TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Informaatika õppekava

Ilja Smirnov

Rämpsposti tõrjumise meetodite võrdlus

Bakalaureusetöö (6 EAP)

Juhendaja(d): Tõnu Tamme, MSc

Tartu 2014

Rämpsposti tõrjumise meetodite võrdlus

Lühikokkuvõte:

Spämm on tänapäeval üks suurimatest probleemidest, millega kasutajad iga päev kokku puutuvad, kuid mitte kõik ei tea, kuidas rämpspostiga õigesti võidelda. Käesolevas bakalaureusetöös võrreldakse erinevaid rämpsposti tõrjumise meetodeid ja otsustatakse, milline neist on kõige efektiivsem ja millist on kõige mõistlikum kasutada. Seejärel viiakse läbi mitu erinevat katset, kusjuures iga meetodit hinnatakse konkreetsete kriteeriumite järgi ning nende põhjal tehakse järeldused. Lõpuks otsustatakse, millist meetodit on kõige mõistlikum kasutada ning milline neist on efektiivsem.

Võtmesõnad:

Rämpspost, spämm, rämpsposti tõrjumise meetodit, kriteeriumit, efektiivsem

Comparison of spam detection methods

Abstract:

Nowadays, spam is one the biggest problems that users face in their everyday life, but not all of them know how to handle it. The purpose of this paper is to compare different spam detection methods, decide which is the best one and when to use them. After that, each method is assessed according to the specific criteria to make conclusion. Finnaly, it is decided which of those methods is the most rational for specific purposes and which is the most effective.

Keywords:

Spam, spam detection methods, criteria, effective

Sisukord

Sissejuhatus

Kiire areng info- ja kommunikatsioonitehniloogias (lühend IKT) on toonud meile mitte ainult häid, vaid ka halbu tulemusi, üks neist on spämm ehk rämpspost. Spämm ehk rämpspost *(spam, junk mail)* on kasutajale e-postiga saadetud soovimatu sisuga, tavaliselt kommertsreklaami sisaldav kiri [1], mille saatmiseks pole meilikonto omanikult luba küsitud. Paljud inimesed on tänapäeval rämpspostiga kokku puutunud, kuid paljud ei tea, kuidas sellega õigesti võidelda, nii et arvuti kahjustamata jääks, sest spämm on üks lihtsamatest allikatest viiruse levitamiseks ja kasutajate andmete varastamiseks. Aasta jooksul saab iga internetikasutaja üle 2000 säärase kirja. Vastavalt läbiviidud uuringule, mida teostas "Kaspersky Lab", selgus, et rämpsposti saadetakse enam arenenud riikidest: Hiina 22,2 %, USA 18,4 %, Lõuna-Korea 14,7 % ja muud riigid 10,6 % [2].

Käesoleva bakalaureusetöö eesmärgiks on võrrelda erinevaid rämpsposti tõrjumise meetodeid ja otsustada, milline neist on efektiivsem ja millist on mõistlikum kasutada.

Töö esimeses peatükis defineeritakse, mis on need määratavad tingimused, tänu millele on võimalik rämpsposti tuvastada.

Teises peatükis kirjeldatakse lühidalt rämpsposti ajalugu ja arengut.

Kolmandas peatükis on ülevaade erinevatest meetoditest ning vaadatakse, kuidas nad töötavad.

Neljandas peatükis võrreldakse kõiki meetodeid ning viiakse läbi kolm erinevat katset, kus hinnatakse klassifitseerimistäpsust, saagist, spetsiifilisust ja täpsust ning lõpuks otsustatakse, milline neist on kõige parem.

1 Rämpspost

1.1 Määratavad tingimused

Kiri ei ole rämpspost, kui see on saadetud tudmatu autori poolt esimest korda ja on täidetud allpool loetletud tingimused:

1. Kiri saadeti ainult teie enda posti aadressile.
2. Kiri sisaldab linki allikale, kust teie aadress võeti (sõbrad, veebileht, foorum jne).
3. Põhejendatud on autori arvamus, miks see kiri võib teile huvi pakkuda.
4. On kasutatud lühikesi lauseid.
5. Puuduvad manused.
6. Kiri sisaldab autori vabandusi.
7. Kiri sisaldab autori reaalseid andmeid.

Nüüd vaatame ilmselgeid tunnused, mis viitavad spämmile:

1. Ei ole määratud allikat, kust kirja saaja aadress võeti.
2. Põhjalik reklaam, koos manusega.
3. Umbes järgmine lause: "Kui te ei soovi enam saada meie informatsiooni, palun saatke meile kiri.".
4. Puuduvad autori vabandused.
5. Saatja aadress on vale või puudub.

Sõltuvalt äriteabe sisaldumisest või mittesisaldumisest tuleb eristada kahte liiki spämmi. Üks neist on kaubanduslik (*unsolicited commercial e-mail*) ja teine on mittekaubanduslik rämpspost (*unsolicited bulk e-mail*).

1.2 Eesmärgid

Enamikul juhtudel kasutatakse spämmi reklaamimiseks. Tavaliselt reklaamitakse mingit toodet või teenust, mõnikord kasutatakse spämmi selleks, et suurendada veebilehtede reitingut ning veelgi harvem saadetakse viiruseid ja troojahobuseid. Peamine eesmärk on jagada informatsiooni võimalikult suure hulga inimestega, tehes samas minimaalseid kuulutusi. Saatja ei hooli konkreetse saajast, vaid peamine on saajate kogus.

