Uppsala universitet
Inst. för informatik och media

Skräppost eller skinka?

En jämförande studie av övervakade maskininlärningsalgoritmer för spam och ham e-mailklassifikation

Spam or ham?

A comparative study of monitored machine learning algorithms for spam and ham e-mail classification.

Simon Bergens & Pontus Frykengård

UPPSALA
UNIVERSITET

Kurs: Examensarbete
Nivå: C
Termin: VT-19
Datum: 2019-06-14
Handledare: Carl-Mikael Lönn

Sammanfattning:

Spam meddelanden i formen e-mails är ett växande problem i dagsläget för verksamheter. Det är ett problem som kostar tid och resurser att motverka. Forskning kring detta har gjorts för att framställa tekniker och verktyg med målet att behandla den växande mängden inkommande spam e-mails. Forskningen angående olika algoritmers förmåga att klassificera e-mails behöver dock uppdateras, detta då både verktyg och spam e-mails blivit mer avancerade. I denna studie har tre olika maskininlärningsalgoritmer utvärderats baserat på deras förmåga att korrekt klassificera e-mails som legitima och spam. Dessa algoritmer är naive Bayes, stödvektormaskin och beslutsträd. Algoritmerna testas i ett experiment med Enron spam dataset och jämförs sedan mot varandra i sina prestationer. Resultatet av experimentet var att stödvektormaskin är den algoritm som klassificerade flest datapunkter korrekt. Men även om stödvektormaskin har störst procentuell andel korrekt klassificerade datapunkter kan övriga algoritmer besitta verksamhetsnytta beroende på uppgift och kontext.

Abstract:

Spam messages in the form of e-mail is a growing problem in today's businesses. It is a problem that costs time and resources to counteract. Research into this has been done to produce techniques and tools aimed at addressing the growing number on incoming spam e-mails. The research on different algorithms and their ability to classify e-mail messages needs an update since both tools and spam e-mails have become more advanced. In this study, three different machine learning algorithms have been evaluated based on their ability to correctly classify e-mails as legitimate or spam. These algorithms are naive Bayes, support vector machine and decision tree. The algorithms are tested in an experiment with the Enron spam dataset and are then compared against each other in their performance. The result of the experiment was that support vector machine is the algorithm that correctly classified most of the data points. Even though support vector machine has the largest percentage of correctly classified data points, other algorithms can be useful from a business perspective depending on the task and context.

Nyckelord:

Maskininlärning, Spam e-mail, Textklassificering, Spam e-mailklassificering

Innehållsförteckning

1 Inledning

Denna studie jämför hur väl tre olika övervakade maskininlärningsalgoritmerna presterar när det kommer till att klassificera spam e-mails från legitima e-mails. Algoritmerna som jämförs är naive Bayes (NB), stödvektormaskin (SVM) och beslutsträd (BT). I detta kapitel presenteras bakgrunden samt problemområdet som studien utgår ifrån.

1.1 Bakgrund

E-mails är idag en central kommunikationskanal för verksamheter både internt samt externt (Kiritchenko & Matwin, 2011; Whittaker et al., 2007). Enligt Kiritchenko & Matwin (2011) uppskattas det att en typisk användare får runt omkring 40–50 e-mails dagligen som ska behandlas. Detta har i sin tur lett till att en stor del av e-mail användarens dag går åt till att behandla e-mails. Mailhantering kan upplevas som stressfyllt av dess användare då mycket tid går åt att kategorisera, sortera och behandla inkommande e-mails. En term som har uppstått på senare tid ur detta är "e-mailöverbelastning" (e-mail overload). E-mailöverbelastning används av Dabbish & Kraut (2006) för att syfta på känslan av att ha tappat kontrollen över sin egen inbox med e-mails då det kommer in fler e-mails än vad användaren klarar av att läsa, svara på, samt kategorisera. En bidragande faktor till e-mailöverbelastning är den ökande mängden av spam som skickas idag.

I samband med att e-mails har blivit ett centralt medel för kommunikation har mängden spam e-mails ökat (Kumar et al., 2012). Spam e-mails är en form av e-mails som ofta definieras av sitt identiska innehåll och massutskick till många personer samtidigt (Kumar et al., 2012). Innehållet av spam e-mails är ofta reklam, falska erbjudande eller oönskat samt skadligt innehåll (Panigrahi, 2012). Problemet med att mängden av spam e-mails har ökat de senaste åren är att det krävs allt mer tid av användaren för att sortera och behandla inkomna e-mails.

Både den växande mängden spamutskick samt det ökade tidskravet för sortering och behandling av spam är något som påverkat verksamheter negativt. Enligt Bujang & Hussin (2013) uppskattades 90 procent av alla e-mails som skickades 2007 att vara av typen spam. Detta har lett till ett behov av att klassificera e-mails för att minska mängden spam. Ytterligare ett problem som är centralt för verksamheter är fel klassificering av e-mails, både spam och legitima. Zhou et al., (2014) och Sahami et al., (1998) påstår att det oftast är skadligare för en verksamhet när ett legitimt e-mail klassificeras som spam jämfört med när ett spam e-mail klassificeras som legitimt. Tekniker och verktyg har utvecklat för att försöka motverka den konstant ökande mängden spam som skickas (Bujang & Hussin, 2013; Panigrahi, 2012; Zhou et al., 2014). Ett av de områden som fått mycket uppmärksamhet inom spamklassificering är maskininlärning.

Maskininlärning (Machine learning; ML) är en central del inom området artificiell intelligens (AI). ML handlar om hur en maskin kan lära sig att automatiskt fatta beslut istället för att en människa ska behöva göra det åt den **(**Michie et al., 1994). Inom maskininlärning är textklassificering ett centralt område som handlar om att klassificera text baserat på dess

innehåll. Detta är en teknik som har använts inom forskning de senaste åren för att se på hur klassificering av spam e-mails kan effektiviseras (Kumar et al., 2012; Panigrahi, 2012). Några av de mest populära typerna av klassificeringsalgoritmerna som används idag är naive Bayes, stödvektormaskin, beslutsträd. Andra tekniker som används är klassificering utifrån bildinnehåll då allt fler spam e-mails har börjat använda bilder som innehåll istället för text. Maskininlärning har blivit en viktig del inom spamklassificering tack vare sin förmåga att snabbt och effektivt lära sig att identifiera och klassificera spam. Detta har visat sig vara värdefullt då formatet på spam e-mails är något som ofta ändras i försök att kringgå spamfilter.

1.2 Problembeskrivning

Ett problem som verksamheter möter i samband med den ökande mängden spam är en negativ inverkan på produktion i form av förlorad tid som går åt att filtrera spam (Bujang & Hussin, 2013; Panigrahi, 2012; Zhou et al., 2014). Det tar tid för en anställd att behandla spam e-mails som klassificerats som legitima men framför allt är det ett problem när legitima e-mails klassificeras som spam. Detta skapar problem för verksamheter då anställda kan gå miste om viktig information från kunder eller medarbetare. Detta kan skapa en ond cirkel där anställda är medvetna om att legitima e-mails blir fel klassificerade och känner då behovet av att gå igenom inboxen för skräppost (Bujang & Hussin, 2013). Om den anställde känner ett behov av att konstant behöva kolla sin inbox för skräppost motverkar det poängen med att ha ett spamfilter. Utöver det har spam en negativ inverkan för verksamheters infrastruktur genom att ta upp både nätverksbandbredd samt lagringsutrymme (Bujang & Hussin, 2013). Detta gör att verksamheter måste spendera resurser på att utöka verksamhetens nätverksbandbredd och serverkapacitet. Ytterligare en kostnad inom detta är behovet av att uppdatera och underhålla spamfilter för att möta den konstanta utvecklingen av spam (Bujang & Hussin, 2013).

