EXAMENSARBETE INOM DATATEKNIK,
AVANCERAD NIVÅ, 30 HP
STOCKHOLM, SVERIGE 2017

Utvärdering av nyckelordsbaserad textkategoriseringsalgoritmer

VIDE KARLSSON

KTH
SKOLAN FÖR DATAVETENSKAP OCH KOMMUNIKATION

Utvädering av nyckelordsbaserade textkategoriseringsalgoritmer

An evaluation of keywordbased text categorization

Vide Karlsson

ragkar@kth.se

Handledare: Pawel Herman & Jussi Karlgren

Examinator: Anders Lasner

Civilingenjörsprogrammet i datateknik

2017 - 01 - 22

Svensk sammanfattning

Maskininlärningsalgoritmer med övervakad inlärning har visat sig ge mycket goda resultat på uppgiften att automatiskt ämneskategorisera text. Övervakad inlärning kräver dock stora mängder manuellt etiketterad träningsdata vilket ofta kan utgöra ett hinder för praktisk tillämpning. Nyckelordsbaserade textkategoriseringsalgoritmer har i litteraturen lyfts fram som ett attraktivt alternativ, eftersom de ej kräver manuellt etiketterad träningsdata. I detta examensarbete undersöks om det existerar andra hinder för att praktiskt tillämpa nyckelordsbaserad textkategorisering. I studien testas även om en lexikal resurs baserad på ords paradigmatiska relationer kan komplettera befintliga metoder för nyckelordsframställning. Ett industriellt motiverat användingsfall togs fram för att mäta praktisk tillämpbarhet. Resultaten visade att ingen av de fem här granskade textkategoriseingsalgoritmerna klarade av att möta kraven i det industriellt moiverade användingsfallet. Men en algoritm föreslagen av Liebeskind, Kotlerman och Dagan (2015) kunde modifieras för att klara kraven. Den nya lexikala resursen gav relevanta nyckelord men algoritmens kategoriseringsförmåga varierade likväl stort mellan olika ämneskategorier och var generellt för låg för de flesta praktiska tillämpningar. Framtida studier behövs för att kartlägga hur kategoriseringsförmågan kan förbättras.

Abstract

Supervised learning algorithms have been used for automatic text categorization with very good results. But supervised learning requires a large amount of manually labeled training data and this is a serious limitation for many practical applications. Keyword-based text categorization does not require manually labeled training data and has therefore been presented as an attractive alternative to supervised learning. The aim of this study is to explore if there are other limitations for using keyword-based text categorization in industrial applications. This study also tests if a new lexical resource, based on the paradigmatic relations between words, could be used to improve existing keyword-based text categorization algorithms. An industry motivated use case was created to measure practical applicability. The results showed that none of five examined algorithms was able to meet the requirements in the industrial motivated use case. But it was possible to modify one algorithm proposed by Liebeskind et.al. (2015) to meet the requirements. The new lexical resource produced relevant keywords for text categorization but there was still a large variance in the algorithm's capacity to correctly categorize different text categories. The categorization capacity was also generally too low to meet the requirements in many practical applications. Further studies are needed to explore how the algorithm's categorization capacity could be improved.

Innehåll

Kapitel 1

Introduktion

De senast decennierna har vi sett en dramatisk ökning i mängden elektroniskt lagrad text. Elektroniskt lagrad text utgör ett samlingsbegrepp för t.ex. webbsidor, text i social media, nyhetsartiklar och e-mail. Ett utmärkande drag hos elektroniskt lagrad text är att den ofta har en mindre formell struktur än tryckt text. Publikationsflödet av elektroniskt lagrad text är också högre än för tryckt text (Baharudin, Lee & Khan, 2010).

Många av de analyser man kan tänkas vilja göra på flödet av elektronisk lagrad text förutsätter dock att man kan strukturera dokumenten och skilja ut texter som tillhör en viss ämneskategori (Baharudin m. fl., 2010; Sebastiani, 2002).

Att kategorisera text för hand har visat sig vara ett resurskrävande arbete. Metoder för att automatisera textkategorisering har därför blivit ett viktigt forskningsområde inom det språkteknologiska forskningsfältet (Baharudin m. fl., 2010).

Tillämpningarna av automatisk textkategorisering, finns inom vitt skilda områden. Effektiv automatisk ämneskategorisering av text är till exempel användbart om man vill analysera opinioner och åsiktsyttringar om ett specifikt ämne i textet tillgängliga på Internet, som sociala medier, recensionstexter eller nyhetstexter (Liebeskind m. fl., 2015). Ett annat finns inom kundtjänstföretag där det är en fördel om frågor sända över mejl automatiskt kan matchas mot redan kända ämneskategorier (Liebeskind m. fl., 2015).

Inom det språkteknologiska forskningsfältet för automatisk textkategorisering har man framförallt fokuserat på att undersöka hur maskininlärningsalgoritmer som använder övervakad inlärning kan användas för att ämneskategorisera text. Att de algoritmerna utgjort fältets fokus kan förklaras av att de uppnått

mycket goda resultat, som i många fall visat sig likvärdiga med den samstämmighet man uppnår när människor utför kategoriseringen (Aggarwal & Zhai, 2012; Baharudin m. fl., 2010; Sebastiani, 2002). Men ett inneboende problem med algoritmer som använder övervakad inlärning är att de kräver stora mängder manuellt framställd träningsdata och att detta i många fall utgör ett reellt hinder för att praktiskt tillämpa algoritmen (Liebeskind m. fl., 2015).

Hur man bäst löser ämneskategorisering av text då det inte finns manuellt etiketterad träningsdata att tillgå är därför ett problem som behöver kartläggas ytterligare. Nyckelordsbaserad textkategorisering har i ett flertal studier lyfts fram som en möjlig lösning på detta problem (Barak, Dagan & Shnarch, 2009; Gliozzo, Strapparava & Dagan, 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu, Zhang, Zhu & Qu, 2009). Syftet med detta examensarbete är att utvärdera om det existerar några hinder för att praktiskt tillämpa de nyckelordsbaserade algoritmer som presenteras i artiklarna ovan. Som grund för utvärderingen kommer ett industriellt motiverat användningsfall användas.

1.1 Uppdragsgivaren

Gavagai är ett företag i Stockholm som säljer produkter relaterade till automatisk textanalys. En av Gavagais viktigare produkter är sentimentanalys eller tonalitetsanalys, där Gavagai analyserar texter som skrivs om ett visst ämne för att härleda skribenternas attityd eller känsla inför det givna ämnet.

De system Gavagai besitter idag är optimerade för att kunna analyser sentiment och tonalitet i de stora textmängder som publiceras kontinuerligt på Internet. Dagens system kan ge en relativt god bild av sentimentet i meningar där ord relaterade till ämneskategorin nämns. Men ett ökande antal kunder har börjat efterfråga en mer noggrann analys av de texter som ingått i sentimentanalysen. Ett exempel kan vara att företag som gjort sentimentanalys på meningar som innehåller företagsnamnet önskar att texterna som ingått i analysen ska kategoriseras baserat på de verksamhetsområden företaget har. Detta är en tjänst som Gavagai inte tillhandahåller idag och som skulle förutsätta ett system för ämneskategorisering av text.

Få kunder besitter stora mängder manuellt etiketterad text för de ämneskategorier de önskar detektera därför vill Gavagai få frågan undersökt hur ämneskategorisering av text kan genomföras då det ej finns manuellt etiketterad träningsdata att tillgå.

1.2 Frågeställning

Nyckelordsbaserad textkategorisering har i litteraturen lyfts fram som ett mer praktiskt tillämpbart och närmast likvärdigt alternativ till textkategorisering med övervakad inlärning (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009). Detta har motiverats med att olika nyckelordsbaserade metoder visat sig ha nära likvärdig kategoriseringsförmåga på klassiska testdataset för textkategorisering.

Faktumet att nyckelordsbaserade textketegoriseringsalgoritmer inte kräver manuellt etiketterad träningsdata och kan ge likvärdiga resultat på vissa kategoriseringsuppgifter, är dock ingen garant för att det inte existerar andra hinder för praktisk tillämpning. Sammantaget är analysen av dagens befintliga nyckelordsbaserade textkategoriseringsmetoder inte tillräcklig för att kunna besvara frågan:

> *I vilka användningsfall är nyckelordsbaserad textkategorisering praktiskt tillämpbart?*

Syftet med detta examensarbete är därför att bidra till att besvara denna övergripande frågeställning.

Examensarbetet genomförs i ett samarbete med företaget Gavagai som arbetar med att ta fram produkter relaterade till automatisk textanalys. Utifrån Gavagais behov har ett industriellt motiverat användningsfall tagits fram. Detta användningsfall utgör examensarbetets definition av praktisk tillämpbarhet och används för att utvärdera den praktiska tillämpbarheten hos de nyckelordsbaserade textkategoriseringsalgoritmer som föreslagits i litteraturen.

Specifikt är målsättningen i detta examensarbete att besvara frågorna:

1. Kan någon av de nyckelordsbaserade kategoriseringsalgoritmerna i de här listade studierna (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009) möta kraven i det framtagna användningsfallet?
2. Kan nyckelord från Gavagais Living Lexicon (Sahlgren m. fl., 2016) användas för att öka kategoriseringsförmågan hos den befintliga nyckelordsbaserade algoritm som visar sig bäst möta kraven i det framtagna användningsfallet?

1.3 Översikt

Denna uppsats syftar till att utvärdera den praktiska tillämpbarheten hos ett antal nyckelordsbaserade textkategoriseringsalgoritmer samt undersöka vilken potential det finns att öka den praktiska tillämpbarheten ytterligare genom att föreslå ett antal algoritmförändringar för befintliga nyckelordsbaserade textkategoriseringsalgoritmer.

Avsnitt 2 nedan beskriver bakgrunden till de experiment som genomförts. Bakgrunden är uppdelad i fem delar:

1. I första avsnittet 2.1 motiveras varför det finns behov av nyckelordsbaserade textkategoriseringsalgoritmer.
2. I avsnittet 2.2 beskrivs de generella gemensamma dragen i nyckelordsbaserad textkategorisering. Här presenteras också de algoritmer som kommer granskas inom ramen för det här examensarbetet.
3. Avsnitt 2.3 beskriver vilka metoder som tidigare använts för att utvärdera textkategoriseringsalgoritmer, därefter diskuteras varför de metoderna inte är tillräckliga för att besvara frågor om algoritmers praktiska tillämpbarhet.
4. I avsnitt 2.4 presenteras det industriellt motiverade användningsfall som tagits fram för att kunna mäta den praktiska tillämpbarheten hos de granskade algoritmerna. Därefter följer granskningen av hur väl de olika algoritmerna uppfyller kraven i användningsfallet.
5. I avsnitt 2.5 Presenteras i detalj Dice algoritmen som är den algoritm som bäst klarar de uppsatta kraven i det framtagna användingsfallet
6. I avsnitt 2.6 beskrivs Gavagai Living Lexicon som kommer användas i experiment för att försöka öka den praktiska tillämpbarheten av nyckelordsbaserad textkategorisering.

Avsnitt 3 utgör studiens metod-avsnitt. Här presenteras ett antal algoritmförändringar för den algoritm som granskningen ovan visat har störst potential att möta kraven i det framtagna användningsfallet. Efter förslagen på algoritmförändringar presenteras de metoder som kommer användas för att utvärdera effekterna av förändringarna.

I avsnitt 4 presenteras resultaten av de ovan föreslagna experimenten.

I avsnitt 5 förs en diskussion kring resultaten av de genomförda experimenten och vilka områden som utgör relevanta spår för framtida forskning.

Kapitel 2

Bakgrund

Intresset för automatisk textkategorisering har växt i takt med att mängden elektroniskt publicerad text ökat dramatiskt de senaste decennierna (Baharudin m. fl., 2010).

En mångfald av algoritmer har prövats för att ämneskategorisera text. Idag råder dock närmast konsensus i forskningsfältet om att kategoriseringsalgoritmen support vector machine (SVM) (tränad på manuellt kategoriserade texter) är den algoritm som ger bäst resultat på ämneskategorisering av text (Aggarwal & Zhai, 2012; Sebastiani, 2002; Yang & Liu, 1999).

Resultaten för automatisk textkategorisering med hjälp av SVM varierar givetvis mellan olika testdataset och är därutöver beroende av hur man valt att representera träningsdata. Men många studier har kunnat presentera resultat som är nära eller likvärdiga med den samstämmighet som uppnås då oberoende människor manuellt ämneskategoriserar text (Aggarwal & Zhai, 2012).

2.1 Varför nyckelordsbaserad textkategorisering?

Övervakad inlärning med manuellt etiketterad träningsdata har visat sig vara den metod som ger bäst kategoriseringsresultat för ämneskategorisering av text. Men det finns flera skäl till att övervakad inlärning kan vara omöjligt att tillämpa i praktiken och flera skäl till varför nyckelordsbaserad textkategorisering utgör ett intressant alternativ att undersöka närmare.

2.1.1 Begränsningar med övervakad inlärning

De goda resultaten för maskininlärnings algoritmer som använder manuellt framställd träningsdata, har sannolikt varit en starkt bidragande orsak till att många studier om automatisk textkategorisering fokuserat på att ytterligare optimera detta tillvägagångssätt.

Men att vara beroende av manuellt etiketterad data utgör av flera skäl ett reellt hinder i många tänkbara tillämpningar av automatisk textkategorisering:

Manuell kategorisering av träningsdata: Att manuellt kategorisera träningstexter är ett mödosamt och tidskrävande arbete. McCallum, Nigam, Rennie och Seymore (1999) framställde manuellt ett test data set för ämneskategorisering av forskningsartiklar baserat på titel och sammanfattning. De rapporterar att det tar en människa ca 90 minuter att kategorisera 100 texter av denna längd.

För att garantera goda resultat krävs mycket träningsdata: När text ska representeras som träningsdata får datan snabbt ett högt antal dimensioner, anledningen är att frekvensen av varje relevant term i normalfallet behöver utgöra en egen dimension (Aggarwal & Zhai, 2012) . Om man har ett högt antal dimensioner på indata så krävs det typiskt sätt ett stort antal träningstexter för att uppnå en god generaralisering hos kategoriseringsalgoritmerna (Marsland, 2015)). Tester på olika annoterade dataset för textkategorisering (så som 20-Newsgroup och Reuters-10) har visat att antalet texter som krävs för att uppnå maximal korrekthet i kategoriseringen variera mycket för olika ämneskategorier. Schohn och Cohn (2000) fann att med kategoriseringsalgoritmen SVM krävdes det minst ett hundra-tal och för vissa kategorier över tusen positiva exempel för att uppnå maximal korrekthet i kategoriseringen av de två kända dataseten 20-Newsgroup och Reuters. Resultaten i i studien av Schohn och Cohn (2000) visar vidare att korrektheten typiskt sätt når en tröskelnivå där ett större antal postiva exempel inte förbättrar resultaten. Korrektheten sjunker dock snabbt om antalet positiva exempel är förre än det antal som krävs för att uppnå tröskelnivån.

