



Examensarbete

Sökmotorlösning för e-handel

Babak Razavi

Siddhartha Pattni

Institutionen för Elektro- och informationsteknik

Lunds Tekniska Högskola, Lunds universitet

SE-221 00 Lund, Sweden

Printed in Sweden

E-huset, Lund, 2011

Abstract

A search function within E-Commerce is a service that may lead the potential customers to products and services they are looking for. A search function that provides the desired result to a customer may facilitate online shopping for its customers. Further, there are many companies who provide E-commerce services and makes competition more intense and for these reasons the search function is an important subject within E-Commerce.

There are many Search Engine solutions that can be used to develop the search function within an E-Commerce site. This thesis presents some of the available search engines and some methods to evaluate them. Further it presents a framework that have been developed and used to evaluate search engines.

During the thesis search engines were divided into two different categories, open source and commercial. This division was then used as a natural divider since the workload was divided between two persons where Babak was responsible for the commercial search engines and Siddhartha for the open source search engines.

Tack till

Handledare Erik Arnwald från Avensia AB

Examinator Anders Ardö från institutionen för Elektro- och informationsteknik.

Mats Berglund på biblioteket i E-huset

Även till följande personer som hjälpt oss under arbetets gång:

Jussi Karlgren från SICS, Christian Tellström, Johan Liljeros, Fredrik Ljungbeck från Avensia AB och slutligen Eva Andersson från studieverkstaden vid Lunds universitet.

Disposition

Kapitel 1

Inledning:

Ger en introduktion till rapporten och presenterar bakgrund, syfte och problemställning. Innehåller även information om viktiga begrepp som nämns i rapporten.

Kapitel 2

Genomförande:

Beskriver hur arbetet har utförts, tillvägagångssätt för kartläggning och urval av sökmotorer och avslutats med en utvärdering.

Kapitel 3

Diskussion och slutsats:

Innehåller reflektioner och slutsatser, förslag på förbättring och vidareutveckling.

Inledning	1
Bakgrund	1
Problembeskrivning.....	1
Syfte.....	1
Mål.....	2
Avgränsningar.....	2
Terminologi	2
E-handel.....	3
Informationsåtervinning.....	4
Sökmotorer.....	4
Genomförande	7
Kravhantering.....	7
Testspecifikation	8
Kartläggning av befintliga sökmotorer	10
Artiklar och tidigare studier.....	11
Verktyg för insamling av information.....	12
Enkät.....	15
Urval av befintliga sökmotorer.....	16
Sökmotorer med öppen källkod.....	17
Kommersiella sökmotorer	20
Första urvalet	26
Utvärdering	34
Utvärderingsverktyg	35

Informationssamlingar	41
Testramverket	44
Testning	51
Avgränsningar.....	52
Andra urvalet.....	53
Lucene	54
SQL Full Text Search	55
Sphinx	57
Diskussion och slutsats	59
Resultat av första urvalet	59
Testresultat från testspecifikationen	63
Slutsats	64
Möjligheter för vidareutveckling av detta projekt	65
Återkoppling och synpunkter	66
Referenser	67
Appendix A. Förstudie	71
Bakgrund	71
Problembeskrivning.....	71
Syfte.....	71
Krav och avgränsningar	72
Genomförande	72
Möten.....	73
Redovisningar	73
Ansvarsfördelning.....	73
Tidsplanering	74

Appendix B. Kravspecifikation	79
Revisionsinformation.....	79
Inledning.....	79
Bakgrund och Målsättningar	79
Systemkrav	81
Kvalitetskrav	92
Tillförlitlighet	94
Prestanda.....	95
Kostnader	96
Appendix C. Tidsplan	97
Appendix D. Prioritering av kraven	99
Appendix E. Tester.....	105
Testspecifikation.....	105
Testinstruktioner	112
Appendix F. Enkät svar	159
Appendix G. Första urvalet.....	161
Sökmotorer med öppen källkod.....	161
Appendix H. Installationsmanualer	165
Installationsguide för Lucene	165
Installationsguide för Sphinx	168
Appendix J. Antal träffar på Altavista	171
Appendix K. Testresultat	185
Lucene	185
SQL Full Text Search	187

Inledning

Detta kapitel presenterar examensarbetet och de grundläggande begrepp som används i arbetet.

Bakgrund

Idag finns det ett behov av att förbättra sökfunktion inom elektronisk handel. Man vill spåra frekvent förekommande sökfrågor och populära produkter, vikta sökresultat efter målgrupp och säljstatistik, hämta relevant information från forum, bloggar, recensioner och omdömen för att förbättra sökfunktionen.

Problembeskrivning

Några av bristerna som uppstår när man använder sökfunktionen är att man inte får träff på de produkter man letar efter, man får irrelevanta svar, produkterna är sorterade på ett sätt som gör det svårt att hitta den produkt man letar efter.

Det finns ett stort utbud av sökmotorer som är utvecklade för olika syften men det är svårt att hitta någon sökmotor som kan uppfylla alla krav som finns inom e-handel. Dessutom saknas någon aktuell och tillgänglig informationskälla som tillhandahåller jämförelser av sökmotorer där det framgår vilka är för- och nackdelar av de befintliga sökmotorerna jämfört med varandra. Detta skulle kunna förorsaka en del svårigheter i urval av en lämplig sökmotor för e-handelsprojekt och i sin tur leder till en del av bristerna som uppstår i samband med sökfunktionen.

Syfte

Syftet med examensarbetet är att komma fram till ett lösningsförslag som dels underlättar urvalet av befintliga sökmotorer och dels

möjliggör en förbättring av sökfunktionen. Vilket i sin tur leder till en bättre kundupplevelse i uppdragsgivarens kommande projekt.

Mål

Målet med examensarbetet är att undersöka vilka sökmotorer som finns tillgängliga och att få värdefulla erfarenheter gällande vilka sökmotorer som är lämpliga att användas inom e-handel.

Avgränsningar

Detta examensarbete har styrts efter att uppfylla uppdragsgivarens önskemål och prioriteringar. I detta examensarbete behandlas inte webbsökmotorer eller användargränssnitt för sökmotorer.

Webbsökmotorer innefattar t.ex. Google och Yahoo vars syfte är att indexera, söka och hämta information från webbsidor på Internet.

Användargränssnitt syftar till det gränssnitt som hanterar kommunikationen mellan människa och dator. Dessutom ingår det inte att implementera en egen sökmotor.

Terminologi

Term (Eng. term): ett ord eller en fras

Dokument (Eng. Document): en samling av termer som beskriver ett ämne

Enterprise Search: att söka information som finns innanför brandväggen på ett företag eller en organisation

Fulltext sökning (Eng. Full Text) innebär att en sökmotor undersöker varje ord i varje dokument och matchar mot

Search):	sökfrågan
Matchning (Eng. Match):	att bedöma likheten mellan sökfråga och relevant dokument[1]
Rankning (Eng. Ranking):	att ranka d.v.s. sortera svar efter en specifik kriterie
Stemming (Eng. Stemming):	att ta fram stammen av en bestämd böjd ordform t.ex. för ordet ”mobiltelefoner” är stammen ”mobiltelefon” [1].
Stoppord (Eng. Stop word):	frekvent förekommande betydelselösa ord som skall filtreras bort som t.ex. ”och”, ”eller” och ”på” m.fl. [2]
Sökfråga (Eng. Query):	en fråga som ställs till t.ex. en sökmotor
Spindel (Eng. Web Crawler):	ett datorprogram som automatiskt hämtar webbsidor från Internet och hittar nya sidor genom att läsa hyperlänkar från bl.a. HTML-sidor[3].

E-handel

Elektronisk handel eller e-handel innefattar olika former av handel som sker med hjälp av datakommunikation, enligt

Nationalencyklopedin[4]. E-handel är ett omfattande begrepp och det finns många tjänster som ingår i elektronisk handel. Detta examensarbete fokuserar på den bakomliggande teknik som används i sökfunktionen i en nätbutik.

För att utveckla en sökfunktion kan man använda sig av sökmotorer. De sökmotorer som används inom e-handel riktar sig mot en specifik grupp av användare som letar efter en speciell produkt på Internet. Dessa sökmotorer används även för att jämföra olika produkter och deras priser[5]. I och med detta finns det speciella krav för sökmotorer för e-handel.

E-handel har utvecklats snabbt de senaste åren, bara i Sverige utgör e-handeln ca 4,7 % [6] av den totala detaljhandeln. Dessutom har antal konkurrenter och mängden information ökat och sökmotorer skulle kunna vara ett intressant ämne inom e-handel.

Informationsåtervinning

Ett av begreppen gällande sökmotorer är (Eng. Information retrieval, IR). På svenska används två termer för att benämna information retrieval, dels informationsåtkomst och dels informationsåtervinning.

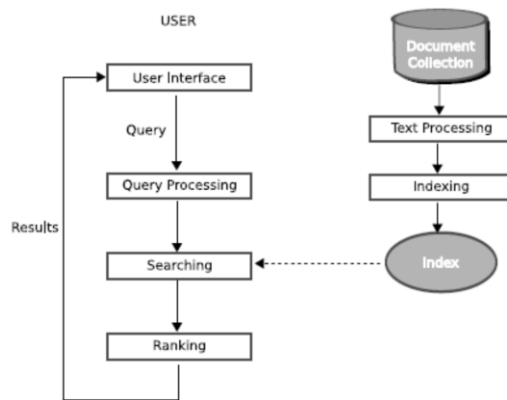
Nationalencyklopedin definierar informationsåtervinning som ”sökning av önskad information från lagrade data. För effektiv informationsåtervinning krävs utveckling av databaser, indexeringsmetoder och klassifikationssystem samt skapande av program för interaktiv informationssökning.” [7]

Informationsåtervinning handlar om att hitta ostrukturerat textmaterial, oftast i form av dokument, från en stor informationssamling, enligt användarens informationsbehov. [8]

Sökmotorer

Nationalencyklopedin definierar en sökmotor som ”ett program som söker lagrad information i strukturerade databaser eller textdatabaser. Sökmotorn tolkar en begäran om information och söker igenom en

eller flera databaser efter förekomster som motsvarar begäran. Ofta ingår sökmotorn som en del i ett större program” [9].



Figur 1: Indexering och sökning i en sökmotor.

- Indexering (Eng. Indexing): Genom att strukturera och representera information kan man på ett snabbt sätt komma åt informationen. Ett s.k. index skapas som kan jämföras med index längst bak i en bok där man genom att slå upp specifika sökord hittar rätt kapitel/del i en större mängd. [8]
- Sökning (Eng. Searching): Att hämta information som uppfyller användarens informationsbehov från indexet. [8]
- Rankning (Eng. Ranking): Att sortera resultatet så att den skulle kunna uppfylla användarens behov på ett bättre sätt. [8]

När det gäller sökmotorer för e-handel spindlas inte information från Internet utan där används interna informationskällor t.ex. databaser innehållande produktinformation istället (Se Spindel under terminologi). Produktinformationen kan vara kategorier som t.ex. pris

och id. Indexeringen kan sedan göras på utvalda kategorier. Användaren har ofta ett sökfält där sökfrågor kan skrivas in och sortering kan ofta göras efter specifika kategorier. Sökmotorn presenterar sedan en lista över relevanta svar som visas till användaren[10].

Genomförande

Arbetet påbörjades med att reda ut de krav som ställs på sökmotorer för e-handel. Dessa krav ligger till grund för de testinstruktioner som används för att utvärdera befintliga sökmotorer. Vidare utfördes en kartläggning dels för att bekanta oss med ämnesområdet och dels för att välja ut ett antal sökmotorer för utvärderingen.

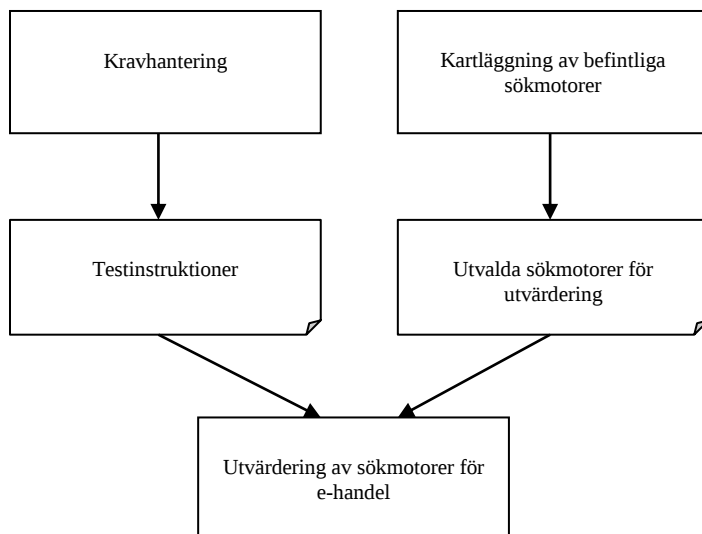


Diagram 1: Genomförande

Kravhantering

En av de centrala delarna av examensarbetet är att ta fram krav som ställs på en sökmotor anpassad för e-handel. Kraven ligger sedan till grund för att utvärdera befintliga sökmotorer. Insamlingsprocessen inleddes med flera möten med experter och kunnig personal inom området med erfarenhet från projektarbeten med bl.a. APPTUS och Google Mini. Under insamlingsprocessen hölls fokusgrupper i form av sprintredovisningar med aktiva inom näringslivet och med aktiva inom kravhantering från LTH för att diskutera vilka metoder som kunde tillämpas på examensarbetet.

Arbetet med att samla in kraven har varit en iterativ process där arbetet med kraven varit som en röd tråd under arbetets gång. Några av de svåraste kraven att definiera gällde användbarhet, tillförlitlighet, utvidgbarhet och prestanda. Svårigheten var att i kraven specificera mätvärden d.v.s. typiska referensvärden, som t.ex. en sökning får maximum ta ett antal millisekunder.

Resultatet av insamlingen låg sedan till grund för att skapa en kravspecifikation. Därefter skapades ett dokument med testinstruktionerna och en testspecifikation utifrån kravspecifikationen. Vidare har uppdragsgivaren prioriterat kraven. Resultatet var arton stycken krav med hög prioritet d.v.s. krav med prioritet 1 (Se appendix B). Testinstruktionerna baserades sedan på de högprioriterade kraven.

Testspecifikation

Resultatet av den utförda prioriteringen låg sedan till grund för att ta fram en testspecifikation och testinstruktioner. I testspecifikationen formulerades testfall och där specificerades vilka krav som testades för respektive testfall. Figur 2 visar ett exempel på hur ett testfall ser ut, där framgår dels funktionens- och kravspecifikationens namn och dels vilken versionsnummer det är. Testspecifikationen är baserad på mallen för en testspecifikation taget från kurskompendium 2 från kursen Programvaruutveckling för stora system Version 1.8 -090616. Samt vad det är som testas och vilket krav det gäller.

Kravspecifikationens namn och version:

Funktionens namn: Sökning på olika språk

Detta funktionstest består av följande testfall:

- 1) En sökfråga formulerad med ett tangentbord på ett annat språk
[41210]

Figur 2: Exempel på ett testfall

Dokumentet testinstruktion som återfinns i appendix E innehåller alla testfall som skapades och där framgår om de är godkända eller inte. Det innehåller 13 stycken testfall och testar följande funktionalitet hos sökmotorerna:

- Sökning och stemming på svenska.
- Rankning
- Externa källor
- Booleska operatorer
- Särskrivningar
- Manuell rankning
- Användarbeteende
- Statistik om sökfrågor
- Användarbeteende och användarhistorik
- Tydligt Licensavtal

Testfallen kontrolleras genom att studera sökmotorernas dokumentation i form av manualer, forum och API:n (Eng. Application Programming Interface). Om dokumentationen innehåller förklaringar och exempel gällande ett testfall godkänns det. En del av testfallen krävde en sökfråga för att kunna testas. Sökfrågan valdes efter att studera datamängden och komma fram till lämpliga sökfrågor som skulle kunna användas för testfallet.