Klassificering av spam e-mails är en kostsam men nödvändig process för verksamheter vilket gör valet av implementerat spamfilter till en viktig aspekt. En av mest använda teknikerna för spamklassificering i dagsläget är maskininlärning. Detta tack vare dess förmåga att korrekt klassificera e-mails samt förmågan av självinlärning. Maskininlärningsalgoritmer är något som konstant utvecklas med nya förbättringar till algoritmernas logik. Enligt (Mujtaba et al., 2017) har de tidigare studier som jämfört maskininlärningsalgoritmer i syfte av spamklassificering blivit utdaterade i dagsläget. Detta har skapat ett kunskapsgap som denna studie försöker täcka genom att utvärdera tre maskininlärningsalgoritmer av typerna naive Bayes, stödvektormaskin och beslutsträd. Detta är viktigt då allt fler verksamheter har börjat implementera egenutvecklade spamfilter utöver de som redan finns i mailklienterna för att minska mängden fel klassificerade e-mails (Cramer et al., 2009).

1.3 Syfte och frågeställning

Syftet med denna studie är att bidra med kunskap till det gap som finns inom litteraturen angående maskininlärningsalgoritmers förmåga att klassificera spam e-mails. Ytterligare önskas även att identifiera hur implementationen av olika algoritmer kan bidra till en verksamhet. Detta görs genom att utvärdera hur väl algoritmerna naive Bayes,

stödvektormaskin och beslutsträd kan klassificera legitima e-mails och spam e-mails korrekt. Frågeställningen för denna studie är:

- ★ Hur presterar algoritmerna naive Bayes, stödvektormaskin och beslutsträd i uppgiften att klassificera legitima e-mails och spam korrekt?

1.4 Avgränsning

Algoritmerna som valts för studien kommer utvärderas utifrån tre mätvärden i sin förmåga att klassificera e-mails, dessa mätvärden är återkallelse, precision och f1. Vad mätvärdena innebär mer i detalj presenteras i kapitel 3.3.2. Mätvärdena har valts då de är lätta att förstå sig på samt ger ett noggrant resultat angående algoritmernas förmåga att klassificera korrekt (Mujtaba et al., 2017). Dessa tre mätvärden kommer användas för att se över antalet av falskt positiva samt antalet av falskt negativa klassificerade datapunkter. Utvärderingen kommer inte se till algoritmernas tekniska uppbyggnad eller hur snabbt de kan utföra uppgiften. Studien kommer heller inte utvärdera implementationsprocessen av algoritmerna eller hur mycket förkunskap hos en användare som krävs för att bruka dem.

1.5 Kunskapsintressenter

Denna studie kan vara intressant för de verksamheter som planerar att implementera en maskininlärningsalgoritm för att filtrera spam e-mails. Ytterligare kan denna studie vara av värde för forskare inom e-mailklassificerings fältet då studien bidrar med uppdaterad kunskap angående prestationen av de vanligaste algoritmerna inom spam e-mailklassificering.

1.6 Disposition

Denna rapport är strukturerad som följande: Kapitel 2 går igenom begreppsförklaring samt tidigare forskning som gjorts inom området. I kapitel 3 beskrivs den metodologi, strategi samt metod som använts för att genomföra studien. Kapitel 4 presenterar det resultat som framkommit ur experimentet**.** I kapitel 5 görs en analys och diskussion angående resultatet. Slutligen i kapitel 6 presenteras slutsatsen för studien samt tankar angående generalisering, begränsningar, implikationer samt vidare forskning.

2 Utökad bakgrund

I detta kapitel presenteras en begreppsanalys för att förklara de olika termerna som används i studien. Ytterligare kommer tidigare forskning inom ämnesområdet att presenteras.

2.1 E-mails och spamfilter

E-mail eller också känt som elektronisk post (e-post) är digitaliserade meddelanden vilket kan innehålla text, bilder och filer. När ett e-mail har skickats och mottagits kan de individer som meddelandet skickats till, ta del av dess innehåll samt vanligtvis även svara med ett nytt e-mail till avsändaren.

Spam e-mail är ett e-mail som grundar sig i massutskick och återkommande utskick, alltså går samma meddelande till samma individ många gånger om. Dessa spam e-mails innehåller oftast liknande innehåll som legitima e-mails men med avsikt att vilseleda eller lura mottagaren att ge ut information, installera skadliga filer på datorn eller bara irritera och ockupera tid hos mottagaren (Panigrahi, 2012). Detta är något som har gjort att spam e-mails till universellt sedda som oönskade. Problem för verksamheter som uppstått på grund av spam tas upp och diskuteras av Bujang och Hussin (2014). Det främsta problemet de tar upp är minskad produktion i samband med den förlorade tiden som går åt att manuellt sortera spam. Varje spam e-mail som inte blir stoppade av spamfiltret kan ta upp till två minuter av en anställds tid att behandla (Bujang & Hussin, 2013). Detta är något som Mark et al., (2008) pekar ut som ett möjligtvis större problem då ett avbrott från den anställdes originella arbetsuppgift kan medföra ett större avbrott än två minuter. I värsta fall kan ett sådant avbrott ta upp till 15 minuter innan det att den anställda lyckas komma tillbaka till sitt arbetsflöde. Ytterligare ett problem med spamfiltrering är kostnaderna av dess underhåll i samband med behovet av uppdaterad teknik. Trots den kostsamma processen av att underhålla spamfilter är det en nödvändig teknik som finns implementerad inom de flesta verksamheter.

Spamfilter är en mjukvara vilket bearbetar inkommande e-mails via algoritmer för att klassificera dem i två kategorier, dessa kategorier är legitima e-mails och spam e-mails. När ett e-mail klassificeras finns det fyra olika indelningar som beskriver resultatet av klassificeringen, sant positiv (SP), sant negativ (SN), falskt positiv (FP) och falskt negativ (FN). Sant positiv och sant negativ är vad som representerar när algoritmen lyckas klassificera ett objekt korrekt. I fallet av denna studie är sant positivt när ett legitimt e-mail blir klassat som legitimt och sant negativt när ett spam blir klassificerat som spam. Falsk positiv och falsk negativ är då motsatsen, när algoritmen klassificerar objekten fel. Falsk positivt uppstår när algoritmen klassificerar ett spam meddelande som legitimt och falsk negativt när algoritmen klassificerar ett legitimt e-mail som spam.

Tidigare forskning har bedrivits runt binär klassificering, dvs att ett e-mail antingen är spam eller inte (Panigri, 2012; Kumar et al., 2012; Metsis et al, 2006; Balamurugan et al., 2007). Fokus för dessa studier ligger ofta i att lyckas klassificera och minska mängden spam som kommer till inboxen till minsta möjliga antal. Detta är en viktig aspekt av spamklassificering för att motarbeta det ökande antalet spam e-mails som skickas. Som tidigare nämnt uppskattades 90 procent av alla e-mails som skickades 2007 att vara av typen spam (Bujang &

Hussin, 2013). Det vanligaste angreppssättet inom binär klassificering har varit att jämföra olika algoritmer för att avgöra vilken det är som presterar bäst i att klassificera spam och legitima e-mails. NB, SVM och BT är ett par av de algoritmer som dyker upp mest inom studier angående binär klassificering (Mujtaba et al., 2017).

Ett problem som har uppstått ur fokuset på träffsäkerhet är en ökad chans för algoritmen att klassificera ett e-mail till fel kategori (Zhou et al., 2014). Dessa scenarion är vad som kallas för falsk positiv samt falsk negativ, då ett legitimt e-mail blir klassat som spam eller ett spam blir klassat som legitimt. Detta är något som medför negativa konsekvenser då om ett legitimt e-mail blir klassad som spam kan viktig information gås miste om medan om ett spam blir klassat som legitimt kostar det tid och resurser att behandlas (Zhou et al., 2014; Sahami et al., 1998). En lösning på detta har undersökts av Zhou et al., (2014) där de föreslår att istället för att använda binär klassificering, göra en trevägsklassificering med skräppost, legitima och en kategori för när algoritmen är osäker. Resultatet av detta var att mindre e-mails blev fel klassificerade utan hamnade i kategorin av osäkra istället.

Målet för alla spamfilter är som tidigare nämnt att i alla situationer kunna avgöra vad som är legitimt och vad som är spam utan att ta fel på varken eller. Men detta drömscenario är ingenting som har lyckats uppnås i dagsläget (Kumar et al., 2012; Bujang & Hussin, 2013). En av de största faktorerna som gör spam till ett problem idag är dess förmåga att utvecklas och anpassa sig för att undgå moderna spamfilter (Panigri, 2012;). Ett spamfilter kan ej implementeras och sedan är alla problem lösta utan måste konstant uppdateras för att inte falla ur relevans (Sahami et al., 1998; Bujang & Hussin, 2013). Detta är en av anledningarna till att forskning om spamklassificering har varit inriktad på att ta nytta av maskininlärning för att möta utmaningen av den snabba utvecklingen av spam.