Om en ämneskategori kräver över 1000 positiva träningsexempel för att uppnå sitt tröskelvärde och samtidigt förekommer sparsamt i den korpus som används för att framställa träningsdata, kommer det krävas att man manuellt etiketterar flera tusentals texter för att uppnå en tillräcklig mängd data för den enskilda ämneskategorin. Utöver detta krävs ytterligare etiketterade texter som kan användas som testdata för att validera att den mängd träningsdata man framställt är tillräcklig.

Etiketterade träningsdata är en färskvara: En nyligen genomförd studie av Rocha m. fl. (2013) undersökte hur tidpunkten då texten skrevs på-

verkade resultatet på ämneskategorisering. De använde sig av två stora dataset som både framställts över ca 20 år och där kategoriseringen gjorts över tid. De kunde se att om man valde träningsdata som skrivits vid en närliggande tidpunkt som testdata förbättrades andelen korrekt kategoriserade träningstexter med upp till 10% jämfört med att använda all tillgänglig träningsdata från samtliga år. Detta är anmärkningsvärt då system för automatiskt kategorisering generellt vinner på att använda stora mängder träningsdata. Samtidigt belyser det att definitionen av ämnen förändras över tid och att det därför finns ett behov av att kontinuerligt byta ut den mängd träningsdata man använder sig av ifall man vill uppnå bra resultat. Praktiskt innebär detta att manuell kategorisering måste göras om frekvent och att man behöver ytterligare system som detekterar när det finns ett behov av ny träningsdata. Med en fullt automatiserad textkategorisering är inget ytterligare system nödvändigt för att detektera förändringar i ämnesdefinitionen eftersom modellen för kategorisering kan uppdateras kontinuerligt i takt med att ny data tillkommer.

Utifrån de här listade punkterna är det lätt att se att kostnaderna för att framställa och underhålla relevanta träningsdataset kan blir höga för flera relevanta tillämpningsfall av automatisk textkategorisering.

Många av de kunder Gavagai träffar ser specifikt ett behov av att relativt frekvent och med kort varsel kunna byta ut den palett av ämneskategorier de önskar detektera. En lösning som bygger på övervakad inlärning innebär därför ständigt tillkommande kostnader för att framställa nya träningssätt för ny tillkomna kategorier.

I många tillämpningarna som är aktuella för Gavagai rör det sig om nyhetstext som ska kategoriseras. I nyhetstext kan sättet att skriva om ett ämne som t.ex. terrorism förändras markant utifrån världsutvecklingen, vilket gör att man även konsekvent behöver lägga manuellt arbete på utvidga och ersätta gamla textter i befintliga träningsdataset, om man väljer övervakad inlärning.

2.1.2 Alternativ till textkategorisering med manuellt etiketterad träningsdata

Som en reaktion på de begränsningar som övervakad inlärning medför har ett växande antal studier börjat underska hur automatisk textkategorisering kan genomföras när det inte finns stora mängder manuellt etiketterad träningsdata att tillgå.

Övergripande kan de lösningar som föreslagits på detta problem delas upp i tre paradigm:

1. **Aktiv inlärning:** I normalfallet vid textkategorisering används en slumpartad mängd texter vid skapandet av ett träningsdataset, vid aktiv inlärning använder man istället metoder för att automatiskt välja ut de exempeltexter som är mest relevanta för algoritmens förmåga att urskilja den sökta kategorin, och kategoriserar enbart de texterna. Aktiv inlärning har tillämpats på textkategorisering och visat sig kunna reducera behovet av manuellt kategoriserade träningstexter betydligt (Schohn & Cohn, 2000; Tong & Koller, 2002)

2. **Klustring för kategorisering:** Adami, Avesani och Sona (2003) presenterade en metod för textkategorisering som bygger på att klustra text och därefter detektera de kluster som bäst representera den kategori av texter som man vill detektera.

3. **Nyckelordsbaserade metoder:** Vid nyckelordsbaserade metoder utgår man typiskt sätt enbart från kategorinamnet för att med olika metoder detektera semantiskt närliggande ord (nyckelord). Nyckelorden används i nästa steg som en representation för kategorin för att avgöra om en text tillhör den givna kategorin eller ej.

I detta examensarbete kommer fokus ligga på att kartlägga, granska och vidareutveckla nyckelordsbaserade metoder.

Nyckelbaserade metoder har flera fördelar jämfört med de två andra föreslagna metoderna för att möta problemet med bristande mängd träningsdata.

Till skillnad från aktiv inlärning inbegriper nyckelordsbaserade metoder inte någon manuell kategorisering av texter. Detta gör att man kan skapa och uppdatera kategorirepresentationerna kontinuerligt utan att det kräver mer än en validering av de nyckelord som tillkommer (Ko & Seo, 2009; Liebeskind m. fl., 2015).

De framtagna nyckelorden utgör även en transparent representation av hur systemet modellerar den givna kategorin. Detta gör det lättare att analysera ev. brister i systemet än vid aktiv inlärning och klustring där det är svårare att analysera varför en given text kategoriserats fel av systemet (se avsnitt 2.3.3 om hur felanalys kan genomföras och användas för att förbättra nyckelordsbaserade metoder) .

De system Gavagai besitter idag har också störst möjlighet att användas för att komplettera och förbättra nyckelordsbaserade kategoriseringsalgoritmer.

2.2 Nyckelordsbaserad textkategorisering

Vid nyckelordsbaserade textkategorisering utgår man typiskt sätt enbart från namnet på den ämneskategori av texter man vill detektera (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009). Istället för att basera kategoriseringen på manuellt etiketterade texter byggs en representation av kategorin upp med hjälp av ordstatistik i ej etiketterad text och/eller andra lexikala resurser. I avsnitt 2.2.1 nedan kommer de övergripande gemensamma dragen för alla nyckelordsbaserade textkategoriseringsalgoritmer beskrivas och i avsnitt 2.2.2 beskrivs de algoritmer som kommer granskas inom ramen för detta examensarbete.

2.2.1 Övergripande algoritmbeskrivning

Övergripande kan alla nyckelordsbaserade algoritmer beskrivas med följande delsteg:

VAL AV KATEGORINAMN: Ta fram ett ämnesnamn för den kategori av texter man vill detektera (manuellt)

GENERERING AV NYCKELORD: Ta fram semantiskt närliggande ord som tillsammans med ämnes-namnet representerar kategorin.

BERÄKNING AV DOKUMENTS NÄRHET TILL NYCKELORD: Närheten mellan framställda nyckelord och texter mäts genom överlapp mellan dokumentens ord och orden i nyckelordsvektorerna. Ett vanligt använt och enkelt mått för att mäta närhet är cosinus av vinklarna mellan nyckelorsdsvektorn och den vektor som representerar orden i dokumentet (Barak m. fl., 2009; Liebeskind m. fl., 2015).

TILLDELNING TILL ÄMNESKATEGORI: Genom närhetsmåttet fås en rankning av hur utmärkande genererade nyckelord är för de dokument som ska analyseras. Utifrån ranknningen behöver ett beslut tas om vilka närhetsmått som ska krävas för kategorisering. Liebeskind m. fl. (2015) kunde i sin studie visa att manuellt och individuellt valda poäng-trösklar för olika ämneskategorier gav en markant förbättring i F1-poäng (ca 0.1-0.2 F1-poäng), jämfört med att kategorisera baserat på en standard kvot av dokumenten med positiva värden på närhetsmåttet. Hur valet av lämplig poäng-tröskel på bästa sätt kan automatiseras återstår för framtida studier att kartlägga.

Många nyckelordsbaserade algoritmer tillämpar också ett femte steg, så kallad bootstrap. Vid bootstrap används samma kategoriseringsalgoritmer som vid

övervakad inlärning men med den avgörande skillnaden att träningsdatan som används är de dokument/textstycken som kategoriserats med hjälp av deras närhet till den givna kategorins nyckelordslista (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009).

2.2.2 Nyckelordsbaserade textkategoriseringsalgoritmer

Här nedan följer en kort presentation av de fem studier som föreslagit nyckelordsbaserad textkategorisering och som kommer granskas inom ramen för det här examensarbetet (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009).

2.2.2.1 Nyckelordsbaserade textkategoriseringsalgoritmer som använder manuellt framställda lexikala resurser

Barak m. fl. (2009), Gliozzo m. fl. (2009), Liebeskind m. fl. (2015) och Qiu m. fl. (2009) är fyra studier som undersökt hur manuellt framställda lexikala resurser som WordNet och Wikipedia kan användas för att framställa nyckelord relevanta för textkategorisering.

WordNet är en stor manuellt framställd lexikal databas över termer som grupperats i set av synonymer där varje grupp representerar ett distinkt koncept. Orden inom en grupp är märkta med vilken relation de har till varandra så som hyponym-relationer och meronym-relationer. Där hyponym-relationen beskriver att ett begrepp är ett underbegrepp, t.ex. är “hund” en hyponym till det överordnade begreppet “ djur”, och där meronym-relationen är det samma som att vara en del av, t.ex. är “finger” en meronym till begreppet “hand”. Specifikt meronym- och hyponym-relationer har använts för nyckelordsframställning.

Wikipedia har använts för att framställa nyckelord genom analyser av vilka ord och länkar som förekommer i wikipedia-artiklar direkt relaterade till kategorinamnet.

I alla tre studiernas algoritmer tillämpas ett bootstrap-steg för att utföra kategoriseringen.

Alla tre artiklarna beskriver resultaten som lovande. Barak m. fl. (2009) och Qiu m. fl. (2009) rapporterar båda att algoritmerna givit goda resultat på de testdataset som prövats. Alla tre studierna rapporterat att för vissa klasser och specifika dataset är resultaten likvärdiga med de som uppnås med övervakad inlärning.

2.2.2.2 Nyckelordsbaserade textkategorisering algoritmer som använder statistiska mått på ords samförekomstfrekvens för nyckelordsframställning

Barak m. fl. (2009); Gliozzo m. fl. (2009); Liebeskind m. fl. (2015) och Ko och Seo (2009) föreslår alla olika metoder för att ta fram nyckelord baserat på hur ofta de förekommer i texter tillsammans med de givna kategorinamnen. I de två förstnämna studierna ses nyckelorden som baseras på samförekomst enbart som ett komplement till nyckelord från lexikala resurser medan Ko och Seo (2009) och Liebeskind m. fl. (2015) även föreslår algoritmer som enbart utnyttjar ords samförekomst i text.

2.2.2.2.1 Latent semantic analysis Barak m. fl. (2009) och Gliozzo m. fl. (2009) föreslår Latent Semantic Analysis (LSA) för att detektera relevanta nyckelord.

Vid LSA utgår man från att ord med liknande mening kommer förekomma i samma texter. Därför skapar man en matris där varje rad representerar en unik term från det dataset som analyseras och varje kolumn representerar en text/stycke i datasetet. Värderna i matrisen definierar hur ofta radens ord förekommer i den text/stycke som kolumnen representerar. Därefter används en oövervakad metod för reducera antalet dimensioner i matrisen. Olika termers semantiska närhet kan sedan beräknas genom att man jämför cosinus av vinkel mellan de två reducerade rad vektorerna (Dumais, 2004).

2.2.2.2.2 Dice-metoden Med anledning av bland annat den mycket beräkningstunga process som krävs för att framställa nyckelord med LSA-metoden, valde Liebeskind m. fl. (2015) att göra en uppföljande artikel som bygger vidare på och modifierar metoderna presenterade av Barak m. fl. (2009) och Gliozzo m. fl. (2009). Liebeskind m. fl. (2015) fann att LSA kunde ersättas med det mycket enklare samförekomstmåttet Dice-koefficienten utan någon försämring i resultat.

Dice-koefficienten är ett enkelt mått på samförekomst som bygger på att man enbart beräknar hur ofta ord samförekommer med kategorinamnet på dokument nivå (se avsnitt 2.5 nedan för en mer detaljerad beskrivning av dice-metoden).

2.2.2.2.3 Nyckelord- och textklustring Ko och Seo (2009) förslår en tredje alternativ algoritm som skapar ett kluster av nyckelord och textstycken för varje ämneskategori.

I algoritmens första steg väljs ett antal nyckelord för varje kategori ut baserat

på hur ofta de förekommer tillsammans med ett givet kategorinamn på dokumentnivå och hur sällan de förekommer i dokument med andra kategorinamn. Därefter styckas texten upp i textstycken som tilldelas ett närhetsmått till olika kategorier baserat på om de framställda nyckelorden förekommer i textstycket samt hur likt textstycket är de stycken som innehåller nyckelorden. Textstycken tilldelas en specifik kategori baserat på vilken kategori de har starkast närhetsmått till. De framställda klustrerna av textstycken används slutligen för att träna en algoritm som använder övervakad inlärning för att känna igen text för en given kategori.

Ko och Seo (2009) rapporterar att algoritmens resultat var nära likvärdiga med de som uppnås på samma dataset med manuellt etiketterad träningsdata och algoritmen Naive Baise.

2.3 Befintliga utvärderingsmetoder för nyckelordsbaserade textkategoriseringsalgoritmer

Utvärderingen av textkategoriseringsalgoritmer har ofta begränsats till att mäta algoritmernas kategoriseringsförmåga på några få kända testdataset. I detta avsnitt kommer vi först beskriva de metoder och dataset som använts för att utvärdera nyckelordsbaserade textkategoriseringsalgoritmer, för att därefter diskutera vilka styrkor och begränsningar de metodvalen har om man önskar utvärdera den praktiska tillämpbarheten hos algoritmerna.

I avsnitt 2.3.1 nedan diskuteras metoder för att mäta kategoriseringsförmåga, i avsnitt 2.3.2 diskuteras de olika öppet tillgängliga testdataseten för textkategorisering och i avsnitt 2.3.3 beskrivs hur felanalys kan genomföras och användas som ett verktyg för vidareutveckling av algoritmer.

2.3.1 Befintliga mått på kategoriseringsförmåga och deras begränsningar

Det mest populära måttet för att mäta kategoriseringsförmågan hos textkategoriseringsalgoritmer är F1-poäng. F1-poäng är definierat som det harmoniska medelvärdet av algoritmens precision och täckning. Typiskt sätt väljs den precisionsnivå och täckningsnivå som man uppnått samtidigt och som maximerar F1-poängen (se ekvation 1).

$$F1 = 2 \cdot \frac{precision \cdot täckning}{precision + täckning} \tag{2.1}$$

Precision och täckning kan beräknas antingen på en mikro-nivå eller en makro-nivå. Mikro-värdet (här betecknat med μ) är den genomsnittliga precision/täckningen för alla individuella kategoriseringsbeslut (se ekvation 2) medan makro-värdet (här betecknat med M) är den genomsnittliga precisionen/täckningen då den först beräknats lokalt för varje ämneskategori (se ekvation 3).