Peamised eesmärgid:

1. Kauba ja teenuse reklaam. Tihti kiidetakse kirjas mingit teenust või kaupa ning viidatakse veebilehtele, kust saab rohkem infot kauba kohta, või telefoninumberile, millele helistades saab tellimust esitada.

2. Veebilehe edendamine. Teave võib olla mitmekülgne. Peamiselt reklaamitakse midagi väga head ja/või tasuta. Link viib leheküljele, millel ei ole absoluutselt mingit pistmist kirjas esitatud teabega. Aga tänu külastajate arvule suureneb veebilehe reiting ehk kasvab populaarsus.

3.Tasulised kõned. Reklaamitakse kaupa ja viidatakse telefoninumbrile, millele helistades lisandub kahtlaselt suur kõnetasu.

4.Turu-uuringud. Varjatud küsitlus või tellimus, mille käigus palutakse täita ankeet ja saata andmed konkreetsele aadressile.

5. Kahju tekitava tarkvara saatmine.

6. Muud eesmärgid.

1.3 Tehnoloogia

Spämmi saatmise tehnoloogia on lihtne ja tõhus. Spämmi võib edastada ise või pöörduda selleks spetsialistide poole.

Kõige olulisem on e-posti aadresside andmebaas. Aadresside kogumiseks kasutatakse eriprogramme, mis koguvad andmeid mingi konkreetse mustri järgi (näiteks, šabloon xxx@xxx). Laialisaatmine teostatakse reeglina anonüümselt või enda e-posti serveritest, kuid siis on saatja aadress vale või võõras. Mõnikord võivad olla andmed ka õiged.

2 Ajalugu

Rämpsposti ajalugu ulatub inimkonna kaugusesse minevikku (tinglikult võime seda arvestada postiteenuse sünniajast peale – alates esimestest valele adressaadile saadetud kirjadest). Antud peatükis vaadeldakse erinevate rämpspostiliikide tekkelugusid ning nende arengut.

2.1 Enne Interneti teket

19. sajandi lõpus lubas Western Union saata telegraafisaadetise korraga mitmesse sihtkoha. Esimene massiline kaubanduslik telegramm saadeti mais 1864, mil mõned Briti poliitikud said pealesunnitud telegrammi, mis reklaamis hambaravi teenuseid.

2.2 Etümoloogia

Sõna spämm pärineb 1970. aastast, kui BBC näitas Spam sketshis komöödiasarja Monty Python‘s Flying Circus. Sündmus algab ühes kohvikus, kus peaaegu iga toit menüüst sisaldab Spam lihakonservi. Kui ettekandja tuleb tellimust vastu võtma, hakkab üks tellijatest vestluse käigus laulma "Spam, Spam, Spam, Spam... lovely Spam! wonderful Spam!". Sõna “SPAM” üleliigne mainimine vestluses tõi kaasa suure lihakonservide importimise Ameerika Ühendriikidesse, eriti konserveeritud sealiha ja singi.

1980-ndatel hakati sõna SPAM seostama kuritahtlike kasutajatega, kes kasutasid hulgaliselt sõna “SPAM” internetis peetavates vestlustes. Kõik jututoad, mis tol aja eksisteerisid, näiteks PeopleLink ja Online America (hiljem tuntud kui America Online või AOL), olid üleujutatud tsitaadiga, mis pärines komöödiasarjast Monty Python‘s Flying Circus. Internetiühendus üle telefoniliini töötas tavaliselt 1200 või isegi 300 bit/s, seega kui saadeti tohutu arv teksti, siis kasutajal tuli päris palju allapoole kerida, et sõnumit läbi vaadata. Seda ärritavalt suurt ja mõttetut tekstilõiku hakati nimetama spämmimiseks.

Hiljem tuli kasutusele Usenet, kus oli lubatud sama sõnumi korduv saatmine. Effekt oli sama, mis pärast BBC esimest Spam sketshi. 31. Märtsil 1993 toimus üks intsident, kus Joel Furr, lõi eksperimentaalse tarkvara, mis hakkas ise saatma sõnumeid erinevate *news.admin.policy* uudisgruppidele. 1998. aastal andis New Oxford Dictionary of English

sõnale SPAM veel ühe tähenduse (enne tähendas sõna SPAM vaid toiduaine kaubamärki): “Ebaoluline või sobimatu saadetud sõnum internetis suurele hulgale kasutajatele või uudisgruppidele.”

2.3 Interneti spämm

Esimene jäädvustatud spämm saadeti 1978. aastal Gary Thuerk’i poolt 393 vastuvõtjale ning see reklaamis *Digital Equipment Corporation*’i uut mudelit. Siis ei eksisteerinud veel sellist mõistet nagu spämm. Esimene ahelkiri pealkirjaga „Teeni raha kiirelt“ on pärit aastast 1988. Esimene suurem kaubanduslik spämmiga seotud vahejuhtum on aastast 1994, kui advokaatidest abielupaar kasutas massilist Useneti postitamist, et reklaamida immigratsiooniseaduse teenuseid. Paari aastaga muutus spämm peamiselt elektronposti teel levivaks soovimatuks teavituseks, milleks ta on jäänud tänaseni [3].