För att kunna utföra testfall 2 som gäller rankning krävs det att man har tillgång till sökfrågor och tillhörande svar. Dock fanns det inga svar att tillgå till den datamängden som används. Därför skapades en metod som tar fram svar och värde på relevans för varje relevant svar tillhörande en specifik sökfråga. Testfallet testades sedan med hjälp av utvärderingsverktyget. För att hitta all information om mobiltelefoner och tillbehör där ordet silver förekommer valdes sökfrågan ”silver”. Kolumnen Global_Color valdes att prioritera så att produkter som har ordet silver i denna kolumn kommer att rangordnas högre.

Testfall 3 som gäller externa källor, t.ex. en källa som innehåller relevanta recensioner om mobiltelefoner, ska användas för att undersöka om en sökmotor kan rangordna information från externa källor lägre än produktinformation från en interndatabas. Efter att ha diskuterat och studerat hemsidor som t.ex. prisjakt.se där produkter och recensioner är sorterade var för sig beslutades att detta krav inte ska testas. Därför har detta krav utgått. Testspecifikationen och testinstruktionerna återfinns också i appendix E.

Kartläggning av befintliga sökmotorer

En kartläggning av befintliga sökmotorer utfördes dels för att bekanta oss med ämnesområdet och dels för att välja ut ett antal kandidater för utvärdering av sökmotorer för e-handel. Kartläggningen påbörjade med att undersöka tidigare artiklar och studier. Därefter letades det efter information som finns tillgänglig på Internet. Det visade sig att SearchTools.com innehåller information om ett stort antal sökmotorer och därför utvecklades ett verktyg för att läsa in information från SearchTools. Dessutom gjordes en enkät för att komplettera kartläggningen med den expertis som finns hos uppdragsgivaren.

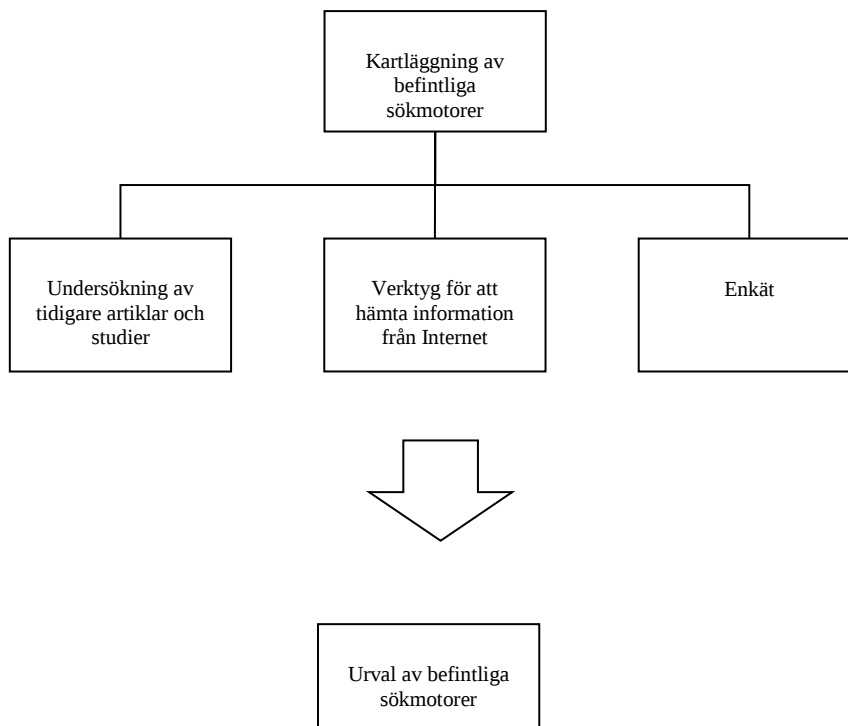


Diagram 2: Kartläggning av befintliga sökmotorer

Artiklar och tidigare studier

Undersökningen om artiklar och tidigare studier inriktar sig i första hand på tidigare studier om sökmotorer för e-handel och därefter studier om befintliga sökmotorer för generella syften. Med hjälp av ELIN (Electronic Library Information Navigator) och Google Scholar hittades följande artiklar. ”*The effectiveness of Web Search Engines for retrieving relevant ecommerce links*” av Bernard J Jason, m.fl. från år 2005 [11] presenterar en utvärdering av fem webbsökmotorer ur ett e-handelsperspektiv. I och med att artikeln behandlar endast webbsökmotorer, bl.a. Google, Exite och Yahoo! Directories, är dess slutsats inte intressant i detta examensarbete. Däremot finns det en del information om ett testverktyg och tillvägagångssättet för att

utvärdera sökmotorer för e-handel med hjälp av verktyget som används i detta arbete (Se utvärderingsverktyg).

Artikeln ”*Performance Evaluation of Desktop Search Engines*” av Chang-Tien Lu, m.fl. [12] presenterar utvärdering av sökmotorer som möjliggör sökning bland och inuti datorns lagrade filer. Det resultat som presenteras i artikeln är inte användbart i utvärdering av sökmotorer för e-handel men det finns en del värdefull information om tillvägagångssättet och metodologin för att utvärdera sökmotorerna. I artikeln presenteras en informationssamling och ett testverktyg som har använts för att utvärdera sökmotorerna. Den informationen ligger till grund för utvärderingen (Se utvärderingsverktyg och informationssamlingar).

Utvecklingen inom sökmotorbranschen har varit väldigt snabb de senaste åren och sökmotorleverantörer har tillverkat nya produkter med korta mellanrum. Dessutom behandlas inte de sökmotorer som är av intresse hos uppdragsgivaren, t.ex. Apptus, Google Mini, Microsoft FAST Search, i de artiklar som hittats. I och med detta gjordes det en egen undersökning för att göra en lista över befintliga sökmotorer.

Verktyg för insamling av information

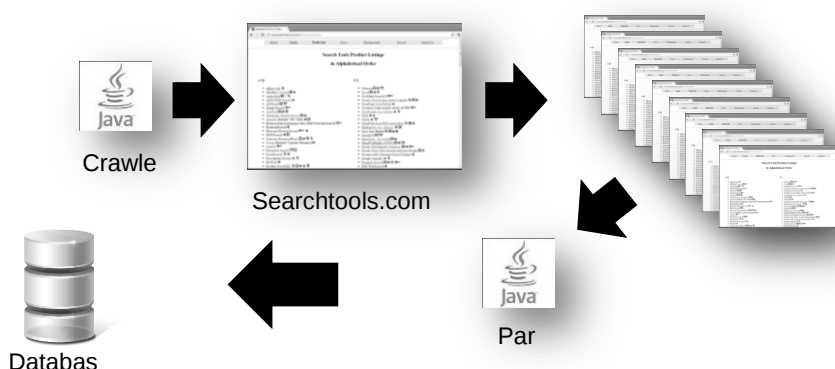
I och med att resultaten från artiklar och tidigare studier som är tillgängliga via ELIN och Google Scholar inte är relevanta för detta examensarbete, valdes andra informationskällor som finns tillgängliga på Internet. Det finns ett fåtal hemsidor som specialiserar sig på sökmotorer, t.ex. searchengineshowdown.com, searchenginewatch.com och SearchTools.com. Det finns även en internetgrupp på Yahoo! som publicerar nyheter om sökmotorer [13].

SearchTools.com innehåller värdefull information om sökverktyg och sökmotorer. Informationen innefattar artiklar, egenskaper, priser, hemsidor där sökverktyget har använts. Det finns dock en stor nackdel med SearchTools, nämligen har den största delen av sidorna

byggts under perioden 2001 till 2007 och SearchTools saknar information om de nya sökverktyg som har utvecklats efter 2007.

Informationen från SearchTools har använts som ett underlag för kartläggning av befintliga sökmotorer. Informationen ligger till grund för en informationsdatabas om alla befintliga sökmotorer. Dessutom har databasen uppdaterats under examensarbetet och kan även användas i uppdragsgivarens framtida projekt.

Det finns information om ett hundratals sökverktyg på SearchTools och därför har det utvecklats ett verktyg för att automatisera insamling av informationen. Detta görs dels för att minska antal arbetstimmar och dels för att minska risken för mänskliga misstag. Figur 3 visar hur verktyget fungerar.



Figur 3: Verktyg för att läsa in information från SearchTools

Web Crawler i Java

En Web Crawler i Java som följer hyperlänkarna på en HTML-sida för att komma åt de länkade sidorna på Internet utvecklades under arbetets gång. Web Crawlern läser hyperlänkar till sidor som handlar om sökverktygen på SearchTools och hämtar dessa sidor.

HTML-Parser

Web Crawlern hämtar HTML-sidorna och skickar dem till en HTML-parser som tar bort HTML-taggar och tolkar dessa sidor till klartext. HTML-parsern har också utvecklats i Java.

Databas

Den information som parseern har samlat in lagras i en databas på en SQL-Server hos uppdragsgivaren. Detta underlättar bl.a. sökning och uppdatering av informationen i samband med utvärdering av sökmotorer för e-handel.

Databasdesign

För att underlätta uppdatering av informationen i samband med utvärderingen av sökmotorer för e-handel har databasdesignen används enligt figur 4. Informationen sparats i flera databastabeller med primära och främmande nycklar som skapar relationer mellan tabellerna. Figur 4 visar tabeller som har använts för att lagra informationen samt relationer mellan tabellerna. Följande tabeller används för att lagra information från SearchTools.

- SearchTools: innehåller information om de befintliga sökverktygen.
- Features: innehåller egenskaper för sökmotorerna.
- Articles: innehåller information om artiklar som har publicerats om sökverktygen.
- Examples: innehåller information om projekt som har använt sig av sökverktygen.

Det finns även en del tabeller som kan användas för att lagra resultat av utvärderingen:

- Criteria: innehåller kriterier som används för att utvärdera sökmotorer

- Evaluations: innehåller betyg och kommentarer för varje kriterium och sökmotor.



Figur 4: Databasdesign

Möjligheter för vidareutveckling

SQL-databasen kan ligga till grund för eventuell vidareutveckling i framtiden. Ett alternativ till vidareutveckling är att skapa en webbapplikation som presenterar information om sökverktygen och utvärderingar från databasen och erbjuder möjligheter för uppdatering av informationen. Dessutom är det möjligt att lagra information om nya sökmotorer och nya utvärderingar i samma databas utan att ändra strukturen.

Enkät

För att kunna ta del av erfarenheter och åsikter från Avensias projekt gällande sökmotorer och att använda dem i urval av sökmotorer skickades ett frågeformulär i form av en enkät ut till samtliga mjukvaruutvecklare.

Svaren innehöll en del intressant information om brister hos och problem med de sökmotorer som använts i de olika projekten. Några brister var bl.a. att dokumentationen för Lucene är för komplicerad och att Google Mini endast kan sortera på datum. Enkäten och de svar som samlades in återfinns appendix F.

Sammanfattningsvis kan nämnas att av resultatet framgår att de som besvarat enkäten känner till Apptus, Avail, Factfinder, Fast, Lucene, Theca, SQL Fulltext Search (SQL FTS) samt i Avensias projekt har Apptus Esales, Avail, FactFinder, Lucene, Google Mini och SQL FTS använts. De sökmotorerna har därför tagits med i det första urvalet.

Urval av befintliga sökmotorer

Efter avslutad insamling av information om befintliga sökmotorer innehöll informationsdatabasen ungefär 230 sökmotorer och en stor del av informationen hade inte blivit uppdaterad sedan 2007. Eftersom examensarbetets tidsramar gjorde det omöjligt att undersöka samtliga av dessa sökmotorer delades sökmotorerna upp i två huvudkategorier, sökmotorer med öppen källkod och kommersiella sökmotorer. Syftet med uppdelningen var att kategorisera sökmotorerna och sedan ta fram de mest populära sökmotorerna inom varje kategori genom att läsa artiklar och tidigare studier om dessa sökmotorer. Därefter togs ett antal kriterier fram för att jämföra dessa sökmotorer i första urvalet.

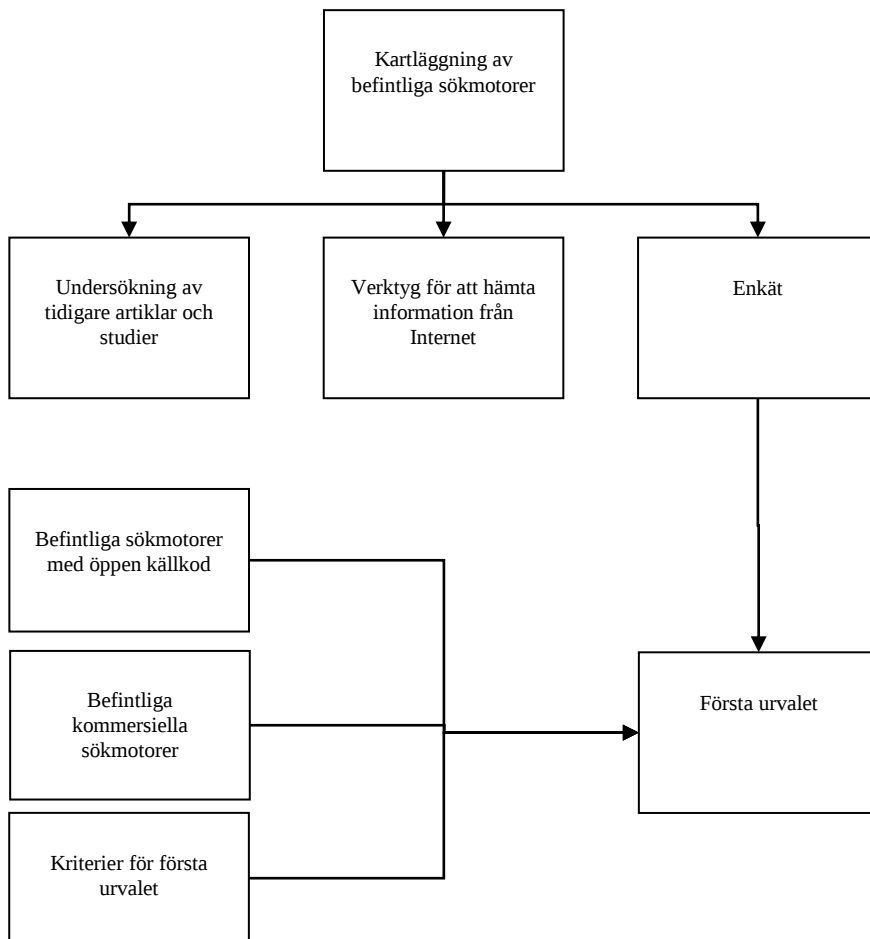


Diagram 3: Urval av befintliga sökmotorer

Sökmotorer med öppen källkod

För att söka efter vetenskapliga artiklar som utvärderar sökmotorer med öppen källkod för e-handel användes dels Google Scholar och dels databaserna COMPENDEX och IEEE. Dock hittades bara två intressanta artiklar.

Den första artikeln ”*Open Source Libraries for Information Retrieval* Vesna Hassler” [14] presenterar fem populära sökmotorer med öppen källkod Xapian, Apache Lucene, Ht://Dig, Swish-e, DataparkSearch enligt tabell 1. Dock framgår inte hur urvalet är gjort och varför de sökmotorerna valdes ut.

Xapian	Apache Lucene	Ht://Dig	Swish-e	DataparkSearch
24 augusti 2010	18 juni 2010	16 juni 2004	4 april 2009	24 jan 2009
v. 1.2.3	v.2.9.3 för Lucene Java 2.x serie baserad på Java 1.4. och v.3.0.2 för Lucene Java 3.x serie baserad på Java 5	v.3.2.0b6	v.2.4.7	v.4.54

Tabell 1: De sökmotorer som presenteras i första artikeln

Resultatet av den första artikeln kan sammanfattas med att alla 5 sökmotorerna stödjer sökfrågor med booleska operander, fraser, parenteser och stemming. Alla stödjer dokument format i form av text- och PDF-filer. Xapian och Lucene kommer dessutom med demo applikationer. Därför tas alla med i det första urvalet utom Ht://Dig bortses då den senaste versionen är uppdaterad år 2004.