Tabell 2.1: Möjliga utfall av kategorisering för kategori k_i

Kategori k_i		Korrekt klacifiering	
		JA	NEJ
Klassificerare	JA	SP_i	FP_i
	NEJ	FN_i	SN_i

Tabell 2.2: Möjliga utfall för dataset av kategorier

Dataset, $C = \{c_1, \dots c_{\lvert c \rvert}\}$		Korrekt klacifiering	
		JA	NEJ
Klassificerare	JA	$SP = \sum_{i=1}^{\lvert C \rvert} SP_i$	$FP = \sum_{i=1}^{\lvert C \rvert} FP_i$
	NEJ	$FN = \sum_{i=1}^{\lvert C \rvert} FN_i$	$SN = \sum_{i=1}^{\lvert C \rvert} SN_i$

$$precision_\mu = \frac{SP}{SP + FP} \qquad täckning_\mu = \frac{SP}{SP + FN} \tag{2.2}$$

$$precision_M = \frac{\sum_{i=1}^{|C|} p_i}{|C|} \qquad täckning_M = \frac{\sum_{i=1}^{|C|} r_i}{|C|} \tag{2.3}$$

där

$$p_i = \frac{SP_i}{SP_i + FP_i} \qquad r_i = \frac{SP_i}{SP_i + FN_i}$$

F1-måttet kan sägas mäta algoritmens maximala kategoriseringsförmåga eftersom utmaningen inom textkategorisering ligger i att uppnå både godtagbar täckning och godtagbar precision samtidigt.

Som följer av ekvation 2 och ekvation 3 ovan tar F1-poäng beräknade med makro-värden mer hänsyn till resultatet för varje enskild ämnedkategori (oberoende av dess storlek), medan F1-poäng på mikro-nivå enbart ser till hur många dokument algoritmen lyckats kategorisera korrekt oberoende av vilken kategori den tillhör. Men i båda fallen är F1-poängen ett genomsnitt av precisionen och täckningen för hela testdatasetet.

Att F1-poäng enbart beskriver genomsnittlig kategoriseringsförmåga för hela datasetet medför att vi inte vet hur täckningen och precisionen varierar mellan de olika testade kategorierna. Detta är problematiskt om målet är att värdera algoritmens praktiska tillämpbarhet. Om fokus enbart ligger på att maximera F1 poäng missas faktumet att en algoritm som har sämre F1-poäng mycket väl kan vara att föredra framför en med högre F1-poäng om vi vet att den förstnämnda ger likvärdiga resultat på de flesta ämneskategorier medan de sistnämnda ger mycket höga resultat på ett antal kategorier och oacceptabelt låga resultat för andra kategorier.

Om inte täckning och precision redovisas separat tillsammans med F1-poängen är det även svårt att avgöra om algoritmen passar för en given tillämpning eftersom man inte kan säga om det är en hög precision eller en hög täckning som givit upphov till F1-poängen. I de flesta fall är det visserligen möjligt att modifiera algoritmen så att den ger en högre precision eller täckning men då till priset att man minskar den andra. Enbart F1-poäng ger dock ingen information om hur mycket resultatet i täckning/precision försämras av att vi höjer kraven på den ena.

Liebeskind m. fl. (2015) argumenterar för att önskad minsta precisionsnivå kan variera mellan olika tillämpningar de väljer därför att redovisa sina kategoriseringsresultat som en "P&R-graf" som speglar uppmätta täckningsnivåer vid specifika valda precisionsnivåer. Men även denna metod innebär att resultaten presenteras som ett genomsnitt för alla kategorier och ger ingen information om variansen i resultat mellan olika kategorier.

2.3.2 Hur valet av testdata påverkar resultaten

Inom textkategorisering forskningen existerar ett fåtal dataset som använts frekvent för att testa föreslagna algoritmer (20-newgroup, Reuters, WebKB). Dessa är framtagna för att möjliggöra jämförelser av olika kategoriseringsalgoritmers kategoriseringsförmåga. Bland annat (Sebastiani, 2002) förespråkar användandet av de här nämnda kända dataseten för att göra resultaten i olika studier jämförbara med varandra. För textkategoriseringsalgoritmer som använder manuellt kategoriserad träningsdata krävs det också typiskt sätt för mycket kategoriserad data för att det ska vara praktiskt genomförbart att framställa ett eget nytt dataset (se avsnitt 2.1.1).

Att samtliga studier använder samma dataset är dock inte oproblematiskt. Lewis (1995) tar i sin artikel upp två problem med att samma dataset används frekvent i många studier:

Ej representativa dataset: De traditionella dataseten har gemensamma egen-

skaper som ofta skiljer sig markant från hur data set ut i praktiska tillämpningar.

För hård anpassning till specifik testdataset: Precis som att att en maskininlärning algoritm kan övertränas av att parametrar anpassas för hårt till slumpartade egenskaper i ett träningsdataset kan hela forskningsfält anpassas för hårt till de kända testdataseten om man enbart vidareutvecklar de algoritmer som råkat prestera väl på existerande testdataset. Detta är ett problem även om vi skulle använda dataset som bra speglar våra tillämpningar. Enbart genom att ibland genomföra tester på nya dataset kan vi verifiera att algoritmerna verkligen har förbättrats.

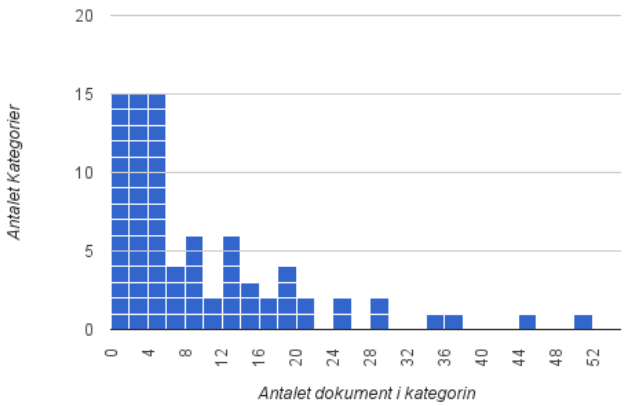
Tre egenskaper som utmärker de ovan listade kända för allmänheten öppna dataseten är:

1. Alla texter ingår i någon av flera namngivna disjunkta kategorier,
2. Varje kategori innehåller ungefär samma antal texter.
3. Antalet kategorier är ofta relativt få i relation till antalet texter

För Reuters-10 som är ett av de frekvent använt dataseten gäller även att texterna är homogena i stil ur bemärkelsen att samtliga texter är välskrivna nyhetstexter.

Att klasserna inte är balanserade och uttömmande i det dataset man vill kategorisera är giltigt för många av de tänkbara tillämpningarna av automatisk textkategorisering (Liebeskind m. fl., 2015).

På grund av ovan nämnda brister i befintliga testdataset valde Liebeskind m. fl. (2015) att framställa ett nytt dataset för sin studie. Målsättningen med det nya datasetet var att detta bättre skulle spegla många praktiska tillämpningsfall. Som grund för datasetet valdes de ca 400000 användargenererade filmbeskrivningar som finns tillgängliga via IMDB:s databas. Med utgångspunkt i filmbeskrivningarna skapades en taxonomi med ca 80 ämneskategorier. Därefter gjordes ett urval av ca 2000 filmbeskrivningar med en minimilängd på 150 tecken, som kategoriserade manuellt in i den komplexa, hierarkiska och ej uttömande taxomin man skapat. Resultatet blev ett dataset där merparten av dokumenten inte tilldelas någon kategori och antalet dokument i de olika kategorierna varierade stort (se figur 1)



Figur 2.1: Fördelning av antal dokument per ämneskategori i test-dataset framställt av Liebeskind m.fl. (2015)

Liebeskind m. fl. (2015) utvärderade tidigare föreslagna nyckelordsbaserade textkategoriserings algoritmer på det nya mer komplexa datasetet. Resultatet visade att algoritmer som presterat väl på klassiska dataseten fick oacceptabelt låga resultat på det nya mer komplexa datasetet. Detta belyser vikten av att utvärdera föreslagna nyckelordsbaserade metoder på dataset som speglar den faktiska tillämpning innan system sätts i produktion.

2.3.3 Felanalys - ett verktyg för algoritmutveckling

Nyckelordsvektorerna utgör transparenta representationer av ämneskategorierna som gör det möjligt att avgöra exakta orsaken till att en text kategoriserats fel. Liebeskind m. fl. (2015) förslår i sin artikel ett antal feltyper som kan användas för att kategorisera de fel som uppstår vid nyckelordsbaserad textkategorisering. Feltyperna utgår från att den algoritm som används delar upp nyckelorden i undergrupperna kontextord och referensord (för förklaring av de begreppen, se avsnitt 2.5.1.2 nedan). Med denna utgångspunkt förslås 4 feltyper:

1. **Passing reference:** En text kategoriseras fel eftersom ett ord som utmärker en annan kategori finns med i texten.
   Exempel:
   Text: *A medical student... moves to Miskatonic University to continue his research*
   Passing reference: University

Korrekt kategori: Medicine
Felaktigt vald kategori: University/Collage

2. **Ambiguity:** En text kategoriseras fel till följd av att ett tvetydigt ord finns med i referensordslistan för en kategori.
   Exempel:
   Text: *A dashing officer of the guard and romantic poet ... Christian, who is also in love...*
   Ambiguity: Christian
   Korrekt kategori: Literature
   Felaktigt vald kategori: Christianity

3. **Noisy Expansion:** En text kategoriseras fel till följd av att referensordslistan innehåller ord som ej refererar till den givna kategorin.
   Exempel:
   Korrekt kategori: Pop/Rock
   Felaktiga referenstermer: machine, mix

4. **Lack of expansion:** En text kategoriseras inte i den sökta kategorin för att att modellen saknar ett relevant referensord.

Analyser av vilka feltyper som dominerar ger viktig information om vilken del av algoritmen som behöver förbättras.

De tre förstnämnda feltyperna kan ses som en brist i den framställda kontextordsvektorn. Om kontextordsvektorn vägt tyngre hade texterna ej räknats till kategorin trots förekomst av ett referensord (Liebeskind m. fl., 2015).

Feltyp tre kan enkelt avhjälpas genom en manuell inspektion av de genererade referensorden för varje kategori. Vilket kan antas vara ett minimalt extra arbete då man vid de flesta praktiska tillämpningar kan tänkas manuellt vilja inspektera kvalitén i de genererade nyckelorden.

Fjärde feltypen påverkar algoritmens täckning och kräver att fler metoder alternativ mer manuellt arbete läggs på att framställa relevanta referensord.

2.4 Algoritmutvärdering med fokus på praktisk tillämpbarhet

Syftet med detta examensarbete är att hjälpa till att besvara frågan när nyckelordsbaserad textkategorisering är praktiskt tillämpbart.

För att definiera praktisk tillämpbarhet och göra det mätbart har ett industriellt motiverat användningsfall tagits fram. Användningsfallet är framtaget i

samarbete med textanalys företaget Gavagai och definierar sex stycken krav för nyckelordsbaserade textkategoriseringsalgoritmer.

I avsnitt 2.4.1 nedan presenteras det framtagna användningsfallet. I avsnitt 2.4.2 utvärderas hur väl dagens befintliga nyckelordsbaserade textkategoriseringsalgoritmer möter kraven i användningsfallet. I avsnitt 2.5 beskrivs i detalj den algoritm som utvärderingen visat har störst potential att möta kraven i användningsfallet.

2.4.1 Användningsfall

Företaget Gavagai önskar ett system för att ämneskategorisera text utan tillgång till manuellt etiketterad träningsdata. I utformandet av systemet finns det flera viktiga krav att förhålla sig till. Här nedan listas sex stycken framtagna minimala krav som Gavagai menar att systemet behöver uppfylla. Till varje krav ges en motivering till varför det specifika kravet kan anses vara generellt giltiga för ett stort antal praktiska tillämpningsfall av nyckelordsbaserad textkategorisering:

1. *Krav:* Manuellt arbete ska minimeras och i de fall det förekommer ska det vara motiverat av en förbättrad kategorisering. Specifikt ska systemet inte förutsätta att det finns tillgång till manuellt etiketterad träningsdata. Det är önskvärt att systemet som föreslås enbart förutsätter ett fåtal manuellt valda ord som representerar det ämnesområde av texter som man önskar att kategorisera och därefter kräver minimal manuell övervakning och feedback.
   Motivering: Detta krav kan antas vara generellt för varje användningsfall av nyckelordsbaserad textkategorisering. Befintliga textkategoriseringsalgoritmer som bygger på övervakad inlärning finns tillgängliga i open source för de flesta programmeringsspråk och ger state-of-the-art resultat för ämneskategorisering av text. Om de alternativa algoritmerna förutsätter lika mycket eller mer manuellt arbete, går vinsten med att inte vara beroende av övervakad inlärning förlorad.

2. *Krav:*Det ska vara möjligt att avgöra ungefärlig täckning och precision för en given ämneskategori utan att det medför mer manuellt arbete än vid algoritmer som använderöverväkad inlärning.
   Motivering: Minsta godtagbara precisions- och täckningsnivå varierar mellan tillämpningar men en minsta precisionsnivå på ca 0.6 brukar ses som ett minimum för många praktiska tillämpningar (Liebeskind m. fl., 2015). Även om kraven alltså varierar är det vid de flesta tillämpningar avgörande att kunna förutsäga hur hög täckning och precision som systemet levererar för en viss ämneskategori. Samma argument som anges krav 1 motiverar att arbetet för att uppnå detta inte överskrida det arbete som krävs för att genomföra övervakad inlärning.

3. *Krav:* Den föreslagna metoden ska ge en mer korrekt kategorisering än den kategorisering man får om man enbart kategoriserar texterna baserat på förekomst av kategorinamnet.
   Motivering: Skälet till att inte slå fast en fix minsta godtagbara nivå för korrekthet i kategoriseringen är att detta kan variera stort mellan olika tänkbara tillämpningsfall. Det varierar också om täckning eller precision är avgörande mellan olika tillämpningsfall. Men om systemet inte ger en bättre kategorisering än den som ges av att kategorisera enbart på förekomst av kategorinamnet är det svårt att motivera implementation av det givna systemet.

4. *Krav:* Det föreslagna systemet ska gå att implementera för godtyckligt språk.
   Motivering: Att algoritmen inte förutsätter ett specifikt språk gör att antalet användningsfall breddas betydligt.

5. *Krav:* Det föreslagna systemet ska gå att implementera för godtycklig ämneskategori givet att det finns texter att tillgå på det aktuella språket som innehåller de manuellt framställda nyckelorden för ämneskategorin.
   Motivering: Ej etiketterade texter är det idag lätt att få tillgång till på de flesta språk. Det är vidare relativt enkelt att säkerställa att ett urval av de texter som algoritmerna får arbeta med innehåller de givna manuellt framtagna nyckelorden. Detta är också strikt nödvändigt eftersom nyckelordsbaserade algoritmer bygger på att de kan använda de manuellt framtagna nyckelorden för att skapa en rikare representation av ämneskategorin.