2.2 Maskininlärning

Maskininlärning är ett begrepp som används för att beskriva en dators förmåga att lära sig utföra en uppgift. Grunden till maskininlärning ligger i förmågan att försöka få en dator att lära sig saker självmant och minska behovet av att förlita sig på en människas ingripande (Michie et al., 1994). Maskininlärningsalgoritmer är ofta applicerade inom området av dataanalys. Detta är ett resultat av det växande behovet av metoder för att behandla de extrema mängderna data inom big data.

Inlärningsprocessen för maskininlärning ser annorlunda ut beroende på vilken typ av inlärningsmetod som används. De vanligaste metoderna som används idag är övervakad inlärning (supervised learning), oövervakad inlärning (unsupervised learning), semi-övervakad inlärning (semi-supervised learning) (Hall et al., 2014; Rouse & Burns, 2018**)**.

2.2.1 Övervakad inlärning

Övervakad inlärning är en metod som använder sig av klassificerad data för inlärning. Tanken är att algoritmen tränas upp på klassificerad data och använder det som mall för att sedan ta beslut och göra förutsägelser (Hall et al., 2014; Rouse & Haughn, 2018). Exempel på inlärningsprocessen kan vara att algoritmen blir matad med ett antal inputs samt vad som är

korrekt output. Dessa inlärda inputs är sedan vad som utgör mallen som den övervakade inlärningsalgoritmen baserar sina statistiska beslut på. Denna typ av algoritm är passande när målet är att algoritmen ska göra förutsägelser baserade på tidigare data av samma typ eller mönster (Hall et al., 2014).

Övervakad inlärning är ofta det första sättet en algoritm tränas på samt om ett projekt har som mål att uppnå semi-övervakad eller oövervakad inlärning i framtiden (Kotsiantis 2007). Övervakad inlärning är den mest träffsäkra metoden för att utföra en inlärning på då en människa kan snabbt lägga sig i processen och korrigera den vid behov vilket vanligtvis resulterar i en bättre inlärd ML-lösning (Kotsiantis 2007).

2.2.2 Oövervakad inlärning

Oövervakad inlärning är en metod som är baserad i att ge en AI inputs utan att den vet vad outputen ska vara (Hall et al., 2014; Rouse & Haughn, 2016). Det blir då algoritmens uppgift att försöka hitta möjliga mönster i de givna datapunkterna, i den data som blir matad till algoritmen. Detta är den inlärningsmetod som kräver minst mänsklig interaktion men istället kräver störst mängd data för inlärning. Oövervakade inlärningsalgoritmer är lämpade att använda sig av när syftet är att hitta okända mönster i en stor mängd av data (Hall et al., 2014; Rouse & Haughn, 2016).

2.2.3 Semi-övervakad inlärning

Semi-övervakad inlärning är en metod som liknar och används inom samma områden som övervakad inlärning på sådant sätt att algoritmen får en input samt en mall för den korrekta outputen (Hall et al., 2014). Skillnaden från övervakad inlärning ligger i mängden data som används för att lära upp algoritmen. Vid upplärningen av en semi-övervakad inlärnings algoritm används det en mindre mängd klassificerad data och en större mängd av oklassificerad data. Anledningen till detta är ofta en fråga angående kostnad då det är billigare att få tag på oklassificerad data. Lämpliga användningsområden för semi-övervakad inlärning är klassificering och förutsägelser då det ej finns resurser för att genomföra en fullskalig övervakad inlärning (Hall et al., 2014).

2.3 Textklassificering

Ett stort fält inom maskininlärning är förmågan att kunna klassificera text. Det är ett fält som funnits sedan 60-talet men blev först populärt inom maskininlärning under 90-talet i samband med att datorns processorkraft ökade mycket (Sebastiani, 2002). Termen “automatisk textklassificering” har använts på olika sätt inom litteratur för att syfta på olika saker (Sebastiani, 2002). Vissa har använt termen för att utifrån textdokument identifiera potentiella kategorier för dokumentet. Andra syftar på att automatisk textklassificering handlar om förmågan att både identifiera kategorier samt att gruppera dokumenten tillsammans under de identifierade kategorierna. Ytterligare har automatisk textklassificering använts för att syfta på processen att enbart gruppera textdokument baserat på dess innehåll mot ett antal förbestämda kategorier (Sebastiani, 2002; Joachims, 1998). För att inte skapa förvirring hos läsaren har valet

att använda den sistnämnda definitionen gjorts för denna studie. Detta val har gjorts då denna studie endast berör en klassificeringsalgoritmers förmåga att gruppera e-mails som legitima eller skräppost.

Två viktiga aspekter inom textklassificering är val av algoritm samt mängd och varians av data som ska användas för att träna upp algoritmen (Drakos, 2018; Scikit-learn, 2019). Detta är viktigt då olika algoritmer är olika känsliga för vad som kallas för överträning. Överträning är när en algoritm inte lyckas generalisera sin inlärning till data utöver träningsdatat. En orsak till detta är när det data som använts för att träna algoritmen är för specifik eller när för mycket av samma typ av data har använts för inlärning. Det motsatta av överträning, underträning, kan också vara ett problem som uppstår vid inlärning av en klassificeringsalgoritm. Precis som det låter är underträning när den data som använts för upplärning för generell vilket gör att algoritmen inte klarar av uppgiften av att klassificera (Drakos, 2018; Scikit-learn, 2019). För att motverka över och underträning kan olika utvärderingsmetoder användas för att öka generaliserbarheten av de upplärda algoritmerna. I denna studie har k-faldig korsvalidering använts för att minska chanserna att algoritmerna som utvärderats blir över eller undertränade. Hur k-faldig korsvalidering har använts presenteras i kapitel 3.3.1.

Som tidigare nämnt är textklassificering en stor del inom maskininlärning och det har tagits fram flera olika algoritmer för att klassificera text. Tre av de mer populära algoritmerna för textklassificering idag är naive Bayes, stödvektormaskin och beslutsträd (Zhou et al., 2014; Kumar et al., 2012; Panigrahi, 2012).

2.3.1 Naive Bayes

NB har använts inom tidigare studier där algoritmen jämförts mot andra algoritmer i uppgiften av att klassificera e-mails (Rennie et al., 2003; Metsis, 2006). I dessa studier dokumenterades NB algoritmens förmåga att klassificera e-mails efter mängd datapunkter den kunde behandla och hur lång tid NB behövde för att klassificera olika mängder data. Resultatet dessa studier kom fram till var att det finns stor potential för NB klassificerare. De är simpla och lätta att implementera samt kan mäta sig med andra erkända algoritmer (Rennie et al., 2003). Dock har dessa studier inte fokuserat på antalet FP samt FN klassificeringar algoritmen gjort jämfört med de andra algoritmerna. Därmed har NB valts som en av algoritmerna som ska utvärderas i denna studie. NB kan bevisas ha ett annat värde än tidigare bedömt med FP och FN som mätvärden i jämförelse mot andra algoritmer.

NB klassificerare tillhör familjen av sannolikhetsbaserade klassificerare. De första NB baserade algoritmerna utvecklades på 60-talet och har ökat i popularitet som val av textklassificerare inom forskning på de senaste åren (McCallum & Nigam, 1998). Eftersom de flesta e-mails och därmed även de flesta spam e-mails innehåller text har MultinomialNB valts att användas då algoritmen är lämplig för bland annat textklassificering (Scikit-learn 2019).

2.3.2 Stödvektormaskin

SVM har precis som NB använts inom tidigare studier för att se på hur klassificering av e-mails kan effektiviseras (Kiritchenko & Matwin, 2011; Kumar et al., 2012; Rennie, 2003). Enligt

Kiritchenko och Matwin (2011) samt Rennie et al, (2003) presterar SVM bättre än NB när semi övervakad inlärning används. Men i dessa studier där SVM utvärderats har inte FP och FN varit mätvärden för att avgöra algoritmens förmåga att klassificera e-mails. Denna studie kommer bedöma SVM utifrån sin förmåga att klassificera e-mails korrekt där antalet FP och FN kommer vara ett mätvärde för att avgöra algoritmens effektivitet jämfört mot andra algoritmer.