3 Meetodid

3.1 Bayesi filtreerimise meetod

Bayesi rämpsposti filtreerimine on statistiline meetod e-posti filtreerimiseks [4]. Oma põhivormina kasutab see naiivset Bayesi klassifitseerijat (ingl *Naive Bayes classifier*), mis on sõnade kogum, tänu millele on võimalik tuvastada rämpsposti, ning on üks lähenemisviisidest, mida kasutatakse tavaliselt tekstiliigitamissüsteemides. Naiivne Bayesi klassifitseerija on tõestusmeetod, mille aluseks on Bayesi teoreem, kusjuures arvutatakse tõenäosus, kas vastav kiri on spämm või mitte. Naiivne Bayesi rämpsposti filtreerimine on põhitehnika, mis tegeleb rämpsposti tuvastamisega ning annab madala valepositiivse rämpsposti avastamise määra, mis on üldiselt vastuvõetav kasutajale. See on üks vanimaid viise rämpsposti filtreerimiseks, mis tuleneb 1990-ndatest.

3.1.1 Bayesi teoreem

Olgu antud hüpoteeside täielik süsteem $H_1, H_2 \dots H_n$ ning olgu teada nende hüpoteeside tõenäosused $P(H_1), P(H_2) \dots P(H_n)$. Tehakse katse, mille tulemuseks on mingi sündmus A, mille tinglikud tõenäosused $P(A \mid H_1), P(A \mid H_2) \dots P(A \mid H_n)$ olgu teada. Siis avaldub hüpoteesi H_i tinglik tõenäosus Bayesi valemi kujul [5].

$$P(H_i \mid A) = \frac{P(H_i) \cdot P(A \mid H_i)}{\sum_{i=1}^{n} P(H_i) \cdot P(A \mid H_i)}$$

Bayesi teoreemi kasutatakse rämpsposti filtreerimiseks kolm korda:

1. Kui on vaja arvutada tõenäosust, et etteantud sõnaga tekst on spämm. Kasutatakse valemit, mis tuleneb Bayesi teoreemist.

$$\Pr(S \mid W) = \frac{\Pr(W \mid S) \cdot \Pr(S)}{\Pr(W \mid S) \cdot \Pr(S) + \Pr(W \mid H) \cdot \Pr(H)}$$

kus

$\Pr(W \mid S)$ - tõenäosus, et spämmi kuuluvas kirjas esineb sõna W

$\Pr(S)$ - spämmi osakaal kõikides kirjades ehk tõenäosus, et kiri on spämm

$\Pr(W \mid H)$ - tõenäosus, et spämmi mittekuuluvas kirjas esineb sõna W

$\Pr(H)$ - vastandsündmuse tõenäosus, et kiri ei ole spämm

2. Kui on vaja arvutada tõenäosust, et kiri on rämpspost, kusjuures arvestatakse kõigi sõnade sagedust. Iga sõna jaoks arvutatakse eraldi esinemise tõenäosus. Juhul kui sõna tõenäosus ületab 0.8, siis seda sõna võib kvalifitseerida kui spämm.
3. Harva esinevatega sõnadega

$$\Pr(S \mid W) = \frac{s \cdot \Pr(S) + n \cdot \Pr(S \mid W)}{s + n}$$

kus

S - saabunud kirja tugevus

N - sõna esinemise sagedus

$\Pr(S \mid W)$ - tõenäosus, et kiri on spämm

3.1.2 Naiivne Bayesi klassifitseerija

Naiivne Bayesi klassifitseerija (ingl *Naive Bayes classifier*) on lihtne tõenäosuslik klassfitseerimisalgoritm. Selle eelisteks on väike vajalike algandmete hulk ja lihtsus, põhiliseks puuduseks on aga eeldus, et kõik sisendid on teineteisest sõltumatud [6].

3.1.3 Klassifitseerija tööpõhimõte

Naiivne Bayesi klassifitseerija on lihtne tõenäosuslik masinõppe algoritm. Klassifitseerija naiivsus seisneb parameetrite omavahelise sõltuvuse eitamises – eeldatakse, et ükski parameeter ei ole teistega korrelatsioonis. Naiivse Bayesi klassifitseerija puhul leitakse iga parameetri järgi tõenäosus, millisesse klassi andmed liigitada. Seejärel korrutatakse tõenäosused läbi ning andmed klassifitseeritakse suurima kogutõenäosusega klassi. Kuigi tegemist on üsna lihtsa lähenemisega, võib naiivne Bayesi