6. *Krav:* Systemet ska inte förutsätta någon särskild ämnesstruktur i det dataset som analyseras.
   Motivering: Anledningen till att man använder automatisk textkategorisering är i de flesta fall att man inte har resurser att analysera de stora dataseten manuellt och att datasetet förändras över tid. Det är därför sällan möjligt att anta någon specifik ämnesstruktur i träningsdata eller den data man önskar kategorisera.

Slutligen ligger det även i Gavagais intresse att utreda om Gavagais Living Lexicon kan användas för att komplettera och förbättra de nyckelordsbaserade algoritmer som förslås i litteraturen.

2.4.2 Utvärdering av praktiska tillämpbarhet hos befintliga nyckelordsbaserade textkategoriseringsalgoritmer

Inom ramen för detta examensarbete har ett antal nyckelordsbaserade kategoriseringsalgoritmer granskats (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo,

2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009). Gemensamt för de för de algoritmer som ingick i granskningen är att de haft målsättningen att göra automatisk textkategorisering mer praktiskt tillämpbar genom att göra manuellt etiketterad träningsdata överflödig. Paradoxalt visar granskningen nedan att undvikandet av manuellt framställt träningsdata ofta kommit till priset av andra eftergifter, både i form av minskad tillämpbarhet och införande av nya manuella steg som inte krävs vid klassisk övervakad inlärning.

Granskningen visade vidare att de hinder för praktisk tillämpbarhet som introducerades i merparten av de nyckelordsbaserade textkatetgoriseringsalgoritmerna var så allvarliga att enbart den algoritm som Liebeskind m. fl. (2015) presenterar som "dice-metoden" har potential att möta kraven i det uppsatta användningsfallet.

De två aspekterna av nyckelordsbaserade textkategorisering som vår granskning visat tydligast påverkar den praktiska tillämpbarheten är vilka resurser som algoritmerna använder för att framställa nyckelord samt om de väljer att tillämpa ett bootstrap-steg eller ej. I avsnitt 2.4.2 diskuterar vi hur dessa aspekter har medfört att algoritmerna föreslagna av Ko och Seo (2009), Barak m. fl. (2009), Gliozzo m. fl. (2009) och Qiu m. fl. (2009) inte har potential att möta kraven i det uppsatta användningsfallet.

I avsnitt 2.5 presenteras och diskuteras dice-metoden.

2.4.2.1 Algoritmer som använder manuellt framställda lexikala resurser

Barak m. fl. (2009), Gliozzo m. fl. (2009) och Qiu m. fl. (2009) föreslår nyckelordsbaserade metoder där nyckelord från manuellt skapade lexikala resurser har en central roll.

Att använda manuellt framställda resurser för nyckelordsframställning utgör klara begränsningar för praktiska tillämpbarheten. Islam, Milios och Kešelj (2012) sammanfattar tre huvudsakliga nackdelar med manuellt framställda lexikala resurser:

1. Att skapa och underhålla lexikala resurser kräver både kunskap och stort engagemang, detta gäller särskilt för en resurs som WordNet.
2. Täckningen för olika begrepp varierar mycket vilket ger upphov till mycket varierande resultat för olika ämneskategorier, detta är giltigt för både WordNet och Wikipedia.
3. Manuellt framställda resurser begränsar vilka språk som algoritmen kan tillämpas på. WordNet är helt begränsat till engelska men även Wikipedia

har klart starkast täckning på engelska vilket gör det omöjligt eller svårt att tillämpa den framtagna metoden för andra språk.

I enlighet med detta kunde Barak m. fl. (2009) och Qiu m. fl. (2009) rapportera mycket varierande resultat för olika kategorier och dataset. Barak m. fl. (2009) såg att även då de uteslut Bootstrap-steget (som kan öka variansen i resultat mellan olika dataset ytterligare, (se avsnitt 2.4.2.4), varierade den genomsnittliga täckning och precision mätt i F1-poäng mellan ca 0.4 och 0.75 för olika dataset. Men även studier som tillämpar övervakad inlärning har kunnat se att olika ämneskategorier får olika kategoriseringsresultat med samma mängd träningsdata och samma algoritm (Schohn & Cohn, 2000). Det är därför svårt att uttala sig med säkerhet om vad som orsakat variansen.

Slutsatsen blir att användande av manuellt framtagna lexikala resurser är problematiskt utifrån det uppsatta användningsfallet. Det bryter mot krav 4 som slår fast att metodens tillämpbarhet inte ska vara begränsad till särskilda språk. Det bryter även mot krav 5 eftersom det är inte godtagbart att implementationen förutsätter uppdatering av Wikipedia eller WordNet för att ha potential att ge godtagbara resultat för många ämneskategorier.

2.4.2.2 Algoritmer med textstycke- och nyckelordsklustring

Ko och Seo (2009) presenterar en nyckelordsbaserad textkategoriseringsalgoritm som skapar kluster av nyckelord och textstycken för varje kategori.

Ko och Seo (2009) hävdar att deras algoritm ger nära likvärdiga resultat på textkategorisering som traditionella metoder baserade på övervakad inlärning. Men i motsats till algoritmer som bygger på övervakad inlärning utnyttjar Ko och Seo (2009) i ett flertal delsteg i sin algoritm att den korpus de kategoriserar (och använder för nyckelord och textklusterframställning) är uppdelad i ett antal disjunkta namngivna kategorier.

Nyckelorden och de textstycken som används för att göra initiala kategoriseringen väljs inte enbart baserat på hur ofta de förekommer tillsammans med kategorinamnet utan även baserat på hur sällan de förekommer i någon av de andra kategorinamnen/ kategoriernas nyckelord. Vetskapen om vilka ämneskategorier som förekommer i korpusen och att ämnena är disjunkta gör att resultaten för en given kategori förbättras. Detta eftersom nyckelord som stämmer in på flera kategori kan väljas bort. Genom denna sållning minskar förväxlingen mellan kategorierna.

Antagandet att data går att dela in i ett diskret antal på förhand kända namngivna disjunkta och uttömande kategorier är sant för klassiska test- och träningsdataset, men det är svårt att finna de praktiska tillämpningar där de hårda

antagandena är uppfyllda. Specifikt är det inte förenligt med krav 6 i det uppsatta användningsfallet som slår fast att vi inte kan förutsätta någon specifik känd ämnesindelning i den korpus som ska kategoriseras eller den korpus som används för nyckelordsframställning.

2.4.2.3 Algoritmer med nyckelord baserade på Latent Semantic Analysis

Barak m. fl. (2009) och Gliozzo m. fl. (2009) kompletterar sina algoritmer med nyckelord framställda med metoden Latent Semantic Analysis (LSA).

Ett inneboende problem med att använda LSA är att det krävs en mycket beräkningstung process för att ta fram de semantiskt närliggande nyckelorden. Beräkningen av nyckelorden måste också göras om i sin helhet om den underliggande korpusen som används för att framställa nyckelorden förändras. ((Liebeskind m. fl., 2015; Sahlgren, 2005)

Även om det givna användningfallet tillåter beräkningstunga algoritmer, kan en algoritm som ger likvärdiga resultat med en mindre beräkningstung process ses som mer praktiskt tillämpbar. Detta gör att dice-metoden som Liebeskind m. fl. (2015) fann gav lika relevanta nyckelord bedöms vara en mer praktiskt tillämpbar algoritm.

2.4.2.4 Algoritmer med bootstrap

Merparten av de studier som studerat nyckelordsbaserade metoder för automatisk text kategorisering har föreslagit algoritmer som använder ett så kallat Bootstrap-steg (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Qiu m. fl., 2009).

Vid bootstrap används samma kategoriseringsalgoritmer som vid textkategorisering med övervakad inlärning men träningsdatan är inte manuellt kategoriserade texter utan dokument som kategoriserats med hjälp av närhetsmåttet mellan de framställda nyckelorden och texten/textstycken (se steg 4 i övergripande algoritmbeskrivningen i avsnitt 2.2.1 ovan)

Ingen av de ovan listade studierna som förespråkar användandet av ett bootstrapsteg ger en klar motivering till detta val. Men antagligen har de mycket goda resultaten från studier som använt övervakad inlärning, varit en motiverande faktor.

Utifrån ett praktiskt tillämpbarhets perspektiv är det dock relevant att reflek-

tera kring vilka antagande om träningsdata som måste gälla för att bootstrap ska ha en chans att ge godtagbara resultat.

Tidigare studier av automatisk textkategorisering har visat att SVM är den kategoriseringsalgoritm som med minst träningsdata uppnår bäst kategoriseringsresultat (Yang & Liu, 1999).

Yang och Liu (1999) och Schohn och Cohn (2000) redovisar i sina studier hur kategoriseringsresultaten för olika ämneskategorier beror av mängden träningsdata. Resultatgraferna för de olika ämneskategorierna kan beskrivas som att det finns ett tröskelvärde där mer träningsdata inte nämnvärt förbättrar resultaten men under den tröskeln sjunker korrektheten i kategoriseringen relativt snabbt. Variansen i hur mycket träningsdata som krävs är dock stor. För vissa ämneskategorier krävs ca 100 positiva exempel medan andra kategorier kräver över 1000 positiva exempel för att uppnå likvärdig korrekthet (Schohn & Cohn, 2000). I fallet med bootstrap krävs därtill sannolikt fler korrekt etiketterade positiva exempel eftersom den initiala kategoriseringen kommer innehålla både falskt positiva och falskt negativa exempeltexter.Att den intiala kategoriseringen innehåller fel kan vi utgå från eftersom den annars redan vore optimal och bootstrap skulle vara överflödigt.

Om bootstrap ska ha en chans tillämpas med goda resultat måste vi alltså kunna garantera att den träningsdata vi kategoriserar med den initiala kategoriseringsmetoden både innehåller en tillräcklig mängd positiva exempel och att vi i den initiala kategoriersingen detekterar tillräckligt många relevanta och korrekta positiva respektive negativa exempel.

Liebeskind m. fl. (2015) tar i sin studie fram ett nytt testdataset för nyckelordsbaserad ämneskategorisering som bättre speglar många praktiska tillämpningar. Detta nya dataset belyser på ett tydligt sätt den problematik som beskrivits ovan. Det nya dataset baserades på ca 2000 filmbeskrivningar från IMDB som kategoriserades in i en komplex hierarki med ca 80 ämneskategorier. Resultatet av kategoriseringen visade att många relevanta ämneskategorier innehöll enbart 1-10 dokument. Den ämneskategori som förekom mest frekvent utgjorde enbart ca 50 av de tvåtusen etiketterade dokumenten.

Liebeskind m. fl. (2015) lät sedan den bootstrap-algoritm som Barak m. fl. (2009) föreslagit kategorisera 120000 ej etiketterade filmbeskrivningar från IMDB. Om vi antar att ämnesfördelningen i det etiketterade testdatasetet är representativt för ämnesfördelningen i det större träningsdatasetet ser vi att även om den initiala kategoriseringen skulle detektera alla positiva exempel (och därmed göra bootstrap egentligen överflödigt) utgör dessa enbart några hundra positiva exempel för de mer ovanliga kategorierna. Detta har Schohn och Cohn (2000) visat inte är ett tillräckligt antal positiva exempel för att uppnå goda kategoriseringsresultat på många ämneskategorier, även om data är manuellt kategoriserad och inte innehåller några felaktigt kategoriserade texter.

I enlighet med resonemanget ovan kunde Liebeskind m. fl. (2015) rapportera att den bootstrap-algoritm som Barak m. fl. (2009) rapporterat gav genomsnittlig F1-poäng på ca 0.5 respektive ca 0.8 på de kända dataseten 20-newsgroup och Reuters-10, fick F1-poäng på ca 0.03 på det nya mer komplexa datasetet. Nyckelordsbaserad kategorisering utan bootstrap med samma mängd träningdata visade sig dock kunna vara en en framkomlig väg även för det nya mer komplexa datasetet.

Av detta kan vi dra slutsatsen att bootstrap-steg kan vara förödande för resultatet om kategorierna inte utgör en stor del av de texter vi vill kategoriersera. Ett bootstrap-steg är därmed inte är förenligt med krav 6 i det framtagna användningsfallet, eftersom det förutsätter att man känner till ämnesfördelningen i det dataset man önskar använda som träningsdata.

Ytterligare ett argument mot att tillämpa bootstrap är att den direkta möjligheterna att analysera varför en text kategoriserats fel då går förlorad (se avsnitt 2.3.3 ovan) (Liebeskind m. fl., 2015).

2.5 Dice-metoden för nyckelordsbaserad textkategorisering

Vår granskning av befintliga nyckelordsbaserade textkategoriseringsalgoritmer visar att Dice-metoden framtagen av Liebeskind m. fl. (2015) är den metod som har störst potential att möta kraven i det framtagna användningsfallet. I detta avsnitt kommer därför delstegen i dice-metoden presenteras i detalj och granskas. Därefter ges en sammanfattande slutsats om vilka krav i användningsfallet som, är uppfyllda, kräver algoritmförändringar för att uppfyllas, respektive kräver ytterligare testning för att det ska vara möjligt att uttala sig om huruvida de är uppfyllda.

2.5.1 Dice-metodens delsteg

De övergripande delstegen i varje nyckelordsbaserad algoritm framgår i avsnitt 2.2.1 ovan. Detta avsnitt kommer därför fokusera på att beskriva vad som utmärker de olika delstegen i dice-metoden.

2.5.1.1 Val av kategorinamen

I den algoritm Liebeskind m. fl. (2015) föreslår används enbart ett ord som "frö" till varje ämneskategori. Detta enda nyckelord har dock valts med omsorg för att inte vara tvetydigt och väl representera den önskade kategorin.

2.5.1.2 Generering av nyckelord

I dice-metoden baseras nyckelorden på Dice-koefficienten, ett generellt mått för att på ett enkelt sätt mäta samförekomst av fenomen i ett dataset. Liebeskind m. fl. (2015) defintion av Dice-koefficienten för ord i texter ges av ekvation 4 nedan.

$$Dice(w_a, w_b) = \frac{D(w_a, w_b)}{D(w_a) + D(w_b)} \tag{2.4}$$

där $D(w_i, w_j)$ är antalet dokument som innehåller både termen w_i och w_j och där $D(w_i)$ är antalet dokument som innehåller termen w_i

Liebeskind m. fl. (2015) föreslår att en korpus med ca 100000 dokument används för att generera k (i deras fall k=100) nyckelord för varje ämneskategori. De k nyckelorden är de k termer (det vill säga ord, bi-gram och tri-gram) som har högst Dice-koefficient för det givna kategorinamnet och som ej är så kallade funktionsord, där funktionsord avser frekvent förekommande ord som främst har en grammatisk funktion. I svenska är *och*, *en*, *till*, och *har* exempel på funktionsord.

För att få ut största möjliga nytta av de genererade nyckelorden föreslår Liebeskind m. fl. (2015) att de genererade nyckelorden delas upp i två grupper referensord och kontextord.