Stödvektormaskin är en linjär modell som används inom övervakad maskininlärning, ofta i syfte av att klassificera text och bilder (Scikit-learn 2019). Linear Support Vector Machine är en algoritm för klassificering som använder en en-mot-resten metod vilket innebär att en datapunkt klassificeras mot resten av hela datasetet innan datapunkten klassificeras inom en kategori (Scikit-learn 2019). Linear Support Vector Machine valdes för att utmana algoritmens goda rykte i fältet när FP och FN blir faktorer som kommer avgöra dess prestation.

2.3.3 Beslutsträd

BT är en typ av algoritm vilket är en övervakad maskininlärningsalgoritm och kan användas för klassificerande uppgifter. Den största motivationen till att använda en algoritm av denna typ är för att bygga och lära upp en träningsmodell som kan förutspå en klass eller ett värde, detta baseras på att modellen lär sig använda urvalsregler från tidigare klassificerad data (Kumar, R et al, 2012). BT algoritmer är lätta att förstå i det att dom har en simpel struktur, alla beslut är tagna efter en ja eller nej frågeställning i noder med olika attribut. Den viktigaste attributen av klassificeringen placeras först i turordning och därefter tillkommer andra attribut vilket till sist ger beslutsträdet möjlighet att klassificera data (Kumar et al., 2012).

CART (Classification and Regression Trees) kommer användas i detta experiment. CART är strukturerad efter binära beslutfattningsträd med funktion för att få ut den största informationsförstärkning vid varje nod (Scikit-learn 2019). Informationsförstärkning innebär att algoritmen ser mönster utefter de val den gör längst med trädet av beslut. Klassificeraren blir säkrare angående ett beslut ju längre ned i trädet en datapunkt färdas. Eftersom CART klassificerar binärt lämpas den att användas i klassificering av datapunkter i två kategorier (Scikit-learn 2019). Tidigare forskning har använt C4.5 i jämförande studier (Kumar et al., 2012; Panigrahi, 2012). I Kumar et al., (2012) visade C4.5 på högre träffsäkerhet än både SVM och BT när den sortera e-mails. CART vilket är en vidareutveckling av C4.5 jämförs sällan i sina prestationer mot andra algoritmer trots att algoritmen är populär att använda för klassificerande uppgifter (Kumar et al., 2012). Därmed har CART valts inför denna studie för att jämföras mot andra algoritmer i dess förmåga att klassificera e-mails korrekt med antalet FP och FN som mätvärden.

2.4 Datasets

Ett dataset är en samling med strukturerade eller ostrukturerade data av en viss typ. Vanliga typer av data är ekonomiska beräkningar, textdokument eller e-mails. Som redan nämnt använder denna studie ett öppet dataset som innehåller spam och legitima e-mails, mer specifikt har Enron-spam valts att användas.

Enron-spam är ett dataset som togs fram och gjordes publikt tillgängligt av Metsis et al. (2006). Datasetet finns tillgängligt som förbehandlat och uppdelat i sex delar, Enron1 - Enron6. Varje indelning av datasetet består av legitima e-mails från Enron datasetet (ska inte blandas ihop med Enron spam datasetet) samt spam e-mails från följande källor: The SpamAssassin corpus, the Honeypot project, the spam collection of Bruce Guenter och spam som insamlats av en av författarna till studien (Metsis et al., 2006). En skillnad på Enron 1–3 och 4–6 är fördelningen av legitima och spam e-mails. I Enron 1–3 är fördelningen 3:1 där legitima meddelande står för 75 procent av antalet e-mails. I Enron 4–6 är fördelningen vänd till 1:3 där spam meddelanden står för 75 procent av alla e-mails. De legitima e-mails i Enron datasetet är meddelanden inom och från det nu nedlagda företaget Enron. Totalt innehåller Enron1 - Enron6 ca 33 000 e-mails varav ungefär hälften är spam e-mails. Datasetet är menat som en resurs för forskare som vill bland annat förbättra verktygen kring och hanteringen av e-mails (Metsis et al., 2006). Enron1 - Enron6 har till exempel använts i en tidigare studie angående klassificering av spam e-mails där NB använts för att uppnå resultatet (Metsis et al., 2006).

3 Design av genomförande

I detta kapitel presenteras och argumenteras de val som gjorts angående forskningsstrategi, datainsamlingsmetod samt metod för dataanalys i studien.

3.1 Forskningsmetodologi

Denna studie har utgått från designforskning som metodologi med design och utvärdering av artefakter för att lösa ett verksamhetsproblem (Hevner et al., 2004; Peffers et al., 2007). Verksamhetsproblemet som denna studie försöker lösa är mängden fel klassificerade e-mails med hjälp av maskininlärning. Denna studie är mer fokuserad i utvärderings aspekten av en artefakt och dess bidrag till verksamhetsnytta än i utvecklingsprocessen. Enligt Hevner et al., (2004) är utvärdering av artefakter av stor vikt inom designforskning då det är genom utvärdering som det visar sig om en artefakt bidrar med ny kunskap eller inte. Utvärdering av artefakter består av två viktiga steg, att testa artefakten mot de krav som bestämts samt att avgöra hur resultat kan generaliseras (Mettler et al., 2014).

3.2 Forskningsstrategi

Forskningsstrategin som används i denna studie är experiment. Experiment är lämpligt att använda sig av när en eller flera artefakter ska jämföras eller utvärderas (Denscombe, 2010; Mettler et al., 2014; Hevner et al., 2004). Anledningen till detta är att experiment grundar sig i att genomföra analyser i en kontrollerad miljö. Detta är något som görs för att minska mängden oförutsägbara faktorer som kan påverka resultatet av ett experiment (Denscombe, 2010). Minskningen av oförutsägbara faktorer är något som anses vara både positivt och negativt. Det positiva är att desto färre oförutsägbara faktorer som påverkar desto lättare blir det att återskapa det genomförda experimentet. Det negativa med det är att trovärdigheten och generaliserbarheten av att experimentet avspeglar ett riktigt scenario minskar (Denscombe, 2010).

Enligt Mettler et al., (2014) finns det tre principer som måste följas när experiment används för att utvärdera artefakter inom designforskning. Dessa principer är följande:

- Kontroll: Kontroll syftar på behovet av att ha en kontrollgrupp i experimentet. Detta är en vanlig förekomst när experiment genomförs och syftar på att det måste finnas en grupp med kontroll data och en grupp med testdata. Skillnaden på de två är att det enbart är testgruppen som bearbetas för att nå ett bättre resultat medan kontrollgruppen hålls undan och används som validering i slutet.
- Slump (randomisation): Slump syftar på att den testdata som används i kontrollgruppen och testgruppen måste vara slumpmässigt utvalt. Detta görs för att minska risken att resultatet påverkas av uppdelningen av data. En viktig aspekt att ha i åtanke när data slumpas är att mängden data måste vara tillräckligt stor för att få en generaliserbar spridning av data. Ett exempel på detta är att om vi har 100 e-mails där 50 är legitima och 50 är spam kan vi utgå från att en slumpmässig indelning till två grupper kommer

ungefär innehålla 50% av respektive typ av e-mail. Om vi däremot bara har 10 e-mails kommer chansen vara högre att det blir en obalans i den slumpmässiga indelningen.
- Manipulation: Manipulation syftar på att artefakten utvärderas mot olika förhållanden. Denna princip följs för att identifiera orsak och effekt inom experimentet. Exempel på detta kan vara att se hur olika mängder av testdata påverkar resultatet.

En alternativ forskningsstrategi som kunde använts i denna studie är fältstudie. En fältstudie är en bra strategi att använda för att utvärdera artefakter enligt Hevner et al., (2004). En fältstudie hade kunnat genomföras genom att gå ut till ett antal verksamheter som använder olika maskininlärningsalgoritmer för spamklassificering. Därifrån samlas data in angående hur väl olika algoritmer presterar i dagsläget i uppgiften av spamklassificering. Anledningen till att detta inte valdes som strategi var främst på grund av tidsramen för studien. Att finna ett flertal verksamheter som använder egen-implementerade spamfilter med olika maskininlärningsalgoritmer för att sedan studera dem är en tidskrävande process. Därför valdes experiment att användas istället för fältstudie då resurserna som krävdes för att genomföra en experimentell strategi var lättare att få tag på.