Den andra intressanta artikeln är ”*A Comparison of Open Source Search Engines*” av Chrisitian Middelton, m.fl. [15] som testar 30 stycken sökmotorer med öppen källkod. Testen utfördes på två olika informationssamlingar från TREC. Dels på informationsamlingen TREC-4 (Se informationssamlingar) från den fjärde i ordningen av konferenser anordnade av TREC (Fourth Text REtrieval Conference[16]) som separerades i 3 olika storlekar, en med 750 MB, en med 1,6GB och en med 2,7GB. Samt på WT10g TREC Web

corpus (WebTREC) på 10.2 GB som också separerades i följande storlekar (2,4; 4,8; 7,2 och 10,2 GB).

Flera av sökmotorerna i artikeln valdes bort p.g.a. att deras senaste uppdatering var innan år 2000. Sökmotorn Nutch bortsågs på grund av att den bygger på samma kärna som Apaches sökmotor, Lucene och XML Query Engine och Zebra på grund av att de fokuserar på XML-strukturerad data. Efter ett initialt test där den minsta informationssamlingen (750 MB) indexerades, valdes även DataparkSearch, mnoGoSearch, Namazu, OpenFTS och WebGlimpse bort för att de inte klarade det.

Därför blev resultatet av den andra artikeln följande: Indri, IXE Toolkit, Lucene, MG4J, OmniFind IBM Yahoo! Edition, Omega, Swish-e, Swish++, Terrier och Zettair.

Några andra sökmotorer som också tas med i första urvalet är Apache Solr, Sphinx och OpenGrok. Anledningen var att de nämndes som tips på frågor gällande sökmotorer med öppen källkod på programmeringsforumet stackoverflow.com[17]. Dessutom valdes Apache Solr för att den har fått pris av webbtidningen InfoWorld för bästa applikation med öppen källkod år 2010[18].

Resultatet av urvalet för sökmotorer med öppen källkod blev därför en kombination av båda artiklarna och de som nämndes ovan vilket ger tabell 2.

Apache Lucene	Apache Lucene.NET	Apache Solr	Indri
IXE Toolkit	MG4J	Omega	OmniFind IBM Yahoo! Edition
OpenGrok	Sphinx	Swish-e	Swish++
Terrier	Xapian	Zettair	

Tabell 2 Resultatet av urvalet för sökmotorerna med öppen källkod

Kommersiella sökmotorer

För att hitta marknadsledande produkter bland kommersiella sökmotorer har artiklar, nyheter och konferenser studerats gällande kommersiella sökmotorer från de senaste åren. Resultatet av studien ligger sedan till grund för urvalet av befintliga sökmotorer.

År 2004 stod Verity, Autonomy, Fast Search & Transfer, Google och Microsoft för hälften av försäljningen av sökmotorlösningar enligt Gary Eastwood [19]. Då företaget Autonomy var känd för sin Idol (Intelligent Data Operating Layer) som kunde hantera fler än 300 dataformat, Google hade levererat tre välkända produkter, Search Appliance, Google Mini, Desktop Search och Microsoft hade levererat MSN Search och MSN desktop samt sökfunktioner för SharePoint.

New Idea Engineering, Inc. publicerade två artiklar som behandlar marknadssituationen under år 2008 [20] respektive år 2009 [21]. Författarna använder sig av två termer för att presentera marknadssituationen, Tier 1 och Tier 2. Tier 1 syftar på de leverantörer som hade stora andelar av marknaden och Tier 2 som syftar på mindre kända leverantörer samt de leverantörer som riktade sig mot ett begränsat geografiskt område.

Den dåvarande norska företaget Fast Search & Transfer fanns med i gruppen Tier 1. Andra leverantör i Tier 1 var Google (som leverede

sin version 5 Search Appliance), Endeca (med sina avancerade administrationsfunktioner) och Autonomy Idol. [20]

I början av 2008 lade Microsoft ett bud på Fast Search And Transfer och på så vis hamnade Microsoft i Tier1-gruppen. Dessutom blev företaget Stellant som levererade CMS – produkter (Content Management System) uppköpt av Oracle, en av de stora aktörerna, och kunde därför hamna i gruppen Tier 1. IBM OminiFind Search Discovery Edition nominerades till årets bästa överraskning år 2006. Verity köpte upp Ultraseek som i sin tur blev uppköpt av Autonomy. Vidare nämns Inxight som köpte in Enterprise Search Assests av Intelliseek i artikeln[20]. Dessutom nämns andra framgångsrika leverantörer som ZyLAB, Vivisimo, Diselpoint, Intellisearch Exalead och Sideran.

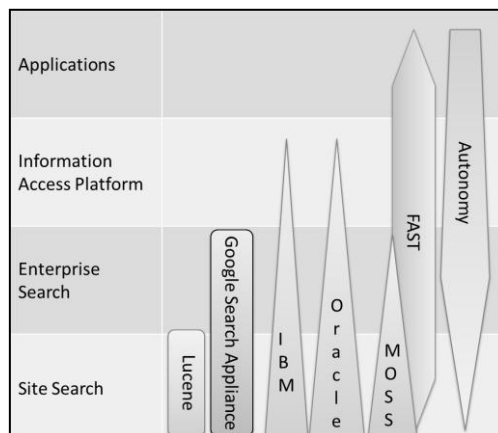
Den andra artikeln från New Idea Engineering, presenterar Autonomy, Endeca, Google, Microsoft/FAST och den franska leverantören, Exalead som Tier 1-leverantörer [21]. Attetio, Diselpoint, IBM, ISYS, Recommind, Thunderstone, Vivisimo och ZyLabs presenteras som Tier 2-leverantörer [21].

I artikeln presenteras även Endeca, FAST, Google, Diselpoint och SLI Systems som lämpade sig för e-handel och Autonomy, FAST, Oracle och IBM som lämpade sig för företagssök (Eng. Enterprise Search) och Intranet. Dessutom presenteras IBM, Vivisimo, Exalead, FAST, Lucene/Solr som lösningar för portaler [21]. Slutligen presenterar Kehoe sin slutsats i tabell 3 nedan [21]:

Compliance/eDiscovery Autonomy Recommind	eCommerce Endeca FAST Google Dieselpoint SLI Systems	Vertical Search/Public Portals IBM Vivisimo Exalead FAST Lucene/Solr
Government/Law Enforcement ISYS Autonomy FAST Vivisimo	Enterprise/Intranet Autonomy FAST Oracle IBM	Customer Support InQuira Endeca FAST Dieselpoint MyRoar
CMS Search Microsoft Search Server Interwoven (Autonomy) Lucene/Solr	OEM/Bundled Exalead Vivisimo Lucene Attivio	Hosted Google SLI
NIE Industry Grid for 2009		

Tabell 3: Klassificering av sökmotorleverantörer enligt Miles Kehoe[21].

Information Retrieval Specialist Group (IRSG) anordnade konferensen Innovations in Web & Enterprise Search år 2010 i London där Chirag Gandhi presenterade befintliga sökmotorer för olika användningsområden enligt figur 5.



Figur 5: Klassificering av sökmotorleverantörer enligt Chirag Gandhi [22].

År 2010 publicerades en artikel på IDG.se som presenterar den svenska sökmotorn Siteseeker. Enligt artikeln bytte Telenor sin lösning byggd på Google Mini mot Siteseeker. Dessutom skulle Apoteket byta sin SharePoint sökmotor på Apoteket.se mot SiteSeeker [22].

2004	Marknadsledande	Övriga				
	Autonomy Fast Search & Transfer Google Microsoft Verity	Endeca Excalibus Inxight Open Text Livelink Discovery Server				
2008	Tier 1	Företag som lockade stora kunder till sig				
	Autonomy Idol Endeca Fast Search & Transfer Google IBM OminiFind Search DE Lucene/Nutch/Solr		Diselpoint Intellisearch Exalead Inxight Microsoft Oracle Sideran Vivisimo ZyLAB			
2009	Tier 1	Tier 2	Stora aktörer	Öppen ällkod	Kostnadsfri	Lågkostnad
	Autonomy Endeca Exalead Google Microsoft/FAST	Attetio Diselpoint IBM ISYS Recommind Thunderstone Vivisimo ZyLabs	dtSearch Siderean Systems X1	Lucene Solr	IBM Omni YAHOO! MS-Express Search Server	Google Site Search
2010						
	IBM Oracle Lucene MOSS	Eurologs Siteseeker				

	Google FAST Autonomy					
--	----------------------------	--	--	--	--	--

Tabell 4: En sammanfattning av artiklarna [19],[20],[21][22]

Tabell 4 visar en sammanfattning av de artiklar och rapporter som har studerats. Dessutom är produkter som Microsoft SQL Full Text Search, Apptus eSales, Google Mini, Google Search Appliance av intresse för uppdragsgivaren. Slutligen användes informationsdatabasen för att ta fram namn på sökverktyg som enligt SearchTools riktar sig mot e-handel. Omniture SiteSearch som blev uppköpt av Adobe, eComm|Search och EasyAsk eCommerce Edition finns i den kategorin.

Efter att ha analyserat informationen ovan har produkter från Adobe, Apptus, Autonomy, Diselpoint, Euroling, Google, IBM, Microsoft, Nextopia, Oracle, Siderean och Thunderstone nominerats för första urvalet. Tabell 5 visar de utvalda produkterna.

Leverantör	Produkt	kommentarer
Adobe	Adobe SiteSearch	(utvecklad för e-handel)
Apptus	eSales	(utvecklad för e-handel)
Autonomy	Autonomy IDOL Server	
Diselpoint	Search Info	(utvecklad för e-handel)
Euroling	SiteSeeker	
Google	Site Search Mini Search Appliance	
IBM	OmniFind Enterprise Edition	
Microsoft	Enterprise Search SQL Full Text Search Fast Search Server Search Server 2010 Express Search Server 2010 Enterprise	(utvecklad för e-handel)
Nextopia	eComm Search™	(utvecklad för e-handel)
Oracle	Secure Enterprise Search	
Siderean	Seamark Faceted Metadata Search	(utvecklad för e-handel)
Thunderstone	Thunderstone Search Appliance	

Tabell 5: Utvalda produkter för första urvalet

Första urvalet

Eftersom examensarbetets tidsramar inte gjorde det möjligt att undersöka samtliga marknadsledande och populära sökmotorer valdes att jämföra sökmotorer utifrån ett antal kriterier (Se kriterier för första urvalet) och välja ut de som uppfyller flest kriterier. Dessutom valdes de sökmotorer som inte tillhandahåller tillräckligt med information om sökmotorn på Internet bort.

Därför användes ett Excel-dokument som innehåller ett antal kriterier för att jämföra de utvalda sökmotorerna. Sökmotorleverantörer kontaktades via e-post och telefon och informationen samlades in för att uppdatera Excel-dokumentet.

Antal genererade träffar från Altavista

Det finns en del sökmotorer som använder sökstatistik och antalet träffar för att bedöma populariteten hos en produkt. Google Trends kan exempelvis visa populära sökord utifrån förhållandet mellan antalet sökfrågor om ett specifikt sökord och det totala antalet sökfrågor som skickas till Google. För att få en indikation på populariteten för sökmotorerna togs det fram antalet träffar från webbsidor på Internet.

Ett Java-program som använder en liknande metod som Google Trends utvecklades. Programmet hämtar namn på sökmotorer från informationsdatabasen, skickar en sökfråga till Altavista, som är en webbsökmotor på Internet, och sparar antalet träffar i informationsdatabasen. Sökfrågan består av en boolesk kombination av namn på sökmotorn och strängen "Search Engine". Det visade sig att resultatet av programmet inte är pålitlig i och med att ett antal sökmotorer som inte vidareutvecklats de senaste åren hamnade högst upp i listan och därför valdes det att inte använda resultatet för att dra någon slutsats. (Se appendix J)

Det finns dock möjligheter för vidareutveckling av programmet. Man kan nämligen undersöka möjligheter för att läsa in information från Google Trends för att bedöma populariteten hos sökmotorerna.

Kriterier för första urvalet

För att jämföra de befintliga sökmotorerna har 45 kriterier tagits fram. En del av dessa kriterier [15] och [23] grundar sig i två artiklar. Förutom artiklarna användes informationsdatabasen för att formulera kriterierna. Tabellen ”Features” i informationsdatabasen innehåller funktioner och egenskaper hos sökmotorerna (Se Verktyg för insamling av information). De mest förekommande egenskaperna i tabellen är viktiga och därför är en del av kriterierna framtagna utifrån dessa.

Tabell 6 Kriterier för första urvalet

Kriterium	Beskrivning
Language	De språk som stöds av sökmotorn.
Did you mean	Rättstavningsfunktion.
Boolean Search	Sökmotorn kan stödja boolesk sökning.
Faceted Search	Sökmotorn kan stödja boolesk sökning i kombination med parenteser.
Relevance Ranking	Rankning av sökresultat.
Platform compatibility	Operativsystem som sökmotorn stödjer och kan installeras på.
Data format compatibility	Fil- och dataformat som sökmotorn kan indexera på.
Database indexing	Databassystem som sökmotorn kan indexera på.
Categorization	Sökmotorn kan kategorisera informationen i olika kategorier.
Exact Match	Sökmotorn kan hitta information som är exakt lik

	sökfrågan.
Fuzzy Matching	Sökmotorn söker på ordstam och olika representationer av ett sökord. Sökmotorn tar även bort stoppord från sökfrasen.
Index persistence	Sökmotorn sparar index-filen på hårddisk, i minnet, o.s.v.
Multi-Site Search	Sökmotorn kan indexera flera källor, t.ex. webbsidor och dokument.
More like this	Sökmotorn söker på innehållet i ett dokument och hittar dokument som liknar det.
Soundex	Sökmotorn använder en fonetisk algoritm för att hitta liknande ord, t.ex. 'Smith' och 'Smythe' har samma Soundex-värde.
Wildcards	Sökmotorn kan söka på sökfras med asterisk (*).
Metaphone	Sökmotorn använder en fonetisk algoritm för att hitta liknande ord, t.ex. 'Erik' och 'Eric' har samma Metaphone-värde.
Acronym	Sökmotorn kan söka på förkortningar av ett ord.
Synonyms	Sökmotorn kan söka på synonymer.
Stemming	Sökmotorn kan söka på ordstam.
Natural Language Processing (NLP)	Sökmotorn kan söka på sökfrågor i naturligt språk.
Ontology	Sökmotorn kan söka på ord med hjälp av en ontologisk algoritm.
Partial Word Matching	Sökmotorn stödjer partiell matchning och kan hitta fraser vars delar liknar sökfrågan.
Phrase Searching	Sökmotorn kan söka på en fras inom citationstecken.

Proximity Searching	Sökmotorn kan hitta två sökord som befinner sig på ett visst avstånd från varandra.
Results List	Sökmotorn skickar sökresultatet i någon form av datastruktur.
Score	Sökmotorn viktar sökresultaten.
Sorting Results	Sökmotorn kan sortera sökresultaten efter olika kategorier.
Thesaurus	Sökmotorn begränsar sökresultatet utifrån kontexten, t.ex. sökmotorn viktar sökresultat för ett speciellt land när man söker på en stad.
Vertical Portals	Sökmotorn kan indexera på information från nischade portaler, t.ex. News Feeds och Chat.
Search Zones	Sökmotorn erbjuder möjlighet att definiera olika zoner för sökresultaten.
Last release	Senaste versionen av sökmotorn.
Fees	Kostnader.
Click tracking	Sökmotorn kan lagra användarbeteende.
Link tracking	Sökmotorn erbjuder referenssökning.
Configuration	Sökmotorn är konfigurerbar.
Auto complete	Sökmotorn kan visa förslag på sökord medan användaren skriver.
API	Programmeringsgränssnitt.
Programming Language	Det programmeringsspråk i vilket sökmotorn har utvecklats.
Web crawler	Teknologin för att hämta information från informationskällor.