Det var Barak m. fl. (2009) som introducerade idén att genererade nyckelord bör delas upp i grupperna referensord och kontextord. Där referensord avser de ord som direkt refererar till kategorinamnet medans kontextord är ord som ofta förekommer tillsammans med kategorinamnet men inte direkt refererar till kategorin. Den bakomliggande teorin är att en text måste innehålla både referensord och kontextord för det givna ämnesområdet för att texten ska anses tillhöra kategorin. Se tabell 3 nedan för exempel på referensord och kontextord för ämneskategorin "hundar".

Tabell 2.3: Exempel på referensord och kontextord för ämneskategorin hundar

Kategori	Referensord	Kontextord
Hundar	hund, hundar, hundens valp, tax, tik ...	koppel, promenad, jakt, bästa vän

Liebeskind m. fl. (2015) delar upp de k genererade nyckelorden i referensord och kontextord genom att definiera referensord som de termer som klarar de fyra nedan listade kriterierna och övriga genererade nyckelord som kontextord. Kriterierna för att att definieras som referensord var följande:

1. **Hög dice-koefficient** Referensord tenderar att vara starkt associerade med kategorinamnet vilket ger de generellt höga dice-koefficient värden. Liebeskind m. fl. (2015) valde att betrakta gränsen för referensord som en parameter och fann att en 0.05 var ett optimalt gränsvärde för att skilja ut referensord och maximera den genomsnittliga kategoriseringsförmågan för de ca 80 ämneskategorier som ingick i deras dataset. Hur väl den satta gränsen matchade det optimala gränsvärdet för varje individuell kategori diskuterar Liebeskind m. fl. (2015) ej.

2. **Uteslut andra kategoriers namn** Liebeskind m. fl. (2015) argumenterar för att kategorinamnen ofta är valda med omsorg för att specifikt benämna en given kategori och därför ej bör kunna utgöra referensord till andra kategorier.

3. **Filtrera multipla expansioner** Utifrån samma resonemang som för kriterium 2, resonerar Liebeskind m. fl. (2015) att det är osannolikt att ett referensord stämmer in på två olika kategorier. Nya referensord bör därför enbart får expandera en kategoris referensordlista, vilket är den där det givna ordet har högst dice-koefficient.

4. **Filtrering av frekventa termer** Liebeskind m. fl. (2015) resonerar att ord som förekommer mycket frekvent i en korpus kan antas inte vara tillräckligt specifika för att tydligt utmärka en enskild ämneskategori, därför filtrerades nyckelord bort som förekom i mer än 4% av dokumenten i korpusen .

Utifrån det uppsatta användningsfallet finns det två problem med de ovan listade kriterierna för att filtrera ut relevanta referensord:

1. **Förutsätter kunskap om ämnesfördelning i data** Samtliga uppsatta kriterier förutsätter kunskap om ämnesfördelning i data för att ge goda resultat, vilket utgör ett brott mot krav 5 i användningsfallet. Om kriterium

2 och 3 som föreslår filtrering av andra kategorinamn och filtrering av multipla expansioner ska göra skillnad krävs kunskap om vilka andra närliggande ämneskategorier som förekommer i den data vi använder för att framställa nyckelord.

På samma sätt görs det antagande om ämnesfördelning när man sätter värden på paramterana i kriterium 1 och 4. Liebeskind m. fl. (2015) valde sina parametervärden så att de optimerade de genomsnittliga F1-poängen för de ca 80 olika kategorierna i det test-dataset de använde. Vad som är en optimal godtagbar frekvens, eller dice-koefficient för att automatiskt filtrera fram referensord kan dock förväntas variera mellan olika ämneskategorier. Om en av de ämneskategorier vi är intresserade av utgör mycket mer än 4% av texterna i den korpus vi använder för nyckelordsframställning är en frekvensgräns på 4% förödande.

2. Parameterar introducerar extra manuellt arbete Liebeskind m. fl. (2015) använde den data som utgör test data i deras studie för att välja parametervärden. För att kunna veta att parametrarna generaliserar och är relevanta för ny data, krävs det att val av parametervärden och validering av de valda värdena sker på olika testdataset. Att ha parametrar förutsättet därmed extra arbete i form mer etiketterad test-data vilket bör vägas mot parametrarnas nytta om krav 1 i användningsfallet ska anses vara uppfyllt.

2.5.1.3 Beräkna närhetsmått mellan nyckelord och dokument

Liebeskind m. fl. (2015) väjer att definiera närheten mellan ett dokument och de genererade nyckelorden enligt följande: Referensorden och kontextorden representeras med var sin vektor. Var vektor har samma antal dimensionen som antalet unika termer som förekommer i den data som ska kategoriseras. Dimensionerna som representerar nyckelord ges värdet 1 övriga dimensioner ges värdet 0. I referensordsvektorn tilldelas enbart de dimensioner som motsvarar referensord värdet 1 medan i kontextordsvektorn tilldelas alla genererade nyckelord värdet 1.

Därefter skapas en vektor för varje dokument där orden som förekommer i dokumentet tilldelas sitt TF-IDF värde och övriga termer tilldelas värdet 0. TF-IDF står för term frequency–inverse document frequencyöch är produkten av ett tf-värde respektive idf-värde (se ekvation 5 nedan).

$$TF - IDF(t, d) = tf(t, d) * idf(t, D) \tag{2.5}$$

där

$$tf(t, d) = 0.5 + 0.5 * \frac{f_{t,d}}{max f_{t',d} : t' \in d} \qquad idf(t, D) = \log \frac{N}{1 + |d \in D, t \in D|}$$

där $f_{t,d}$ är frekvensen av term t i dokument d och N är antal dokument i kopusen D.

Ett givet dokument d närhet till nyckelorden för kategori c beräknas därefter som en produkt av de två faktorerna, referensords-vektorns cosinuslikhet med dokumentvektorn och kontextords-vektorns cosinuslikhet med dokumentvektorn (se ekvation 6 nedan).

$$Närhet(d, c) = \frac{r_c}{||r_c|| * ||d||} * \frac{k_c}{||k_c|| * ||d||} \tag{2.6}$$

där r_c är referensorden för kategori c och k_c är kontextorden för kategori c

Genom att använda produkten av dokumentordsvektorns närhet till referensordsvektorn respektive kontextordsvektorn säkerställs att enbart dokument som innehåller referensord tilldelas ett närhetsmått högre än 0 för kategorin c. Hur högt närhetsmåttet blir ett mått på hur ofta nyckelorden förekommer i dokumentet normerat på textlängd.

2.5.1.4 Välja ut texter tillhörande kategorin baserat på närhetsmåttet mellan nyckelord och dokumentet

Liebeskind m. fl. (2015) prövar i sin studie tre olika metoder för att avgöra vilka texter som tillhör en given ämneskategori. Varje metod är vald för att den ska ge en möjlighet att välja en ungefärlig önskad precisionsnivå i det dataset som kategoriseras:

1. Välj ut de p procent högst rankade texterna för varje kategori. Ett lägre p ger högre precision men lägre täckning
2. Beräkna för varje kategori, differensen d i närhetsmått mellan det högst rankade dokumentet och det lägst rankade dokuementet för den givna kategorin. Välj en tröskel t för kategorisering baserat på en procentandel av d adderat till närhetsmåttet för det lägst rankade dokuementet:

   $$t(c) = min_c + d \cdot p$$

   där
   t(c) = tröskeln för att tillhöra kategorin c
   min_c = närhetsmåttet för den lägst rankade texten med en positiv närhet till kategorin c.
   p = en procentsats vald utifrån ungefärlig önskad precision
3. Simulera en mänsklig redaktör genom att manuellt sätta den lägsta tröskel som bibehåller den önskade precisionsnivån i dokumenten med närhetsmått över tröskeln.

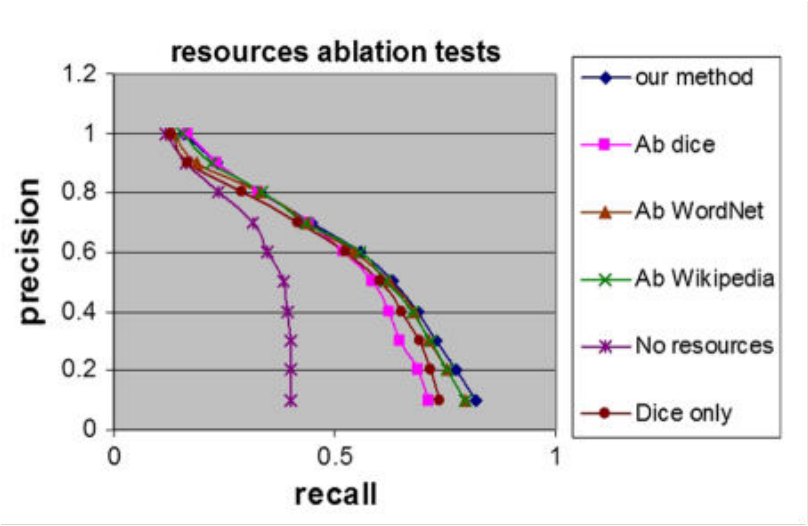
Liebeskind m. fl. (2015) fann att metod 2 gav bättre resultat än metod 1 men metod 3 gav den klart bästa kategoriseringsförmågan.

Ett problem med metod 3 är att Liebeskind m. fl. (2015) använde testdata för att välja optimala trösklar för respektive kategori och ingen diskussion förs kring om hur väl de manuellt satta trösklarna skulle generalisera till annan test-data. Innan uppföljande studier fastlagt i vilken utsträckning olika valda trösklarna generaliserar innebär nyckelordsbaserad textkategorisering därför att man behöver införa en mänsklig redaktör som manuellt måste kontrollera vilka av de texter som systemet kategoriserat korrekt och vilka de ej kategoriserat korrekt för att kunna säkerställa en viss precisionsnivå för en given kategori.

Det är möjligt att anta att man med tillräcklig mängd testdata som är representativ för den data som ska kategorisering kan förutspå relevanta tröskelvärden. Att manuell kontinuerligt utvärdera resultatet krävs även vid övervakad inlärning. Givet att det är möjligt att finna tröskelvärden som generaliserar till ny kan därför inte införandet av en mänsklig editor som tar fram tröskelvärden anses bryta mot krav 1 i det uppsatta användningsfallet som slår fast att algoritmen inte får kräva mer manuellt arbete än vid övervakad inlärning.

2.5.2 Dice-metodens kategoriseringsförmåga

Liebeskind m. fl. (2015) undersökte i sin studie hur nyckelord från olika resurser kunde komplettera varandra samt hur textkategorisering med enbart dice-nyckelord presterar i jämförelse med metoder som utnyttjar även manuellt framställda lexikala resurser. Algoritmerna utvärderades på det dataset baserat på IMDB data som presenteras i detalj i avsnitt 2.3.2 ovan. Resultatet valde Liebeskind m. fl. (2015) att redovisa som en graf som visar uppnådd täckning när tröskelvärden för kategorisering valts manuellt för att uppnå en specifik precisionsnivå (e figur 2 nedan).



Figur 2.2: Resultat för Dice-metoden (only dice) i jämförelse med andra nyckelordsresurser från Liebeskind m.fl.(2015)

2.5.3 Slutsats om dice-metodens potential att möta kraven i användningsfallet

Användningsfallet specificerar sex stycken krav för praktisk tillämpbarhet. Av granskningen ovan framgår att det är specifikt de filter som Liebeskind m. fl. (2015) föreslår för att automatiskt skilja referensord från kontextord som medför att algoritmen direkt bryter mot kraven i det uppsatta användningsfallet. Om algoritmen ändras på ett sätt som gör filtren överflödiga utan att introducera nya brott mot kraven är dock de problemen undanröjda.

För övriga delar av algoritmen gäller följande:

Krav 1 kan antas vara uppfyllt då det enbart krävs testdata för att validera om dice-metoden är lämplig för den givna tillämpningen.

Krav 2 i användningsfallet specificerar att det ska vara möjligt att förutse algoritmens precisions- och täckningsnivå utan att det kräver mer arbete än vid övervakad inlärning. Med en tillräcklig mängd testdata är det sannolikt möjligt att uppskatta precisions- och täckningsnivån för en specifik ämneskategori.

Krav 3 specificerar att kategoriseringsresultatet ska vara högre än om vi bara kategoriserar baserat på frekvens av kategorinamnet. Som framkommer av avsnitt 2.4.3.2 ovan tyder de resultat Liebeskind m. fl. (2015) redovisat att detta

krav är uppfyllt.

Krav 4-6 specificerar att algoritmen inte ska förutsätta någon specifik ämnesstruktur i träningsdata och gå att implementera för godtycklig språk och ämneskategori om det finns texter på det givna språket som innehåller den givna ämneskategorins manuellt framtagna nyckelord. Delstegen i algoritmen (exkluderat referensordsfiltren) utgör inga hinder för att uppfylla de kraven. Men vi vet från utvärderingar av algoritmer som använder övervakad inlärning att olika ämneskategorier tenderar att kräva olika mycket träningsdata för att uppnå likvärdiga resultat. Variansen i resultat mellan olika ämneskategorier kan vara lika stor vid nyckelordsbaserad ämneskategorisering. Eftersom resultaten av Dice-metoden enbart redovisats som genomsnittlig precision och täckning för samtliga kategorier kan vi inte uttala oss om och i så fall ungefär hur vanligt det är att algoritmen inte alls klarar av att kategorisera en given ämneskategori. Det krävs ytterligare testning för att en sådan uppskattning ska vara möjlig.

2.6 Gavagai Living Lexicon - ett nytt verktyg i nyckelordsbaserad textkategorisering

De samförekomstmått som använts för att framställa nyckelord i de befintliga algoritmer som presenterats och granskats ovan bygger alla på beräkningar av hur ofta nyckelorden förekommer i samma dokument/text-stycken som det givna kategorinamnen.

Relationen som tidigare studiers samförkomstmått speglar benämns inom lingvistiken, en syntagmatisk relation. Där syntagm är ett samlingsbegrepp för sekvenser av lingvistiska entiteter som bokstäver, ord, och meningar (Sahlgren, 2006).

Inom lingvistiska forskningsfältet har man kunnat visa att ord med likartad semantisk mening inte enbart utmärks av att de förekommer i sekvens utan att ett minst lika relevant kännetecken för ord med likartade semantiska egenskaper är att de tenderar att vara utbytbara i samma sekvenser. Ett exempel kan vara: “jag är glad” där ordet “glad” kan ersättas med andra ord som beskriver mänskliga egenskaper så som “ledsen”, “lång”, “hungrig” och “törstig”, medan ordet “jag” kan ersättas med andra ord för levande varelser som kan inneha egenskapen glad. Relationen som orden “ledsen”, “lång”, “hungrig” och “törstig” har till varandra brukar inom lingvistiken benämnas en paradigmatisk relation. Där paradigm står för ömsesidigt utbytbara lingvistiska entiteter .