3.3 Metod

Metoden som använts i denna studie för att genomföra experimentet är samma som används av Kumar (2012), Panigrahi (2012) samt Balamurugan et al., (2007). Metoden som används i dessa tre studier grundar sig i de följande sex elementen:

- Tränings dataset
- Förbehandling
- Attribut urval
- Inlärning
- Testning
- Utvärdering

Överskådlig fungerar metoden på följande sätt: Först sker insamlingen av data som kommer användas för att träna samt testa maskininlärningsalgoritmerna. Inom spamklassificerings området är det vanligt att studier använder sig av publika dataset som finns tillgängliga online. Efter avgörs det om och hur förbehandlingen av datasetet ska göras. Detta steg görs för att minska mängden av onödig data som är orelevant eller störande för experimentets resultat. Exempel på detta kan vara att ta bort stoppord eller genomföra trunkering (att böja ett ord till dess morfologiska rot) på texten. Detta är en process som kan vara av stort värde i fallet av att en textklassificeringsalgoritm används för klassificeringen. När det är gjort genomförs attribut urval för att avgöra vad för element som ska utgöra grunden för klassificeringsalgoritmen. Det vill säga; vad är det som algoritmen ska basera sina beslut på när den klassificerar objekt. En vanlig teknik för attribut urval inom textklassificerare är bag of words (Mujtaba et al., 2017). Bag of words är en modell där ord ur en text samlas och presenteras enskilt utan sammanhang till dess originella mening (Zhang et al., 2010). Därefter kan en klassificeringsalgoritm bli upplärd på orden baserat på hur ofta de används i texterna för varje kategori. När attribut urvalet är klart appliceras de klassificeringsalgoritmer som ska tränas. De sista två stegen i metoden är testning av den upplärda klassificeringsalgoritmen samt utvärdering av resultatet. Testningen av den upplärda algoritmen sker ofta genom att algoritmen blir matad med ny data som den ej

haft tillgång till tidigare. Sedan mäts dess förmåga att lyckas klassificera de nya mängderna av data mot olika värden. Exempel på vanliga värden att mäta för spamklassificering är träffsäkerhet, återkallelse, precision och f1-värde (Mujtaba et al., 2017). Slutligen sammanställs och utvärderas resultatet mot varandra.

3.3.1 Datainsamlingsmetod

I denna studie används k-faldig korsvalidering för att samla in ett mätbart värde av prestationen från de olika klassificeringsalgoritmerna. Korsvalidering är en teknik som ofta används för att utvärdera hur väl en maskininlärningsalgoritm klarar av att klassificera tidigare osedd data (Brownlee, 2018; Drakos, 2018; Scikit-learn, 2019) Valet av k-faldig korsvalidering gjordes då tekniken tar i beräkning två av de tre principer som används inom en experimentell forskningsstrategi. Principerna som k-faldig validering använder sig av är kontroll och slump. k-faldig korsvalidering fungera på följande sätt:

1. Blanda datasetet slumpvis
2. Dela upp datasetet i k antal grupper
3. För varje unik grupp:
    a. Välj gruppen till att vara testgruppen
    b. Välj resterande grupper till att vara träningsgrupper
    c. Lär upp algoritmen på träningsgrupperna och utvärdera sedan mot testgruppen
    d. Behåll värdet av utvärderingen och kassera den upplärda modellen
4. Sammanfatta den upplärda modellens förmåga att klassificera objekt med hjälp av utvärderings värden

Det vanligaste antalet av k-grupper som används för denna typ av utvärdering är fem och tio. Enligt (James et al., 2013) är valet av fem eller tio k-grupper standard då det har empiriskt bevisats att det ger ett resultat som varken lider av att vara partiskt eller av stor varians i sitt resultat.

En annan typ av analysmetod som övervägdes var holdout validering. Holdout validering är också en typ av korsvalidering likt k-faldig. Skillnaden på k-faldig och holdout är att holdout enbart gör en delning av datasetet. Denna delning består då av två grupper, en grupp med träningsdata och en grupp med testdata. En vanlig delning för holdout är att dela på datasetet 80–20 där 80 procent av datasetet blir träningsdata och 20 procent blir testdata (Drakos, 2018). Anledningen till att denna studie använder k-faldig istället för holdout är att holdout vanligtvis lider av en högre varians samt större chans för överträning. Då 80–20 delningen enbart görs en gång när holdout används kommer de 20 procent som valts ut som testdata aldrig att användas som träningsdata. Det vill säga, det är svårt att avgöra om delningen av datasetet är tillräckligt slumpartat och går att generalisera mot osedd data utanför datasetet. Detta är vad som kallas för överträning, när klassificeringsalgoritmen presterar bra på träningsdata men dåligt på tidigare osedd data (Drakos, 2018).

3.3.2 Dataanalysmetod

För att jämföra hur väl de olika klassificeringsalgoritmerna presterar i uppgiften av spam e-mail klassificering har ett antal mätvärden valts ut. De mätvärden som används i denna studie för att jämföra de olika algoritmerna är som tidigare nämnt precision, återkallelse samt f1 värden. Alla mätvärden är baserade på hur väl en algoritm klarar av att klassificera sann positiv, sann negativ, falsk positiv, falsk negativ. Sann positiv (SP) är när en algoritm lyckas klassificera ett legitimt e-mail som legitimt. Sann negativ (SN) är när algoritmen lyckas klassificera ett spam meddelande som spam. Falsk positiv (FP) samt falsk negativ (FN) är när algoritmen klassificerar ett legitimt e-mail som spam respektive ett spam e-mail som legitimt. Återkallelse, Precision samt f1 beräknas på följande sätt:

1. Återkallelse = SP / (SP+FN). Återkallelse är algoritmens förmåga att klassificera positiva objekt som positiva utifrån totalen av ett objekt. Ett exempel på detta skulle vara om ett dataset med 200 e-mails varav 100 är legitima e-mails och 100 är spam e-mails där algoritmen lyckas klassificera de 100 legitima e-mails som legitima kommer testet ha 100 procent återkallelse även om algoritmen klassificerade alla 200 e-mails som legitima.
2. Precision = SP / (SP + FP). Precision representerar värdet av korrekt klassificerade positiva objekt. Detta mätvärde är viktigt att använda för att fastställa hur väl algoritmen kan identifiera positiva objekt. Om exemplet som togs upp ovanför i förklaringen för återkallelse används skulle algoritmen ha 100 procent återkallelse men enbart 50 procent precision. Detta då 50 procent av de e-mails som klassificerats som legitima e-mails egentligen var spam e-mails.
3. f1-värden = 2 * (precision * återkallelse) / (precision + återkallelse). f1-värdet är ett sätt att mäta både precision och återkallelse genom att beräkna deras sammanslagna medelvärde. Både precision och återkallelse väger lika tungt i uträkningen av f1-värden.

Alla mätvärden har skalan 0–1 där 0 är det lägsta och 1 är det högsta. Vid utvärdering av algoritmerna bör alla mätvärden vara närmare 1 för att algoritmen ska värderas högre.

Genom att beräkna precision och återkallelse kan ett resultat dras angående hur väl respektive klassificeringsalgoritmerna lyckades identifiera legitima och spam e-mails.

3.4 Kvalitetskriterier och etik

Som tidigare nämnt finns det tre viktiga aspekter av att genomföra en experimentell forskningsstrategi, kontroll, slump och manipulation (Mettler et al., 2014). Dessa tre aspekter nämns även av Oates (2006) och bör följas för att experimentets resultat ska vara trovärdigt. Metoden som har använts i denna studie har följt dessa tre aspekter. Kontrollgrupper har använts med hjälp av korsvalidering som slumpat data i grupper. Manipulation av enskilda faktorer har skett då Enron spam dataset som användes i denna studie innehåller olika antal legitima samt spam e-mails. Ytterligare en viktig punkt som nämn av Oates (2006) är att data som används i ett experiment bör vara riktigt data och ej egenkonstruerad. Detta bör göras för att resultatet av experimentet ska bli mer generaliserbart. Denna punkt har följts genom användningen av riktiga e-mails och spam e-mails som finns i datasetet, Enron spam.

