebo reálných hodnot.

3.2 Normalizace příznaků

Normalizace je jednou z nejdůležitějších technik transformace příznaků. Převádí hodnoty numerických příznaků na nový rozsah hodnot, typicky $[-1; 1]$ nebo $[0,0; 1,0]$. Jedním z důvodů pro použití této techniky je měření vzdálenosti mezi dvěma vektory u mnoha algoritmů strojového učení, kde mohou jednotlivé složky vektoru, jež se řádově liší, negativně ovlivnit výsledek, [24].

3.2.1 Min-max normalizace

V rámci min-max normalizace je provedena lineární transformace na vstupních datech. Uvažujme, že min_P a max_P jsou minimum, respektive maximum příznaku P. Potom je hodnota příznaku p_i zobrazena na novou hodnotu p'_i v intervalu $[new_{min}; new_{max}]$ spočtením

$$p'_i = \frac{p_i - min_P}{max_P - min_P}(new_{max} - new_{min}) + new_{min}. \tag{3.1}$$

3.2.2 z-score normalizace

Při z-score normalizaci jsou hodnoty příznaku P normalizovány na základě průměru a standardní odchylky P. Hodnota p_i příznaku P je normalizována na hodnotu p'_i pomocí

$$p'_i = \frac{p_i - \bar{P}}{\sigma_P}, \tag{3.2}$$

kde $\bar{P}$ a σ_P jsou průměr, respektive standardní odchylka hodnot příznaku P. Tuto metodu je vhodné použít, když předem neznáme minimum a maximum hodnot příznaku nebo jsou v množině vstupních dat přítomné instance, u nichž je hodnota tohoto příznaku výrazně odlišná od zbytku, [23].

3.3 Volba podmnožiny příznaků

Při návrhu množiny příznaků popisujících jednotlivé instance je nutné dbát na dvě základní kritéria. Příznaky by měly instance jednak popisovat co nejpřesněji, jinými slovy by se při převodu nezpracovaných dat na vektor příznaků

měla ztratit co nejmenší část obsažené informace. Na druhou stranu je potřeba se vyvarovat zahrnutí příznaků, které jsou nerelevantní, redundantní nebo by mohly přenášet do vektoru data, která jsou v původních instancích zašuměná a mohla by tudíž vést ke zbytečnému zkomplikování klasifikace, [23].

Obvykle není možné vybrat ideální podmnožinu příznaků, jelikož počet všech podmnožin množiny atributů odpovídá velikosti potenční množiny, jenž je $2^{|A|}$, kde A je původní množina příznaků. Existuje proto řada metod, které slouží k redukci dimensionality vstupních dat. Tyto techniky dělíme na wrapper metody, embedded metody a filter metody, [25].

3.3.1 Wrapper metody

Jde o heuristické metody, které využívají algoritmus strojového učení a na základě změřené chyby tohoto algoritmu na testovacích datech rozhodují, zda bude množina příznaků modifikována. Časté jsou dva typy strategií:

- Forward selection. Výchozí množina příznaků je prázdná. V každé iteraci je přidán příznak, který nejvíce zvýší přesnost klasifikace.

- Backward selection. Přistupuje k řešení obráceně než předchozí metoda. Ve výchozí množině jsou všechny příznaky a v každé iteraci je odebrán příznak, při jehož odebrání se nejvíce zvýší přesnost klasifikace.

Oba přístupy iterují, dokud není splněna ukončovací podmínka. Ta může spočívat například v nezlepšení výsledku klasifikace mezi dvěma iteracemi, [24].

3.3.2 Embedded metody

Jde o označení metod vestavěných přímo do učícího algoritmu. Redukce dimensionality tedy neprobíhá ve fázi předzpracování dat, ale v rámci učení. Příkladem mohou být rozhodovací stromy, což je název pro klasifikační metodu, při které je konstruována stromová struktura. Ve fázi učení je strom budován od kořene, přičemž je pro každý uzel zvolen takový atribut, který data co nejpřesněji rozdělí do výstupních tříd. Tento atribut je volen podle určitého kritéria, například informačního zisku. Ve výstupním stromu tudíž nedochází k rozhodování podle všech atributů, ale pouze podle těch, které mají na výsledek největší vliv. Tím se eliminuje riziko přeučení a zároveň dochází k implicitní selekci příznaků, [25].

3.3.3 Filter metody

Filter metody fungují nezávisle na algoritmu strojového učení. Jejich výstup je seznam příznaků z původní množiny seřazený podle významnosti. Výsledná podmnožina vzniká zvolením prvních n položek z tohoto seznamu. Mezi tyto

metody můžeme zahrnout například algoritmy založené na měření informačního zisku jednotlivých atributů, na poměru informačních zisků nebo χ^2 testu, [25].

3.4 Příznaky extrahované z phishingových e-mailů

Tato sekce obsahuje výčet příznaků, jež mohou být získány z phishingových zpráv. Návrhy příznaků čerpám z literatury popsané ve druhé kapitole. Pro přehlednost je dělím do čtyřech kategorií na příznaky popisující klíčová slova, odkazy v e-mailu, strukturální příznaky a příznaky získané pokročilejšími metodami.

U každého příznaku uvádím označení, popis, typ příznaku a zdroj, ve kterém byl tento příznak popsán.

3.4.1 Klíčová slova

Zanášejí informaci o přítomnosti slov, jež se ve phishingových zprávách nalézají častěji než v běžných zprávách. Někteří autoři ve svých pracích neuvádějí explicitní výčet hledaných klíčových slov, [12].

Příznaky sledující přítomnost klíčového slova jsou v dalším textu označeny předponou **kw** doplněnou o klíčové slovo, například pro slovo account **kwAccount**.

- Četnost klíčových slov account, access, bank, credit, click, identity, inconvenience, information, limited, log, minutes, password, recently, risk, social, security, service a suspended. Numerické příznaky, [21].
- Přítomnost klíčových slov account, update, confirm, verify, secur, notif, log, click, inconvenien (detekovány jsou v některých případech úmyslně jenom kořeny, aby nebyl příznak obejit ohnutým tvarem slova). Binární příznaky, [10].
- Frekvence klíčových slov měřená metodou TF-IDF. Autor volil sledovaná slova výběrem 60 nejfrekventovanějších slov v množině sesbíraných phishingových e-mailů. Numerické příznaky, [12].

3.4.2 Odkazy v e-mailu

Jelikož právě odkaz na podvodnou stránku je jedním z nejcharakterističtějších znaků phishingové zprávy, mnoho řešení na ně dává zvláštní důraz. Většinou je analyzováno formátování odkazu a podoba URL cílové stránky. Některé příznaky zanášejí i informace o stáří domén a podobě cílové stránky. To je ale spíše výjimečné, jelikož systémy závislé na komunikaci s externími zdroji informací mohou vykazovat nižší rychlost a menší spolehlivost, [22].

- **linkIP**. Přítomnost odkazu ve formátu IP adresy. Binární příznak, [10] [12][20][22].
- **linkWHOIS**. Přítomnost odkazu na doménu registrovanou před méně než 60 dny. Binární příznak, [20].
- **linkNonmatching**. Přítomnost podvodně naformátovaného linku. Text představující odkaz reprezentuje jinou URL než hodnota HTML atributu href. Binární příznak, [10][12][20][22].
- **linkKW1**. Přítomnost odkazu na nemodální doménu, jenž je reprezentován textem obsahujícím slova link, click nebo here. Jako modální označuje autor doménu, na níž odkazuje největší podmnožina odkazů v e-mailu. Binární příznak, [20].
- **linkKW2**. Přítomnost odkazu, jenž je reprezentován textem obsahující slova click, here, login nebo update. Binární příznak, [10].
- **linkCount**. Počet odkazů v e-mailu. Numerický příznak, [10][12][20][22].
- **linkDomainsCount**. Počet různých domén v odkazech v e-mailu. Numerický příznak, [20][22].
- **linkMaxDotsCount**. Počet teček v URL odkazu s nejvíce tečkami. Numerický příznak, [10][12][20][22].
- **linkInternalCount**. Počet interních odkazů. Jde o odkazy vedoucí na určité místo v e-mailu. Numerický příznak, [10].
- **linkExternalCount**. Počet externích odkazů. Jde o odkazy vedoucí na web. Numerický příznak, [10].
- **linkImages**. Počet odkazů reprezentovaných obrázkem. Numerický příznak, [10][22].
- **linkImageMaps**. Počet obrázků, které jsou využity jako klikací mapy. Příznak je měřen počtem využití HTML tagu map, jenž umožní definovat v obrázku oblasti, jež odkazují na různé URL. Numerický příznak, [22].
- **linkPort**. Odkaz vede na nestandardní port. Za nestandardní port se považuje port jiný než 80 nebo 443. Binární příznak, [12][22].
- **linkRedirect**. Přítomnost odkazu na web, který uživatele přesměruje na jinou doménu. Binární příznak, [12].
- **linkEncoding**. Přítomnost odkazu obsahujícího znaky kódované v ISO-latin kódování. Jde o detekce podřetězců v URL složených ze znaku procenta a dvou následujících znaků reprezentujících hexadecimální číslice. Binární příznak, [12][22].