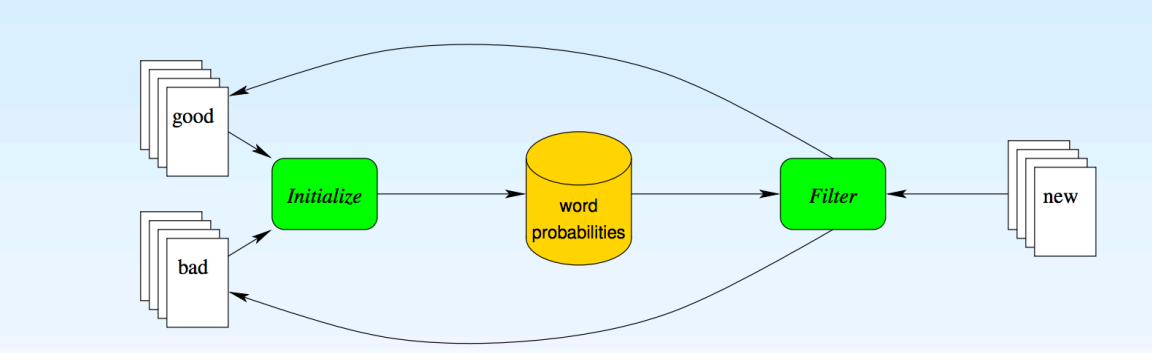
klassifitseerija olla efektiivsem kui mitmed keerulised algoritmid. Üldjuhul, mida väiksem on parameetrite omavaheline korrelatsioon, seda parema tulemus on võimalik naiivse Bayesi klassifikaatoriga saavutada.

Kasutatakse näiteks spämmifiltrites, võttes tunnusteks sõnumisse kuuluvad sõnad. Kuigi need on üksteise suhtes kõike muud kui sõltumatud, on Paul Grahami implementatsiooni saavutatud täpsus 99,75% soovimatute sõnumite korral ja 100% tavaliste kirjade korral [17].

3.1.4 Paul Graham implementatsioon

2002. aastal modifitseeris Paul Graham Bayesi filtreerimise meetodid, millest ta kirjutas oma artklis “A Plan for Spam”.

See koosneb kahest etapist(vt. Joonis 1).



Joonis 1. Paul Graham implementatsioon [16].

kus

1. Initsialiseerimine (ingl *Initialize*)
2. Filtreerimine (ingl *Filter*)

3.1.4.1 Initsialiseerimine

Peab olema andmebaas headest ja halbadest kirjadest, kusjuures iga sõna jaoks arvutatakse tema esinemise tõenaosus.

Kui me teame

kiriHalb – koguarv "halb" sõnumeid

kiriHea – koguarv "hea" sõnumeid

sõnaHalb – sõna esinemised "halb" kirjas

sõnaHea – sõna esinemised "hea" kirjas

siis on võimalik arvutada tõenäosust iga sõna(w) sagedust heade ja halbade kirjades.

$$P(w) = \max\left(\min\left(\frac{r_b}{r_g + r_b}, 0.99\right), 0.01\right)$$

kus

$$r_b = \min\left(\frac{sõnaHalb}{kiriHalb}, 1.0\right),\ r_g = \min\left(\frac{2sõnaHea}{kiriHea}, 1.0\right)$$

Mida kaugemal on P(w) väärtusest 0.5, seda lihtsam on arvutada, kas käesolev kiri on rämpspost või mitte.

Valemi realisatsioon Java keeles.

```
public void finalizeProb() {
    if (rGood + rBad > 0) pSpam = rBad / (rBad + rGood);
    if (pSpam < 0.01f) pSpam = 0.01f;
    else if (pSpam > 0.99f) pSpam = 0.99f;
}
```

3.1.4.2 Filtreerimine

Saabunud postkasti kirjas valitakse 15 nö “huvitatavat” sõnu, mille tõenäosused on keskelt (0,5) kaugemale, kusjuures iga sõna jaoks arvutatakse tõenäosus.

$$p = \frac{prod_1}{prod_1 + prod_2}$$

kus

$$prod_1 = \prod_{i=1}^{n} P(w_i)\,,\ prod_2 = \prod_{i=1}^{n} (1 - P(w_i))$$

Mida lähemal on p väärtus ühele, seda suurema tõenäosusega võib öelda, et kiri on spämm.

On mitmeid erinevaid viise, kuidas leida 15 huvitavamat sõna. Käesoleval juhul kasutatakse admestruktuuriks listi, kuhu kogutakse sõnade tõenäosused ning sorteeritakse kasutades pistemeetodit. Pistemeetodi keerukus halvimal ja keskmisel juhul on $O(n^2)$ ning parimal juhul $O(n)$ [19].

Huvitavate sõnade erinevuse keskelt arvutamine:

```
public float interesting() {
    return Math.abs(0.5f - pSpam);
}
```

3.1.5 Eelised ja puudused

Bayesi rämpsposti filtreerimise meetodi eelised:

- Kontseptuaalselt väga lihtne aru saada.
- Väga efektiivne.
- On võimalik enda järgi seadistada.
- Paljud e-posti kliendid juba otseselt või kaudselt toetavad Bayesi filtreerimist.

Bayesi rämpsposti filtreerimise puudused:

- Ressursimahukas.
- Korpus heade ja halbade kirjadega.
- Ei tuvasta pilte.
- Initsialiseerimine on aeganõudev.

3.2 Winnow

3.2.1 Winnow

Winnow on staatiline algoritm rämpsposti filtreerimiseks, mille pakkus välja Nick Littlestone [7]. Oma põhivormina kasutab see regulaaravaldisi. On väga sarnane Perceptron algoritmiga.