Simultaneous update and searching	Online-uppdatering
Available Documentation	Dokumentation
Technical Support packages	Teknisk support
Number of documents	Antalet dokument som sökmotorn kan indexera på.

Resultat

En risk för felkällor med första urvalet är att information som uppfyller ett specifikt kriterium inte hittats betyder inte det att sökmotorn i fråga inte stödjer den funktionen, utan då kan det bero på att den inte hittats.

Sökmotorer med öppen källkod

I den framtagna listan som är resultatet från det första urvalet står kriterierna till vänster och varje sökmotor har en egen kolumn som innehåller den information hittats (Se tabell 7). Tomma celler anger att information som uppfyller kriterierna inte har hittats.

Av sammanställningen framgår det att Swish++, Open-Grook och Zettair är de sökmotorer som hittats minst information om. Omega och Apache.Net bortses från eftersom det bara hittade begränsad information som uppfyllde kriterierna.

Tabell 7 visar resultatet för samtliga sökmotorer med öppen källkod och där framgår vilka sökmotorer som hittades överlägset mest information om. Utifrån tabellen valdes Apache Lucene, Apache Solr, Xapian och Sphinx.

[illegible]

Kommersiella sökmotorer

Trots att samtliga av de utvalda leverantörerna har kontaktats för att samla in information om deras produkter saknas det information om vissa produkter, t.ex. SiteSeeker har inte levererat någon information om sin sökprodukt. Trots att produkten har varit populär på den svenska marknaden har den inte valts ut den i första urvalet.

För att beräkna antalet uppfyllda kriterier för varje sökmotor (Se bilaga G) användes funktionen countIf() i Microsoft Excel och resultatet visar att Autonomy och MS Fast med 26/45 respektive 24/45 uppfyller flest kriterier bland de kommersiella sökmotorerna. Tabell 8 visar resultatet för samtliga sökmotorer. Utifrån tabellen valdes Oracle Secure Enterprise Search, MS Fast, Dieelpoint och MS Search Server 2010 express.

Criteria	Oracle Secure Enterprise Search	MS FAST	Dissepoint	MS_Search_Server_2010_Express	MS_Search_Server_2010	Google Search Appliance	Autonomy IDOL Server	eComm Search	Apptus	MS FULL TEXT SEARCH	Adobe SiteSearch	Thunderstone Webinator	Google Site Search	Google Mini	EasyAsk eCommerce Edition	Siderean Seamark
Language	All	40+	40	51	22	27	100+	.	All	Multi	Multi	Multi	Multi	28	.	.
Did you mean	.	☒	☒	☒	☒	☒	.	☒	☒	.	☒	.	.	☒	.	☒
Boolean Search	☒	☒	☒	☒	☒	☒	☒	.	.	.	.	.	.	☒	.	.
Faceted Search	.	☒	☒	☒	☒	☒	.	.	.	.	.	.	.	☒	.	.
Relevance Ranking	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	.	☒	☒	☒
Platform compability	All	.	Cross	Win	Win	.	Win	.	.	Win	.	.	.	.	.	EPISer
Data format compatibility	150+	400	6+	31	.	220	1000	.	.	15+	15+	100+	6+	All	☒	.
Database indexing	☒	☒	☒	.	☒	☒	☒	.	.	☒	☒	.	.	☒	.	.
Categorization	☒	☒	☒	.	manag	.	☒	☒	.	☒	☒	.	☒	.	.	☒
Exact Match	.	☒	☒	☒	.	.	.	.	.	☒	.	.	.	☒	.	.
Fuzzy Matching	☒	☒	☒	.	.	.	☒	☒	.	☒	.	☒	.	No	.	.
Index persistence	Disk	.	.	SQL	SQL	.	.	.	.	SQL	.	.	.	☒	.	.
Multi-Site Search	☒	☒	☒	☒	☒	☒	.	.	☒	.	☒	☒	☒	☒	.	.
More like this	☒	.	.	☒	☒	.	.	.	.	.	☒	.	.	.	.	.
Soundex	.	☒	.	☒	.	.	.	☒	.	.	☒	.	.	.	.	.
Wildcards	☒	☒	☒	☒	☒	.	☒	.	.	.	.	.	☒	.	.	.
Metaphone	.	.	.	.	.	.	.	☒	.	.	.	.	.	.	.	.
Acronym	.	☒	.	☒	☒	☒	.	.	.	.	☒	.	.	.	.	.
Synonyms	☒	☒	☒	.	.	☒	☒	☒	☒	☒	☒	.	☒	☒	.	.
Stemming	☒	☒	☒	.	.	☒	☒	☒	☒	☒	☒	.	☒	.	.	☒
Natural Language Processing (NLP)	☒	.	.	.	.	.	☒	.	.	.	.	☒	.	.	☒	.
Ontology	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
Partial Word Matching	.	.	.	.	.	.	.	☒	.	.	.	.	.	.	.	.
Phrase Searching	☒	☒	☒	☒	☒	☒	☒	.	.	☒	.	.	.	☒	.	.
Proximity Searching	☒	☒	☒	☒	☒	☒	☒	.	.	☒	☒	☒	.	.	.	.
Results List	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	☒
Score	☒	☒	.	☒	☒	☒	.	.	.	☒	☒	☒	☒	☒	.	.
Sorting Results	☒	.	☒	☒	.	☒	.	☒	☒	☒	☒	☒	☒	☒	.	☒
Thesaurus	☒	.	☒	.	.	.	☒	.	.	☒	.	.	.	.	.	.
Vertical Portals	☒	.	.	.	.	☒	.	.	.	.	.	.	.	.	.	.
Search Zones	☒	☒	.	.	.	.	.	.	.	.	.	☒	.	.	.	.
Last release	R. 11.1	.	.	.	.	V. 6	.	.	.	R. 2	V. 6.0.	V. 5.1	.	.	.	.
Click tracking	☒	☒	☒	.	.	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒
Link tracking	☒	☒	☒	.	.	☒	.	.	.	.	.	☒	.	.	.	.
Configuration	☒	☒	☒	☒	☒	☒	☒	.	☒	☒	☒	☒	☒	☒	☒	☒
Autocomplete	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	.
API	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	☒
Programming Language	.	.	Java	.	.	.	.	.	Java	.	.	.	.	.	.	.
Web crawler	☒	☒	☒	☒	☒	.	.	.	.	☒	.	.	.	.	.	.
Simultaneous update and searching	☒	.	☒	☒	☒	.	☒	.	☒	.	☒	☒	☒	☒	.	.
Available Documentation	Online	.	FAQ	TechN	TechN	Online	.	.	.	☒	.	.	Online	Online	.	.
Technical Support packages	☒	.	.	☒	.	.	.	.	.	.	☒	☒	.	.	.	.
Number of documents	1M	.	.	10M	10M	10M	.	.	.	Unlimi	200k	500k	300k	.	.	.
Fees	.	245,668 kr per processor	.	Free	.	.	.	.	.	.	10k docs Free, 20k docs \$700	10k docs 3.500€, 20k docs 6000€, 50k \$250	100k docs 3.500€	.	.	.
Points	26	24	17	16	14	14	12	12	11	11	10	9	9	9	5	6

Tabell 8: Sammanställning av kommersiella sökmotorer

Utvärdering

Vår undersökning om tidigare studier gällande utvärdering av sökmotorer visar att det finns färdiga verktyg och informationssamlingar som kan användas för att utvärdera bl.a. sökmotorer (Se Artiklar och tidigare studier). Detta ligger till grund för två undersökningar dels om befintliga utvärderingsverktyg och dels om befintliga informationssamlingar. Dessa undersökningar i sin tur ligger till grund för utveckling av ett testramverk för utvärdering av sökmotorer för e-handel.

Det finns två olika intressanta aspekter att beakta när det gäller utvärdering av IR-system, nämligen effektivitet (Eng. Effectiveness) och verkningsgrad (Eng. Efficiency). När det gäller verkningsgraden kan man t.ex. mäta hur många millisekunder och hur mycket utrymme i minnet som behövs för att genomföra en sökning. Man kan även mäta hur många bytes utrymme per dokument som behövs för att skapa ett index. En viktig egenskap som diskuteras i samband med verkningsgraden är svarstiden. Svarstiden är den tid som en användare måste vänta för att få svar på en sökfråga. Som nämnts innan, beror svarstiden på verkningsgraden hos ett IR-system men den kan även påverkas av antal användare som samtidigt skickar sökfrågor till IR-systemet. Begreppet genomströmning (Eng. Throughput) används för att uttrycka antalet sökfrågor ett IR-system ska kunna besvara per sekund.

Det är svårare att mäta effektiviteten hos ett IR-system än att mäta verkningsgraden. Anledningen är att effektivitet grundar sig på mänskliga bedömningar. Ett dokument kan anses vara relevant till en sökfråga när det innehåller den information som efterfrågas på sökfrågan. Relevansen kan uttryckas med ett binärt värde, t.ex. relevant eller irrelevant. Den kan även uttryckas med en tre- eller femgradig skala.

Verkningsgrad är en viktig aspekt vid utvärdering av IR-system men det kräver stora arbetsinsatser. Detta examensarbete behandlar en del av kraven som handlar om effektiviteten hos sökmotorer för e-handel.

Dessa krav är prioriterade av uppdragsgivare (Se bilaga C). De krav som handlar om verkningsgraden är inte inom ramen av detta examensarbete.

Utvärderingsverktyg

Det finns en del artiklar som visar att man kan använda sig av befintliga verktyg för att utvärdera effektiviteten hos sökmotorer. Artikeln "*Performance Evaluation of Desktop Search Engines*" [12] presenterar resultat från ett utvärderingsverktyg från TREC (Text REtrieval Conference). Dessutom presenterar artikeln "*A comparison of Open Source Search Engines*" [15] värdena som TREC_EVAL, ett standard testverktyg från NIST (Se TREC_EVAL), har genererat [24].

Med hjälp av dessa artiklar har ett antal verktyg som kan användas för att utvärdera sökmotorer hittats, t.ex. TREC_EVAL och IReval. Båda verktygen är kostnadsfria och källkoden är tillgänglig på Internet. Dock finns inte någon väldokumenterad användarmanual tillgänglig som visar hur man går tillväga för att använda verktyget för att analysera sökresultatet. I och med detta har samtliga författare kontaktats från flera artiklar där man har använt TREC_EVAL för att utvärdera sökmotorer och fått informationen på detta vis.

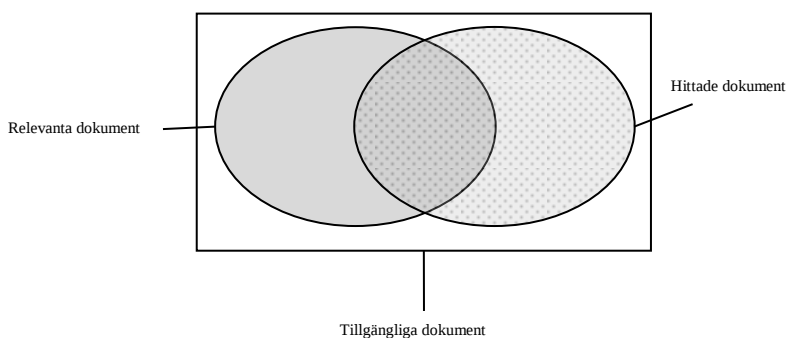
IReval

IReval är ett utvärderingsverktyg och beräknar bl.a. Precision och Recall i olika nivåer samt MAP (*Mean Average Precision*) för ett informationsåtervinningssystem (Se TREC_EVAL). IReval kan även beräkna värdena för statistiska tester som är utanför ramarna för detta examensarbete. Källkoden för IReval är tillgänglig och verktyget är utvecklat i Java vilket gör det möjligt att använda verktyget i Windows-miljö och anpassa till ramverket som har utvecklats i Java.[25]

TREC_EVAL

TREC_EVAL är ett standardverktyg som används av TREC för att utvärdera informationsåtervinningssystem. Man kan använda TREC_EVAL för att jämföra sökresultatet med ett facit och analysera effektivitet och informationsåtervinning hos en sökmotor. [26]

TREC_EVAL presenterar effektiviteten hos ett informationsåtervinningssystem, i termer av Precision och Recall. Precision är en kvot mellan antalet relevanta dokument som finns i en informationssamling och antalet dokument som ett informationsåtervinningssystem har hittat. Recall är en kvot mellan antalet dokument en sökmotor har hittat och antalet dokument som finns tillgänglig för sökmotorn.



Figur 5 Venndiagram om hur man beräknar Recall och Precision

TREC_EVAL presenterar värdena Precision och Recall i olika nivåer för att utvärdera rankningsfunktionen hos ett informationsåtervinningssystem. Precision@K och Recall@K presenterar Precision respektive Recall för de k första svaren i ett

sökresultat. TREC_EVAL presenterar även Mean Average Precision (MAP) som visar medelvärdet för Precision i olika nivåer.

Discounted Cumulative Gain (DCG)

Ett sätt att utvärdera sökmotorer är att räkna ut DCG (Eng. Discounted Cumulative Gain) som är ett mått på effektiviteten av en sökmotor. DCG mäter användbarheten (gain) av ett dokument i en resultatlista. När en sökfråga ställs producerar en sökmotor en resultatlista i relevansordning och ju högre upp i listan ett dokument är desto högre är DCG. För att beräkna DCG måste två förutsättningar uppfyllas:

- Relevanta dokument med högre rank är mer användbara.
- Den interna ordningen är: att relevanta dokument är mer relevanta än, de marginellt relevanta som i sin tur är mer relevanta än de som är irrelevanta. DCG beräknas enligt formeln nedan.

Discounted Cumulative Gain

$$DCG_p = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i}$$

Figur 6: Formeln för att beräkna DCG

Där rel_i står för den betygsatta relevansen för ett resultat på position i , i en resultatlista. Ett exempel taget från Wikipedia [27] om hur DCG beräknas är om man har sex stycken dokument $D_1, D_2, D_3, D_4, D_5, D_6$ med följande relevans 3,2,3,0,1,2. [27]

Förutsatt att man har värdena för kvoten $rel_i/\log_2 i$ beräknas DCG_6 beräknas enligt figur 7:

$$DCG_6 = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i} = 3 + (2 + 1.887 + 0 + 0.431 + 0.772) = 8.09$$

Figur 7 Beräkna DCG för dokument nr 6 i en resultatlista

<i>i</i>	<i>rel_i</i>	<i>log₂ i</i>	$\frac{rel_i}{\log_2 i}$
1	3	N/A	N/A
2	2	1	2
3	3	1.59	1.887
4	0	2.0	0
5	1	2.32	0.431
6	2	2.59	0.772

Tabell 9 Data som behövs för att beräkna DCG

En av de utvärderingar av sökmotorer med öppen källkod som hittats är en utvärdering som utfördes på följande sökmotorer med öppen källkod: *Lucene*, *Zettair*, *Indri*, *Sphinx*, *RDBMS* och *Xapian*. I utvärderingen används informationssamlingen TREC-9 Filtering track som är en 300 MB mängd av medicinska journaler innehållande omfattande information om olika sjukdomar och symptom. Där framgår dels titel, författare, abstract, nyckelord och även 63 sökfrågor i formen <uppgift (task), dokument (document), 2|1|0 bedömning (rating)> där 2 är väldigt relevant, 1 något relevant och 0

inte värderad. För att mäta effektiviteten har medel DCG (Average DCG) räknats ut på resultatet av sökfrågorna i position 1-10. [28]

Det utvärderingsverktyg som använts i utvärderingen fanns tillgänglig via en github [29] på Internet. En github är en webbaserad tjänst för mjukvaruutveckling som använder versionshanteringsverktyget GIT. En av bristerna med utvärderingen är att den är publicerad i en blogg på Internet och inte publicerad inom den akademiska världen och därför har resultatet av utvärderingen inte varit intressant utan bara tillvägagångssättet.