- **linkAt**. Přítomnost znaku @ v některém z URL přítomných ve zprávě. Binární příznak, [22].
- **linkSenderDomain**. Doména odesílatele zprávy je rozdílná s modální doménou zprávy. Modální doména má zde stejný význam jako u příznaku linkKW1. Binární příznak, [26].
- **linkSSLSS**. Počet odkazů, které vedou na stránky, jež šifrují spojení pomocí self-signed certifikátů. Numerický příznak, [22].
- **linkReverseDNS**. Počet doménových jmen, pro které nebyl nalezen odpovídající reverzní DNS záznam. Numerický příznak, [22].
- **linkNonAscii**. Počet URL, jež obsahují znaky nespadající do ASCII znakové sady. Numerický příznak, [22].
- **linkTargetCountries**. Vnáší geolokační údaje o IP adresách přítomných v odkazech ve zprávě. V citovaném řešení jde o sadu 51 příznaků, které reprezentují 50 různých zemí (51. příznak slouží pro IP adresy, u nichž nemohla být země určena). Všechny tyto příznaky jsou numerické a reprezentují počet IP adres uvedených ve zprávě přidělených dané zemi, [22]. Autor v článku nezmiňuje, z jakého zdroje čerpá geolokační informace o IP adresách, jelikož ale uvádí, že k vyčíslení daných příznaků není nutné volání vzdálených zdrojů, budu ve spojitosti s těmito příznaky pracovat s volně dostupnou databází IP2Lite, která převádí IPv4 adresy na kódy zemí, [27].
- **linkStatusBar**. Počet webů, na něž je z textu zprávy odkazováno, které ve svém kódu obsahují pokus o manipulaci se stavovým řádkem prohlížeče prostřednictvím JavaScriptu. Numerický příznak, [22]. V současné době již nelze v nejpopulárnějších prohlížečích stavový řádek modifikovat [28], čímž tento příznak ztrácí smysl.

3.4.3 Strukturální příznaky

Zanášejí do vektoru příznaků informaci o formátování zprávy, použitém slovníku a stylu, jakým je e-mail napsaný.

- **structHTML**. E-mail formátovaný pomocí HTML. Příznak je pozitivní, pokud je v metainformacích zprávy detekován MIME typ text/html. Binární příznak, [10][22][20].
- **structJS**. E-mail obsahuje JavaScript. Binární příznak, [10][12][22][20].
- **structOnClickEvents**. Počet OnClickEvents definovaných ve vloženém a nalinkovaném JavaScript kódu. Numerický příznak, [22].

- **structLinkedJS**. Počet domén, z nichž je nalinkovaný JavaScript kód a zároveň se nevyskytují v dalších odkazech přítomných ve zprávě. Numerický příznak, [22].
- **structImages** Do e-mailu je vložen obrázek (detekce HTML tagu img v těle zprávy). Binární příznak, [12].
- **structScripting**. E-mail obsahuje skriptovací jazyk (detekce HTML tagu script v těle zprávy). Binární příznak, [10].
- **structForms**. E-mail obsahuje formuláře (detekce HTML tagu form v těle zprávy). Binární příznak, [10][12][22].
- **structCharCount**. Celkový počet znaků v textu e-mailu. Numerický příznak, [21].
- **structUniqueWords**. Celkový počet unikátních slov v textu e-mailu. Numerický příznak, [21].
- **structVocabRichness**. Bohatost slovníku měřená počtem slov děleno počtem znaků v textu e-mailu. Numerický příznak, [21].
- **structFunctWords**. Jde o poměr mezi počtem klíčových slov nalezených v textu e-mailu a celkovým počtem slov. Numerický příznak, [21].
- **structSubject**. Podobnost předmětu zprávy s předměty u phishingových e-mailů. Binární příznak, [21]. Autor článku neuvádí, jakou metodou podobnost měří, dále v práci s tímto příznakem tedy nepracuji.
- **structGreeting**. Podobnost oslovení ve zprávě s phishingovými e-maily. Binární příznak, [21]. Obdobně jako v případě předchozího příznaku autor neuvádí, jak podobnost měří.
- **structMIMECount**. Celkový počet MIME částí v těle zprávy. Numerický příznak, [10].
- **structMIMECompositeCount**. Celkový počet kompozitních MIME částí v těle zprávy. Jako kompozitní části jsou označeny všechny MIME části typu multipart/*. Numerický příznak, [10].
- **structMIMEDiscreteCount**. Celkový počet diskrétních MIME částí použitých v těle zprávy. Za diskrétní MIME část je považována jakákoliv část s typem jiným než multipart/*. Numerický příznak, [10].
- **structMIMEAlternativeCount**. Celkový počet alternativních MIME částí použitých v těle zprávy. Jako alternativní MIME část je označena část s typem multipart/alternative. Numerický příznak, [10].

- **structSubjectLength**. Počet znaků v předmětu zprávy. Numerický příznak, [22].
- **structSenderLength**. Počet znaků v uvedené adrese odesílatele. Numerický příznak, [22].
- **structMessageSize**. Udává velikost zprávy v bytech. Numerický příznak, [22].
- **structSigned**. Přítomnost digitálního podpisu ve zprávě. Binární příznak, [22].

3.4.4 Pokročilé příznaky

Tyto příznaky jsou specifické tím, že k získání jejich hodnot jsou nezbytné komplikovanější algoritmy, případně je nutné volat externí aplikaci.

- **advSpamFilter**. E-mail byl spamovým filtrem SpamAssassin označen jako spam. Binární příznak, [10][20][22].
- **advSpamScore**. Skóre, kterým SpamAssassin ohodnotil e-mail. Numerický příznak, [10].
- **advMarkov**. Výstup modelů postavených na Markovových řetězcích, jež měří, zda zpráva přišla ze zdroje běžných, respektive phishingových zpráv (měří se jednak výsledky klasifikace obou modelů a zároveň spočtené pravděpodobnosti). Binární a numerické příznaky, [10]. Blíže je metoda popsána v kapitole 2.
- **advCLTOM**. Výstup CLTOM modelu popisujícího sémantiku zprávy. Numerický příznak, [10]. Blíže je metoda popsána v kapitole 2.
- **advSearchSenderDomain**. Počet výsledků při zadání domény odesílatele do vyhledávače. Numerický příznak, [22].
- **advSearchLowDomain**. Nejnižší počet výsledků, který vyšel při zadávání jednotlivých domén uvedených ve zprávě do vyhledávače. Binární příznak, [22].
- **advSearchNotMatch**. Počet linků uvedených ve zprávě, u nichž se při zadání domény do vyhledávače neshoduje doména s doménou ani jednoho z prvních 4 výsledků. Numerický příznak, [20].
- **advSearchKeywords**. Pro spočtení tohoto příznaku je využit Classifier4J software, který vytvoří na základě analýzy obsahu zprávy čtyři klíčová slova. Ta jsou následovně jednotlivě i naráz odeslána do vyhledávače. Hodnota příznaku je počet odkazů v e-mailu, které neležely na žádné z deseti nejvýše umístěných domén ani v jednom z pěti dotazů na vyhledávač. Numerický příznak, [22].

Klíčové slovo	Běžné zprávy	Phishing
ebay	0,6 %	22,2 %
paypal	0,3 %	31,9 %
protect	4,5 %	45,3 %
fraud	0,6 %	31,0 %

Tabulka 3.1: Naměřená relativní četnost výskytu navržených klíčových slov.

3.4.5 Návrh nových příznaků

V této sekci budu prezentovat návrhy na další příznaky, jež by mohly být užitečné při popisu phishingové zprávy. Příznaky jsem vytvořil na základě analýzy obsahu běžných a phishingových e-mailů. Zdroj a počet těchto zpráv je blíže popsán v kapitole 5.

První skupina příznaků se týká klíčových slov a vychází z analýzy nejfrekventovanějších slov ve phishingových e-mailech. Jde o slova ebay, paypal, protect a fraud. V případě prvních dvou slov jde o názvy společností, na které cílí výrazná část phishingových útoků. Při použití těchto klíčových slov sice vzniká riziko falešné detekce e-mailů zaslaných skutečně těmito společnostmi, nicméně v kombinaci s dalšími příznaky by mohla přítomnost uvedených slov znamenat užitečnou informaci pro algoritmy strojového učení. Klíčová slova protect a fraud jsou spojena s jednou z častých technik sociálního inženýrství, kdy útočník tvrdí příjemci zprávy, že došlo ke kompromitaci jeho účtu a tudíž je nutné obnovit přihlašovací údaje. Relativní četnost navržených slov, kterou jsem naměřil v běžných a phishingových zprávách, je uvedena v tabulce 3.1. Ve všech případech se jedná o binární příznaky.

Další příznak se týká odkazů ve zprávách a je inspirovaný metodou analýzy odkazů v článku [15]. Spočívá v detekci URL, jež obsahují více než jeden podřetězec, jenž je kombinací doménového jména a TLD. Phisheři této techniky využívají pro zmatení uživatele, který si, když vidí, že skutečná doména společnosti je součástí odkazu, může myslet, že vstupuje na bezpečný web, ačkoliv se jedná o phishingový útok. Jde o binární příznak, který označuji newMoreDomains.

Poslední tři příznaky se týkají anomálií objevených ve formátování zpráv a informacích uváděných v metadatech zasílaných společně s textem zprávy. První spočívá v detekci MIME části typu multipart/alternative, která však obsahuje pouze jedinou podčást, což není u běžné pošty obvyklé. Typ multipart/alternative slouží k označení souboru částí zprávy s identickým obsahem, jež jsou naformátovány více způsoby, aby mohla být zpráva zobrazena například v emailovém klientu, který neumí vykreslovat HTML (častá kombinace typů zasílaných v multipart/alternative je text/plain a text/html). Tento binární příznak označuji názvem newSingleAlternative. Další dva příznaky reprezentují přítomnost nekorektně vyplněného pole Content-Type, respek-

Příznak	Běžné zprávy	Phishing
newMoreDomains	3,8 %	57,0 %
newSingleAlternative	0,0 %	31,8 %
newUnknownMime	0,0 %	1,3 %
newDamagedEncoding	0,0 %	0,4 %

Tabulka 3.2: Naměřená relativní četnost pozitivního výskytu navržených příznaků.

tive charset v metadatech uvedených u některé z MIME částí e-mailu. Za nekorektně vyplněné pole považuji takové, jež nesplňuje požadavky uvedené v RFC 2045, [29]. Jedná se o dva binární příznaky, které dále nazývám newUnknownMime a newDamagedEncoding. Naměřenou relativní četnost pozitivního výskytu výše popsaných příznaků příznaků u běžných a phishingových zpráv uvádím v tabulce 3.2.