Skillnaden mellan paradigmatiska och syntagmatiska relationer illustreras tydligt genom att representera likartade meningar i en 2-d matris där ord i samma

kolumn har en paradigmatisk relation till varandra och ord på samma rad har en syntagmatisk relation till varandra, se tabell 4 nedan.

Tabell 2.4: Exempelmeningar där orden i samma kolumn har en paradigmatisk relation och orden på samma rad har en syntagmatisk relation

Barnet	gillar	den	blå	leksaken
Hunden	biter	den	röda	bollen
Biet	gillar	de	stora	blommorna

Meronymer och hyponymer är exempel på ord som ofta har en stark paradigmatisk relation till varandra. Hyponymer avser ord som kan sägas vara underbegrepp till ett övergripande begrepp, t.ex. är hund och katt exempel på hyponymer till det övergripande begreppet djur. Meronymer är istället ord som är en del av det överordnade begreppet t.ex. är finger en meronym till hand och cykelhjul är en meronym till cykel.

Liebeskind m. fl. (2015) undersökte i sin studie hur ord från de lexikala resurserna Wikipedia och WordNet kunde komplettera nyckelorden genererade utifrån samförekomst i text med kategorinamnet. De fann att ord från Wikipedia inte gav någon mätbar förbättring för de ca 80 kategorier som testades men specifikt hyponymer och meronymer valda från WordNet gav en klar förbättring i kategoriseringsresultat på ca 0.05 F1-poäng. Eftersom användandet av en lexikal resurs som WordNet begränsar tillämpningarna till enbart engelska spårket är är det relevant att finna alternativa metoder för att berika nyckelordsrepresentationerna med hyponymer och meronymer.

Gavagai Living Lexicon är ett system för att bland annat detektera ord med stark paradigmatisk relation till varandra. Tyvärr har det dock visat sig svårt att på ett automatiserat sett skilja ut specifikt hyponymer och meronymer från antonymer (d.v.s motsatsord) och andra semantiskt relaterade ord som också tenderar att ha en stark paradigmatisk relation (Lenci & Benotto, 2012; Roller, Erk & Boleda, 2014) . Den lösning Gavagai valt att tillämpa är att bygga ett användarvänligt gränssnitt där användaren först ger ett "frö-ord" och därefter kan välja bland de paradigmatiskt närliggganande orden, och förfina sin sökning.

2.6.1 Random Indexing - en metod för att effektivt detektera paradigmatiskt närliggande ord

För att på ett effektivt sätt kunna detektera vilka ord som har en stark paradigmatisk relation till varandra krävs analys av stora mängder data. Den algoritm Gavagai Living Lexicon använder för att på ett skalbart och effektivt sätt kunna analysera ords paradigmatiska relation till varandra i stora datamängder, kallas

Random Indexing (RI).

Likt LSA är RI en metod för att i ett vektorrum representera hur ord samförkommer i text. Men RI skiljer sig från LSA i tre viktiga avseenden: (1) RI är en skalbar metod där minnesåtgången växer linjärt (istället för kvadratiskt) med antal unika kontexter och termer, (2) RI kräver ingen beräkningstung process för att ge information om vilka termer som är distributionellt relaterade. (3) Vektorrummet som byggs upp vid RI byggs upp sekvensiellt i takt med att ny data blir tillgänglig detta innebär att all data inte måste finnas tillgänglig innan analys kan göras (Sahlgren, 2005).

Detta gör RI mer praktiskt tillämpbart än LSA-metoden. Eftersom RI enbart förutsätter ej etiketterade texter är den även helt förenlig med det uppsatta användningsfallet.

Den grundläggande algoritm-idén vid RI är att skapa ett "ordrum" där varje term representeras av en kontext-vektor, och där ord med likartad distributionella egenskaper får kontextvektorer som har likartad riktning i ord-rummet. Algoritmen för att skapa ordrummet kan beskrivas i två huvudsteg (Sahlgren, 2005):

1. Tilldela varje unik kontext (dokument, stycke, "sliding-window" av termer, eller enskilda termer) en index-vektor med fix dimension d (ca 1000 dimensioner) där ett litet antal slumpmässigt valda dimensioner tilldelas värdet 1 eller -1 och övriga dimensioner tilldelas värdet 0.
2. Skapa en kontext-vektor för varje unik term genom att skanna av texter och för varje kontext där en given term förekommer addera kontextens index-vektor eller index-vektorerna för termerna som ingår i kontexten, till den givna termens kontext-vektor.

Som framgår av algoritmbeskrivningen ovan kan ordrum byggas för att spegla olika typer av distributionella relationer. För att skapa ett ordrum där vektorer med likartad riktning har en paradigmatisk relation beräknas uppdateringen av kontext-vektorerna enligt ekvation 7 nedan.

$$v(a) = v(a) + \sum_{t_v \cdots, -1, +1 \cdots t_h} w(b_j) \Pi r(b_j) \tag{2.7}$$

där
v(a) är kontext-vektorn för termen a som ska uppdateras
j är index för de termer (utöver a) som ingår i det fönster av termer som skannats, (term t_v till term t_h)
w(b) är en vikt funktion som viktar hur relevant termen b som ingår i fönstret är, typiskt har ord som förkommer mycket frekvent en lägre vikt för att inte blir för tongivande.

$r(b)$ är indexvektorn för term b
Π = är en permutation som roterar index vektorn för term b och används för att koda vilken position term b har i relation till term a i fönstret.

Resultatet blir ett ord-rum där ord med en paradigmatisk relation till varandra har kontextvektorer som sannolikt är mer parallella med varandra än ord som inte har en paradigmatisk relation. Denna effekt uppnås eftersom randomvektorerna med hög sannolikhet från början är närmast ortigonala med varandra (Sahlgren, 2006).

Utöver analys av ordvektorernas grad av parallellitet arbetar Gavagai utifrån hypotesen att ordrummets inre strukturer går att analysera närmare för att utvinna ytterligare relevant information om hur olika termer är relaterade, t.ex. är det intressant att kartlägga hur en terms grannar i ordrummet relaterar till varandra, då det kan antas delvis spegla olika betydelser av en term (Karlgren, Holst & Sahlgren, 2008).

Kapitel 3

Metod

Inom ramen för detta examensarbeta har ett användningsfall tagits fram som använts för att utvärdera den praktiska tillämpbarheten hos nyckelordsbaserade textkategorisering algoritmer från fem olika studier (Barak m. fl., 2009; Gliozzo m. fl., 2009; Ko & Seo, 2009; Liebeskind m. fl., 2015; Qiu m. fl., 2009). Utvärderingen som presenteras i avsnitt 2.4 ovan visar att dice-metoden (som presenterats i detalj i avsnitt 2.5) är den algoritm som har störst potential att möta kraven i det framtagna användningsfallet. Dice-metoden är därför den nyckelordsbaserade textkategoriseringsalgoritm som kommer ligga till grund för de experiment som här presenteras.

3.1 Algoritmförändringar för att öka den praktiska tillämpbarheten i Dice-metoden

Ur avsnitt 2.5.3 ovan framgår att det finns ett antal kvarvarande hinder för att dice-metoden ska ha en chans att möta kraven i det framtagna användningsfallet. Här presenteras därför tre algoritmförändringar för att undanröja de kvarvarande hindren och samtidigt försöka öka algoritmens precision och täckning.

3.1.1 Ersätt automatisk filtrering av referensord med manuella val

Som framgår i avsnitt 2.5 ovan förespråkar (Liebeskind m. fl., 2015) ett antal referensordsfilter som samtliga är problematiska ur ett praktiskt tillämpbarhets perspektiv.

Ett enkelt sätt att komma till rätta med de problem automatiskt filtrering av referensord skapar, är att introducera ett manuellt val av referensord. I de flesta praktiska tillämpningar är det troligt att en manuell kontroll av de förslagna nyckelorden ändå kommer genomföras för att säkerställa att systemet funnit en godtagbar representation av de valda ämneskategorierna. För en människa är det oftast vid detta tillfälle en lätt uppgift att avgöra vilka av de genererade nyckelorden som refererar direkt till kategorinamnet och vilka som enbart är ord som ofta förekommer frekvent med kategorinamnet. Detta ger också ett tillfälle att sortera bort föreslagna referensord som speglar icke relevanta betydelser av de givna kategorinamnet, ett exempel är kategorin "science" där systemet föreslog referensordet "science fiction" och "fiction" då de termerna samförekommer utmärkande ofta, men fortfarande inte är relevanta.

Optimalt bör valet av referensord genomföras av ett antal oberoende personer. I detta examensarbete fanns ej möjlighet till detta, därför utförde jag personligen de manuella valen av referensord. Detta introducerar en större subjektivitet än om valet utförts av flera personer. Valet utfördes dock utan kunskap om innehållet i texterna som utgjorde testdata, valet redovisas även i bilaga A för att möjligöra läsarens granskning.

3.1.2 Använd flera manuellt framtagna nyckelord för generering av kontextord

Samtliga nyckelordsbaserade algoritmer som studerats inom ramen för detta examensarbete har valt att använda en enskild term som manuellt framtaget "frö" för det kategorier de önskar detektera. Men t.ex. Liebeskind m. fl. (2015) nämner tt bättre valt kategorinamn i vissa fall skulle ge mer relevanta kontextord och därmed bättre kategoriseringsresultat.

Från ett perspektiv som har praktisk tillämpbarhet i fokus är det svårt att se någon motivering till att man skulle begränsa sig till ett enskilt kategorinamn. Här föreslås därför en mindre förändring av dice-algoritmen så att flera referensord kan ges som frö till algoritmen, se ekavation 8 nedan.

$$Dice([w_0, w_1, w_2..., w_i], w_j) = \frac{D([w_0, w_1, w_2..., w_i], w_j)}{D([w_0, w_1, w_2..., w_i]) + D(w_j)} \tag{3.1}$$

där $D([w_0, w_1, w_2..., w_i], w_j)$ är antalet dokument som innehåller både termen w_j och någon av termerna i $[w_0, w1, w2..., wi]$ och $D(w_j)$ är antalet dokument som innehåller termen w_j

Att de genererade kontextorden baseras på mer data kan antas bidra till att mer relevanta nyckelord genereras.

Nedan presenteras två förslag på referensord som tillsammans med kategorinamnet kan användas som "frö-ord" vid framställandet av dice-nyckelord.

3.1.2.1 Komplettera kategorinamnet med grammatiska böjningsformer

För de flesta kategorier är det lämpligt att använda alla böjningsformer av kategorinamnet som givna referensord. Om vi som ett exempel. önskar att detektera kategorin "hundar" bör ord med en hög dice-koefficient för dokument som innehåller ordet "hund", "hundens", "hundarnas" "hundar" värderas som lika relevanta nyckelord.

I detta skede föreslås att de grammatiska böjningsformerna väljs manuellt eftersom det först behöver klargöras hur mycket de ökar kategoriseringsförmågan. Metoder för automatisering är befogat först om vi kan bekräfta att det ger en klar förbättring i resultat.

3.1.2.2 Komplettera kategorinamnet med manuellt valda paradigmatiskt närliggande ord

I avsnitt 2.6 ovan presenteras begreppet paradigmatiskt relaterade ord och hur Gavagai Living Lexicion kan användas för att ta fram ord som har en paradigmatisk relationen till kategorinamnet.

Bland de paradigmatiska grannarna till en term återfinns ofta synonymer hyponymer och meronymer. Liebeskind m. fl. (2015) betonar i sin artikel att hyponymer och meronymer från WordNet utgjorde ett relevant komplement till nyckelord genererade med Dice-metoden. Det finns därför skäl att anta att samma typ av ord genererade via Gavagai Living Lexicon kan utgöra ett relevant komplement.

Tidigare studier har visat att det är svårt att med hög korrekthet på ett automatiserat vis skilja ut specifikt meronymer och hyponymer från andra paradigmatiska grannar i ett ordrum likt det Gavagai Living Lexicon bygger på (Lenci & Benotto, 2012; Roller m. fl., 2014). I detta skede föreslås därför att de pa-

radigmatiska granarana väljs manuellt bland de förslag Gavagai Living Lexicon föreslår för kategorinamnet och dess böjningsformer.

3.2 Utvärdering av kategoriseringsresultat

I avsnitt 2.3.1 ovan beskrivs de mått som tidigare använts för att mäta textkategoriseringsalgoritmers kategoriseringsförmåga. I samma avsnitt riktas kritik mot de befintliga utvärderingsmetoderna. Kärnan i kritiken mot befintliga mått ligger i att de inte ger någon information om hur kategoriseringsförmågan skiljer sig åt mellan olika kategorier.

Krav 2 i det framtagna användningsfallet säger att önskad minsta nivå på precision och täckning kan variera mellan olika tillämpningar men att det alltid är önskvärt att kunna förutspå algoritmens täckning och precision för de olika kategorierna som ska detekteras.

Givet att det inte existerar ett etiketterad testdataset som är representativt för det dataset som ska kategoriseras är det omöjligt att uttala sig med säkerhet om en viss algoritm kan uppnå en önskad täcknings- och precisionsnivå för en given ämneskategori. Den nya metod som här nedan föreslås kan dock ge en bättre förståelse för kategoriseringsförmågan hos den algoritm som utvärderas än tidigare föreslagna mått.

Förslaget är att täckning och precision redovisas i en matris, där Y-axeln visar precisionsnivå, X-axeln visar täckningsnivå och värdena i matrisen visar antalet ämneskategorier ur testdatasetet där det var möjligt att uppnå den givna täcknings och precisionsnivån samtidigt (se tabell 5, 6 och 7 i avsnitt 4 nedan).

Två olika algoritmer kan jämföras genom att man jämför antalet kategorier som klarar den för tillämpningen önskade täckning- och precisionsnivån. Utifrån hur många kategorier som algoritmen klarar de uppsatta kraven för, kan man ta beslut om det är värt att pröva algoritmen för den givna tillämpningen.

Det föreslagna måttet blir särskilt relevant för nyckelordsbaserad kategorisering. Här kan vi dra nytta av att det bara krävs ett etiketterat testdataset och inte någon etiketterad träningsdata som vid övervakad inlärning. Detta gör det praktiskt möjligt att skapa testdataset med ett stort antal kategorier. Om en algoritmen testats på ett stort antal olika kategorier och klarar de önskade kraven på täckning och precision för majoriteten av kategorierna, säger detta oss att denna algoritm är värd att pröva. Om tvärtom algoritmen endast klarar att uppnå de önskade resultaten på ett fåtal kategorier kan detta vara grund för att avfärda den givna algoritmen.

Test på ett stort antal kategorier möjliggör också analys av vilka egenskaper som utmärker de ämneskategorier som algoritmen inte klarar av att kategorisera. Detta kan utgöra viktig information både för vidareutveckling av befintliga algoritmer och för att avgöra om en befintlig algoritm sannolikt är tillämpbar på en specifik uppsättning ämneskategorier eller ej.