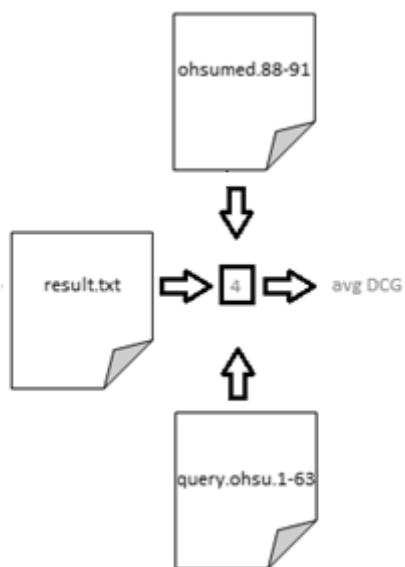
Utvärderingsverktyget är uppbyggt enligt figur 8 där program 1 och 3 ser till att generera filer med rätt format för att användas av index.java för indexering och search.java för sökning.

Program 2 räknar ut totala storleken i bytes av en valfri mapp, i detta exempel har mappen index valts.



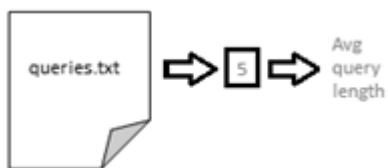
Figur 8: Program för att beräkna storleken på en mapp

Program 4 använder sig av datamängden, sökfrågor och svar för att beräkna medelvärdet av DCG.

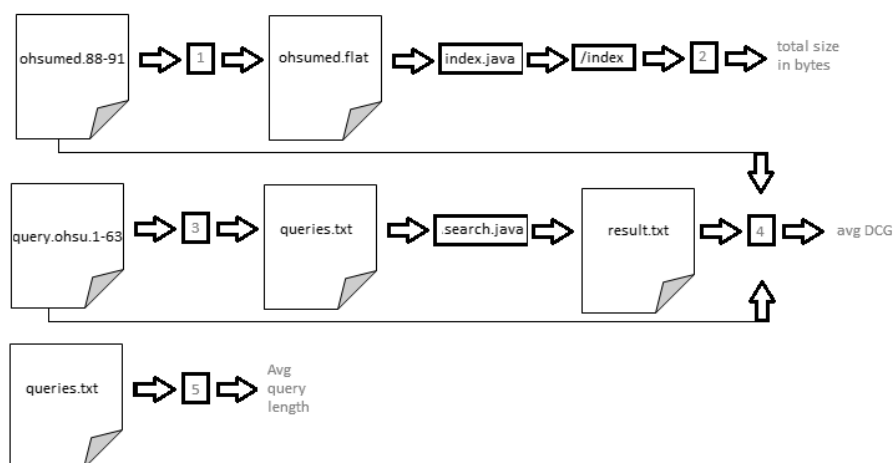


Figur 9 Program för att beräkna medelvärdet av DCG

Program 5 räknar ut medellängden av sökfrågorna vilket kan ge en uppfattning om hur långa sökfrågorna är. Detta har inte använts i examensarbetet.



Figur 10 Program som beräknar medellängden av sökfrågor



Figur 11 Utvärderingsverktyget

Informationssamlingar

Genom att undersöka artiklar och studier om utvärdering av sökmotorer utficks kännedom om att det finns färdiga informationssamlingar som kan användas för att testa sökmotorer. ”Performance Evaluation of Desktop Search Engines” [12] och ”A comparison of Open Source Search Engines” [19] är två artiklar som båda presenterar informationssamlingar från TREC (Se Trec). Syftet med undersökning av informationssamlingar var att bekanta oss med befintliga informationssamlingar och att hitta en lämplig informationssamling som kan användas för att utvärdera sökmotorer för e-handel.

En informationssamling innehåller en datamängd med tusentals dokument från en relevant miljö för att utvärdera sökmotorer. Man kan exempelvis använda sig av en datamängd som innehåller medicinska journaler för att utvärdera sökmotorer för ett sjukvårdssystem. Dessutom kan informationssamlingen innehålla sökfrågor och facit till sökfrågorna. Det finns t.ex. en informationssamling med HTML, Excel, Word- och PDF-filer och

100 sökfrågor samt namn på de relevanta filer där sökfrågorna förekommer. Denna informationssamling har använts för att mäta effektiviteten hos olika sökmotorer i artikeln *"Performance Evaluation of Desktop Search Engines"*[12].

TREC

År 1992 anordnade NIST (National Institute of Standards and Technology) och IRPA (Intelligence Advanced Research Projects Activity) konferensen, Text REtrieval Conference (TREC), vars syfte var att stödja och uppmuntra forskning inom informationsåtervinning genom att erbjuda en infrastruktur för utvärdering av teknologier inom informationsåtervinning. [30] Konferensen som har pågått sedan dess har diskuterat de svårigheter och problem som uppstått i olika områden inom informationsåtervinningen, t.ex. skräppostfilter (Eng. Spam filtering) och bloggsökning (Eng. Blog Search). TREC har även levererat informationssamlingar för att testa och utvärdera olika produkter. Dessa Informationssamlingar innehåller tusentals dokument, sökfrågor (Eng. Queries) och facit till sökfrågorna som kallas för qrels. Informationssamlingarna, som även kallas för Test Collections, innehåller information som hittills inte har publicerats, t.ex. information från Federal Register. Informationen och sökfrågorna är i XML-format och innehåller XML-taggar som presenterar bl.a. syftet med testet och detaljer om sökfrågor. Figur 12 visar sökfrågor och beskrivning från en informationssamling.

```

<topic number="6" type="ambiguous">
  <query>kcs</query>
  <description>Find information on the Kansas City Southern railroad.
  </description>
  <subtopic number="1" type="nav">
    Find the homepage for the Kansas City Southern railroad.
  </subtopic>
  <subtopic number="2" type="inf">
    I'm looking for a job with the Kansas City Southern railroad.
  </subtopic>
  <subtopic number="3" type="nav">
    Find the homepage for Kanawha County Schools in West Virginia.
  </subtopic>
  <subtopic number="4" type="nav">
    Find the homepage for the Knox County School system in Tennessee.
  </subtopic>
  <subtopic number="5" type="inf">
    Find information on KCS Energy, Inc., and their merger with
    Petrohawk Energy Corporation.
  </subtopic>
</topic>

```

Figur 12 sökfrågor och beskrivning från en informationssamling.

Efter att ha tagit kontakt med NIST och sökt information på TREC:s hemsida har vi kommit fram till att TREC inte har levererat någon informationssamling för e-handel och i och med detta valdes att inte använda Informationssamlingar från TREC för att utvärdera sökmotorerna. Däremot användes informationssamlingen Ohsumed-88-89 från TREC för att utveckla testramverket (Se Testramverket).

Produktdatabas

I början av examensarbetet beslutades att en informationssamling från något av uppdragsgivarens projekt skulle användas för utvärderingen av sökmotorerna.

Informationssamlingen som fanns tillgänglig hade använts för en nätbutik som säljer mobiltelefoner och tillbehör.

Informationssamlingen bestod av en matris med 7281 rader med produkter i form av mobiltelefoner och tillbehör och 198 kolumner med kategori, pris, produkt id etc. Till skillnad från TREC som innehåller både sökfrågor och svar innehåller informationssamlingen inga svar utan bara sökfrågor och produktinformation. Sökfrågorna som togs fram från loggar var frågor som hade ställts på hemsidan. Nackdelen med frågorna var att de är väldigt enkla d.v.s. sökfrågorna saknar några komplicerade booleska uttryck och stämmer inte

överens med testfallen. Majoriteten av sökfrågorna är produktnamn med modellbeteckning enligt Nokia E51.

Något som diskuterades under sprintredovisningar och möten var en eventuell möjlighet att från loggar komma åt antal klick som leder till köp från hemsidan, när en kund klickat t.ex. tre gånger och sedan köpt en produkt. Efter undersökning visade det sig dock att det inte fanns tillgängligt i loggarna. Så för att kunna komponera egna svar på sökfrågorna undersöktes möjligheten att från loggarna komma åt de sökfrågor som leder till köp, t.ex. när en kund söker på Sony Ericsson W995 och sedan trycker på köp. Dock fanns detta inte heller tillgängligt.

Testramverket

Ett testramverk i Java har utvecklats för att utvärdera olika sökmotorer genom att analysera resultatet. För att skicka sökfrågor till en sökmotor och ta emot resultatet behöver man använda ett programmeringsgränssnitt. De programmeringsgränssnitt(Eng. Application Programming Interface) som används för att kommunicera med sökmotorer är olika för olika sökmotorer och därför delades testramverket upp i olika moduler. På så vis behöver man inte genomföra stora förändringar i ramverket för att testa en ny sökmotor. Man behöver endast lägga till en ny modul som kommunicerar med sökmotorn i fråga för att söka och generera sökresultatet. Omvandlingsmoduler i ramverket kan därefter läsa in och analysera sökresultatet. En annan fördel med uppdelningen är att man kan byta ut en modul mot ett externt verktyg och på så vis kan man använda sig av de befintliga verktygen i stället för modulen i fråga. I skrivande stund finns följande moduler i ramverket.

Omvandlingsmodul för sökfrågor

Filer som innehåller sökfrågor kan förekomma i flera olika format. Det finns både tabb- och kommaseparerade textfiler men även XML-filer som innehåller sökfrågor. Omvandlingsmodulen för sökfrågor används för att omvandla dessa filer till ett enhetligt format som testmodulen kan läsa in. (Figur 14 visar en fil i detta format) Formatet på dessa filer är olika i olika informationssamlingar och i och med detta ska ramverket utvecklas för varje filformat (Figur 14 visar det format som används i TREC) och man kan lägga till ytterligare metoder för övriga filer.

query_id	iter	doc	rank
1	0	511	1
1	0	513	1
2	0	80	1
2	0	90	1

Figur 14: Sökresultat från en sökmotor

query_id	Iter	doc_id	rank	sim	run_id
1	0	15	20	0.0197334	0
1	0	1	21	0	0
2	0	71	1	0.213504	0
2	0	68	2	0.158238	0

Figur 15: Facit till sökfrågor (qrels)

Testmodul

Testmodulen innehåller metoder som kommunicerar med sökmotorer för att skapa ett index på informationssamlingen och ta emot sökresultat från dessa sökmotorer under sökningen. Det finns t.ex. en modul i ramverket som kommunicerar med

programmeringsgränssnittet för Lucene. Testmodulen läser också in sökfrågorna från en fil som omvandlingsmodulen har genererat. Vidare sparas sökresultaten i en fil som i sin tur används av omvandlingsmodul för sökresultat.

Det finns en möjlighet att lägga till nya moduler för att testa flera sökmotorer i framtiden. Man kan även utföra sökningar utanför ramverket och spara sökresultatet i en fil. Vidare kan man utveckla metoder som omvandlar resultatet till det format som utvärderingsmodulen läser in och analysera resultatet med hjälp av utvärderingsmodulerna.

Omvandlingsmodul för sökresultat

Denna modul har till uppgift att läsa in sökresultat från en fil, som testmodulen har genererat, och omvandla det till ett format som utvärderingsmodulerna kan läsa in. Man kan även lägga till metoder som omvandlar filerna till ett format som ett externt utvärderingsverktyg kan läsa in. På så vis kan man använda utvärderingsmodulerna i ramverket såväl som externa utvärderingsverktyg som TREC_EVAL.

Utvärderingsmodul

Utvärderingsmodulerna i ramverket har till uppgift att jämföra det resultat som en sökmotor har genererat med ett s.k. facit för att utvärdera och analysera resultatet. Utvärderingsmodulerna läser in två filer, sökresultat som omvandlingsmodulen har genererat och ett facit som kallas för ”qrels” (query-relevance sets). Qrels innehåller information om relevanta dokument i informationssamlingen för varje sökfråga (figur 15 visar qrles-filen). Följande utvärderingsmoduler och verktyg kan användas i samband med ramverket för att beräkna värdena för effektivitet och informationsåtervinning hos en sökmotor.

TREC_EVAL

TREC_EVAL är ett standardverktyg som används av TREC (Text REtrieval Conference) för att utvärdera

informationsåtervinningssystem(Se TREC_EVAL). Man kan använda TREC_EVAL för att jämföra sökresultatet med det önskade resultatet och analysera effektivitet och informationsåtervinning hos en sökmotor. TREC_EVAL är ett C-program som läser in två textfiler, qrels.txt och en fil som innehåller sökresultat. Figur 16 visar ett exempel på de beräknade värdena från TREC_EVAL och tabell 9 visar vad de olika värdena står för.

```
$ ./trec_eval test/qrels.test test/results.test
num_q          all      3
num_ret        all     1500
num_rel        all     561
num_rel_ret    all     131
map            all     0.1785
gm_ap         all     0.1051
R-prec        all     0.2174
bpref         all     0.1981
recip_rank    all     0.4064
ircl_prn.0.00 all     0.4665
ircl_prn.0.10 all     0.3884
ircl_prn.0.20 all     0.3186
ircl_prn.0.30 all     0.2732
ircl_prn.0.40 all     0.2666
ircl_prn.0.50 all     0.2184
ircl_prn.0.60 all     0.0822
ircl_prn.0.70 all     0.0348
ircl_prn.0.80 all     0.0312
ircl_prn.0.90 all     0.0312
ircl_prn.1.00 all     0.0312
P5            all     0.2667
P10           all     0.3000
P15           all     0.3111
P20           all     0.3667
P30           all     0.3333
P100          all     0.2467
P200          all     0.1600
P500          all     0.0873
P1000         all     0.0437
```

Figur 16: Analys av sökresultat utförd av TREC_EVAL.

Value	Beskrivning
num_ret	Number of retrieved documents
num_rel	Number of relevant documents
num_rel_ret	Number of relevant retrieved documents
map	Mean Average Precision (MAP)
gm_ap	Average Precision. Geometric Mean, $q_score = \log(\text{MAX}(\text{map}, .00001))$
R-prec	R-Precision (Precision after R (= num-rel for topic) documents retrieved)
bpref	Binary Preference, top R judged nonrel

recip_rank	Reciprical rank of top relevant document
ircl_prn.0.00	Interpolated Recall - Precision Averages at 0.00 recall
ircl_prn.0.10	Interpolated Recall - Precision Averages at 0.10 recall
ircl_prn.0.20	Interpolated Recall - Precision Averages at 0.20 recall
ircl_prn.0.30	Interpolated Recall - Precision Averages at 0.30 recall
ircl_prn.0.40	Interpolated Recall - Precision Averages at 0.40 recall
ircl_prn.0.50	Interpolated Recall - Precision Averages at 0.50 recall
ircl_prn.0.60	Interpolated Recall - Precision Averages at 0.60 recall
ircl_prn.0.70	Interpolated Recall - Precision Averages at 0.70 recall
ircl_prn.0.80	Interpolated Recall - Precision Averages at 0.80 recall
ircl_prn.0.90	Interpolated Recall - Precision Averages at 0.90 recall
ircl_prn.1.00	Interpolated Recall - Precision Averages at 1.00 recall
P5	Precision after 5 docs retrieved
P10	Precision after 10 docs retrieved
P15	Precision after 15 docs retrieved
P20	Precision after 20 docs retrieved
P30	Precision after 30 docs retrieved
P100	Precision after 100 docs retrieved
P200	Precision after 200 docs retrieved
P500	Precision after 500 docs retrieved
P1000	Precision after 1000 docs retrieved

Tabell 10: De värden utvärderingsmoduler kan beräkna

IReval

IReval är ett Java-program av Trevor Strohman [31] vid University of Massachusetts. Programmet är ett utvärderingsverktyg och beräknar Average Precision, Precision, Recall, Geometric och BPREF för en sökmotor (Se TREC_EVAL). Dessa värden är bland de mest använda för att mäta informationsåtervinning enligt Strohman [32]. IReval består av följande Java-klasser:

- **RetrievalEvaluator:** Klassen innehåller metoder för att beräkna värdena för BPREF, GMAP, MAP, Precision och Recall (Se TREC_EVAL) samt presenterar information om relevanta

dokument som ett IR-system har hittat respektive de relevanta dokument som IR-systemet inte har hittat.