3.4.6 Příznaky sledované v česky psaných zprávách

Pro účely detekce česky psaných phishingových zpráv je nezbytné upravit příznaky, které jsou závislé na jazyku zprávy. To se týká zejména volby vhodných klíčových slov, pro jejichž zjištění je nutné provést analýzu česky psaných zpráv a získat tak nejčetnější slova.

Vzhledem k nízkému počtu dostupných česky psaných phishingových zpráv v plné podobě, tedy včetně metadat a informací o formátování, nebylo možné pracovat s příznaky sledujícími strukturu zprávy nebo obsažené odkazy, jelikož bych postrádal dostatek dat k natrénování modelu. Rozhodl jsem se proto přistoupit k tvorbě množiny příznaků, které sledují rozdíly ve slovníku běžných a phishingových zpráv. K tomu účelu jsem využil strategii uvedenou v publikaci [12].

Tato strategie spočívá v konstrukci příznaků vytvořením rejstříku použitých slov v textech phishingových e-mailů, odstranění stopslov[17] a následné aplikaci metriky TF-IDF, která sleduje relevanci slov v rejstříku dokumentů. Tato metrika sestává ze dvou složek. První je četnost sledovaného slova:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}, \tag{3.3}$$

kde $n_{i,j}$ je počet výskytů slova t_i v dokumentu d_j. Jmenovatel vyjadřuje počet slov v dokumentu. Druhá je převrácená četnost slova v rejstříku dokumentů:

$$idf_i = \log \frac{|D|}{|\{j : t_i \in d_j\}|}, \tag{3.4}$$

[17] Jde o slova, které nenesou informaci, například spojky nebo předložky. V anglicky psané literatuře se seznam těchto slov označuje jako stopword list.

Sledované slovo	Četnost
účet	42,7 %
odkaz	24,4 %
kliknout	20,7 %
internetový	19,5 %
servis	19,5 %
spořitelna	18,3 %
bezpečnostní	18,3 %
bankovnictví	18,3 %
informace	17,1 %
potvrzení	17,1 %
aktualizovat	17,1 %
karta	15,9 %
přístup	14,6 %
aktualizace	13,4 %
seznam	13,4 %
potvrdit	12,2 %
adresa	12,2 %
obnovit	12,2 %
transakce	9,8 %
banka	8,5 %
platební	7,3 %
kreditní	4,9 %

Tabulka 3.3: Naměřená relativní četnost sledovaných slov v textech česky psaných phishingových e-mailů.

kde D je množina všech dokumentů a jmenovatel vyjadřuje počet dokumentů d_j, ve kterých se nachází sledované slovo t_i. V kontextu mé práce je množina D složena z textů phishingových zpráv trénovací množině. Výsledná hodnota TF-IDF pro dané slovo je výsledkem součinu těchto dvou složek:

$$tf\text{-}idf_{i,j} = tf_{i,j} \cdot idf_i. \tag{3.5}$$

V tabulce 3.3 jsou uvedena vybraná slova a jejich relativní četnost v získaném souboru textů českých phishingových zpráv. Při výběru slov byla na texty zpráv aplikována tokenizace, která spočívá v rozdělení textu na jednotlivá slova a následně lemmatizace, jež slouží k převodu slov na základní tvary. Z textů byla rovněž odstraněna slova kratší než 3 znaky, které typicky nemají žádnou informační hodnotu, protože jde ve většině případů o spojky nebo předložky. Z výsledné množiny slov byla vybraná ta s vysokou četností a se souvislostí s obvyklou strategií phishingových útoků. Pro lemmatizaci byla použita knihovna JLemmaGen, [30].

Výsledný vektor příznaků pro česky psanou zprávu tedy obsahuje 22 hodnot, z nichž každá reprezentuje hodnotu TF-IDF pro jedno klíčové slovo v rámci textu dané zprávy.

Kapitola 4

Strojové učení

Strojové učení je disciplínou informatiky, jež se zabývá systémy, které dokáží zlepšovat svoji přesnost na základě poskytnutých dat. Definice uvedená v [31] říká, že počítačový program se učí ze zkušenosti E s respektem k třídě úkolů T a výkonnostní míře P, pokud jeho výkonnost P nad řešením úkolů T roste se zkušeností E. V kontextu mé práce je řešeným úkolem filtrování phishingových e-mailů od ostatní pošty, obecněji řečeno binární klasifikace, jenž je popsána v sekci 4.1. Zkušenost E je poskytnuta prostřednictvím trénovacích dat, jejichž popis je obsažen v kapitole 5. Výkonnost P je měřena přesností klasifikace, pro níž existuje několik metrik, kterými se zabývá sekce 4.3.

Oblast strojového učení můžeme rozdělit do dvou kategorií. První z nich je strojové učení bez učitele (unsupervised learning), při kterém předem neznáme počet ani význam tříd, do kterých chceme vstupní data třídit. Druhou je potom strojové učení s učitelem (supervised learning), v níž tuto informaci o počtu a významu tříd máme, [23].

4.1 Binární klasifikace

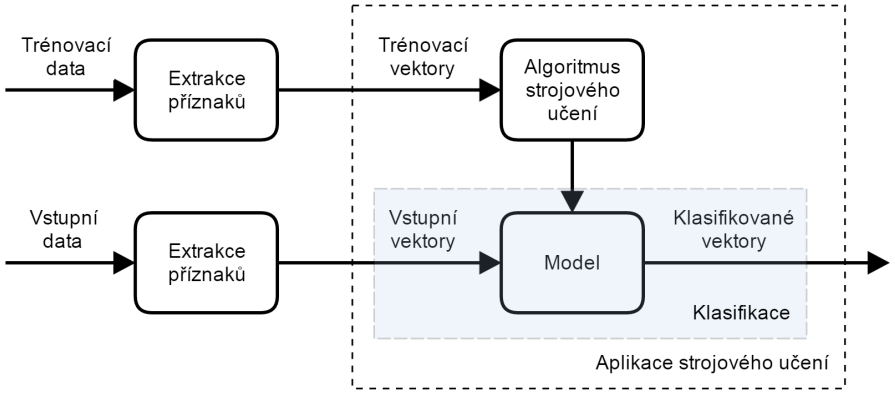
Binární klasifikace je jednou ze základních úloh strojového učení s učitelem. Probíhá ve dvou krocích, [23].

V učící fázi (learning step) je za pomocí klasifikačního algoritmu a trénovací množiny dat vytvořen klasifikační model. Trénovací množina je sestavena z vektorů příznaků, kterým je přiřazena třída učitelem. Výsledný model by měl pracovat po vzoru mapovací funkce

$$f((x_1, x_2, ..., x_n)) = C_i, \tag{4.1}$$

která pro každý vstupní vektor příznaků určí jednu ze dvou tříd C_i, do které daná instance spadá. Jednotlivým klasifikačním algoritmům je věnována podkapitola 4.2.

Ve druhé fázi využijeme model k samotné klasifikaci. Před ostrým nasazením je nezbytné vytvořený model otestovat. Za tímto účelem je potřeba vytvo-



Obrázek 4.1: Diagram znázorňující aplikaci strojového učení, [32].

řit testovací množinu instancí, které mají rovněž určenou třídu učitelem. Testovací množina by měla být disjunktní s trénovací množinou, abychom mohli odhalit případný problém přeučení. K této negativní vlastnosti dochází, když je vytvořený model naučený zahrnovat do charakteristik tříd drobné anomálie specifické pro trénovací množinu, ne však obecně pro celou třídu.

V případě, že v rámci testování dosáhne model dostatečné přesnosti, je možné ho nasadit do provozu a využít ke klasifikaci nově příchozích dat, [32].

4.2 Klasifikační algoritmy

Pro tvorbu klasifikačních modelů existuje velké množství algoritmů. S respektem k omezenému rozsahu své bakalářské práce se budu v následujících sekcích zabývat popisem jednak dvou základních algoritmů, k-NN a Naïve Bayes, dále potom algoritmem SVM, který byl úspěšně aplikován v několika řešeních popsaných ve druhé kapitole, [10][21][22]. Popsané klasifikátory budou využity pro testování v následujících kapitolách.

Klasifikační algoritmy můžeme rozdělit do dvou základních kategorií, [23].

- U Lazy learning neprobíhá generalizace až do okamžiku, kdy dojde ke klasifikaci neznámé instance. Učící fáze tedy spočívá obvykle v pouhém uložení trénovacích dat bez dalšího zpracování. Výhodou těchto metod je jejich snadná úprava, při které se můžeme pokusit zlepšit přesnost klasifikace pouhým přidáním nových trénovacích instancí. Nevýhodou jsou vysoké výpočetní a prostorové nároky. Příkladem je algoritmus k-NN.
- Opakem je Eager learning, při kterém je v učící fázi z trénovacích dat odvozen generalizující model. Teprve po vytvoření tohoto modelu můžeme přistoupit ke klasifikaci neznámých instancí. Výhodou jsou menší nároky na výpočetní výkon a paměť, nevýhodou komplikovaná až nemožná

úprava výsledného modelu. Příkladem jsou algoritmy Naïve Bayes nebo SVM.

4.2.1 k-NN

Metoda k nejbližších sousedů je založena na porovnání klasifikovaného vektoru s nejpodobnějšími trénovacími vektory.

Každý trénovací vektor s n atributy představuje bod v n-dimenzionálním prostoru. Klasifikovaný vektor je porovnán s k trénovacími vektory, které jsou v daném prostoru nejblíže. Vzdálenost můžeme měřit různými metrikami. Jednou z nejobvyklejších je Euklidovská vzdálenost

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2}, \tag{4.2}$$

kde $X_1 = (x_{11}, x_{12}, ..., x_{1n})$ a $X_2 = (x_{21}, x_{22}, ..., x_{2n})$ jsou instance, mezi kterými měříme vzdálenost. Klasifikovaný vektor je následně označen třídou, do které patří největší počet z k nejbližších sousedů, [23].