3.3 Data

För att utvärdera hur kategoriseringsförmågan i dice-algoritmen påverkas av ovan beskrivna algoritmförändringar användes det nya testdataset som Liebeskind m. fl. (2015) föreslår och som presenteras i detalj i avsnitt 2.3.2 ovan. Ett antal mindre förändringar av det ursprungliga datasetet genomfördes:

1. Kategorier med färre än fem dokument uteslöts från analysen eftersom ett enskilt korrekt respektive inkorrekt dokument annars kan göra allt för avgörande roll för den uppnådda täckningen och precisionen i den kategorin.
2. Kategorin “the environment” uteslöts ur datasetet eftersom funktionsord däribland ordet ”the” filtrerarades bort i pre-processning av data.
3. Dokumenten i kategorin “world war 2” adderades direkt till kategorin war och kategorinnamnet “world war 2” uteslöts eftersom alla tecken som ej är bokstäver filtrerades bort i pre-processning av data.

3.3.1 Pre-processning

Texterna i både träningsdata och testdata genomgick ett antal filter av pre-processning innan vidare analys tillämpades. Efter att texten hade delats upp i en lista av termer (ord, bi-gram och tri-gram) genomgick listan av termer följande 3 filter:

1. **Alla bokstäver omvandlades till gemener** För majoriteten av termerna är giltigt att det är samma term som avses oavsett om den är skriven med gemener eller delvis/enbart versaler.
2. **Termer med andra tecken än bokstäver filtrerades** Detta filter introducerades i samband med initiala tester med ett dataset som innehöll årtal och datum som visade sig förekomma frekvent i specifika ämneskategorier. I det dataset som slutligen valdes för att testa kategoriseringförmågan, är det möjligt att detta filter var överflödigt.

3. **Filtrering av funktionsord** Natural Language Tool Kit (NLTK) är ett frekvent använt Python ramverk i implementationer där mänskligt språk processas. I NLTK ingår stöd för att filtrera funktionsord. Detta filter motiverades av att Liebeskind m. fl. (2015) beskrivit att detta tillämpades i deras implementation av dice-algoritmen. Även om manuellt framställda listor med funktionsord kan betraktas som att man introducera ett språkberoende i algoritmen, ansågs detta godtagbart då det är möjligt att framställa listor av sannolika funktionsord ur Gavagais lexicon som bygger på ordförekomst-statistik ur mycket stora textdatamängder.

.

Kapitel 4

Resultat

I detta avsnitt presenteras en utvärdering av hur de algoritmförändringar av dice-metoden som föreslagits i avsnitt 3.1 ovan påverkar algoritmens kategoriseringsförmåga. För att mäta kategoriseringsförmåga används den metod som föreslås i avsnitt 3.2 ovan. Som jämförelse används både dice-metoden orginalversionen av dice-metoden så som den beskrivs av Liebeskind m. fl. (2015) och en kategorisering baserat på enbart frekvens av kategorinamnet, se tabell 5,6 och 7 nedan. För att möjliggöra direkt jämförelse med resultaten i Liebeskind m. fl. (2015) studie beskrivs även algoritmernas uppmätta genomsnittliga kategoriseringsförmågan med en precisions-täckningsgraf, se figu X nedan.

Tabell 4.1: Resultat för kategorisering baserat på enbart kategorinamn: tabellvärden motsvarar antalet kategorier där den aktuella täcknings- och precisionsnivån var möjlig att uppnå samtidigt

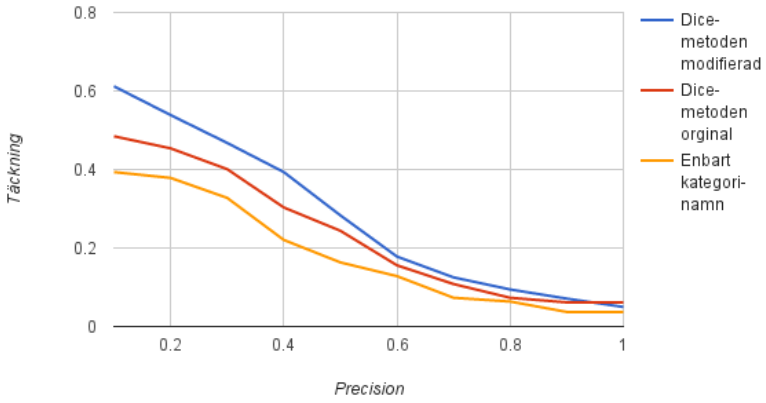
precision/täckning	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	44	44	44	44	44	44	44	44	44	44
0.1	33	22	20	17	14	12	8	7	2	1
0.2	32	21	20	16	14	12	6	6	2	1
0.3	26	17	15	14	11	9	4	2	1	1
0.4	23	14	12	8	7	6	3	2	1	1
0.5	21	13	9	8	5	3	2	2	1	1
0.6	17	9	7	7	1	1	1	1	0	0
0.7	10	7	4	2	1	1	1	1	0	0
0.8	8	7	3	2	1	0	0	0	0	0
0.9	6	4	2	2	1	0	0	0	0	0
1.0	6	4	2	2	1	0	0	0	0	0

Tabell 4.2: Resultat för kategorisering med originalversionen av dice-metoden: tabellvärden motsvarar antalet kategorier där den aktuella täcknings- och precisionsnivån var möjlig att uppnå samtidigt

precision/täckning	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	44	44	44	44	44	44	44	44	44	44
0.1	36	27	23	19	17	15	15	12	5	3
0.2	36	26	22	19	15	12	12	8	4	3
0.3	32	23	21	17	14	11	8	6	3	2
0.4	29	21	17	13	11	7	6	4	1	1
0.5	26	19	13	10	7	4	3	2	1	1
0.6	18	11	6	4	3	2	2	2	0	0
0.7	15	8	5	2	2	2	1	0	0	0
0.8	13	4	2	1	1	1	0	0	0	0
0.9	12	4	2	1	1	0	0	0	0	0
1.0	12	4	2	1	1	0	0	0	0	0

Tabell 4.3: Resultat för kategorisering med dice-metoden med förslagna algoritmförändringar : tabellvärden motsvarar antalet kategorier där den aktuella täcknings- och precisionsnivån var möjlig att uppnå samtidigt

precision/täckning	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	44	44	44	44	44	44	44	44	44	44
0.1	40	38	35	32	28	25	20	16	7	3
0.2	38	36	29	25	23	17	12	9	5	3
0.3	38	31	26	23	19	14	10	5	3	2
0.4	36	27	21	19	13	10	6	4	2	2
0.5	32	21	14	14	7	7	5	3	1	1
0.6	20	13	10	9	6	5	3	3	0	0
0.7	14	10	7	5	4	3	2	0	0	0
0.8	11	7	5	4	2	2	2	0	0	0
0.9	10	6	3	2	1	1	1	0	0	0
1.0	9	5	3	2	1	1	1	0	0	0



Figur 4.1: Genomsnittlig uppmätt täckning vid givna precisionsnivåer.

Kapitel 5

Diskussion

Nyckelordsbaserad textkategorisering har i ett flertal artiklar lyfts fram som ett mer praktiskt tillämpbart alternativ än de kategoriseringsalgoritmer som kräver manuellt etiketterad träningsdata. I kontrast till detta visar den utvärdering som presenterats ovan att var och en av fem granskade nyckelordsbaserade kategoriseringsalgoritmerna introducerar nya betydande hinder för praktisk tillämpbarhet som ej lyfts upp och diskuteras i de aktuella artiklarna.

För att definiera praktisk tillämpbarhet har i detta examensarbete ett industriellt motiverat användningsfall tagits fram (se avsnitt 2.4.1 ovan). Specifikt visade utvärderingen att ingen av de fem granskade algoritmerna, i sin nuvarande utformning, kunde möta kraven i det framtagna användningsfallet.

Den algoritm som utifrån utvärderingen bedömdes ha störst potential att möta kraven i det framtagna användningsfallet var Dice-algoritmen föreslagen av Liebeskind m. fl. (2015). Denna algoritm valdes därför som grundalgoritm för fortsatta experiment

5.1 Utvärdering av dice-algoritmens kategoriseringsförmåga

I avsnitt 3.2 ovan föreslås en ny metod för att mäta kategoriseringsförmåga. Syftet med det nya måttet är att till skillnad från tidigare studiers mått kunna visualisera hur mycket kategoriseringsförmågan varierar mellan olika ämneskategorier.

Resultatet visar att den ursprungliga dice-algoritmens kategoriseringsförmåga varierar betydligt mellan de olika ämneskategorierna. För över 10% av kategorierna var det inte möjligt att uppnå en (samtidig) täcknings- och precisionsnivå över 0.1, och för enbart ett fåtal kategorier uppnåddes en precisionsnivå över 0.6 som ses som ett ungefärligt minimikrav för många praktiska tillämpningar (Liebeskind m. fl., 2015).

I ovan presenterade resultat bör dock noteras att jag ej lyckats reproducera de resultat Liebeskind m. fl. (2015) rapporterade för samma dataset och samma testade kategoriseringsalgoritmer (dice-metoden och kategoriserade baserat enbart på kategorinamn).

En möjlig förklaring till skillnadena i resultat är att det existerar ett fel i någon av studiernas implementationer. Kategorisering baserat på enbart kategorinamn bör dock betraktas som en enkel algoritm. Med anledning av de stora skillnader i resultat genomfördes utöver enhetstester av funktionerna inblandade i kategoriseringen, en manuell inspektion av kategoriseringen för ett antal kategorier och inga fel kunde upptäckas. Detta talar för att det antingen existerar i den implementation (Liebeskind m. fl., 2015) använt eller att det existerar en skillnad mellan den testdata Liebeskind m. fl. (2015) använde och den testdata som använts i denna studie.

På grund av upphovsrättsliga skäl kunde Liebeskind m. fl. (2015) ej publicera sitt datasetet i sin helhet. Testdatasetet som använts i detta examensarbete är därför konstruerat utifrån IMDB:s öppet tillgängliga datafiler och ett dokument som specificerar titel, år och annotering för ca 2000 filmbeskrivningar. Tillgång till detta dokument fick jag genom kontakt med artikelförfattarna i Liebeskind m. fl. (2015). En möjlighet är därför att de filmbeskrivningarna som IMDB har lagrade har modifierats för några av de annoterade aktuella filmerna, eller att det dokument som låg till grund för mitt testdatasetet inte korrekt speglar den testdata Liebeskind m. fl. (2015) använde i sin studie.

5.2 Effekter av föreslaga algoritmförändringar

De algoritmförändringar som föreslagits ovan syftade till att både byta ut de delar av dice-algoritmen som bryter mot det framtagna användningsfallet och höja korrektheten i kategoriseringen.

Resultatet för dice-algoritmen med de föreslagna algoritmförändringarna visar sig vara i stort likvärdigt med resultat för den ursprungliga dice-algoritmen föreslagen. Enbart en trend av förbättrade täckning kunde observeras för de låga precisions-nivåerna.

Inga test genomfördes för att utvärdera om de testade algoritmförändringarna medförde någon signifikant förbättring i resultat. Men värdet av ett sådant signifikanstest kan också ifrågasättas. Skillnaderna på de precisions- och täckningsnivåer som är relevanta för de flesta praktiskt tillämpningarna var mycket små.

Det är också viktigt att observera att värdena i resultat-matrisen bör tolkas som värden med ett konfidensintervall. Testdatasetet utgör enbart ett slumpmässigt urval av 2000 filmbeskrivningar från en databas med nära 400 000 filmbeskrivningar. Om andra dokument hade valts som testdata hade kategoriseringsresultatet för många kategorier sannolikt varit annorlunda. Valet av ämneskategorier påverkar även det resultatet. Andra ämneskategorier för samma data skulle sannolikt ge ett annorlunda utfall. Resultatet talar dock för att dice-algoritmen behöver vidareutvecklas ytterligare för att både minska variansen i resultat mellan olika ämneskategorier och generellt höja kategoriseringsförmågan.

Ytterligare en begränsning för praktisk tillämpning av nyckelordsbaserad textkategorisering som framkommit i och med den stora variansen i resultat för olika ämneskategorier är att det kommer krävas ett manuellt framställt representativt tillämpnings-specifikt testdataset. Ett sådant testdataset krävs för att säkerställa algoritmens ungefärliga täckning och precision för de valda ämneskategorierna. Detta kan antas vara sant även om vi förbättrar den generella kategoriseringsförmågan om inte variansen i resultat minskar betydligt. Det nya dataset Liebeskind m. fl. (2015) belyser att detta är en betydande begränsning i tillämpbarhet. I framställandet av det nya datasetet etiketterade manuellt ca 2000 dokument vilket gav ett dataset där ca hälften av 80 utvalda ämneskategorier innehöll färre en 5 dokument. Tillämpningar där vi önskar känna till ungefärlig täcknings- och precisionsnivå, och där vi önskar detektera en relativt sällsynt ämneskategori finns det alltså bara möjlighet att dra nytta av nyckelordsbaserad textkategorisering ifall det är klart mer än några tusentals dokument som ska analyseras.

5.3 Förslag på framtida forskning

Slutsatsen från både utvärderingen av befintliga nyckelordsbaserade textkategoriseringsalgoritmer och de algoritmförändrings-experiment som genomförts ovan är att metoderna för nyckelordsbaserad textkategorisering behöver vidareutvecklas för att möta kraven i många praktiska tillämpningar. Här nedan presenteras ett antal områden som jag finner viktiga att studera vidare för att kunna komma tillrätta med ovan listade problem.

5.3.1 Hur ökar vi kategoriseringsförmågan

Den avgörande faktorn för kategoriseringsresultatet är kvalitén på nyckelorden. I avsnitten nedan diskuteras därför olika metoder för att höja kvalitén på de genererade nyckelorden

5.3.2 Utnyttja frekvens av givna frö-ord i beräkningen av dice-nyckelord

I dice-algoritmens grundutförande görs ingen skillnad på ord som samförekommer i ett dokument där de givna frö-orden förekommer endast vid ett tillfälle och dokument där de givna frö-orden förekommer frekvent. Men dokument där de givna frö-orden förekommer mer frekvent har högre sannolikt att verkligen tillhöra den givna ämneskategorin. I framtida studier vore det därför intressant att undersöka om en uppviktning av kontextord som förekommer i dokument där det givna nyckelordet förekommer frekvent, skulle ge en förbättring i kategoriseringsresultat.