- `SetRetrievalComparator`: Klassen innehåller metoder som beräknar värdena för statistiska tester som inte behandlas i detta examensarbete.
- `SetRetrievalEvaluator`: beräknar de önskade värdena för `averagePrecision`, `ndcg`, `bpref`, `P10` och `P20` (Se `TREC_EVAL`).

num_ret	301	500
num_rel	301	474
num_rel_ret	301	71
map	301	0,0688
ndcg	301	0,1584
ndcg15	301	0,0000
R-prec	301	0,1498
bpref	301	0,0292
recip_rank	301	0,0069
P5	301	0,0000
P10	301	0,0000
P15	301	0,0000
P20	301	0,0000
P30	301	0,0000
P100	301	0,0000
P200	301	0,3400
P500	301	0,1420
P1000	301	0,0710
num_ret	302	500
num_rel	302	77
num_rel_ret	302	50
map	302	1,0936
ndcg	302	0,6617
ndcg15	302	0,1706
R-prec	302	0,6494
bpref	302	0,4748
recip_rank	302	1,0000
P5	302	0,2000
P10	302	0,7000
P15	302	0,4667
P20	302	0,3500
P30	302	0,3333
P100	302	0,5000
P200	302	0,2500
P500	302	0,1000
P1000	302	0,0500
num_ret	303	500
num_rel	303	10
num_rel_ret	303	10
map	303	0,1041
ndcg	303	0,3862
ndcg15	303	0,0000
R-prec	303	0,0000
bpref	303	1,0000
recip_rank	303	0,0526
P5	303	0,0000
P10	303	0,0000
P15	303	0,0000
P20	303	0,0500
P30	303	0,0333
P100	303	0,1000
P200	303	0,0500
P500	303	0,0200
P1000	303	0,0100
num_ret	a11	1509
num_rel	a11	561
num_rel_ret	a11	131
map	a11	0,4222
ndcg	a11	0,4021
R-prec	a11	0,2664
bpref	a11	0,3013
recip_rank	a11	0,3532
P5	a11	0,0667
P10	a11	0,2333
P15	a11	0,1556
P20	a11	0,1333
P30	a11	0,1222
P100	a11	0,2000
P200	a11	0,2133
P500	a11	0,0873
P1000	a11	0,0437

Figur 8: Resultat från IReval

Testning

Detta kapitel presenterar tillvägagångssättet för samt resultatet av testning och utvärdering av sökmotorer för e-handel.

Avgränsningar

Innan testningen påbörjades, utfördes ett antal avgränsningar för att hinna med att testa de utvalda sökmotorerna inom examensarbetets tidsramar. Avgränsningarna gällde informationssamlingar, utvärderingar och testning.

Informationssamling

Sökning på flera språk är en viktig aspekt i fråga om utvärdering av sökmotorer för e-handel, men den aspekten är inte inom tidsramen för detta arbete. Dessutom är det inte möjligt att använda denna informationssamling för att utvärdera en sökmotor ur denna aspekt eftersom produkt databasen innehåller information om mobiltelefoner och deras tillbehör från en svensk nätbutik.

Sökhistoriken från nätbutiken vilken innehåller ett antal sökfrågor som skickats till sökmotorn fanns även tillgänglig. Dessa sökfrågor används inte i utvärderingen eftersom det inte finns någon tydlig association mellan sökfrågorna och testfallen. En alternativ metod för att formulera sökfrågor för testfallen är att använda de sökfrågor som ledde till försäljning av produkter.

Utvärdering

Detta examensarbete handlar om utvärdering av effektiviteten hos sökmotorer (Se utvärdering) för e-handel. Utvärderingen har utförts utifrån de prioriterade kraven från uppdragsgivaren och för de sökmotorer som var tillgängliga och var av intresse för uppdragsgivaren. De krav som handlar om verkningsgrad (Eng. efficiency) och användargränssnitt för sökmotorer inte är inom ramen av detta examensarbete.

Testning

Det finns en del avgränsningar gällande testfallen som skulle påverka vilka testfall som skulle utföras.

Testfallet som gällde indexering och sökning av externa källor utgick eftersom det inte var aktuellt att ranka information från externa källor och visa det ihop med sökresultatet från datamängden.

Testfallet som gäller rankning där findAll() funktionen i Microsoft Excel användes för att hitta alla produkter och kolumner innehållande sökordet ”silver” har en del felkällor. Eventuella fel kan uppstå om sökordet stavas med versaler eller gemener.

Det enda alternativet var att för hand gå igenom varje rad och se om produkten innehöll sökordet ”silver”. Detta var dock inte aktuellt då informationssamlingen innehöll över 7000 rader.

Andra urvalet

De sökmotorer som arbetades fram i det första urvalet presenterades till uppdragsgivaren Avensia som fick utföra en prioritering, bestämma vilka som var mest intressanta att testa. Det andra urvalet påverkades även av vilka sökmotorer som var tillgängliga dels via nerladdning från deras hemsidor och dels vilka som var tillgängliga som testversioner/demos. Snittet mellan de sökmotorerna som var prioriterade och de som var tillgängliga var de som skulle testas.

Av sökmotorerna med öppen källkod var alla de som valdes ut i första urvalet tillgängliga via deras hemsidor. Av de prioriterade sökmotorerna var däremot bara Apptus och SQL Server Fulltext Search (SQL FTS) tillgängliga. Det dröjde dock några veckor innan sökmotorn från Apptus var tillgänglig. De sökmotorer som prioriterades högst av Avensia var Apptus, Avail, SQL FTS, Microsoft Fast och Lucene. Därtill valdes Sphinx på grund av att den klarat sig mycket bra i det första urvalet. Därför blev resultatet av det andra urvalet Lucene, SQL FTS och Sphinx.

Lucene

I examensarbetet har Lucene 2.4.1 använts för att skapa ett index på en tabbseparerad textfil som innehåller informationssamlingen. Därefter användes testramverket för att genomföra testfallen enligt testspecifikationen (Se testspecifikation).

Enligt den informationen som är tillgänglig bl.a. på Apache.org, stödjer Lucene sökning och stemming på svenska men stemmingfunktionen kräver en del förbättringar. Ett test har utförts som visar att Lucene kan ta fram grund formen av ”hörlurarna” (se testspecifikationen för Lucene).

Lucene stödjer booleska operander som ”AND”, ”OR” och ”NOT” och de kan kombineras med sökord som skickas till sökmotorn (Se testfall 4,5,6 och 7).

Det finns en funktion i Lucene som kan användas för att söka på särskrivningar men funktionen är inte utvecklad för svenska språket (Se testspecifikationen för Lucene, testfall 9). Man kan dock utveckla ett Java-program som kan användas vid sökning på särskrivna ord.

Lucene stödjer manuell rankning och man kan använda en inbyggd metod för att prioritera både dokument och kategori (Se testspecifikationen för Lucene, testfall 10). Det framgår inte av dokumentationen om det finns metoder för att lagra användarbeteende, statistik om sökfrågor eller historik för användarbeteende (Se testspecifikationen för Lucene, testfall 11, 12, 13).

Tabell 11: Testresultat för Lucene

Testfall	Rubrik	Godkänt
1	Sökning och stemming på svenska	Nej
2	Rankning	Se tabell 20 Lucene Appendix K
3	Indexering på externa källor	Utgått
4,5,6,7	Boolesk sökning	Ja
8	Särskrivning	Nej
9	Manuell rankning	Ja
10	Användarbeteende	Delvis
11	Statistik om sökfrågor	Delvis
12	Användarhistorik	Delvis
13	Tydligt licensavtal	Ja

SQL Full Text Search

Testning av SQL FTS (SQL full-text Search) utfördes dels utifrån informationen på MSDN (Microsoft Development Network) och dels genom att genomföra tester på en MS-SQL Server hos uppdragsgivaren. MS SQL Server Management Studio har använts för att skapa ett s.k. Full Text Index på en tabell som innehåller informationssamlingen. Därefter har testramverket används för att genomföra testfallen enligt testspecifikationen (Se testspecifikation).

Enligt den informationen som är tillgänglig bl.a. på MSDN (se testspecifikationen för MS-SQL FTS), stödjer SQL FTS sökning och stemming på svenska. Ett test har utförts som visar SQL FTS kan hitta alla produkter där hörlur förekommer som t.ex. ”Sony Ericsson HPM-88 hörlur med bullerreducering” om man söker på ”hörlurarna” (se testspecifikationen för MS-SQL FTS).

I SQL-FTS kan man använda booleska operander som "AND", "OR" och "NOT" för att kombinera flera SQL-funktioner (Se testspecifikationen för MS-SQL FTS, testfall 4, 5, 6 och 7). Dessutom finns det en del SQL-funktioner som tar emot en kombination av nyckel- och sökord som argument. (Se testfall 4,5,6 och 7).

SQL Server stödjer särskrivning och sammansatta ord enligt MSDN.(Se testspecifikationen för MS-SQL FTS, testfall 9). Det framgår dock inte av dokumentationen om det finns någon inbyggd metod för manuell rankning, men man skulle kunna implementera manuell rankning med hjälp av en kolumn i databasen. (Se testspecifikationen för MS-SQL FTS, testfall 10).

Det finns en inbyggd historikfunktion i SQL Server som lagrar alla sökningar och ändringar som sker i en databas (Se testspecifikationen för MS-SQL FTS, testfall 11). Dessutom finns det inbyggda attribut för att lagra information om det användarkonto med vilket man har kopplat sig mot SQL-databasen. Man kan även skapa en tabell i databasen och utveckla ett program som lagrar användarhistoriken i tabellen (Se testspecifikationen för MS-SQL FTS, testfall 12).

För att utvärdera rankningssystemet i SQL FTS har testfall 2 skickats till SQL Servern via en testmodul som har utvecklats i Java. Vidare har sökresultaten skickats till en omvandlingsmodul och därefter till utvärderingsmodulen, IReval.

Tabell 12 Testresultat för SQL FTS

Testfall	Rubrik	Godkänt
1	Sökning och stemming på svenska	Ja
2	Rankning	Se tabell 19 Lucene Appendix K
3	Indexering på externa källor	Utgått
4,5,6,7	Boolesk sökning	Ja
8	Särskrivning	Ja
9	Manuell rankning	Delvis
10	Användarbeteende	Delvis
11	Statistik om sökfrågor	Delvis
12	Användarhistorik	Ja
13	Tydligt licensavtal	Ja

Sphinx

Under testningen av Sphinx användes primärt två stycken källor, dels referensmanualen för betaversionen 1.10 i HTML-format [33] och dels kapitlet om Sphinx innehållande en överblick för att hitta information som visar om det stödjer testfallen. [34]

Sphinx stödjer svenska och stemming på svenska genom ett tredjepartsbibliotek (snowball stemmer).

Sphinx stödjer de booleska operatorena ”AND”, ”OR”, ”-” och ”NOT” däremot framgår inte hur ”+” används. Det framgår tydligt vilken licens som gäller. Det finns stöd för att komma åt statistik för sökfrågor men däremot stöds inte särskrivningar på svenska för tillfället utan bara på kinesiska, japanska och koreanska. Stöd för användarhistorik och historik för varje användare hittades inte och testfallet gällande indexering av externa källor har utgått.

För att testa rankningen har ett javaprogram skrivits som kopplar upp sig till SQL servern och kan hämta de rader ur datamängden som innehåller ordet ”silver”. I skrivande stund har detta testfall inte testats.

Testfall	Rubrik	Godkänt
1	Sökning och stemming på svenska	Ja
2	Rankning	inte testats
3	Indexering på externa källor	Utgått
4,5,6,7	Boolesk sökning	Delvis
8	Särskrivning	Nej
9	Manuell rankning	Nej
10	Användarbeteende	Nej
11	Statistik om sökfrågor	Ja
12	Användarhistorik	Nej
13	Tydligt licensavtal	Ja

Tabell 13: Testresultat för Sphinx

Diskussion och slutsats

I detta kapitel presenteras resultat från det första urvalet samt en del svårigheter och möjligheter för vidareutveckling och forskning. Slutligen presenteras återkoppling och synpunkter från examensarbetet.

Resultat av första urvalet

Något som diskuterats fram med Avensia är att dela in sökmotorerna inom både kommersiella och sökmotorer med öppen källkod i tre olika kategorier. Kategorierna är:

- Stora projekt
- Medelstora projekt
- Små projekt

Tanken med indelningen är att välja ut lämpliga kandidater framtagna i det första urvalet för de tre kategorierna genom att analysera de Excel-dokument som producerats och se vilka sökmotorer som uppfyller följande kriterier. Givetvis har kriterierna pris och antal dokument en del brister t.ex. då alla sökmotorerna med öppen källkod är kostnadsfria och då antal dokument beror på vilken licens som valts. Men detta bortses från då kombinationen av alla fem kriterierna är tänkt att fungera som en fingervisning om vilka sökmotorer som är lämpliga inom respektive kategori utifrån den information som var tillgänglig för varje sökmotor.

- Rankning
- Boolesk sökning
- Synonymer
- Pris
- Antal dokument

För sökmotorerna med öppen källkod hittades följande data i Excel-dokumentet.

Tabell 14: Vilka kriterier sökmotorerna med öppen källkod klarar.

Sökmotor	Rankning	Boolesk sökning	Synonymer	Pris	Antal dokument
Indri	Nej	Ja	Ja	Gratis	Upp till 50 miljoner dokument
Omnifind	Ja	Ja	Ja	Gratis	Upp till 500 000 dokument
Sphinx	Ja	Ja	Ja	Gratis	Upp till 1 000 000 dokument
Xapian	Ja	Ja	Ja	Gratis	Databasfiler >2GB

Thunderstone Webinator och Google Site Search bortses också från då de inte har stöd för ranking och boolesk sökning som är kritiskt för en sökmotor för e-handel. Dessutom har Indri inte stöd för rankning och Adobe SiteSearch inte stöd för boolesk sökning.

Google Mini är relativt dyr med tanke på antal dokument den klarar och lämpar sig kanske inte för små projekt. MS Search Server 2010 och MS Search Server 2010 Express har inte stöd för synonymer vilket kanske kan bortses för när det gäller MS Search Server 2010 Express då den är gratis.

Tabell 15: Vilka kriterier de kommersiella sökmotorerna klarar.

Sökmotor	Rankning	Boolesk sökning	Synonymer	Pris	Antal dokument
Oracle Secure Enterprise Search	Ja	Ja	Ja	245,668 kr per processor	1 miljon
MS Search Server 2010	Ja	Ja	-	-	10 miljoner
MS Search Server 2010 Express	Ja	Ja	-	Gratis	mer än 300 000
Adobe SiteSearch	Ja	-	Ja	-	obegränsat
Google Search Appliance	Ja	Ja	Ja	-	mer än 10 miljoner
Google Mini	Ja	Ja	Ja	6000 €	300 000
Google Site Search	Nej	Nej	Ja	250 \$	500 000
Thunderstone Webinator	Nej	Nej	Nej	700 \$	mer än 200 000

Sökmotorer som klarar av mer än 1 miljon dokument passar för stora projekt. För medelstora projekt är det upp till en miljon och för små projekt upp till 300 000 dokument. Då förutsätts att sökmotorer har stöd för rankning, boolesk sökning, synonymer och att priset passar för respektive kategori.

För stora projekt är följande sökmotorer lämpliga.

- Google Search Appliance
- MS Search Server 2010

För medelstora projekt är följande sökmotorer lämpliga.

- Sphinx
- Omnifind

För små projekt är följande sökmotorer lämpliga.