4.2.2 Naïve Bayes

Naïve Bayes je příkladem statistického klasifikátoru, jenž pracuje na základě určení pravděpodobností, že klasifikovaná instance přísluší k jednotlivým třídám.

Samotná klasifikace probíhá následovně:

1. Označme jako D množinu n-rozměrných trénovacích vektorů s označenou třídou.

2. Máme m tříd $C_1, C_2, ..., C_m$. Klasifikovanému vektoru $X = (x_1, x_2, ..., x_n)$ bude přidělena třída C_i s maximální aposteriorní pravděpodobností. Bude tedy platit

$$P(C_i|X) > P(C_j|X) \qquad \text{pro } 1 \leq j \leq m,\ j \neq i. \tag{4.3}$$

   Dále se výpočet řídí Bayesovou větou

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \qquad \text{pokud } P(X) > 0, \tag{4.4}$$

   která popisuje výpočet aposteriorní pravděpodobnosti.

3. Jelikož $P(X)$ je pro všechny třídy konstantní, je potřeba maximalizovat pouze $P(X|C_i)P(C_i)$. Pokud nejsou pravděpodobnosti jednotlivých tříd známé, můžeme uvažovat $P(C_1) = P(C_2) = ... = P(C_m)$, případně můžeme vycházet z jejich podmíněného zastoupení v trénovací množině.

4. Pro snížení výpočetní složitosti se dále uvažuje, že jednotlivé atributy jsou nepodmíněné. Proto můžeme pro jednotlivé třídy určit

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i). \tag{4.5}$$

Pravděpodobnost $P(x_k|C_i)$ u nominálních příznaků spočítáme podle zastoupení hodnoty x_k v trénovacích vektorech označených třídou C_i. U numerických příznaků uvažujeme, že mají Gaussovo normální rozdělení definované vztahem

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{4.6}$$

Pravděpodobnost tudíž určíme s využitím odhadu střední hodnoty a standardní odchylky pomocí vztahu

$$P(X_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}). \tag{4.7}$$

5. Pro klasifikaci X je nutné určit $P(X|C_i)P(C_i)$ pro všechny třídy C_i. Vektor X je následovně označen třídou C_i, pro kterou platí

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \qquad \text{pro } 1 \leq j \leq m,\ j \neq i. \tag{4.8}$$

U klasifikátoru Naïve Bayes rozlišujeme jeho parametrickou a neparametrickou verzi. Zatímco u parametrické verze je funkce určující rozdělení hodnot příznaků zadaná fixně jako parametr algoritmu, u neparametrické probíhá v rámci učící fáze odhad těchto funkcí pro jednotlivé příznaky, [23].

4.2.3 Support Vector Machine

Jde o velmi účinnou metodu binární klasifikace, která staví na řešení dvou problémů. Prvním je převod úlohy na problém optimální lineární separace.

Ten spočívá v uvažování o testovacích vektorech jako o bodech v prostoru. V případě, že nemůžeme rozdělit tyto body nadrovinou takovým způsobem, že v každém z poloprostorů leží instance pouze jedné třídy, je nezbytné provést nelineární transformaci, která vektory převede do prostoru vyšší dimenze, kde separovatelné jsou.

V druhé části je provedena optimalizace, jejímž cílem je přiblížit se k maximum marginal hyperplane (nadrovina s maximálními okraji, MMH). Jde o nadrovinu, která nejlépe separuje jednotlivé třídy. Jinak řečeno, je maximalizována hodnota minima vzdáleností od nadroviny.

Název metody vychází z označení pro podpůrné vektory nejbližší nadrovině, které jsou nezbytné pro vytvoření finálního modelu. Jednou z největších výhod použití SVM je kvalitní generalizace trénovacích dat a tudíž nízká tendence k přeučení, [23].

4.3 Testování modelu

Před ostrým nasazením klasifikačního modelu je nezbytné provést jeho ohodnocení. Pro tento účel existuje řada různých přístupů, jejichž popisem se zabývá tato sekce.

4.3.1 Výběr testovacích dat

Jak již bylo zmíněno na začátku kapitoly, je nezbytné, aby byl model testován na množině dat, jenž je rozdílná s množinou trénovací. Jednou z možností je tedy náhodným výběrem vstupní data rozdělit, jinou strategií je křížová validace, v rámci které jsou data rozdělena do n shodně velkých částí (folds). Přesnost modelu posléze testujeme n-krát, přičemž v každé iteraci použijeme rozdílnou část dat jako testovací množinu a všechny zbylé jako trénovací. Výsledná chyba se spočte podle vztahu

$$err = \frac{\sum_{i=1}^{n} err_i}{n}. \tag{4.9}$$

Zvláštním případem je one-leave-out křížová validace, kde je n rovno počtu instancí ve vstupních datech, což může vést ke zjemnění testů. Křížová validace slouží k testování rozdílných modelů, [33].

4.3.2 Měření úspěšnosti modelu

Instance z testovací množiny dělíme na pozitivní (positives) a negativní (negatives), podle třídy, kterou jsou označeny. V kontextu detekce phishingových e-mailů jsou pozitivní podvodné e-maily a negativní všechny ostatní.

V rámci testování používáme model ke klasifikaci testovacích instancí. Spočtenou třídu instance porovnáme s její skutečnou třídou, čímž nám vzniknou čtyři kategorie, [33].

- True positives (TP) označují správně klasifikované pozitivní instance (správně klasifikované phishingové zprávy).
- True negatives (TN) označují správně klasifikované negativní instance (správně klasifikované běžné zprávy).
- False positives (FP) označují špatně klasifikované negativní instance (špatně klasifikované běžné zprávy).

Tabulka 4.1: Matice záměn

		Klasifikováno jako:	
		Pozitivní	Negativní
Označeno jako:	Pozitivní	Počet TP	Počet FN
	Negativní	Počet FP	Počet TN

- False negatives (FN) označují špatně klasifikované pozitivní instance (špatně klasifikované phishingové zprávy).

Počty testovacích instancí spadajících do jednotlivých kategorií můžou být zapsány do matice záměn, jejíž podoba je znázorněna tabulkou 4.1.

Základní míra úspěšnosti modelu je přesnost (accuracy), která dává do poměru správně klasifikované instance vůči celkovému počtu instancí.

$$accuracy = \frac{TP + TN}{P + N} \tag{4.10}$$

Obdobně můžeme měřit poměr špatně klasifikovaných instancí.

$$error\ rate = \frac{FP + FN}{P + N} \tag{4.11}$$

Tyto míry ovšem mají omezenou vypovídající hodnotu, jelikož jsou závislé na poměru pozitivních a negativních instancí v testovacích datech. Větší informační hodnotu proto mají proto senzitivita (sensitivity, někdy také recall) a specificita (specificity), které měří poměr správně detekovaných pozitivních instancí, respektive negativních instancí vůči celkovému počtu pozitivních, respektive negativních instancí.

$$sensitivity = \frac{TP}{TP + FN} \tag{4.12}$$

$$specificity = \frac{TN}{TN + FP} \tag{4.13}$$

Řadí se k nim ještě třetí příbuzná míra, přesnost (precision), která porovnává počet správně klasifikovaných pozitivních instancí, vůči celkovému počtu instancí, jež byly klasifikovány pozitivně.

$$precision = \frac{TP}{TP + FP}. \tag{4.14}$$

Alternativní a často používaná míra je potom f-míra (f-measure), která dává do souvislosti přesnost a sensitivitu.

$$f\text{-}measure = \frac{2 * precision * sensitivity}{precision + sensitivity}. \tag{4.15}$$

4.3.3 Vizualizace přesnosti modelu

Přesnost modelu je možné vizualizovat pomocí ROC (receiver operating characteristic) křivky, jež znázorňuje vztah mezi množstvím TP a FP. Pro vynesení grafu je nutné znát pravděpodobnost, se kterou model udělil instanci danou třídu. Klasifikované instance seřadíme podle pravděpodobností a postupně považujeme prvních n instancí za pozitivní a zbylé za negativní, přičemž počítáme TP a FP, které vynášíme na křivku.

Plocha pod křivkou (area under curve, AUC), je potom další metrikou, která se používá pro zhodnocení výkonnosti modelu, [33].

Kapitola 5

Návrh softwarového řešení

V této kapitole popíši aspekty implementace vlastního řešení. Nejdříve popisuji postup řešení, tedy jednotlivé fáze, ve kterých probíhal vývoj aplikace. Dále se zmiňuji o použitém softwaru a knihovnách, jež jsou do řešení zahrnuty. Následující sekce popisují výslednou architekturu aplikace a data využitá pro trénování a testování aplikovaných algoritmů. Kapitola je završena popisem frontendu, který byl pro aplikaci vytvořen.

5.1 Strategie řešení

Pro zjednodušení práce jsem se rozhodl rozdělit implementační práce do níže uvedených celků.

V první fázi se budu zabývat vývojem části aplikace zodpovědné za zpracování příchozího e-mailu a jeho transformaci na vektor příznaků. Tato část práce je klíčová pro vytvoření datasetu, na kterém budu mít možnost testovat různé techniky předzpracování příznaků a výběru jejich nejvhodnější podmnožiny. Tato část aplikace je naprogramována s použitím knihoven jsoup a JavaMail.

V dalším kroku použiji software RapidMiner, pomocí kterého vyberu podmnožinu příznaků, která maximalizuje přesnost phishingového filtru. Rovněž vyberu vhodný klasifikační algoritmus.

V rámci třetí fáze omezím ve vyvíjené aplikaci počet extrahovaných příznaků s respektem k výsledkům z předchozího kroku a implementuji klasifikační model. Přesnost vytvořeného filtru otestuji na testovací množině zpráv.

Nakonec se zaměřím na modifikaci řešení pro český phishing. Za tímto účelem bude vytvořen druhý model detekující česká klíčová slova. Následně bude porovnána úspěšnost dvou vytvořených klasifikátorů na množině testovacích česky psaných phishingových zpráv. Důvod, který zamezil realizaci komplexnějšího modelu specializovaného na češtinu, je popsán v sekci 5.4.1 zabývající se zdrojem dat.