Det finns två svagheter i denna studie i samband med att en experimentell strategi har använts. Den första svagheten som identifierats utifrån Oates (2006) är behovet av att ha en hypotes. Denna studie har ingen hypotes angående vilken algoritm som kommer prestera bäst. Detta för att flera tidigare studier visat olika resultat vid jämförelse av algoritmer som förekommer i experimentet (Kumar et al., 2012; Kiritchenko & Matwin, 2011). Den andra svagheten med den valda metoden är som tidigare nämnt den negativa aspekten av att genomföra experiment i en kontrollerad miljö. Detta påverkar generaliserbarheten negativt då en kontrollerad miljö inte återspeglar en verklig miljö (Oates, 2006; Mettler et al., 2014).

All forskning måste förhålla sig till etiska ramverk (Oates 2006). Dessa ramverk handlar om det etiska ansvaret som forskare måste förhålla sig till mot deltagare av studien. I denna studie har inga individer deltagit förutom studiens författare, därav har inga extra åtgärder gjorts utifrån ett etiskt perspektiv. Alla verktyg som förekommer i studien är publikt tillgängliga via internet för fri användning, därmed har heller ingen åtgärd gjorts utifrån ett etiskt perspektiv kring verktygen.

4. Resultat

Resultatet som presenteras i tabellerna nedan är det genomsnittliga återkallelse-värdet, precisions-värdet och f1-värdet av den 10-faldiga korsvalideringen. Resultatet är uppdelat i två delar för varje algoritm och mätvärde. Den första delen av tabellen visar varje algoritms prestationer för individuella dataset. Den andra delen av tabellen har tre grupperingar, det totala genomsnittliga resultatet för respektive algoritm och mätvärde för Enron 1–3, Enron 4–6 och Enron 1–6. Denna indelning görs då fördelningen av legitima och spam e-mails skiljer sig för Enron 1–3 och Enron 4–6 (se kapitel 2.4). Resultatet visas i både dess originella form, mätvärdet mellan 0 och 1 samt i motsvarande procentform.

Nedan är tabellen för mätvärdet återkallelse hos samtliga algoritmer, i genomsnitt presterar SVM bättre än de övriga algoritmerna.

Dataset	**Naive Bayes**		**Stödvektormaskin**		**Beslutsträd**	
Enron1	0,594	59,4%	0,990	99,0%	0,903	90,3%
Enron2	0,504	50,4%	0,989	98,9%	0,906	90,6%
Enron3	0,477	47,7%	0,988	98,8%	0,869	86,9%
Enron4	0,998	99,8%	0,998	99,8%	0,980	98,0%
Enron5	1,000	100%	0,999	99,9%	0,976	97,6%
Enron6	0,998	99,8%	0,998	99,8%	0,968	96,8%
Enron1 - Enron3	0.525	52,5%	0,989	98,9%	0,892	89,2%
Enron4 - Enron6	0,999	99,9%	0,998	99,8%	0,975	97,5%
Enron1 - Enron6	0,762	76,2%	0,994	99,4%	0,934	93,4%

Tabell 1 - Genomsnittligt Återkallelse-värde

Nästa tabell är för algoritmernas precisions-värde, återigen är SVM den algoritm som i genomsnitt presterar bäst.

Dataset	**Naive Bayes**		**Stödvektormaskin**		**Beslutsträd**	
Enron1	1,000	100%	0,969	96,9%	0,899	89,9%
Enron2	1,000	100%	0,993	99,3%	0,931	93,1%

Enron3	1,000	100%	0,989	98,9%	0,912	91,2%
Enron4	0,907	90,7%	0,989	98,9%	0,978	97,8%
Enron5	0,952	95,2%	0,992	99,2%	0,975	97,5%
Enron6	0,899	89,9%	0,984	98,4%	0,971	97,1%
Enron1 - Enron3	1,000	100%	0,984	98,4%	0,914	91,4%
Enron4 - Enron6	0,919	91,9%	0,988	98,8%	0,975	97,5%
Enron1 - Enron6	0,960	96,0%	0,986	98,6%	0,944	94,4%

Tabell 2 - Genomsnittligt Precision-värde

Slutligen är tabellen med det genomsnittliga värdet av f1 hos algoritmerna. SVM är även här den mest effektiva och bäst presterande algoritmen.

Dataset	**Naive Bayes**		**Stödvektormaskin**		**Beslutsträd**	
Enron1	0,745	74,5%	0,980	98,0%	0,894	89,4%
Enron2	0,670	67,0%	0,991	99,1%	0,919	91,9%
Enron3	0,645	64,5%	0,989	98,9%	0,889	88,9%
Enron4	0,951	95,1%	0,994	99,4%	0,978	97,8%
Enron5	0,975	97,5%	0,996	99,6%	0,977	97,7%
Enron6	0,946	94,6%	0,991	99,1%	0,969	96,9%
Enron1 - Enron3	0,686	68,6%	0,987	98,7%	0,901	90,1%
Enron4 - Enron6	0,957	95,7%	0,994	99,4%	0,975	97,5%
Enron1 - Enron6	0,822	82,2%	0,990	99,0%	0,938	93,8%

Tabell 3 - Genomsnittligt F1-värde

5. Analys

Resultatet av algoritmernas prestationer visar i en procentuell andel av hur många e-mails för respektive algoritm som klassificerades korrekt. Detta förenklar men förtydligar också algoritmernas prestationer för en utvärdering. En uppskattning av vilken algoritm som bäst lyckas identifiera och klassificera vad som är spam och legitima e-mails kan framställas och frågeställningen för denna studie kan besvaras.

Resultatet för de olika algoritmerna skiljer sig olika mycket beroende på vilket mätvärde som använts. Det mätvärde där algoritmerna skiljer sig mest är återkallelse som presenteras i tabell 1. Naive Bayes står ut till skillnad från SVM och BT då det är extrema skillnader i algoritmens testresultat baserat på vilket av Enron grupperingarna som testats. I Enron 1–3 hade NB ett snittvärde på 52,5 procent återkallelse medan i Enron 4–6 hade NB ett snittvärde på 99,9 procent. Detta betyder att fördelningen av legitima e-mails mot spam som algoritmen tränas på har stor inverkan på hur väl den lyckades identifiera spam från legitima e-mails. Enligt resultatet i tabell 1 var det mer givande för NB att tränas på mer spam än legitima meddelanden. Detta gällde även SVM och BT även om deras förbättringar inte var lika extrema som för NB. Återkallelse snittvärdet för SVM förbättrades 0,9 procentenheter och BT förbättrades med 8,3 procentenheter, vilket fortfarande är en tydlig förbättring även om den ej är lika extrem som för NB.

Resultatet för precision, som presenteras i tabell 2, är mer jämnt mellan de tre algoritmerna jämfört med resultatet av återkallelse. De flesta resultaten ligger över 93 procent med ett par avstickare under 90 procent. Ett resultat som sticker ut är NB:s precision för Enron 1–3 där den lyckades ha 100 procent precision i alla testerna. Det vill säga, NB klassificerade inte ett enda e-mail fel oavsett om det var spam eller legitimt. Detta är intressant med tanke på att enligt NB:s resultat för återkallelse i tabell 1 lyckades den i snitt enbart identifiera 52 procent av alla e-mails i Enron 1–3. Ytterligare ett intressant resultat är att jämföra NB:s återkallelse och precision för Enron 4–6. NB:s återkallelse blev betydligt bättre genom att få tillgång till mer spam vid upplärning än legitima e-mails men i fallet av precision fick NB ett sämre resultat. Precisionen för Enron 4–6 sjönk från 100 procent till ett snittvärde på 91,9 procent. Det som är viktigt att påpeka här är att i Enron 4–6 lyckades NB identifiera 47,4 procentenheter fler e-mails än i Enron 1–3, detta motsvarar 2850 e-mails i datasetet. Detta resultat pekar på två olika saker, antingen är det svårare för NB att klassificera en större mängd e-mails eller att NB får en större nytta inom precision genom att tränas på fler legitima e-mails än spam. Resultatet av precision för SVM och BT är liknande det resultat som de fick på återkallelse. Både algoritmerna förbättrades när en större mängd spam användes vid inlärning. SVM fick enbart en ökning på 0.4 procentenheter från Enron 1–3 till 4–6 vilket pekar på att SVM:s precision inte påverkas lika mycket av antalet spam jämfört med NB och BT. BT gjorde däremot en ökning på 5.9 procentenheter i dess precision när den fick tillgång till fler spam vid inlärning.