- MS Search Server 2010 Express

Testresultat från testspecifikationen

Tabell 16: Sammanfattning om vilka testfall sökmotorerna uppfyller

Testfall	Rubrik	Lucene	SQL FTS	Sphinx
1	Sökning och stemming på svenska	Nej	Ja	Ja
2	Rankning	Se tabell 19 Appendix K	Se tabell 20 Appendix K	Inte testats
3	Indexering på externa källor	Utgått	Utgått	Utgått
4,5,6,7	Boolesk sökning	Ja	Ja	Delvis
8	Särskrivning	Nej	Ja	Nej
9	Manuell rankning	Ja	Delvis	Nej
10	Användarbeteende	Delvis	Delvis	Nej
11	Statistik om sökfrågor	Delvis	Delvis	Ja
12	Användarhistorik	Delvis	Ja	Nej
13	Tydligt licensavtal	Ja	Ja	Ja

Resultatet i Tabell 16 sammanfattar testningen av sökmotorerna Lucene, SQLFTS och Sphinx. Där framgår om sökmotorerna uppfyller kraven fullt, delvis eller inte alls. Med delvis menas att sökmotorn har stöd för funktionen men att den har brister.

Slutsats

Resultatet av detta examensarbete kan sammanfattas i följande punkter:

- Kravspecifikation
- Informationsdatabas om befintliga sökmotorer
- Informationssamling
- Urval av befintliga sökmotorer
- Testramverk
- Testresultat

Kravspecifikation innehåller alla krav som har fångats under arbetets gång och kan användas för att välja ut de viktigaste kraven i samband med ett e-handelsprojekt. Med andra ord kan man använda kravspecifikationen som en lista med tänkbara krav som ställs på sökmotorer för e-handel och utifrån den välja de krav som är av störst betydelse för projektet. Vidare kan man jämföra sökmotorerna utifrån de prioriterade kraven och sedan väja ut de sökmotorer som uppfyller flest kriterier.

Under första urvalet (Se Appendix G) har det tagits fram ett antal sökmotor som anses vara lämpliga för e-handelsprojekt. Resultatet av första urvalet visar hur dessa sökmotorer kan lämpa sig till olika projekt utifrån projektets omfattning (Se Resultat av första urvalet). Man kan även jämföra sökmotorerna utifrån informationen som presenteras i Appendix G.

Informationsdatabasen innehåller information om befintliga sökmotorer. Databasen kan dels ligga till grund för en kartläggning av nya sökmotorer och dels kan användas för att jämföra dessa sökmotorer med varandra. Informationsdatabasen kan uppdateras kontinuerligt och användas för urval av sökmotorer i uppdragsgivarens kommande e-handelsprojekt.

Informationssamlingen och testramverket kan användas för att analysera sökresultat från en specifik sökmotor. På så vis kan man

välja sökmotorer som ger ett bättre resultat i praktiken. Informationssamlingen och ramverket kan även användas för att trimma sökresultatet för en specifik sökmotor. Man kan konfigurera sökmotorn och sedan använda testramverket för att jämföra sökresultatet med ett facit.

Utvärdering av sökmotorer är ett väldigt brett ämne som kräver stora insatser i form av tid och resurser. Detta examensarbete är ett begränsat exempel på de metoder som kan användas för att utvärdera sökmotorer för e-handel vilket i sin tur underlättar urval av befintliga sökmotorer för e-handelsprojekt och leder till en förbättring av sökfunktionen.

Möjligheter för vidareutveckling av detta projekt

Som nämnts tidigare, har en informationsdatabas utvecklats för att lagra information om befintliga sökmotorer och sökverktyg (Se Verktyg för insamling av information). Databasen innehåller tabeller för att lagra resultatet av det första urvalet men på grund av tidsbrist användes ett Excel-dokument. Det finns möjligheter för att läsa in Excel-dokument och lagra dess information i informationsdatabasen. Dessutom kan man utveckla ett webbgränssnitt som kommunicerar med databasen och gör informationen tillgänglig på uppdragsgivarens interna webbportal. Man kan även vidareutveckla funktioner som läser in statistik från Google Trends för att mäta populariteten hos sökmotorer och presentera informationen med hjälp av webbgränssnittet.

Man kan vidareutveckla testramverket genom att lägga till nya testmoduler för att testa fler sökmotorer. Dessutom kan man lägga till nya utvärderingsmoduler som t.ex. mäter prestandan hos sökmotorer för varje sökning, mäter hur lång tid det behövs för att skapa ett index för ett antal dokument eller mäter storleken på indexeringsinformationen.

Vidare kan en annan informationssamling som innehåller andra typer av information användas, t.ex. en produkt-databas från en nätbutik där man kan ställa komplicerade sökfrågor och mäta effektiviteten hos

sökmotorn utifrån dessa frågor. Man kan t.ex. utvärdera sökmotorn genom att söka på ”gaffelpärm” vilket syftar på en speciell typ av pärm och undersöka om sökmotorn misstolkar sökfrågan som gaffel.

Återkoppling och synpunkter

Eftersom uppdragsgivaren tillämpar SCRUM, den lätttrörliga (eng. Agile) arbetsmetodiken för mjukvaruutveckling på sitt projekt rekommenderades det att använda SCRUM även för examensarbetet.

Där arbetet delas in i sprintar som är 14 dagar långa och som inleds med ett planeringsmöte (eng. Sprint Planning) och avslutas med en sprintredovisning. Under sprintredovisningen presenteras det utförda arbetet under sprinten. Ett dokument (eng. Backlog) skapas för varje sprint och innehåller alla deluppgifter som skall ingå i den kommande sprinten.

Genom sprintredovisningarna har presentationsteknik övats och de har även fungerat som möten. Under mötena har det funnits tid att diskutera och få återkoppling gällande arbetet.

Under arbetet med kravspecifikationen, testspecifikationen och testinstruktionen har kurskompendium 2 från kursen Programvaruutveckling för stora system Version 1.8 -090616 används frekvent.

Något som hade underlättat vårt examensarbete hade varit om det funnits en hemsida likt SearchTools.com som innehåller utvärderingsverktyg, sökmotorer och datamängder specifikt för e-handel.

Referenser

[1] <http://www.eit.lth.se/fileadmin/eit/courses/eitn01/L1-handout4.pdf>

[2] http://www.nada.kth.se/utbildning/grukth/exjobb/rapportlistor/2006/rapporter06/johansson_peter_06008.pdf

[3] <http://sv.wikipedia.org/wiki/Spindel> (internet)

[4] http://www.ne.se/elektronisk-handel?i_h_word=e-handel

[5] http://faculty.ist.psu.edu/jjansen/academic/pubs/jansen_ecommerce_ipm2006.pdf

[6] <http://e-handelsbloggen.nu/>

[7] <http://www.ne.se.ludwig.lub.lu.se/informationsåtervinning>

[8] <http://web.simmons.edu/~benoit/LIS466/Baeza-Yateschap01.pdf>

[9] <http://www.ne.se.ludwig.lub.lu.se/lang/sökmotor>

[10] <http://uu.diva-portal.org/smash/get/diva2:291812/FULLTEXT01>

[11] <http://portal.acm.org/citation.cfm?id=1142110>

[12] Performance Evaluation of Desktop Search Engines

http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4296606

[13] http://tech.groups.yahoo.com/group/search_dev/

[14] http://www.ece.udel.edu/~hfang/lucene/Lucene_exp.pdf

[15] A comparison of Open Source Search Engines

<http://wrg.upf.edu/WRG/dctos/Middleton-Baeza.pdf>, s.7

[16] <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=2EB134373A9397ED97ED07261D7ADE23?doi=10.1.1.86.9332&rep=rep1&type=pdf>

[17] <http://stackoverflow.com/questions/602787/which-are-the-best-alternatives-to-lucene>

[18] <http://www.infoworld.com/d/open-source/bossie-awards-2010-the-best-open-source-applications-150¤t=10&last=3>

[19] Gary Eastwood, Enterprise search suppliers and their products, Fast Search & Transfer, Google Inc. and Microsoft Corp., Computer Weekly, December 2005

[20] <http://www.ideaeng.com/tabId/98/itemId/150/2008-Enterprise-Search-Vendors-The-new-Fab-4.aspx>

[21] <http://www.ideaeng.com/tabId/98/itemId/181/Overview-of-the-Enterprise-Search-Market-2009.aspx>

[22] <http://irsg.bcs.org/SearchSolutions/2010/presentations/mPhasis%20I%20Still%20Haven't%20Found%20What%20I%20am%20Looking.ppt>, s. 10

[23] Open Source Libraries for Information Retrieval, Vesna Hassler”, September 2005, IEEE Computer Society Press”

<http://www.computer.org/portal/web/csdl/doi/10.1109/MS.2005.131>

[24] Stefan Bütcher, m.fl. Implementing and Evaluating Search Engines, 2010, s. 38

[25] <http://ciir.cs.umass.edu/~strohman/code/>

[26] Stefan Bütcher, m.fl. Implementing and Evaluating Search Engines, 2010, s. 8

- [27] http://en.wikipedia.org/wiki/Discounted_cumulative_gain
- [28] A comparison of Open Source Search Engines
<http://zooie.wordpress.com/2009/07/06/a-comparison-of-open-source-search-engines-and-indexing-twitter/>
- [29] <https://github.com/zooie/opensearch/>
- [30] http://en.wikipedia.org/wiki/Text_Retrieval_Conference
- [31] <http://ciir.cs.umass.edu/~strohman/>
- [32] <http://ciir.cs.umass.edu/~strohman/code/>
- [33] <http://sphinxsearch.com/docs/1.10/>
- [34] <http://sphinxsearch.com/about/sphinx/>

Appendix A. Förstudie

Bakgrund

Det finns flertal olika befintliga sökmotorer som används för olika syften, men de uppfyller inte de viktigaste önskemålen när de används i samband med e-handel. Några av bristerna som uppstår är att man inte får träff på rätt sökfråga, får irrelevanta svar, viktar enligt fel kriterier t.ex. viktar på datum istället för försäljningsstatistik och är inte konfigurerbar d.v.s. att man inte kan ställa in sökresultatet om det behövs. Anledningen är att befintliga sökmotorer är skapade för att leta efter information på webbsidor medan e-handel kräver en sökmotor som letar efter produkter som har olika egenskaper som namn, pris o.s.v.

Problembeskrivning

Några av de viktigaste egenskaper som ställs på en sökmotor för e-handel är att den ska kunna konfigureras d.v.s. det går att trimma in den efter frekvent förekommande sökfrågor. Sökmotorn skall snabbt svara på sökfrågor. Det skall finnas stöd för viktning t.ex. efter användarbeteende, målgrupp och säljstatistik. Skall indexera upp relevant information från t.ex. forum, bloggar, recensioner och omdömen. Dessa egenskaper är viktiga för att underlätta sökning för kunder och maximera försäljning. Uppdragsgivaren har ett stort behov av en lösning som uppfyller de ovan nämnda kraven.

Syfte

Syftet med projektet är att ta fram en tillfredställande lösning vilket ökar omsättningen för e-handeln och förbättrar sökfunktionen vilket ger en bättre kundupplevelse.

Krav och avgränsningar

Akademiska krav

- Opposition vid ett seminarium där ett annat examensarbete presenteras.
- En presentation vid ett offentligt seminarium vid LTH.
- En skriftlig rapport på svenska eller engelska med en sammanfattning på engelska.
- En separat sammanfattning som kan vara populärvetenskaplig eller ha formen av en vetenskaplig artikel.

Uppdragsgivarens krav

- Presentationer av utfört arbete och resultat.
- Kontinuerliga redovisningar av arbetet som utförts under en två veckors period.
- Ta fram en kravlista för utvärdering av sökmotorer för e-handel.
- Att utvärdera befintliga sökmotorer utifrån kravlista.

Genomförande

Arbetet inleds med att få en överblick om ämnet och försätter sedan genom att studera och samla in information gällande tekniska detaljer för sökmotorer. Informationsinsamlingen sker genom intervjuer, möten med experter inom flera olika ämnesområden och studera vetenskapliga artiklar samt genom kontinuerlig handledning och mailkorrespondens med experter som är verksamma både inom näringslivet och på universitetets sida.

Arbetsmetodiken SCRUM tillämpas och vid dagliga statusmöten besvaras frågorna:

- Vad som har gjorts sedan igår?
- Vad som skall åstadkommas tills i morgon?
- Vad hindrar oss?

Anledningen är att få en överblick hur arbetet fortlöper och att åtgärda eventuella problem.

Möten

Möten med examinator för att verifiera att examensarbetet håller en rimlig vetenskaplig nivå.

- Dagliga statusmöten med handledare för att gå igenom gårdagens och dagens uppgifter, problem och tankar.
- Möten med föreläsare och professorer på Lunds Tekniska Högskola för att diskutera tekniska detaljer gällande sökmotorer.

Redovisningar

- Redovisningen av examensarbetet utförs muntligt inför examinator och opponenter.
- Sprintredovisning: redovisning av avslutat delmål som sker var annan vecka.

Ansvarsfördelning

Resurserna består av två personer där ansvarsfördelningen är jämnt fördelad och arbetsfördelningen bestäms inför varje sprint.

Tidsplanering

Tabell 17 visar vilka aktiviteter som utförs under projektet, respektive arbetstimmar och en kort beskrivning av aktiviteten.

Aktivitet	Arbetstimmar	Beskrivning
Morgonmöte	60	Dagliga statusmöten med handledare
Tidsplan	5	Skriva en tidsplan för projektet samt bestämma milstolpar
Problembeskrivning	10	Reda ut vad problemet innebär.
Förstudie	40	Bestämma bakgrund, syfte och metodologi för projektet.
Fördjupning	121	Studera tekniska detaljer för att utvärdera och undersöka en lämplig lösning på problemet.
Planering	16	Planera och bestämma arbetsmetodik.
Skriva kravlista	16	Fastställa en kravlista för problembeskrivningen
Undersöka kriterier för utvärdering	80	Komma fram till en lämplig metodologi som ligger till grund för utvärderingen.
Utvärdering av tio sökmotorer	1000	Analysera och utvärdera lämpliga sökmotorer.
Sprintredovisning 1	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 2	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 3	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 4	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 5	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 6	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 7	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 8	6	Redovisning av utfört arbete i pågående sprint.
Skriva rapport	140	Skriftlig redovisning av resultatet.
Korrigera rapport	40	Eventuella korrigeringar efter granskning.
Redovisning	30	Muntlig redovisning av arbetet.

Summa antal arbetstimmar

1606

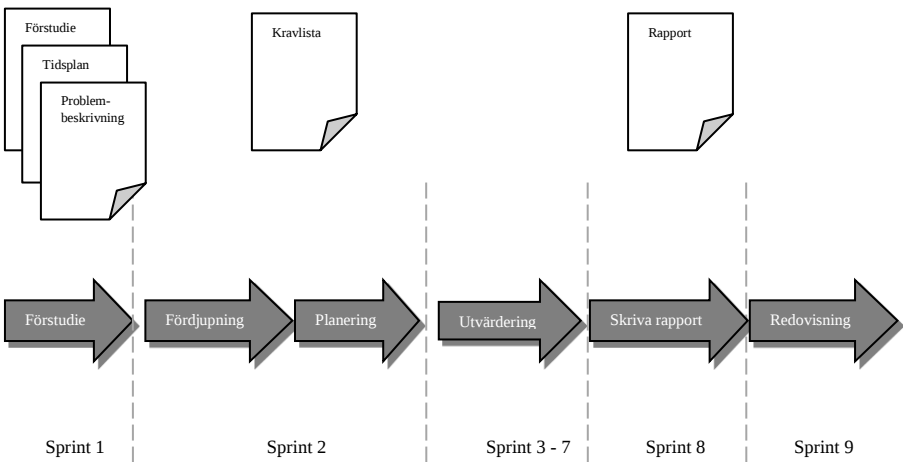
Vilket stämmer bra med antal arbetstimmar för 20
veckors arbete.

20x5x8x2=1600 h

Tabell 17: Antal arbetstimmar specifikt för varje aktivitet

Projektmodell

Figur 18 visar projektets flöde i förhållande till varje sprint. Arbetet fördelas i sprintar som är på två arbetsveckor och en redovisning hålls i slutet av varje sprint.