Výsledky a diskuse všech výše zmíněných testů jsou náplní kapitoly 6.

5.2 Použité knihovny a aplikace

V rámci realizace aplikace jsem se rozhodl využít existujících knihoven, které implementují dílčí části algoritmu. Knihovny byly voleny hlavně podle rozsahu nabídnutých funkcí. Rovněž jsem se rozhodl využít aplikaci RapidMiner, pro usnadnění volby techniky předzpracování dat a klasifikačního algoritmu.

5.2.1 JavaMail API

JavaMail je sada abstraktních API, jejichž primární funkcí je definice rozhraní podle platných standardů pro emailové aplikace. Oracle rovněž poskytuje open-source referenční implementaci těchto rozhraní. JavaMail API je licencováno pod CDDL a GPLv2.

Pro implementaci mého řešení je klíčová část aplikace určená pro parsování e-mailů ve standardních formátech. Knihovna tak umožňuje snadný přístup k informacím uvedeným v hlavičkách e-mailových zpráv, dokáže detekovat jednotlivé MIME části zpráv a předávat je společně s metainformacemi. Ve svém kódu používám verzi 1.5.1, [34].

5.2.2 jsoup

Jelikož většina v současnosti zasílané elektronické pošty je formátována s pomocí HTML a velká část příznaků popisovaných ve phishingových zprávách cílí na hyperlinkové odkazy či jiné prvky, jež jsou realizovány pomocí HTML, je vhodné využít parser, jenž dokáže tento kód rychle zpracovávat a extrahovat jednotlivé skupiny tagů a jejich atributy ze zprávy.

K tomu poslouží knihovna jsoup, která je implementovaná v Javě a podporuje extrakci prvků HTML z dokumentu. Největšími výhodami této knihovny je velká flexibilita v možnostech selekce různých částí dokumentu a schopnost pracovat i s nevalidním HTML. Knihovna jsoup je licencována pod MIT license. Ve svém řešení využívám knihovnu ve verzi 1.8.1, [35].

5.2.3 RapidMiner

RapidMiner je open-sourcový software pod licencí AGPL, který zprostředkovává sofistikované prostředí pro analýzu dat, zpracování dat a jejich následnou vizualizaci.

Jde o rozsáhlý nástroj založený na vizuálním programování, které spočívá v tvorbě procesů sestávajících se z jednotlivých bloků, které mají definovanou funkci a množinu přípustných vstupů a výstupů, [36]. Tento nástroj umožňuje komplexní ladění parametrů algoritmů strojového učení a předzpracování dat, což je hlavní důvod, proč jsem se rozhodl využít tento nástroj pro optimalizaci prezentovaného algoritmu.

5.3 Architektura aplikace

Aplikace pro filtrování zpráv sestává ze tří modulů. První modul provádí základní zpracování zprávy, které spočívá v převodu formátu EML na instanci třídy, jež obsahuje extrahovaná metadata, informace o struktuře zprávy a extrahovaný text, který je v e-mailových klientech zobrazován uživateli.

Druhý model přijímá tento objekt a na základě uložených informací vyčíslí hodnoty jednotlivých příznaků, které uloží do vektoru příznaků. Implementoval jsem většinu příznaků uvedených v kapitole 3 s několika výjimkami. První z nich jsou příznaky pracující s informacemi získanými z webových stránek, na které je v obsahu zpráv odkazováno. Od extrakce těchto příznaků jsem upustil, jelikož většina phishingových webů má krátkou životnost, a proto jsou odkazy obsažené v trénovací množině zpráv prakticky všechny nefunkční. Mezi další vypuštěné příznaky patří ty, které využívají výsledků vyhledávačů. Tyto příznaky nebyly implementovány, jelikož většina API fulltextových vyhledávačů je při větším počtu dotazů zpoplatněna. Poslední výjimku tvoří dvojice příznaků advMarkov a advCLTOM. Ty byly vynechány, jelikož složitost složitost jejich implementace by přesahovala očekávaný rozsah této práce.

Výsledný počet extrahovaných příznaků je 89, z čehož 24 spadá do kategorie klíčových slov, 17 se týká odkazů obsažených ve zprávě, 26 popisuje geolokační údaje o IP adresách přítomných v těle zprávy, 20 příznaků je strukturálních a 2 pokročilé (jedná se o výstup aplikace SpamAssassin)[18].

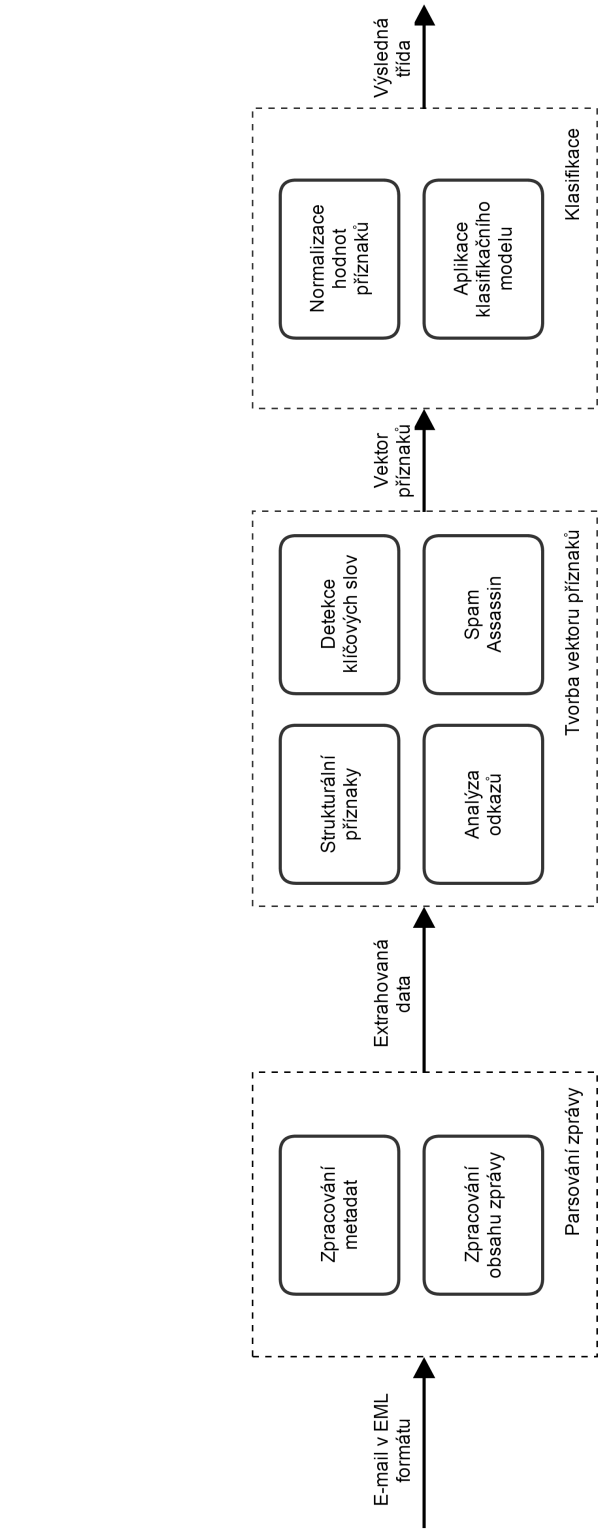
Poslední modul je zodpovědný za normalizaci hodnot příznaků a aplikaci klasifikačního modelu na vstupní vektor. Pro normalizaci jsem zvolil metodu z-score, protože u některých příznaků neznám předem rozsah hodnot u neznámých instancí a tato technika problém částečně eliminuje. Výstupem je třída, kterou model instanci přiřadil. Výsledná architektura aplikace je vizualizována na diagramu 5.1.

5.4 Zdroje dat pro strojové učení

Jako první část vstupních dat pro strojové učení jsem zvolil korpus anglických phishingových zpráv, které na své webové stránce zveřejnil Jose Nazario, [37]. Tento korpus se skládá ze čtyř částí. Celkově obsahuje 4553 zpráv, které byly získány v letech 2004 až 2007. Obsahují průřez mnoha phishingovými technikami.

Druhou nezbytnou částí vstupních dat je průřez běžnou poštou, který bude v trénovací množině reprezentovat ty zprávy, jež by měly být filtrem označeny jako bezpečné. K tomu poslouží veřejně dostupný korpus běžné pošty vzniklý v rámci projektu SpamAssassin, [38]. Obsahuje celkem 4150 zpráv získaných do roku 2006, které jsou rozděleny na lehké a těžké. Toto dělení vychází z míry podobnosti zpráv s nevyžádanou poštou a tudíž náročností správné klasifikace

[18] Do příslušných kategorií jsou připočteny i nově navržené příznaky.



Obrázek 5.1: Architektura aplikace

Typ zpráv	Počet	Zdroj
Běžné	4150	[38]
Phishingové	4553	[37]

Tabulka 5.1: Anglické phishingové zprávy

těchto zpráv spamovými filtry. Typickým znakem těchto zpráv je například frekventované využití HTML k formátování obsahu zprávy nebo využití slovních spojení, které se často vyskytují v hromadné nevyžádané poště. Jako obtížných označili autoři korpusu 250 zpráv.

Z obou korpusů čerpali zprávy i autoři řešení [12] a [20]. Nevýhodou je větší stáří dat a tudíž i riziko, že vytvořený model nebude dostatečně aktuální.

Souhrn informací o těchto datech je uveden v tabulce 5.1.