5.3.3 Introducera mer komplexa samförekomst mätningar för tvetydiga ämneskategorinamn

Ämneskategorierna i denna studie var aktivt valda för att i yttersta möjliga mån inte ha tvetydiga kategorinamn. I många praktiska tillämpningar är det dock sannolikt att man kan önska använda sig av ämneskategorier där kategorinamnen är tvetydiga. Ett exempel kan t.ex. vara företagsnamn som "Apple" eller engelska ordet "rock" i bemärkelsen musikgenre. Här vore det önskvärt att kunna utesluta dokument som innehåller t.ex. ordet "fruit" eller "climbing". En annan möjlighet är att addera en lista med andra ord som måste förkomma tillsammans med det givna kategorinamnet för att dokumentet ska ingå i analysen

5.3.4 Ersätt dice-koefficienten med ett syntagmatiskt Random Indexing ordrum

Vid experimenten var det möjligt att genom manuell inspektion se en trend mot fler relevanta nyckelord när fler ord användes som "frö-ord" till dice-algoritmen. Exempel på förbättrad kvalité fanns bland de kategorier som präglats av att för få dokument låg till grund för nyckelordsberäkningen och som därför hade kontextordlistor som dominerades av tri-gram (som sannolikt enbart förkommit i en eller ett fåtal av de dokument där det givna nyckelordet förekommit). När

fler “frö-ord” användes byttes en del av dessa tri-grams ut till termer som upplevdes som mer utmärkande för ämneskategorin. En möjlighet för att upptäcka relevanta frö-ord är därför att jobba iterativt och för varje iteration välja ut nya lämpliga referensord som tillsammans med dessas relevanta paradigmatiska grannar används som frö till nästa iteration.

Ett problem med att använda dice-algoritmen för iterationer är att varje iteration kräver en ny genomgång av de texter som innehåller någon av de de givna nyckelorden. Detta blir en tidsödande beräkning om antalet kategorier är stort.

Här föreslås att framtida studier prövar att använda metoden Random Indexing (RI) (som presenteras i avsnitt 2.6 ovan), för att kartlägga även syntagmatiska relationer. Förslaget är att man bygger ett ord-rum som likt dice-koefficienten lagrar information om termers syntagmatiska närhet på dokumentnivå. Detta är möjligt genom att varje dokument tilldelas en unik slumpgenererad index-vektor och termernas kontext-vektorer byggs upp genom addering av index-vektorerna för de dokument där termen förekommer. RI innebär visserligen en komprimering av informationen som dice-koefficienten förmedlar men metoden har flera fördelar förutom att det underlättar snabba iterationer. RI gör det lättare att uppdatera modellen med ny data, eftersom ordrummet byggs upp sekventiellt och kan lätt uppdateras utan några tyngre beräkningar. Det kan även vara intressant att analysera den inre-strukturen ord-rummet då den kan ge intressant information om hur de olika syntagmatisktgrannarna är relaterade till varandra. Om de syntagmatitska granarna grupperas utifrån deras släktskap sinsemellan är det sannolikt möjligt att detektera när ett ord har flera lite olikartade betydelse. Om vi använder exemplet med engelska ordet “rock” är det sannolikt att ordet “climbing-gear” förekommer i vissa dokument som även “rock” förekommer i men ordet “music” förekommer inte i samma dokument som “climbing-gear”. Detta gör att vi kan välja de nyckelord som ingår i klustret med både “rock” och “music” men utesluta nyckelorden som förekommer enbart i klustret med “rock” och “climbing-gear”.

5.3.5 Hur hanterar vi variansen i resultat mellan olika ämneskategorier?

Även om den generella kategoriseringsförmågan i de nyckelordsbaserade textkategoriseringsalgoritmerna förbättras är det ett hinder för den praktiska tillämpbarheten om den stora variansen i resultat kvarstår. Om variansen i resultat är stor blir det en chanstagning om algoritmen kommer fungera för den givna tillämpningen. Bara det manuella arbetet att framställa relevant test-data för de ämneskategorier man önskar att detektera kan bedömmas vara en för stor investering om algoritmen ger ej godtagbara katetgoriseringsresultat för flera tidigare testade ämnes-kategorier.

Om vi på förhand skulle kunna förutspå vilka ämneskategorier som kommer uppnå bra respektive ej godtagbara resultat med en given algoritm skulle detta ha stor betydelse för den praktiska tillämpbarheten. Ett viktigt område för framtida studier är därför att försöka kartlägga vad som utmärker de ämneskategorier som får utmärkande bra respektive dåligt resultat. Nyckelordsbaserad textkategorisering lämpar sig särskilt bra för denna typ av analyser eftersom det kräver mycket mindre arbete att framställa dataset med ett stort antal ämneskategorier,

Ett intressant experiment vore att låta en oberoende person värdera kvalitén på de framställda nyckelorden, för att därefter undersöka hur väl denna mänskliga kvalitetsbedömning korrelerar med uppnådda kategoriseringsresultat.

Ett annat sätt att uppnå bättre förståelse för vad som ger bra respektive dåliga resultat är att analysera vilka feltyper som dominerar i de ämneskategorier som presterar bra respektive dåligt (se avsnitt 2.3.3 ovan om felanslys).

5.3.6 Hur minskar vi mängden manuellt arbete?

Fokus för vidareutveckling bör ligga på förbättra kvalitén i kategoriseringen innan ytterligare algoritm-steg automatiseras. Om önskvärd kvalité uppnås finns det dock ett kvarvarande område som är särskilt intressant att automatisera.

Det steg i den modifierade dice-algoritmen som fortfarande kräver betydande manuellt arbete är valet av vilka dokument som ska tilldelas en specifik kategori. Idag är tanken att detta görs genom en mänsklig editor som drar en gräns där de rankade dokumenten understiger en önskad precisionsnivå. I framtida studier kan det därför vara intressant att undersöka hur väl de manuellt valda trösklarna generaliserar till ny test-data.

5.4 Slutsatser

Den viktigaste slutsatsen i detta examensarbete är att alla de nyckelordsbaserade textkategoriseringsalgoritmerna som här granskats har betydande inneboende hinder för att fungera i många praktiska tillämpningar. Specifikt kunde ingen av algoritmerna möta kraven i det industriellt motiverade användingsfall som tagits fram för att mäta praktisk tillämpbarhet.

Som grund för de algoritmförändrings-experiment som genomfördes användes dice-metoden föreslagen av Liebeskind m. fl. (2015) eftersom det var den av de granskade algoritmerna som bäst mötte kraven i det framtagna anvädningsfallet.

Med de förslagna algoritmförändringar var det möjligt att undanröja de punkter där dice-algoritmen direkt bröt mot det framtagna användingsfallet. Men ur det genomförda experimenten framgick även att det ej gick att reproducera de resultat Liebeskind m. fl. (2015) redovisade för samma dataset och algoritmer.

Genom att föreslå och tillämpa en ny metod för att redovisa kategoriseringsresultat belystes att det existerade en stor varians i resultat för olika ämneskategoirer och att detta utgör ett betydande kvarvarande hinder för praktisk tillämpbarhet av den modiferade dice-metoden.

Sammanfattningsvis visar resultaten från detta examensarbete problemen på att många akademiska studier av textkategorisering helt fokuserat på att maximera genomsnittliga kategoriseringsförmågan på några få kända dataset, och därmed missat aspekter som är avgörande för att föreslagna algoritmer ska fungera i många praktiska tillämpningar.

Framtida studier av nyckelordsbaserad textkategorisering förslås därför undersöka hur vi kan öka generella kategoriseringsförmågan utan att göra avkall på kraven om praktisk tillämpbarhet.

Ett annat viktigt område för framtida studer är att undersöka om det är möjligt att på förhand förutså vilka ämneskategorier som kommer vara enkla respektive svåra att kategorisera, t.ex. genom analys av genererade nyckelord . Kunskap om vilka kategorier som funger väl respektive mindre bra skulle kunna utgöra viktig guidning för valet att tillämpa nyckelordsbaserad textkategorisering eller ej.

Referenser

Adami, G., Avesani, P. & Sona, D. (2003). Bootstrapping for hierarchical document classification. I *Proceedings of the twelfth international conference on information and knowledge management* (s. 295–302).

Aggarwal, C. C. & Zhai, C. (2012). A survey of text classification algorithms. I *Mining text data* (s. 163–222). Springer.

Baharudin, B., Lee, L. H. & Khan, K. (2010). A review of machine learning algorithms for text-documents classification. *Journal of advances in information technology*, *1*(1), 4–20.

Barak, L., Dagan, I. & Shnarch, E. (2009). Text categorization from category name via lexical reference. I *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics, companion volume: Short papers* (s. 33–36).

Dumais, S. T. (2004). Latent semantic analysis. *Annual review of information science and technology*, *38*(1), 188–230.

Gliozzo, A., Strapparava, C. & Dagan, I. (2009). Improving text categorization bootstrapping via unsupervised learning. *ACM Transactions on Speech and Language Processing (TSLP)*, *6*(1), 1.

Islam, A., Milios, E. & Kešelj, V. (2012). Comparing word relatedness measures based on google n-grams.

Karlgren, J., Holst, A. & Sahlgren, M. (2008). Filaments of meaning in word space. I *Advances in information retrieval* (s. 531–538). Springer.

Ko, Y. & Seo, J. (2009). Text classification from unlabeled documents with bootstrapping and feature projection techniques. *Information Processing & Management*, *45*(1), 70–83.

Lenci, A. & Benotto, G. (2012). Identifying hypernyms in distributional semantic spaces. I *Proceedings of the first joint conference on lexical and computational semantics-volume 1: Proceedings of the main conference and the shared task, and volume 2: Proceedings of the sixth international workshop on semantic evaluation* (s. 75–79).

Lewis, D. D. (1995). Evaluating and optimizing autonomous text classification systems. I *Proceedings of the 18th annual international acm sigir conference on research and development in information retrieval* (s. 246–254).

Liebeskind, C., Kotlerman, L. & Dagan, I. (2015). Text categorization from category name in an industry-motivated scenario. *Language Resources and Evaluation*, *49*(2), 227–261.

Marsland, S. (2015). *Machine learning: an algorithmic perspective*. CRC press.

McCallum, A., Nigam, K., Rennie, J. & Seymore, K. (1999). A machine learning approach to building domain-specific search engines. I *Ijcai* (vol. 99, s. 662–667).

Qiu, Q., Zhang, Y., Zhu, J. & Qu, W. (2009). Building a text classifier by a keyword and wikipedia knowledge. I *Advanced data mining and applications* (s. 277–287). Springer.

Rocha, L., Mourão, F., Mota, H., Salles, T., Gonçalves, M. A. & Meira Jr, W. (2013). Temporal contexts: Effective text classification in evolving document collections. *Information Systems*, *38*(3), 388–409.

Roller, S., Erk, K. & Boleda, G. (2014). Inclusive yet selective: Supervised distributional hypernymy detection. I *Coling* (s. 1025–1036).

Sahlgren, M. (2005). An introduction to random indexing. I *Methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering, tke* (vol. 5).

Sahlgren, M. (2006). *The word-space model* (opublicerad doktorsavhandling). Citeseer.

Sahlgren, M., Gyllensten, A. C., Espinoza, F., Hamfors, O., Holst, A., Karlgren, J., ... Viswanathan, A. (2016). The Gavagai Living Lexicon. I *10th edition of the language resources and evaluation conference, 23-28 may 2016, portoroz (slovenia).*

Schohn, G. & Cohn, D. (2000). Less is more: Active learning with support vector machines. I *Icml* (s. 839–846).

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

Tong, S. & Koller, D. (2002). Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, *2*, 45–66.

Yang, Y. & Liu, X. (1999). A re-examination of text categorization methods. I *Proceedings of the 22nd annual international acm sigir conference on research and development in information retrieval* (s. 42–49).

Appendix

Manuellt valda referensord för ämnes-kategorier i testdata

airplanes air force, aircraft, airline, airliner, airliners, airlines, airplane, airplanes, airport, aviation, bussiness class, cockpit, flight, jet, pilot, pilots, plane, planes, stewardess

aliens alien, aliens, cyborg, cyborgs, extraterrestrial, ufo

animals animal, animals, apes, birds, butterflies, chimps, cows, elephants, frogs, horses, insects, mammals, monkeys, parrots, pigs, primates, rabbits, reptiles, turtles

arts art, artist, artists, arts, footage, gallery, studio

baseball baseball, major league, pitcher baseball, major league, pitcher

basketball basketball, nba, ncaa

boxing boxer, boxers, boxing, knockout, muhammad ali, rocky, sparring

business business, businessman, businessmen, company

christianity biblical, catholiscism, chatolic, christianity, christians, crucified, crucifixion, evangelical, jesus, messiah, mormonism, pastor, satan

cinema cinema, cinemas, cinematic

college/university college, graduate school, high school, highschool, professor, undergraduate, university

comic-book batman, comic, comic book, comic books, comics, superman

dance ballet, choreography, dance, dancer, dancers, dances, dancing

disability adhd, alzheimer, alzheimers, autism, autist, brain damage, cerebral palsy, cognitive impairment, dementia, disabilities, disability, disabled, disabled children, disabled people, dyslexia, epilepsy, hydrocephalus, intellectually disabled, ocd, rett syndrom, schizophrenia, special needs, spinal injury, wheelchair

drugs amphetamine, cannabis, cocaine, crack cocaine, dealer, drug, drugs, ecstasy, heroin, illegal substances, meth, narcotics, opiates, overdose

fraud bribe, bribery, bribes, corrupt, corruption, forgery, fraud, frauds, money laundering, perjury

gambling bingo, blackjack, casino, casinos, gambler, gambling, poker, roulette

hip hop hip hop

history historical, history

journalism editor, journalism, journalist, journalists, news, newsman, newsmen, photographer, photographers, reporter, reporters

legal arrested, court, first amendment, law, lawmaker, lawyer, legal, legalism, legislation

literature fiction, limericks, literature, literatures, nonfiction, poem, poet, poetic, poetics, poetry

mafia gangster, gangsters, mafia, mafias, mafiosa, mafioso, mobster, organized crime

martial-arts black belt, bruce lee, karate, kung fu, martial art, martial arts, martial arts expert, mixed martial arts, mma

medicine cardiology, dentistry, doctor, doctors, epideiology, medicine, neurology, nurse, patient, patients, pharmacology, psychiatry, surgery

military air force, air forces, army, general, generals, militarily, military, navy, officer, officers, soldier, soldiers, war

music music, musician, musicians

mythology greek myth, mythologie, mythologies, mythology

nature forest, forests, landscape, mountain, mountains, nature, ocean, oceans, wild life

pets cat, cats, dog, dogs, kittens, pet, pets, puppy

political congress, democrats, elections, geo political, geopolitcs, governing party, left wing, leftwing, liberalism, marxist, political, politician, politicians, politics, reformist, republicans, right wing, rightwing, senator, social democrats

pop/rock pop, rock

prison jail, prison, prisoner, prisoners, prisons

psychology cognitive science, counselor, mental helth, psychiatry, psychologist, psychology, suicide

rasicm apartheid, civil rights movement, martin luther king, racial inequality, racism, racist, racists, segregation, skinheads, slavery, white privilege

religion church, god, religion, religions, religiosity, religious, spiritual

school school, schools

science laboratory, science, scientific, scientist, scientists, university

showbiz entertainment, showbiz

sport sport, sports

theater broadway, shakespeare, theater, theaters

traditions folklore, tradition, traditions

trawel journey, journeys, travel, traveling, travels, treveler, trips

war army, battle, military, soldiers, war, wars