Iga tekst jagatakse juppideks, kasutades regulaaravaldisi, kusjuures võetakse arvesse nii numbreid kui ka nimed ja pärast pannakse eri andmestruktuuridesse. Lõpuks võrreldatakse olemasolevaid sõnu andmestruktuuris ja kirjas.

Winnow algoritmi kasutatakse siis, kui on vaja tuvastada, kas antud kiri on rämpspost või mitte.

4.2.2 Winnow algoritm

1. Koguda kõik failid ja juhuslikult neid järjestada
2. Iga faili:
    3. Võta tekst failist
    4. Jaga tekst juppideks
        5: Iga aktiivsele omadusele:
            Kui sõnum vastab eelnevalt salvestatud funktsioonile, aktiivne.
            Kui ei ole jätame ära
        6: Iga aktiivsele omadusele:
                7: Arva, kas see on hea või halb kiri
                8a: Kui tuleb, et kiri on “halb” kuid faili nimi on “hea”
                    Suurenda “hea” kirja kaalu
                    Alanda “halb” kirja kaalu
                8b: Kui tuleb, et kiri on “hea” kuid faili nimi on “halb”
                    Suurenda “halb ” kirja kaalu
                    Alanda “hea” kirja kaalu
                9: Rakenda saadud kaalud
                    Kui hea ja halbu kirja kaalud on liiga lähedal,ei ole spämm
        10: Lõpeta punkt 5:
            Kõigi sõnumit, mis ei sobinud eelnevalt salvestatud omadustele:
                Loo funktsiooni nende sõnumite jaoks ja salvesta neid

4.2.3 Regulaaravaldised

Regulaaravaldised on võimas vahend teksti uurimiseks ja muutmiseks. Nende abil on võimalik leida tekstist teatud tüüpi sõnu, lauseid ja muid täheühendeid ning neid paindlikult lahti harutada ja teha asendusi [8]. Winnow puhul on regulaaravaldised peamine vahend, mille abil tekst juppideks jagatakse ja siis salvetatakse iga sõna eraldi eri anmestruktuuri nimega "*feature*". Allolevas tabelis on toodud erinevad regulaaravaldiste tüübid, mida on võimalik kasutada (Tabel 1).

Nimed	Regulaaravaldised
Tavaline tekst	[^\p{Z}\p{C}]+
CRM114	[^\p{Z}\p{C}][-.,:\p{L}\p{M}\p{N}]*[^\p{Z}\p{C}]?
Lihtsustatud CRM114	[^\p{Z}\p{C}][-\p{L}\p{M}\p{N}]*[^\p{Z}\p{C}]?
HTML	[^\p{Z}\p{C}][/!?#]? [-\p{L}\p{M}\p{N}]*(?:[”’=;]|/?>|:/*)?

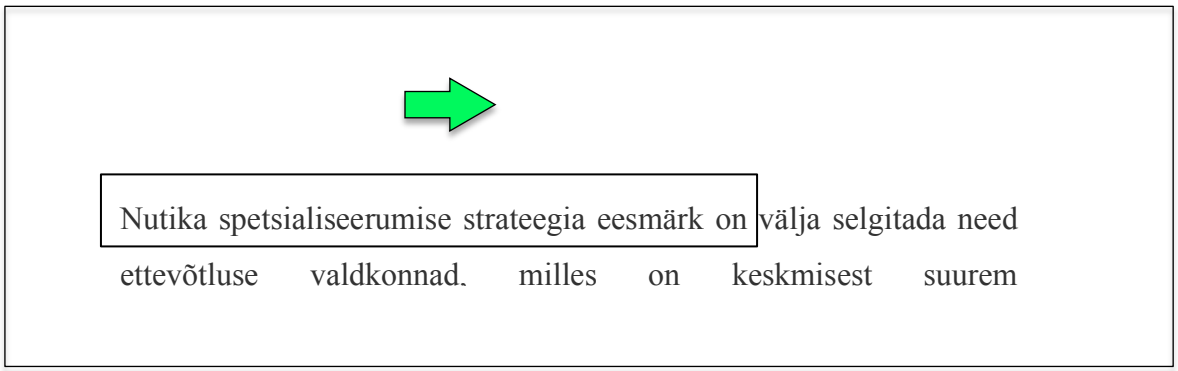
Tabel 1. Regulaaravaldised [18].

Java keeles regulaaravaldis:

```
[\p{Cntrl}\p{Space}]
```

4.2.4 Teksti valimine

Kasutaja valib palju sõnu võib mahtuda ühte kasti. Arvesse lähevad nii numbrid kui ka tähemärgid. Optimaalne akna suurus on viis.



Siin on akna suurus viis ehk aknase mahub täpselt viis sümbolit. Iga kord liigutatakse akent vasakult paremale. Järgmine positsioon algab sõnaga "spetsialiseerumise" ja lõppeb sõnaga "välja", kusjuures sõna "Nutika" jäetakse ära jne.