5.4.1 Český phishing

Sběr českých phishingových zpráv je o mnoho komplikovanější, jelikož neexistuje veřejně přístupný korpus, jenž by taková data obsahoval. Nejblíže k tomu má český portál Hoax.cz [39], jenž se zabývá sběrem různých druhů nevyžádané pošty a vzděláváním uživatelů v oblasti práce s elektronickou poštou. Mimo jiné obsahuje sekci o phishingu, ve které je zveřejněno přibližně sto česky psaných phishingových zpráv. Problém s těmito zprávami je takový, že obsahují pouze textovou část, tedy část, která je viditelná příjemci zprávy v poštovním klientovi. Tím pádem ztrácíme mnoho cenných informací z původní zprávy.

Zdrojem dalších zpráv byl manuální sběr. Celkový počet získaných česky psaných phishingových zpráv v plném formátu je 21, což je pro natrénování klasifikačního modelu nedostačující.

Rozhodl jsem se proto udělat ústupek, který spočívá v natrénování modelu pro české zprávy pouze na větší množině textů zpráv, přičemž budou extrahovány jenom příznaky, jež je možné z těchto dat získat, tedy informace o přítomnosti a frekvenci klíčových slov, jež byly popsány v sekci 3.4.6. Vynechány jsou naopak příznaky popisující strukturu zprávy, obsažené odkazy nebo výstup filtru SpamAssassin.

Pro doplnění korpusu o běžnou poštu jsem použil 100 e-mailů z osobní schránky, které poslouží jako data reprezentující běžné zprávy posílané v českém jazyce. Množina zpráv tak obsahuje jak obchodní poštu zasílanou většími společnostmi, kterou se snaží phisheři napodobit, tak i běžnou korespondenci mezi uživateli freemailových služeb. Celkové počty českých zpráv jsou uvedeny v tabulce 5.2.

Typ zpráv	Počet	Včetně metadat	Zdroj
Běžné	100	Ano	
Phishingové	21	Ano	
Phishingové	82	Ne	[39]

Tabulka 5.2: České phishingové zprávy

5.5 Frontend aplikace

Práce s vytvořenou aplikací je umožněna dvěma způsoby. Prvním je její spuštění z příkazové řádky, přes kterou je možné aplikaci předat zprávu v EML formátu, pro niž je následovně predikována třída. Při tomto spuštění je nutné na příkazové řádce specifikovat dva parametry v daném pořadí. První parametr -s specifikuje cestu ke spustitelnému souboru, jenž slouží jako frontend k filtrovacímu skriptu aplikace SpamAssassin. Tento skript je v aplikaci použit pro vyčíslení hodnoty příznaku závislého na výstupu aplikace SpamAssassin[19]. Druhým parametrem -m je specifikován EML soubor, ve kterém je uložena EML zpráva, jenž má být zpracována filtrem. Následuje příklad použití aplikace z příkazové řádky:

```
   $ java -jar PhishingFilter.jar -s ./path/spamassassin
-m message.eml
   Testing message.eml
   NOT_PHISH_MESSAGE
```

Ve výpisu aplikace je uživatel informován o aktuálně testovaném souboru a výsledku samotného testování.

Aplikace dále podporuje volitelný parametr -d, při jehož použití aplikaci kromě výsledné třídy vypíše do konzole i podrobné informace o vypočtené hodnotě jednotlivých příznaků a průběhu klasifikace.

Druhým způsobem práce s aplikací je využití API rozhraní. To spočívá ve volání statické metody classifyMessage třídy PhishingFilterAPI. Při volání metody je nutné předat dva parametry typu String, jejichž význam je shodný s významem parametrů při volání aplikace z příkazové řádky. Výsledkem volání je hodnota výčtového typu MessageType, jež specifikuje, zda byla zpráva označena jako phishingová či nikoliv.

[19]Cestu ke spamovému filtru SpamAssassin je při použití aplikace nutné zadat, jelikož uvedený filtr neexistuje v podobě multiplatformní knihovny. Pro účely vývoje a testování aplikace jsem využil portu SpamAssassinu pro Windows, SAwin32, [40]. Bližší popis binárního souboru, ke kterému je nutné uvést při použití aplikace cestu je popsán v dokumentaci na `https://spamassassin.apache.org/full/3.4.x/doc/spamassassin-run.txt`.

Kapitola 6

Ladění parametrů a testování

Tato kapitola popisuje proces ladění parametrů jednotlivých komponent prezentovaného algoritmu a jeho závěrečné testování. Uvádím jednak výsledky hledání optimální podmnožiny příznaků, volby klasifikačního algoritmu a měření přesnosti finálních modelů pro český a anglický phishing.

Pro účely měření byl dataset vytvořený z anglicky psaných zpráv rozdělen v poměru 70:30 na trénovací a validační množinu. Měření přesnosti klasifikace pro různé podmnožiny příznaků proběhlo na množině trénovacích zpráv s využitím 10-fold cross validace. Validační množina byla následně využita pro testování finálních modelů, abych předešel riziku přeučení pro trénovací množinu.

Model pro filtrování česky psaných zpráv byl trénován na textech 160 zpráv a následně testován na testovací množině obsahující 41 zpráv. Zmíněných 41 zpráv je v plné podobě včetně metadat, což umožnilo porovnání přesnosti s filtrem určeným pro anglické zprávy, který potřebuje k práci kompletní zprávy.

6.1 Volba podmnožiny příznaků

V rámci optimalizace množiny atributů jsem testoval tři různé přístupy. Prvním z nich bylo vypočtení informačního zisku jednotlivých parametrů a následné trénování modelu s využitím vektorů, které obsahovaly prvních n příznaků s největším informačním ziskem. V tabulce 6.1 je uvedeno 10 příznaků s nejvyšším informačním ziskem. Z uvedené tabulky vyplývá, že vysoký vliv na přesnost klasifikace mají příznaky získané s pomocí spamového filtru SpamAssassin. Z toho důvodu jsem zkusil rovněž optimalizovat množinu příznaků, ze které byly ty spojené se SpamAssassinem vyňaty, abych otestoval, zda by byl filtr použitelný i v případě extrakce příznaků, které lze získat přímo ze zprávy bez použití externích aplikací.

Z měření informačního zisku dále vyplynulo, že velký přínos mají příznaky týkající se odkazů ve zprávě, konkrétně měření počtu odkazů a unikátních

Příznak	Informační zisk
advSAScore	1,000
advSAClass	0,957
structHTML	0,802
linkCount	0,779
linkMaxDots	0,775
linkExternal	0,675
linkDomainCount	0,667
structFunctWord	0,587
linkSender	0,486
kwAccount	0,381
linkNonmatch	0,342
newMoreDomains	0,306

Tabulka 6.1: Měření nejpřínosnějších příznaků informačním ziskem. Informační zisky jednotlivých příznaků jsou standardizované. Uvedeno je pouze 10 příznaků s největším informačním ziskem z celkového počtu 89 příznaků, pro které měření proběhlo.

domén v nich obsažených. Naopak příznaky nesoucí geolokační informaci o IP adresách přítomných v e-mailu mají informační zisk téměř nulový.

Z nově navržených příznaků mají největší informační zisk příznaky newMoreDomains a newSingleAlternative.

Další dvě metody využité pro volbu příznaků spadají do kategorie wrapper. Testoval jsem jak forward, tak i backward selection. Klasifikačním algoritmem, pro který jsem měřil přesnost, byl Naïve Bayes, jež byl zvolen, protože umožňuje rychlejší trénování a následnou klasifikaci, než algoritmy k-NN a SVM. Výsledky měření různých podmnožin příznaků jsou uvedeny v tabulce 6.2.

Nejlepší podmnožiny příznaků bylo dosaženo metodou Forward Selection, a proto jsem se rozhodl ve finální aplikaci redukovat množinu extrahovaných atributů na 26 prvků, mezi které patří structFunctWord, structHTML, linkCount, structUnqWords, linkExternal, structCharCount, advSAClass, structVocabRichness, structImages, linkAt, structSenderLen, structForms, structSize, linkIP, structMIMECount, structSubjLen a měření přítomnosti klíčových slov update, limit, social, service, bank, ebay, log, credit, access a protect. Při použití těchto příznaků dosahuje filtr přesnosti 99,0 %.

Dalším podstatným zjištěním je, že i při vypuštění příznaků spojených se SpamAssassinem dosahuje vytvořený model přesnosti 97,7 %. Výhodou vypuštění příznaků advSAClass a advSAScore by bylo zejména zrychlení celé aplikace, jelikož při profilování aplikace bylo zjištěno, že aplikace při zpracování nové zprávy většinu času (přes 80 % doby běhu) čeká na výsledné skóre poskytované právě aplikací SpamAssassin.

Aplikovaná metoda	Zahrnutí SA	Počet příznaků	Přesnost
Forward selection	Ano	26	99,0 %
Backward selection	Ano	38	98,9 %
Informační zisk	Ano	8	98,7 %
Backward selection	Ne	38	97,7 %
Forward selection	Ne	12	97,6 %
Informační zisk	Ne	8	97,3 %

Tabulka 6.2: Porovnání různých optimalizačních metod a jejich výsledků. Pro každou z metod je uveden nejlepší výsledek.

Zvolené k	Přesnost klasifikace
1	99,2 %
2	99,0 %
3	99,1 %
4	99,0 %
5	99,1 %
6	98,9 %
7	98,9 %
8	98,8 %
9	98,8 %
10	98,8 %

Tabulka 6.3: Optimalizace k-NN algoritmu. Pro vyšší počet sousedů měla přesnost sestupnou tendenci.

6.2 Volba klasifikačního algoritmu

Dalším krokem je volba klasifikačního algoritmu. Testovány byly algoritmy k-NN, Naïve Bayes a SVM. U prvního algoritmu, k-NN, bylo nejdříve vybráno vhodné k. Výsledky tohoto měření jsou uvedeny v tabulce 6.3. Největší přesnost byla získána pro algoritmus 1-NN, nicméně jsem se rozhodl dále pracovat s verzí 5-NN, jenž měla nepatrně nižší přesnost a zároveň poskytuje lepší generalizaci, čímž snižuje riziko přeučení.