f1-värdena som presenteras i tabell 3 är som tidigare nämnt ett värde som representerar en sammanslagning av återkallelse och precision. Resultaten för BT och SVM skiljer sig väldigt lite från de tidigare resultat de fått när precision och återkallelse jämförts separat. Detta då ingen av de två algoritmerna haft speciellt stor varians i mätvärdena baserat på vilket Enron dataset som använts. BT låg runt omkring 90 procent i både återkallelse och precision för Enron 1–3 samt ett medelvärde av 97,5 procent i både återkallelse och precision för Enron 4–6. Detta gör att BT får ett f1-värde på 90,1 procent för Enron 1–3 och ett f1-värde på 97,5 procent för Enron

4–6. SVM har till och med mindre skillnad i dess f1-värde med 98,7 procent för Enron 1–3 och 99,4 procent för Enron 4–6. Anledningen till detta är SVM:s höga mätvärden av precision och återkallelse där det enbart finns ett tillfälle då SVM gick under 98 procent i något mätvärde. Slutligen som för precision och återkallelse varierar f1-värdet mellan de olika Enron dataseten för NB. Det som är intressant att se på för NB:s f1-värde är att det är lågt för Enron 1–3 trots att den hade 100 procent precision i testerna. Anledningen till detta är att NB enbart hade 52,5 procent i snittvärde för återkallelse i Enron 1–3. Utifrån de f1-värden som sammanställts i tabell 3 kan slutsatsen dras att alla tre algoritmerna presterar genomsnittligt bättre i uppgiften av spamklassifikation om datat som använts för inlärning innehåller mer spam än legitima e-mails.

Innehållet av data i de olika dataseten påverkade resultatet för alla mätvärden. Utöver den uppenbara skillnaden med fördelningen av spam och legitima e-mails hade även innehållet av varje dataset inom de fördelningarna påverkan på resultatet. Ännu en gång var det naive Bayes som hade den största variansen av resultatet inom fördelning Enron 1–3 och 4–6. Den största mellanskillnaden för naive Bayes var dess återkallelse mellan Enron 1 och 3 där skillnaden hamnade på 11,7 procentenheter. Men som tidigare nämnt var detta det område där naive Bayes presterade sämst. SVM och BT påverkades inte lika mycket av innehållet i dataseten 1–3 och 4–6 då spridningen av deras resultat sällan gick över 2 procentenheter. Utöver det identifierades enbart ett samband med resultatet och hur datan var indelad och det var att nästan alla algoritmer fick sitt bästa f1-värde i Enron 5. Den enda algoritmen som inte fick sitt bästa resultat var BT som fick ett bättre resultat med 0,1 procentenheter i Enron 4.

5.1 Diskussion

Resultatet av denna studie visar att SVM är den algoritm vilket presterar bäst i att klassificera legitima e-mails och spam e-mails. Detta resultat har tagits fram utifrån f1-värdet som representerar hur många av datapunkterna i ett dataset som identifierats och klassificerats korrekt. I samtliga tester presterade SVM bättre än NB och BT baserat på sitt f1-värde. Andra attribut av algoritmen kan påverka värderingen, men en argumentation kan föras angående förmågan om att klassificera datapunkter efter återkallelse och precision som mest värdefull attribut (Kumar et al., 2012; Rennie, 2003). Därför har SVM bedömts som den algoritm vilket presterar bäst av dem som utvärderats i denna studie. Eftersom 90 procent av alla e-mails som skickas uppskattas att vara spam kostar detta tid och resurser för verksamheter att behandla. Ett spamfilter som kan behandla många e-mails är därav något som önskas för att tillföra verksamhetsnytta (Kiritchenko & Matwin, 2011; Zhou et al, 2014). Men e-mails som klassificerats fel kan komma till att kosta mer tid och mer resurser för en verksamhet vilket styrker vikten i att värdera en algoritm efter sin förmåga att klassificera datapunkter korrekt (Kumar et al., 2012; Zhou et al, 2014). Då legitima e-mails med viktig information inte når användare och när spam e-mails kommer igenom filtret förstör detta förtroendet användare har för spamfiltret (Zhou et al, 2014; Bujang & Hussin, 2013). Detta leder till att användare börjar manuellt behandla sina e-mails och motverkar syftet med spamfiltret. En verksamhet vilket har ett icke fungerande spamfilter måste manuellt behandla e-mails vilket kostar tid och resurser för verksamheten som ifall de aldrig haft ett filter (Zhou et al, 2014).

I denna studie nåddes ett liknande resultat till vad Kiritchenko och Matwin (2011) gjorde i deras jämförande studie av maskininlärningsalgoritmer. Resultatet de kom fram till var att SVM presterade bäst i uppgiften av spamklassificering. Detta är något som skiljer sig från vad Kumar

et al., (2012) kom fram till i deras studie där BT presterade bättre än både SVM och NB. I denna studie har BT ett jämt resultat genom alla testningar men presterade under SVM och över NB i snitt. Ytterligare visar denna studies resultat att NB inte riktigt har kommit ifatt de andra algoritmernas prestanda ännu. Detta resultat går att likna till vad Rennie (2003) kom fram till i och med att NB har potential att mäta sig med algoritmer som SVM. Denna studie bevisar att det stämmer då NB hade högst mätvärde men inom specifika områden. Det som skiljer de tidigare studierna från denna är att de främst fokuserat på träffsäkerhet som mätvärde och inte precision, återkallelse och f1-värde. Detta är dock något som stärker denna studie genom att visa på att algoritmerna kan prestera bra inom andra mätvärden än bara träffsäkerhet.

En fråga som uppstod efter studiens experiment är om en så kallad multi-layer algoritm hade tillfört mer nytta till verksamheter. Det finns potential med NB då resultatet i denna studie visar att NB klassificerade vissa dataset med 100 procent precision även om SVM överlag presterar bättre. Om NB skulle arbeta tillsammans med SVM för samma dataset hade möjligtvis ett bättre resultat framställts. Ett ideal för verksamhetsnyttan i att klassificera e-mails vore såklart att både kunna behandla alla inkommande e-mails och klassificera dem korrekt. Enligt Kumar et al., (2012) är detta inget som ser ut att vara möjligt inom en snar framtid**.** Men vidare forskning kring multi-layer algoritmer rekommenderas då det kan leda till att en dag uppfylla 100 procent identifierade och korrekt klassificerade datapunkter.

I behandling av dataset där fler spam e-mails förekommer presterar samtliga algoritmer bättre i denna studie. I experimentet för denna studie visar NB och BT störst skillnad i sitt resultat efter att ha blivit presenterade med dataset vilket innehåller mer spam än legitima e-mails. Detta kan vara relaterat till att algoritmerna är under- eller övertränade och skulle behöva använda mer eller mindre testdata under inlärningsprocessen. Detta stödjs av att SVM som inte är lika känslig för över- samt underträning visade ett jämnare resultat oavsett antal spam e-mail i datasetet. Detta stärker algoritmens förtroende för sin förmåga att klassificera e-mails korrekt, en större förmåga att klassificera datapunkter korrekt betyder en större verksamhetsnytta (Zhou et al, 2014; Bujang & Hussin, 2013).

6. Slutsats

Följande kapitel är slutsatsen av denna studie där en sammanställning nämns följt av generaliserbarheter, begränsningar, implikationer för forskning och praktiken samt avslutande ord kring vidare forskning i området.

6.1 Sammanställning

Denna studie besvarar frågeställningen kring vilken algoritm bland NB, SVM och BT som presterar bäst i klassificeringen av e-mails. Resultatet efter studiens experiment tyder på att SVM presterade bäst jämfört mot NB och BT. SVM utför sin uppgift med ett snittvärde av 99,4 procent återkallelse, 98,6 procent precision och därmed ett f1-värde på 99 procent. Antalet korrekt klassificerade datapunkter har identifierats som den viktigaste aspekten i utvärderingen av ett spamfilters verksamhetsnytta. Det har konstaterats att ett spamfilter motverkar överbelastning av e-mails samt minskar mängden inkommande spam e-mails vilket är givande för både resurs och tidssparande inom en verksamhet. Den insparade tiden och resurserna kan användas för att utbilda och underlätta arbetet för personal.