4.2.5 Ortogonaalne hõre bigramm

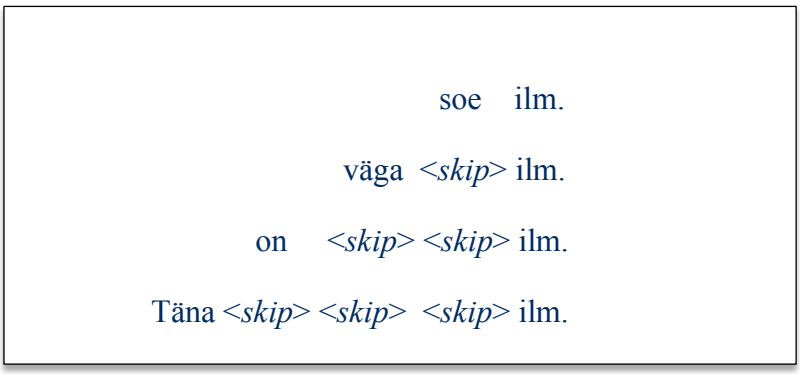
Ortogonaalne hõre bigramm(ingl *Orthogonally Sparse Bigram,* lühend *OSB*) on üks populaarsematest andmestruktuuridest. Selle eelisteks on võimalus vältida liiasust ja üldiselt täiustada paljusid rämpsposti filtreerimise algoritme [9]. Lisaks sellele aitab see lahendada ühte väga olulist probleemi – duplikaatide esinemist.

Teksti algusest võetakse viis sõna ja paigutatakse *OSB* andmestuktuuri.

Kusjuures $w_1, w_2 \ldots . w_n$ on sõnade jada ja nende põhjal tehakse neli erinevat kombinatsiooni. Kõige mõistlikum ja mugavam on seda teha tabelina.

				*w*4 *w*5
			*w*3	<*skip*> *w*5
		*w*2	<*skip*>	<*skip*> *w*5
*w*1	<*skip*>	<*skip*>	<*skip*>	*w*5

Näide: Täna on väga soe ilm.



4.2.6 Eelised ja puudused

Allpool on toodud Winnow'i eelised:

- Saab hakkama müradega.
- Lihtne.

Vaatame nüüd Winnow'i puudused:

- Ei tuvasta pilte.
- Ei ole stabiilne.
- Initsialiseerimine on aeganõutav.
- Suure andmete puhul ei ole väga efektiivne.

4 Katsed

4.1 Lähteandmed

Katse eesmärgiks on teada saada, millist meetodit on kõige mõistlikum kasutada ning otsustada kumb neist on parem.

Käesoleval juhul kasutasin heade ja halbade kirjade andmebaasiks Enroni korpust. Enron on suur andmebaas kirjadest , mis on loodud 158 kasutaja poolt, kes töötasid Enroni korporatsioonis [14]. Viisin läbi kolm katset ning hindasin igat meetodit nelja kriteeriumi järgi: klassifitseerimistäpsus(ingl *Accuracy*), täpsus(ingl *Precision*), saagis(ingl *Recall*) ja spetsiifilisus (ingl *Specificity*) (Tabel 2).

Mõõt	Valem
Klassifitseerimistäpsus	$\frac{TP+TN}{TP+TN+FP+FN}$
Täpsus	$\frac{TP}{TP+FP}$
Saagis	$\frac{TP}{TP+FN}$
Spetsiifilisus	$\frac{TN}{TN+FP}$

Tabel 2. Kriteeriumid koos valemitega

Käeoleval juhul kasutasin teiste kasutajete poolt kirjutatud koode [9, 13].

Kõik failid on lisatud tööle (vt. Lisa 1).

5.2 Kriteeriumid

5.2.1 Klassifitseerimistäpsus

Klassifitseerimistäpsus (ingl *Accuracy*) on arv, mis näitab, kui suur osa testandmetest sai antud mudeli järgi õigesti klassifitseeritud [10].

$$\frac{TP + TN}{TP + TN + FP + FN}$$

TP - kiri oli märgistatud kui spämm, pärast katse läbiviimist osutus, et kiri on spämm.

TN - kiri oli märgistatud kui unikaalne, pärast katse läbiviimist osutus, et kiri on unikaalne.

FP - kiri oli märgistatud kui spämm, pärast kaste läbiviimist osustus, et kiri on unikaalne.

FN - kiri oli märgistatud kui unikaalne, pärast kaste läbiviimist osustus, et kiri on spämm.

5.2.2 Täpsus

Iga klassi kohta eraldi mõõdetav *täpsus (*ingl *Precision*) näitab, kui suur osa selle klassi hulka loetud näidetest ka tegelikult sinna kuuluvad [10].

$$\frac{TP}{TP + FP}$$

TP - kiri oli märgistatud kui spämm, pärast katse läbiviimist osutus, et kiri on spämm.

FP - kiri oli märgistatud kui spämm, pärast kaste läbiviimist osustus, et kiri on unikaalne.

5.2.3 Saagis

Saagis (ingl *Recall)* näitab, kui suur osa antud klassi näidetest korrektselt sinna kuuluvaks loeti ehk kirjad, mis oli märgistatud kui head ning katse läbiviimise käigus osutusid headeks [11].

$$\frac{TP}{TP + FN}$$

TP - kiri oli märgistatud kui spämm, pärast katse läbiviimist osutus, et kiri on spämm.

FN - kiri oli märgistatud kui unikaalne, pärast kaste läbiviimist osustus, et kiri on spämm.