Tabulka 6.4 poskytuje porovnání přesnosti rozdílných klasifikátorů. Vzhledem k nejvyšší dosažené přesnosti byl pro finální model vybrán algoritmus 5-NN.

6.3 Přesnost klasifikace anglických zpráv

V této sekci popisuji výsledky testování pro finální model, který je přítomen ve výsledné aplikaci. Uvedená měření proběhla na validační množině obsahující 1366 phishingových a 1245 běžných zpráv.

Klasifikační algoritmus	Přesnost
5-NN	99,1 %
Naïve Bayes	99,0 %
SVM	98,9 %

Tabulka 6.4: Porovnání přesnosti klasifikace anglických zpráv při použití různých klasifikačních algoritmů.

		Klasifikováno jako:	
		Běžná	Phishingová
Označeno jako:	Běžná	1236	9
	Phishingová	18	1348

Tabulka 6.5: Matice záměn při měření přesnosti výsledného modelu pro anglické zprávy.

Řešení	Přesnost
Klasifikační model	99,0 %
SpamAssassin	97,2 %

Tabulka 6.6: Srovnání přesnosti SpamAssassinu a vytvořené aplikace při klasifikaci anglických zpráv.

Matice záměn 6.5 znázorňuje počty správně a špatně klasifikovaných instancí. Pozitivním výsledkem je nízký počet běžných zpráv zaměněných za phishingové (false positives). Tomu odpovídá i výsledná specificita uvedená spolu s dalšími mírami v tabulce 6.7.

Některá řešení uvedená v rešeršní části práce mají přesnost vyšší, což může být způsobeno různými okolnostmi, například extrakcí komplexnějších příznaků ze zpráv nebo rozdílným výběrem testovací množiny zpráv. Rovněž je nutno brát v potaz, že množina navržených příznaků ve výše popsaném řešení nespoléhá na žádné externí zdroje informací, jinými slovy, všechny příznaky jsou extrahovány lokálně ze zprávy, což umožňuje rychlejší filtrování.

Posledním měřením je porovnání navrženého řešení se spamovým filtrem SpamAssassin. Pro klasifikaci ve SpamAssassinu byly využity pouze lokálně extrahované příznaky. Porovnání výsledků je uvedeno v tabulce 6.6. Ačkoliv naměřené výsledky ukazují vyšší přesnost nově vytvořeného modelu, je nutné zmínit, že SpamAssassin je filtr určený pro detekci jakékoliv nevyžádané pošty, ne jenom phishingu. Tomu odpovídá i skutečnost, že z 53 chybně klasifikovaných instancí SpamAssassinem šlo v 47 případech o phishing a pouze v 6 případech o běžnou zprávu.

Míra	Výsledek
Přesnost	99,0 %
Error rate	1,0 %
Sensitivita	98,7 %
Specificita	99,3 %
Precision	99,3 %
f-míra	99,0 %

Tabulka 6.7: Výsledky testování modelu pro anglicky psané zprávy.

Řešení	Přesnost
Model pro anglické zprávy	65,8 %
Model pro české zprávy	85,4 %

Tabulka 6.8: Porovnání přesností modelů při klasifikaci českých zpráv.

6.4 Přesnost klasifikace českých zpráv

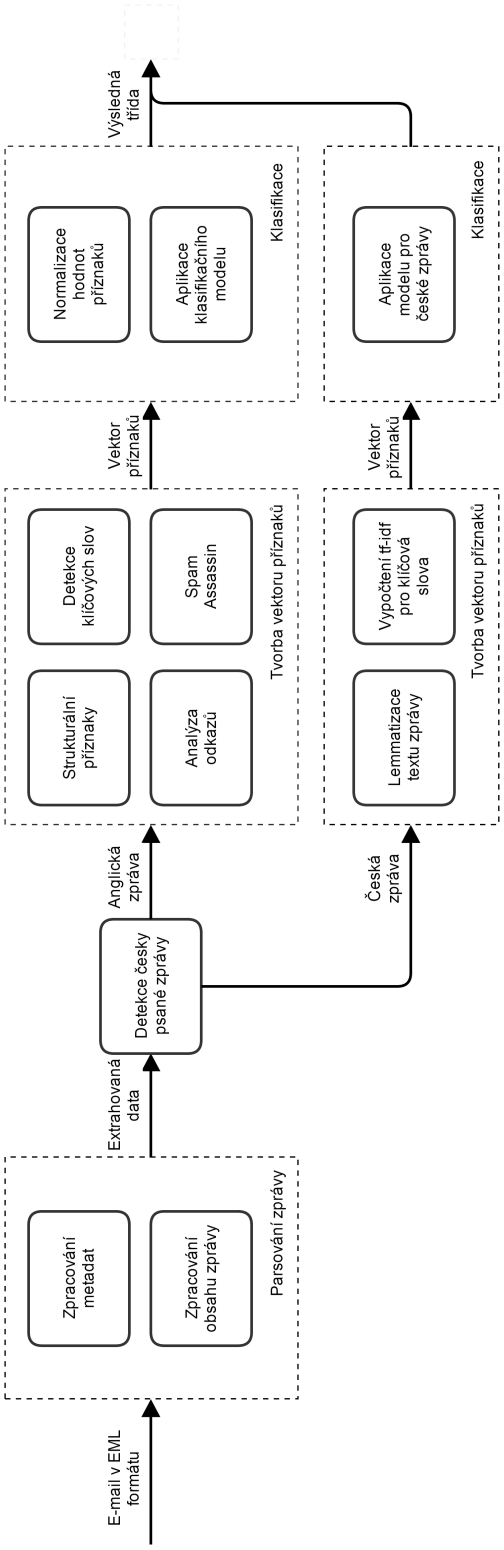
Pro účel klasifikace česky psaných zpráv byl vytvořen model založený na frekvenční analýze textu zprávy s pomocí metriky TF-IDF. Následně byl v rámci optimalizace parametrů strojového učení vybrán klasifikátor 10-NN. V tabulce 6.8 je uvedeno porovnání přesnosti modelu natrénovaného na textech českých zpráv s přesností komplexnějšího modelu natrénovaného na anglických zprávách, který analyzuje ve vstupní zprávě i přítomné odkazy a její strukturu. Z výsledků vyplývá, že se vyplatí pro účel klasifikace českých zpráv přidat do aplikace speciální model natrénovaný na textech česky psaných zpráv.

V rámci testování byl rovněž učiněn experiment, který spočíval v natrénování klasifikačního modelu na anglicky psaných zprávách, jež by pracoval jenom s příznaky, které jsou nezávislé na jazyku vstupní zprávy. Model tedy nebyl ovlivněn očekávanou absencí anglických klíčových slov v českých zprávách. Při testování tohoto modelu na česky psaných zprávách bylo dosaženo přesnosti 56,1 %, což poukazuje na skutečnost, že české phishingové zprávy mají rozdílnou podobu od anglického phishingu.

Na základě uvedených měření byl do aplikace přidán modul, jenž rozpoznává česky psané zprávy, které následně klasifikuje speciálním modelem natrénovaným na textech českých zpráv.

Detekce česky psaných zpráv funguje na principu pokusu o lemmatizaci slov obsažených v textu e-mailu. V případě, že je úspěšně lemmatizací pozměněn tvar alespoň 40 % slov[20], je zpráva označena jako česká. Výsledná architektura je znázorněna na diagramu 6.1.

[20]Hranice byla určena experimentálně tak, aby docházelo k minimu falešných detekcí. Modul pro detekci jazyka přesto negativně ovlivnil přesnost filtrování českých zpráv, jejíž hodnota je ve výsledném filtru pro testovací data 80,5 %.



Obrázek 6.1: Architektura aplikace s modelem pro české zprávy.

Závěr

Cílem této bakalářské práce bylo vytvoření aplikace pro programovou detekci phishingových zpráv s využitím metod data miningu a text miningu.

V rámci řešení jsem se seznámil se strukturou phishingových útoků a popsal jsem typické vlastnosti phishingové zprávy. Analyzoval jsem způsoby potírání phishingu v jednotlivých fázích útoku.

Vytvořil jsem seznam příznaků, jež byly v minulosti prezentovány v cizích pracích a rozdělil je do čtyřech kategorií na příznaky popisující klíčová slova, strukturu zprávy, odkazy v e-mailu a pokročilé příznaky.

Sesbíral jsem vzorové phishingové zprávy a na základě analýzy jejich obsahu jsem navrhl nové příznaky, které se týkají klíčových slov a struktury zprávy. Rovněž jsem představil přístup k detekci česky psaných zpráv založený na frekvenční analýze textu zprávy.

Seznámil jsem s technikami předzpracování dat, jmenovitě standardizací hodnot příznaků, výběrem vhodné podmnožiny příznaků a s klasifikačními algoritmy k-NN, Naïve Bayes a SVM.

Navrhl jsem architekturu filtru pro detekci phishingových zpráv. Navrženou architekturu jsem implementoval v Javě a funkčnost filtru ověřil jak na českých, tak i na anglických zprávách. Změřil jsem přesnost filtru a zhodnotil výsledky.

Hlavním výsledkem práce je algoritmus implementovaný v Javě, který s využitím algoritmů strojového učení klasifikuje phishingové zprávy. Řešení rovněž obsahuje specializovaný model pro česky psané zprávy.

Pro ostré nasazení aplikace do provozu by bylo nezbytné získat novější trénovací data, která by jednak lépe reprezentovala průřez elektronickou poštou, jež je v dnešní době zasílána a hlavně umožnila spolehlivé natrénování dalších příznaků, které sledují podobu phishingových webů, jež jsou v prezentované trénovací množině povětšinou nedostupné. S tím jsou spojeny i další cesty možného vývoje aplikace.

Kromě trénování klasifikátoru na novějších datech by pravděpodobně šlo zlepšit úspěšnost klasifikátoru pokročilými optimalizacemi parametrů strojo-

vého učení, například aplikací evolučních algoritmů pro selekci příznaků. Dalším možným směrem by byl vývoj klasifikátoru, jenž by implementoval ternární klasifikaci mezi běžnou poštu, spam a phishing.