6.2 Generaliserbarhet

Som tidigare nämnt är en experimentell strategi en bra strategi att använda sig av för att jämföra olika artefakter då målet är att minska mängden oförutsägbara faktorer som spelar in på resultatet. Men genom att minska mängden oförutsägbara faktorer påverkas även generaliserbarheten av resultatet från experimentet (Denscombe, 2010).

Enligt Lee och Baskerville (2003) finns det fyra olika sätt att generalisera en studie. Det första sättet är från empiriska uttalanden till empiriska uttalanden. Det andra är från empiriska uttalanden till teoretiska uttalanden. Det tredje är från teoretiska uttalanden till empiriska uttalanden och slutligen det fjärde sättet, från teoretiska uttalanden till teoretiska uttalanden. I denna studie har det förstnämnda sättet använts för att generalisera data till ett mätbart värde och observationer.

Två faktorer som stärker generaliserbarheten i denna studie är den slumpmässiga indelningen av data för träningen av algoritmerna samt att testning gjordes för båda spridningarna av data i dataseten. Det vill säga, att tester gjordes där legitima e-mails stod för majoriteten av data samt tvärtom. Dessa faktorer är något som tas upp av Lee och Baskerville (2003) som viktiga när empiriska uttalanden generaliseras till empiriska uttalanden.

6.3 Begränsningar

Det finns två begränsningar som identifierats i denna studie. Den första begränsningen är mängden av algoritmer som utvärderats i denna studie, ett större antal hade framställt en bredare

insikt bland algoritmernas förmågor och ett säkrare resultat. Det finns fler maskininlärningsalgoritmer som kan användas för uppgiften av textbaserad klassificering än de som utvärderades i denna studie. Anledningen till att enbart tre algoritmer valdes var på grund av tidsramen som studien hade. Den andra begränsningen inom denna studie var författarnas kunskap angående hur logiken bakom algoritmerna mer i detalj fungerar. Detta har gjort att djupare analyser angående implementation ej har kunnat utföras, istället har fokus placerats kring analys av testresultaten i samband med verksamhetsnyttan algoritmerna tillför.

6.4 Implikationer för forskning och praktiken

Verksamhetsnyttan som en algoritm tillför en organisation beror på vilket syfte algoritmen ska fylla och vad som är verksamhetskravet för algoritmen. Det betyder att värdet av en algoritm är subjektivt från ett verksamhetsperspektiv. Om en verksamhet kan acceptera att ett visst antal datapunkter klassificeras fel eller att alla datapunkter inte behandlas påverkar detta hur verksamheten värderar algoritmen för just deras process av klassificering. Studien har endast värderat algoritmer efter en aspekt av många fler som kan komma till att spela roll för en verksamhets egen utvärdering. Dock kan argument föras över att antalet rätt klassificerade datapunkter vilket är vad denna studie fokuserat på väger mer i de flesta fall jämt med andra aspekter.

6.5 Framtida arbete

För framtida arbeten relaterat till klassificering av e-mails rekommenderas att multi-layer algoritmer testas och används. Med det kan en bedömning ske över vilken kombination av algoritmer som fungerar mest effektiva tillsammans för att klassificera datapunkter. Framtida liknande studier bör även utföra experiment med fler algoritmer, andra dataset och ytterligare mätvärden än vad som dokumenterades i denna studie. Detta för att få en bredare bild av vilken potential algoritmernas prestanda når och därmed få en tydligare bild över algoritmernas förmåga av spamklassificering i dagsläget.

7. Källförteckning

Balamurugan, S. A., Rajaram, D. R., Athiappan, G., & Muthupandian, M. (2007). Data mining techniques for suspicious email detection: A comparative study. In *IADIS European Conference Data Mining*.

Bujang, Y. R., & Hussin, H. (2013, March). Should we be concerned with spam emails? A look at its impacts and implications. In *2013 5th International Conference on Information and Communication Technology for the Muslim World (ICT4M)*(pp. 1-6). IEEE.

Cramer, H. S., Evers, V., Van Someren, M. W., & Wielinga, B. J. (2009, April). Awareness, training and trust in interaction with adaptive spam filters. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 909-912). ACM.

Dabbish, L. A., & Kraut, R. E. (2006, November). Email overload at work: an analysis of factors associated with email strain. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work* (pp. 431-440). ACM.

Denscombe, M. (2014). *The good research guide: for small-scale social research projects*. McGraw-Hill Education (UK).

Georgios Drakos. (2018-08-16) *Cross-Validation.* Towards Data Science. [Blog] Hämtad 2019-05-13 från: https://towardsdatascience.com/cross-validation-70289113a072

Hall, P., Dean, J., Kabul, I. K., & Silva, J. (2014). An overview of machine learning with SAS® enterprise miner™. *SAS Institute Inc*.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.

Jason Brownlee (2018-05-23) *A Gentle Introduction to k-fold Cross-Validation.* Machine learning mastery. [Blog] Hämtad 2019-05-13 från: https://machinelearningmastery.com/k-fold-cross-validation/

Joachims, T. (1998, April). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning* (pp. 137-142). Springer, Berlin, Heidelberg.

Kiritchenko, S., & Matwin, S. (2011, November). *Email classification with co-training.* Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research (pp. 301-312). IBM Corp.

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, *160*, 3-24.

Kumar, R. K., Poonkuzhali, G., & Sudhakar, P. (2012, March). Comparative study on email spam classifier using data mining techniques. In *Proceedings of the International MultiConference of Engineers and Computer Scientists* (Vol. 1, pp. 14-16).

Lee, A. S., & Baskerville, R. L. (2003). Generalizing generalizability in information systems research. *Information systems research*, *14*(3), 221-243.

Mark, G., Gudith, D., & Klocke, U. (2008, April). The cost of interrupted work: more speed and stress. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*(pp. 107-110). ACM.

McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization* (Vol. 752, No. 1, pp. 41-48).

Mettler, T., Eurich, M., & Winter, R. (2014). On the Use of Experiments in Design Science Research: A Proposition of an Evaluation Framework. *CAIS*, *34*, 10.

Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006, July). Spam filtering with naive bayes-which naive bayes?. In *CEAS*(Vol. 17, pp. 28-69).

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). Machine learning. *Neural and Statistical Classification*, *13*.

Mujtaba, G., Shuib, L., Raj, R. G., Majeed, N., & Al-Garadi, M. A. (2017). Email classification research trends: Review and open issues. *IEEE Access*, *5*, 9044-9064.

Oates, B. J. (2006). *Researching information systems and computing*. London;Thousand Oaks, Calif;: SAGE Publications.

Panigrahi, P. K. (2012, November). A comparative study of supervised machine learning techniques for spam e-mail filtering. In *2012 Fourth International Conference on Computational Intelligence and Communication Networks* (pp. 506-512). IEEE.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, *24*(3), 45-77.

Rennie, J. D., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the poor assumptions of naive bayes text classifiers. In *Proceedings of the 20th international conference on machine learning (ICML-03)* (pp. 616-623).

Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998, July). A Bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the 1998 workshop* (Vol. 62, pp. 98-105).

Scikit-learn. (2019). *Cross-validation: evaluating estimator performance.* Hämtad 2019-05-13 från: https://scikit-learn.org/stable/modules/cross_validation.html

Scikit-learn. (2019). *Scikit-learn: machine learning in Python*. Hämtat 2019-05-16 från https://scikit-learn.org/stable/

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, *34*(1), 1-47.

Von Alan, R. H., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, *28*(1), 75-105.

Whittaker, S., Bellotti, V., & Gwizdka, J. (2007). Email and PIM: Problems and possibilities. *Availiable at: http://www. researchgate. net/publication/246050153_Email_and_PIM_Problems_and_Possibi lities*.

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, *1*(1-4), 43-52.

Zhou, B., Yao, Y., & Luo, J. (2014). Cost-sensitive three-way email spam filtering. *Journal of Intelligent Information Systems*, *42*(1), 19-45.