5.2.4 Spetsiifilisus

Spetsiifilisus (ingl *Specificity*) näitab, kui suur osa antud klassi näidetest loeti negatiivselt sinna kuuluvaks ehk kirjad, mis oli märgistatud kui spämm ning katse käigus osutusid spämmideks [12].

$$\frac{TN}{TN + FP}$$

TN - kiri oli märgistatud kui unikaalne, pärast katse läbiviimist osutus, et kiri on unikaalne.

FP - kiri oli märgistatud kui spämm, pärast kaste läbiviimist osustus, et kiri on unikaalne.

4.2 Katse läbiviimine

Viisin läbi kolm erinevat katset, kusjuures iga meetodit hindasin nelja kriteeriumi järgi. Andmebaasiks kasutasin Enroni korpust [15]. Esimese katse puhul oli 60 kirja, teise katse puhul oli 120 ja viimases katses oli 200 kirja.

4.3 Kokkuvõte katsetulemusest

Kokku oli 380 kirja, kusjuures 190 kirja oli märgistatud kui halvad ja ülejäänud kui head. Esimese katse puhul selgus, et Bayesi filtreerimise meetodi klassifitseerimistäpsus on 62,5% , mis on 1.5 korda suurem kui Winnow'l (vt. Tabel 3). Teise ja kolmanda katse puhul me näeme, et nende meetodite tulemused on suht lähedased ja klassifitseerimistäpsus ei ületa 80%. Vaadates teisi kriteeriumeid, siis Bayesi puhul spetsiifilisus on 86,6% , täpsus on 74,1% ning Winnow puhul spetsiifilisus on 50% ja täpsus on 49,1%. Mõlema meetodi puhul saagid on 38,3% (vt. Tabel 3).

	Bayesi filtreerimine	Winnow
Klassifitseerimistäpsus	62,5%	49,5%
Spetsiifilisus	86,6%	50%
Täpsus	74,1%	49,1%
Saagis	38,3%	38,3%

Tabel 3. Meetodite võrdlus 60 kirja

Teise katse puhul osutus, et Bayesi filtreerimise meetodi klassifitseerimis täpsus on 61,2%, mis on 0,15 korda väiksem kui Winnow'l (vt. Tabel 4). Vaadates teisi kriteeriumeid, siis Bayesi puhul spetsiifilisus on 86,6% , täpsus on 74,1%, saagis on 35,8% ning Winnow puhul spetsiifilisus on 50%, täpsus on 49,1% ja saagis on 66,6% (vt. Tabel 4).

	Bayesi filtreerimine	Winnow
Klassifitseerimistäpsus	61,2%	70%
Spetsiifilisus	87,5%	63,6%
Täpsus	73,6%	65,6%
Saagis	35,8%	66,6%

Tabel 4. Meetodite võrdlus 120 kirja

Kolmanda katse puhul tuli välja, et Bayesi filtreerimise meetodi klassifitseerimistäpsus on 65,5%, mis on 0,2 korda väiksem kui Winnow'l (vt. Tabel 5). Vaadates teisi kriteeriumid siis Bayesi puhul spetsiifilisus on 86,6% , täpsus on 74,1%, saagis on 43,5% ning Winnow puhul spetsiifilisus on 50%, täpsus on 49,1% ja saagis on 79% (vt. Tabel 5).

	Bayesi filtreerimine	Winnow
Klassifitseerimistäpsus	65,5%	77%
Spetsiifilisus	87,5%	75%
Täpsus	77,6%	75,9%
Saagis	43,5%	79%

Tabel 5. Meetodite võrdlus 200 kirja

Läbi viidud katsete puhul, võib näha, et kui kirjade arv suureb siis väheneb Bayesi klassifitseerimistäpsus, kuid Winnow'i puhul kasvab.

Sellest võib järeldada, et Bayesi filtreerimise meetodeid on mõistlikum kasutada siis, kui kirjade arv ei ületa 60, vastasel juhul ei ole vahet kumba meetodit kasutada, sest nii Bayesi kui ka Winnow rämpsposti tuvastamise määr ei erine teineteisest olulisel määral. Kui aga vaadata ühte Bayesi filtreerimismeetodi eelistest, milleks on võimalus seadistada seda enda järgi siis sel juhul on efektiivsus liigikaudu 99%. Kui inimest huvitavad ainult need kirjad, mis on seotud tema erialaga, siis muud kirjad lihtsalt ei jõua tema postkasti, sest need märgistatakse kui rämpspost või kustutatakse ära. Kuid Winnow'i puhul rämpsposti tuvastamise määr on liigikaudu 75%, mis on kasutajatele üldiselt vastuvõetav.

Seega mõlemad meetodid võivad olla kasuks rämpspostiga võitlemiseks ning millist meetodid kasutada on juba tarkvaraarendaja otsustada.

Kokkuvõte

Töö eesmärgiks on võrrelda erinevaid meetodeid rämpsposti filtreerimiseks ning otsustada kumb neist on parem ja millal on neid mõistlik kasutada.

Töö esimeses peatükis selgitatakse, mida kujutab endast spämm ja millised on need määratavad tingimused, mis vastavad spämmile.

Töö teises peatükis käsitletakse kolme erinevat teooriat kust sõna spämm võiks tuleneda.

Kolmandas peatükis võrreldakse erinevaid meetodeid rämpsposti filtreerimiseks ja valitakse Bayesi filtreerimise meetodi ja Winnow, kusjuures selgitatakse kuidas nad töötavad ja vaadatakse nende eelised ja puudused .