Literatura

[1] JAKOBSSON, Marcus a Steven MYERS: *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft.* New Jersey: Wiley, 2006, ISBN 978-0-471-78245-2.

[2] HATTON, Les: *Email forensics: Eliminating Spam, Scams and Phishing.* UK: Bluespear Publishing, 2011, ISBN 978-1-908422-00-2.

[3] RAMZAN, Zulfikar: Phishing Attacks and Countermeasures. In *Handbook of Information and Communication Security*, editace STAVROULAKIS, P. a M. STAMP, Springer Berlin Heidelberg, 2010, ISBN 978-3-642-04116-7, s. 433–448.

[4] NOVINKY.CZ: Platnost internetového bankovnictví končí, zkouší podvodníci nový trik. [online]. 9.12.2014, [cit. 2014-12-13]. Dostupné z: `http://www.novinky.cz/internet-a-pc/bezpecnost/355897-.html`

[5] DŽUBÁK, Josef: Aktualizace účtu - Česká spořitelna. [online]. ©2000-2014, [cit. 2014-12-13]. Dostupné z: `http://www.hoax.cz/phishing/aktualizace-uctu---ceska-sporitelna--20141207/`

[6] SYMANTEC CORPORATION: Cílený phishing: podvod, ne sport. 8.6.2011, [cit. 2014-12-15]. Dostupné z: `http://cz.norton.com/spear-phishing-scam-not-sport/article`

[7] ANTI PHISHING WORKING GROUP, Inc: APWG Public Education Initiative. [2010], [cit. 2014-12-15]. Dostupné z: `http://phish-education.apwg.org/`

[8] WARDMAN, Bradley: *A series of methods for the systematic reduction of phishing.* Dizertační práce, The University of Alabama at Birmingham, Birmingham, USA, 2011.

[9] WOMBAT SECURITY TECHNOLOGIES, Inc.: Educating Users to Improve Awareness, Change Behaviors, and Reduce Risk. [2015], [cit. 2015-2-15]. Dostupné z: https://www.wombatsecurity.com/security-education/educate

[10] BERGHOLZ, A., J. DE BEER, S. GLAHN aj.: New Filtering Approaches for Phishing Email. [online]. 2009, [cit. 2015-2-17]. Dostupné z: http://www.antiphishresearch.org/downloads/journal-08-12-16-final.pdf

[11] ALMOMANI, A., B. GUPTA, S. ATAWNEH, A. MEULENBERG a E. ALMOMANI: A Survey of Phishing Email Filtering Techniques. IEEE, 2013, s. 2070–2090.

[12] ABU-NIMEH, Saeed: *Phishing Detection Using Distributed Bayesian Additive Regression Trees*. Dizertační práce, Dallas, USA, Southern Methodist University, 2008.

[13] NETCRAFT Ltd.: Anti-Phishing Services. [2015], [cit. 2015-2-19]. Dostupné z: http://www.netcraft.com/anti-phishing/

[14] GOOGLE Inc.: Upozornění na phishing a malware. [2015], [cit. 2015-2-20]. Dostupné z: https://support.google.com/chrome/answer/99020

[15] GARERA S., N. PROVOS, M. CHEW a A. D. RUBIN: A Framework for Detection and Measurement of Phishing Attacks. *WORM*, Alexandria, USA, 2007.

[16] OPENDNS: PhishTank. [2015], [cit. 2015-2-20]. Dostupné z: http://www.phishtank.com/

[17] DUNLOP M., S. GROAT a D. SHELLY: GoldPhish: Using Images for Content-Based Phishing Analysis. In *The Fifth International Conference on Internet Monitoring and Protection*, Barcelona: IEEE, 5 2010, s. 123–128.

[18] WORLD WIDE WEB CONSORTIUM: What is the Document Object Model? ©1997-2005, [cit. 2015-2-23]. Dostupné z: http://www.w3.org/DOM/

[19] PAN, Ying a Xuhua Ding: Anomaly Based Web Phishing Page Detection. In *Computer Security Applications Conference, FL, USA, 2006. ACSAC '06. 22nd Annual*, IEEE, s. 381–392.

[20] FETTE I., N. SADEH a A. TOMASIC: Learning to Detect Phishing Emails. *Proceedings of the 16th International Conference on World Wide Web*, New York, USA, 2007: s. 649–656.

[21] CHANDRASEKARAN M., K. NARAYAN a S. UPADHYAYA: Phishing E-mail Detection Based on Structural Properties. In *Proceedings of the NYS Cyber Security Conference*, New York, 2006.

[22] GANSTERER, Wilfried N. a David PÖLZ: E-Mail Classification for Phishing Defense. In *Advances in Information Retrieval*, *Lecture Notes in Computer Science*, ročník 5478, 2009, ISBN 978-3-642-00957-0, s. 449–460.

[23] HAN J., M. KAMBER a J. PEI: *Data mining: Concepts and techniques*. Morgan Kaufmann Publishers, třetí vydání, 2012, ISBN 9780123814791.

[24] KORDÍK, Pavel: Vytěžování znalostí z dat: Předzpracování dat. [přednáška]. Praha: FIT ČVUT, 5. března 2015.

[25] VACULÍK, Karel: *Selekce příznaků pomocí nekorelovaných charakteristik*. Diplomová práce, Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřící techniky, 2013, 42s.

[26] TOOLAN, Fergus a Joe CARTHY: Feature Selection for Spam and Phishing Detection. In *eCrime Researchers Summit (eCrime)*, 10 2010, s. 1–12.

[27] IP2LOCATION: LITE IP-COUNTRY Database. [online databáze]. ©2011-2015, [cit. 2015-4-18]. Dostupné z: `http://lite.ip2location.com/database-ip-country`

[28] REFSNES DATA: Window status property. ©1999-2015, [cit. 2015-4-20]. Dostupné z: `http://www.w3schools.com/jsref/prop_win_status.asp`

[29] FREED, N. a N. BORENSTEIN: Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. Technická zpráva, RFC Editor, 11 1996, rFC 2045. [cit. 2015-4-24]. Dostupné z: `http://www.ietf.org/rfc/rfc2045.txt`

[30] HLAVÁČ, Michal: jLemmaGen. [knihovna]. Březen 2014, [cit. 2015-4-25]. Dostupné z: `https://bitbucket.org/hlavki/jlemmagen`

[31] MITCHELL, Tom. M.: *Machine Learning*. USA: McGraw-Hill, 1997, ISBN 0-07-115467-1.

[32] FLACH, Peter: *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. New York: Cambridge University Press, 2012, ISBN 9781107096394.

[33] KORDÍK, Pavel: Vytěžování znalostí z dat: Hodnocení kvality modelu. [přednáška], Praha: FIT ČVUT, 12. března 2015.

[34] ORACLE CORPORATION: JavaMail API. [knihovna]. 3 2014, [cit. 2015-4-22]. Dostupné z: https://java.net/projects/javamail/pages/Home

[35] HEDLEY, Jonathan: jsoup: Java HTML Parser. [knihovna]. 9 2014, [cit. 2015-4-21]. Dostupné z: http://jsoup.org/news/release-1.8.1

[36] RAPIDMINER: RapidMiner: Data Mining, ETL, OLAP, BI. [software]. 10 2014, [cit. 2015-4-17]. Dostupné z: http://sourceforge.net/projects/rapidminer

[37] NAZARIO, Jose: Phishing Corpus. [korpus]. 7 2007, [cit. 2015-4-1]. Dostupné z: http://monkey.org/~jose/wiki/doku.php

[38] THE APACHE SOFTWARE FOUNDATION: SpamAssassin public mail corpus. [korpus]. 1 2006, [cit. 2015-4-1]. Dostupné z: https://spamassassin.apache.org/publiccorpus/

[39] DŽUBÁK, Josef: Hoax | Phishing. [online databáze]. 2 2015, [cit. 2015-4-3]. Dostupné z: http://www.hoax.cz/phishing/databaze/

[40] POP3PROXY: SpamAssassin for Win32. Duben, 2013, [cit. 2015-4-7]. Dostupné z: http://sourceforge.net/projects/sawin32/

Příloha A

Seznam použitých zkratek

AGPL Affero General Public License

API Application Programming Interface

ASCII American Standard Code for Information Interchange

AUC Area under the curve

BART Bayesian Additive Regression Trees

CBART Classification Bayesian Additive Regression Trees

CDDL Common Development and Distribution License

CLTOM Class-Topic Model

CSS Cascading Style Sheets

DKIM DomainKeys Identified Mail

DNS Domain Name System

DOM Domain Object Model

EML Formát pro ukládání elektronické pošty v plain textu

FN False negative

FP False positive

GPL General Purpose License

HTML HyperText Markup Language

IM Instant messaging

IP Internet Protocol

ISO International Organization for Standardization

k-NN k-Nearest Neighbours

MIME Multi-Purpose Internet Mail Extensions

MIT Massachusetts Institute of Technology (označení licence)

MMH Maximum Marginal Hyperplane

MTA Message transfet agent

OCR Optical character recognition

ROC Receiver Operating Characteristic

SMS Short Message Service

SPF Sender Policy Framework

SQL Structured Query Language

SVM Support vector machine

TAN Transaction authentication number

TF-IDF Term frequency–inverse document frequency

TLD Top Level Domain

TN True negative

TP True positive

URL Uniform Resource Locator

Příloha **B**

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD a návod k použití aplikace
- `data`
  - `datasets` ..... vytvořené datasety
  - `messages` ..... sesbírané zprávy použité pro strojové učení
- `examples`
  - `app` ..... spustitelná forma aplikace
  - `data` ..... testovací data
  - `testEngMessages.bat` ..... skript ukazující práci s aplikací
- `src`
  - `impl` ..... zdrojové kódy implementace
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text`
  - `assignment.pdf` ..... zadání práce ve formátu PDF
  - `thesis.pdf` ..... text práce ve formátu PDF