Neljandas peatükis viiakse läbi kolm erinavat katset, kusjuures hinnatakse iga meetodi puhul klassifitseerimistäpsust, spetsiifilisust, saagist ja täpsust.

Katse tulemusena leitakse, et Bayesi filtreerimise meetodid on mõistlikum kasutada siis kui kirjade arv ei ületa 60 kirja. Kui kirjade arv on üle 60, siis ei ole vahet kumba meetodit kasutada, sest nii Bayesi kui ka Winnowi rämpsposti tuvastamise määr ei erine üksteisest.

Kasutatud kirjandus

[1] Priit (2010), "Mis on spämm", URL http://arvutiturve.wordpress.com/2010/03/21/mis-on-spamm/, (14.05.2014).

[2] Kaspersky Lab, "Спам в первом квартале 2014", URL https://www.securelist.com/ru/analysis/208050841/Spam_v_pervom_kvartale_2014, (14.05.2014).

[3] Kai Tootsi (2009), Referaat aines "Rakendustarkvara: Internet", Tartu Ülikool, URL http://courses.cs.ut.ee/2009/internet/Main/KuidasVõideldaSpämmiVastu? (14.05.2014)

[4] Wikipedia, "Bayesian spam filtering", URL http://en.wikipedia.org/wiki/Bayesian_spam_filtering, (14.05.2014)

[5] Ako Saug (2006), "Statistika ja tõenäosusteooria", Audentese Ülikool, URL http://www.sauga.pri.ee/audentes/download/stait.pdf, (14.05.2014).

[6] Wikipedia, "Naive Bayes classifier", URL http://en.wikipedia.org/wiki/Naive_Bayes_classifier, (14.05.2014).

[7] Wikipedia, "Winnow (algorithm)", URL http://en.wikipedia.org/wiki/Winnow_(algorithm), (14.05.2014).

[8] Regulaaravaldised PHP-s (2003), URL http://phpcenter.eu/opetused.php?id=92, (14.05.2014).

[9] Neural Networks, "Winnow: Spam Recognition", URL http://mnemstudio.org/neural-networks-winnow-example-1.htm, (14.05.2014).

[10] Wikipedia, "Accuracy and precision", URL http://en.wikipedia.org/wiki/Accuracy_and_precision, (14.05.2014).

[11] Wikipedia, "Precision and Recall", URL http://en.wikipedia.org/wiki/Precision_and_recall, (14.05.2014).

[12] Wikipedia, "Specificity", URL http://en.wikipedia.org/wiki/Specificity_(statistics), (14.05.2014).

[13] Daniel Shiffman, “Bayesian Filtering”, URL http://shiffman.net/teaching/a2z/bayesian/, (14.05.2014).

[14] Wikipedia, “Enron Corpus”, URL http://en.wikipedia.org/wiki/Enron_Corpus, (14.05.2014).

[15] “Enron dataset”, URL http://nlp.cs.aueb.gr/software_and_datasets/Enron-Spam/index.html, (14.05.2014).

[16] Zhang Hongxin (2009), Slaidid “Naïve Bayes Classification”, Zhejiang University, China, URL http://www.cad.zju.edu.cn/home/zhx/ML/ML2009-2-3-naive_bayes_classification.pdf, (14.05.2014).

[17] Andres Erbsen (2012), Uurimistöö “Masinõppe kasutamine oluliste kirjade tuvastamiseks”, Tallinna Reaakool, URL http://minitorn.tlu.ee/~jaagup/oma/too/12/04/kirjavordlus/mlmail.pdf, (14.05.2014).

[18] Christian Siefkes, Fidelis Assis, Shalendra Chhabra, William S. Yerazunis, “Combining Winnow and Orthogonal Sparse Bigrams for Incremental Spam Filtering”, URL http://www.cs.ucr.edu/~schhabra/winnow-spam.pdf, (14.05.2014).

[19] Wikipedia, “Insertion sort”, URL http://en.wikipedia.org/wiki/Insertion_sort, (14.05.2014).

Lisad

I. Lisa 1. Bakalauresetöö juurde kuuluvate failid

Bakalaureusetöö juurde kuuluvad järgmise failid:

Bayesi filtreerimise meetod:

A2ZFileReader – faili lugemine

A2ZFileWriter – faili kirjutamine

Bayesian – peamine klass

SpamFilter – initsialiseerimine ja filtreerimine

Word – sõnade tõenäosuse arvutamine

Messages1 – esimene katse: 60 kirja

Messages2 – teine katse: 120 kirja

Messages3 – kolmas katse: 200 kirja

Good.txt – andmebaas headest kirjadest

Bad.txt – andmebaas halbadest kirjadest

Winnow:

Winnow – peamine klass

Samples – kirjade andmebaasid

Messages1 – esimene katse: 60 kirja

Messages2 – teine katse: 120 kirja

Messages3 – kolmas katse: 200 kirja

II. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina **Ilja Smirnov** (sünnikuupäev: 02.08.1992)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose **Rämpsposti tõrjumise meetodite võrdlus**,

   mille juhendaja on Tõnu Tamme,

   1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
   1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace´i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, **14.05.14**