Figur 18: Projektflöde

Risikanalys

Tabell 18 visar de identifierade riskerna samt förslag på åtgärder som kan vidtas. Tabellen är sorterad efter sannolikhet och efter effekt på arbetet och det framgår tydligt att regelbunden återkoppling från

Risk	Sannolikhet	Effekt på arbetet	Förslag på åtgärd
Att missat att granska en ny och viktig produkt	80 %	90 %	Regelbunden uppföljning av utvecklingen inom ämnesområdet
Avsaknad av relevant vetenskaplig källa	45 %	90 %	Regelbunden återkoppling från examinator
Dålig planering	35 %	80 %	Regelbunden återkoppling från handledare och examinator
Felaktigt resultat	30 %	90 %	Regelbunden återkoppling från examinator
Avsaknad av tydliga mål	30 %	90 %	Genom regelbunden uppföljning och granskning med handledare och examinator
Ingen av de befintliga sökmotorerna uppfyller kraven	30 %	70 %	Ta fram en rimlig kravlista
Fel kriterier vid utvärdering	30 %	70 %	Regelbunden återkoppling från handledare och examinator
Ointressant resultat för företaget	30 %	65 %	Regelbunden återkoppling från handledare
Brist på motivation	10 %	70 %	Olika motivationsmetoder och lyfta fram motiven för att uppnå målet

examinator och handledare kan åtgärda de största problemen i projektet.

Tabell 18: Identifierade risker som kan påverka projektet

Förslag till lösning

Genom att studera vetenskapliga artiklar och utföra intervjuer med experter fastställs en kravlista som gäller för sökmotorer för e-handel.

Vidare används kravlistan för att ta fram en utvärderingsmetod för att jämföra befintliga sökmotorer för e-handel. Slutligen genom att definiera ett antal scenarier kan den lämpligaste lösningen sedan väljas.

Appendix B. Kravspecifikation

Revisionsinformation

Utfärdat av	Datum	Version	Ändringar
Siddhartha Pattni	2010-10-15	1.0	Baseline
Babak Razavi	2010-12-09	1.1	Uppdatera formatering
Babak Razavi	2010-12-10	1.2	Uppdatera formatering

Inledning

Detta dokument presenterar de krav som fastställts efter undersökning och intervjuer för sökmotorer som ska användas i samband med e-handel. Kravspecifikationen skall ligga till grund för en utvärdering av sökmotorlösningar för e-handel.

Bakgrund och Målsättningar

Bakgrund

Det finns flertal olika befintliga sökmotorer som används för olika syften, men de uppfyller inte de viktigaste önskemålen när de används i samband med e-handel. Några av bristerna som uppstår är att man inte får träff på rätt sökfråga, får icke-relevanta svar, sökresultatet rankas enligt felaktiga kriterier t.ex. rankas på datum istället för försäljningsstatistik och är inte konfigurerbar d.v.s. att man inte kan ställa in sökresultatet om det behövs. Anledningen är att

befintliga sökmotorer är skapade för att leta efter information på webbsidor medan e-handel kräver en sökmotor som letar efter produkter som har olika egenskaper som namn, pris osv.

Huvudsyfte och Användarperspektiv

Huvudsyftet med kravgranskning är att fånga de tänkbara krav som ställs på en sökmotor i ett e-handelsprojekt. Kraven ställs från utvecklarna på Avensia som ska leverera, t.ex. en nätbutik till sina kunder enligt e-handelsmarknadens behov.

Uppdragsgivare

Avensia AB arbetar med att skapa marknadsledande och effektiva e-handelssidor och använder Microsoft Commerce Server som plattform för deras e-handelslösningar.

Projektgrupp

Projektgruppen består av två studenter, Babak Razavi och Siddhartha Pattni, från civilingenjörsprogrammen Infocom respektive Datateknik på LTH. Erik Arnwald från Avensia har blivit utsedd som handledare och Anders Ardö från institutionen EIT på LTH som examinator.

Systemavgränsning

Krav gällande gränssnitt och design behandlas inte i detta dokument.

Tidsplan

Bilaga A: Tidsplan.xls

Systemkrav

Funktionella krav

Språk

Aktörer:

A: En potentiell kund som letar efter en produkt eller information om produkten med hjälp av sökmotorlösningen.

S: Sökmotorlösningen.

"Sökfunktion på flera språk"

Förutsättning: En sökfras formuleras på ett av språken som nämns i krav 41211~41219. Sökfrasen innehåller en kombination av produktnamn, produktkategorier och stoppord.

A skickar en sökfras till S. S identifierar språket, filtrerar bort stopporden. S skickar sökresultatet till A.

"Sökning av synonymer"

Förutsättning: Sökfrasen innehåller synonymer för produktnamn eller kategori och är formulerade på ett av 4121 till 4128.

A skickar en sökfras till S. S identifierar språket. S Identifierar synonymer i sökfrågan och letar efter dem i databasen. S skickar sökresultatet till A. A skickar t.ex. en sökfras som "kameramobil", "businessmobil" eller "designmobil" till S.

"Sökning på svenska med ett danskt tangentbord"

Förutsättning: A är en svensktalande användare som sitter vid en dator med ett danskt tangentbord.

A skickar en sökfras till S. S identifierar språket trots att sökfrågan innehåller danska tecken, t.ex. "Skønheden och odjuret"

"Sökning av termer och begrepp"

Förutsättning: S har tillgång till information om termer och begrepp som används i produktbeskrivningar.

A skickar en sökfras på svenska som även innehåller ord på främmande språk, t.ex. "Kameramobil exchange". S identifierar språket. S letar efter facktermen i databasen. I och med att facktermen finns i databasen ändrar S inte termen. Funktioner som rättstavning och översättning avaktiveras för denna term så att "Exchange" inte översätts till "växling". S skickar tillbaka svaren till A.

"Stemming, identifiera ordstam"

Förutsättning: En sökfras formuleras med en ändelse för t.ex. plural. A skickar en sökfras som har en ändelse för plural t.ex. "mobiler". S identifierar ändelse och vilken stam ordet tillhör, i detta fall stammen "mobil". S skickar tillbaka svaren till A.

"Sökning/indexering bland olika filformat"

Förutsättning: Sökmotorn har tillgång till information lagrad bland olika filformat.

A skickar en sökfras. S identifierar den. S söker och indexerar bland flera format t.ex. PDF, html och skickar tillbaka svaren till A.

41211. Sökmotorlösningen ska klara sökning på svenska.
41212. Sökmotorlösningen ska klara sökning på danska.
41213. Sökmotorlösningen ska klara sökning på finska.
41214. Sökmotorlösningen ska klara sökning på norska.
41215. Sökmotorlösningen ska klara sökning på engelska.
41216. Sökmotorlösningen ska klara sökning på estniska.
41217. Sökmotorlösningen ska klara sökning på isländska.
41218. Sökmotorlösningen ska klara sökning på europeiska språk.
41219. Sökmotorlösningen ska klara sökning på ett av språken i krav
41211~41218
41220. Sökmotorlösningen ska klara sökning där sökfrågan är
formulerad med ett tangentbord med vilket språket inte är
samma som hos själva sökfrågan.
41221. Sökmotorlösningen ska filtrera bort stoppord för samtliga språk
(Se Krav 4121~4125).
41222. Sökmotorlösningen ska ha en rättstavningsfunktion så att när
användaren stavar fel får användaren fler korrekta
alternativ.
41223. Sökmotorlösningen ska klara sökning av synonymer för samtliga
språk (Se Krav 4121 till 4125).
41224. Sökmotorlösningen ska stödja "stemming" på samtliga språk.
41225. Sökmotorlösningen ska kunna indexera flera filformat, t.ex. html,
text och PDF.

Rankning

Aktörer:

A: En potentiell kund som letar efter en produkt eller information om produkten med hjälp av sökmotorlösningen.

B: Systemadministratör

S: Sökmotorlösningen.

"Rankning utifrån förekomsten av sökfras"

Förutsättning: Sökfrasen innehåller ett eller flera ord som förekommer i produktinformationen som finns i databasen.

A skickar en sökfras till S. S letar efter sökfrasen och rankar produkter utifrån vilket fält ordet förekommer i, t.ex. när sökordet förekommer i rubriken ska produkten rankas högre än när det förekommer i beskrivningen.

"Rankning utifrån kontext"

Förutsättning: Sökfrasen innehåller ett eller flera ord som förekommer i produktinformationen men produkten är i en icke-relevant kontext, t.ex. skall "Gaffelpärm A4" rankas högre än "Kalender gaffeltyp" när man söker på "Gaffel".

A skickar en sökfras till S. S letar efter sökfrasen och rankar produkter som inte är i samma kontext lågt. S skickar sökresultatet till A.

"Rankning utifrån säljstatistik"

Förutsättning: S har tillgång till säljstatistik för varje produkt i databasen.

A skickar en sökfras till S. S letar efter sökfrasen och rankar sökresultatet efter kriterier från scenario 7 och 8. S rankar det sorterade resultatet efter säljstatistik så att produkter som säljer mest kommer först i prioritetsordningen.

"Rankning utifrån användarkategori"

Förutsättning: S har tillgång till information om A:s användarkategori om t.ex. A är privat- eller företagskund. S har även tillgång till information om varje produkts målgrupp t.ex. när man söker på "X10" kan "X10 mini Xperia" och "X10 Xperia" vara mer intressanta för företagskunder.

A skickar en sökfras till S. S rankar de produkter utifrån den användarkategori som A tillhör. S skickar sökresultatet till A.

"Rankning utifrån konfiguration"

Förutsättning: B kan t.ex. konfigurera S så att en viss produkt rankas högre än de andra. S har tillgång till konfigurationen. B har konfigurerat S så att "iPhone 4" rankas högre än "iPhone 3G". Anledningen är att "iPhone 4" kommer efter "iPhone 3G" eftersom sökresultatet sorteras i kronologisk ordning.

B skickar en sökfråga till S. S rankar sökresultatet utifrån konfigurationen. S skickar sökresultatet till A.

"Rankning utifrån användarhistorik"

Förutsättning: S har tillgång till användarhistorik för varje användare. Användarhistoriken innehåller information om de produkter som en användare har köpt och de sökfrågor användaren har skickat till S.

A skickar en sökfråga till S. S rankar sökresultatet utifrån användarhistorik. S skickar sökresultatet till A.

"Rankning - Identifiera populära produkter m.h.a. Google"

Förutsättning: S har tillgång till Googles sökresultat. Antalet träffar hos Google kan användas för att ranka populariteten hos produkter.

A skickar en sökfråga till S. S rankar sökresultatet utifrån antalet träffar. Varje produkt som ska visas rankas utifrån antal träffar hos Google.

41371. Sökmotorlösningen ska kunna ranka sökresultat.
41372. Sökmotorlösningen ska ranka utifrån vilket fält attributet förekommer i.
41373. Sökmotorlösningen ska ranka utifrån kontext så att produkter i en icke-relevant kontext ska rankas lågt.
41374. Sökmotorlösningen ska klara av att ranka utifrån säljstatistik.
41375. Sökmotorlösningen ska klara av att ranka utifrån användarkategori.
41376. Sökmotorlösningen ska klara av att ranka efter vald konfiguration.
41377. Sökmotorlösningen ska klara av att ranka efter användarhistorik.
41378. Sökmotorlösningen ska klara av att ranka efter antal träffar hos Google.

Konfiguration

"Konfigurera önskade sökresultat för sökkriterier"

Förutsättning: B önskar att manuellt kunna ställa in sökresultat för en eller flera kriterier, t.ex. B önskar att "Exchange" ska ge träff på "Sony Ericsson Naite" som stödjer MS-Exchange.

B konfigurerar ett sökresultat för en sökfras. S visar resultatet när A skickar sökfrasen till S.

"Konfigurera relationer"

Förutsättning: B önskar att definiera en relation mellan olika produkter. B önskar att "X10" ska ge träff på alla tillbehör med relation till "X10", t.ex. tillhörande batterier, hörlurar, etc.

B konfigurerar relationer mellan olika produkter.

"Konfigurera manuell rankning (Force Position)"

Förutsättning: B önskar att manuellt ranka en produkt, t.ex. B önskar att ranka "iPhone 4" högst.

B konfigurerar en manuell rankning för en produkt i S.

"Konfigurera externa källor för informationsinsamling"

Förutsättning: B önskar att konfigurera externa källor för varje produkttyp t.ex. omdömmen, forum och recensioner.

B konfigurerar vilka de externa källorna för informationsinsamling är för S.

"Konfiguration av termer och begrepp"

Förutsättning: B ska lägga till facktermer och uttryck i databasen, t.ex. MMS, Exchange och TrackID.

B konfigurerar S och lägger till nya termer i databasen. S använder informationen när A skickar en sökfråga till S.

"Konfigurera identifiering av populära produkter m.h.a. Google:s sökresultat"

Förutsättning: B ska identifiera populära produkter med hjälp av Google. Man kan jämföra antalet träffar för att bedöma hur populär en produkt är.

B konfigurerar S och lägger till nya termer i databasen. S använder informationen när A skickar en sökfråga till S.

41461. Sökmotorlösningen ska erbjuda möjligheten att konfigurera sökresultat efter specifika sökkriterier.

41462. Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.

41463. Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att information indexeras upp från externa källor.

41464. Sökmotorlösningen ska erbjuda möjligheten att konfigurera facktermer och begrepp.

41465. Sökmotorlösningen ska erbjuda möjligheten att konfigurera identifiering av populära produkter m.h.a. Google:s sökresultat.

Relationer

"Sökresultat utifrån relationer"

Förutsättning: S har tillgång till relationer mellan olika produkter, t.ex. en USB laddare kan ladda flera olika modeller och har därför en relation till X10.

A skickar en sökfråga till S. S visar även produkter som har en relation till sökfrasen.

Användarbeteende/Statistik

"Användarbeteende i samband med varje sökfråga"

Förutsättning: S har tillgång till A:s IP-nummer eller användarnamn.

S lagrar sökfrågor och användarbeteende i samband med varje sökfråga d.v.s. de sidor som A besökt och hur lång tid som har spenderats på varje sida.

"Användarhistorik"

Förutsättning: S har tillgång till A:s IP-nummer eller användarnamn

S lagrar alla produkter A har köpt.

"E-handelsspårning"

Förutsättning: S lagrar statistik om användarbeteenden i samband med kampanjer.

S ska koppla alla transaktioner till en produkt och på så sätt identifiera lönsamma inkomstkällor. B kan få en bra bild av vilka sökord och kampanjer som är ekonomiskt lönsam.

"Identifiera avsaknad av lämpligt sökresultat"

Förutsättning: S lagrar A:s sökfrågor, antal träffar och de sidor A besöker.

B kan gå in i systemet och om det finns sökfrågor som inte har gett träffar.

41641. Sökmotorlösningen ska kunna lagra statistik om alla sökfrågor och de användare som har skickat dem till sökmotorn.

41642. Sökmotorlösningen ska kunna lagra användarbeteende i samband med varje sökfråga.

41643. Sökmotorlösningen ska kunna lagra statistik om olika sökfrågor.

41644. Sökmotorlösningen ska kunna lagra historik för varje användare.

41645. Sökmotorlösningen ska kunna utföra E-handelsspårning.

Externa källor

"Samla in information från externa källor".

Förutsättning: B har konfigurerat externa källor för varje produkttyp.
S har tillgång till konfigurationen.

S söker igenom och sparar undan information från forum, recensioner och omdömen angivna i konfigurationen. A skickar en sökfråga till S. S skickar sökresultatet och resultatet från externa källor till A.

Sökmotorlösningen ska indexera upp information från externa källor.

Indexering

"Indexering efter uppdatering av databas".

Förutsättning: B har uppdaterat produktdatabasen.

B skickar en förfrågan till S. S uppdaterar sina index.

En administratör ska manuellt kunna uppdatera sökmotorlösningens index.

Kvalitetskrav

Användarvänlighet

”En sökfråga uttrycks i kombination av ”

Förutsättning: En sökfras formuleras med en booleska operander.
A skickar en sökfras som har en booleska operand. S identifierar den booleska operander. S skickar tillbaka svaren till A.

42111. Sökmotorlösningen ska erbjuda möjligheten för live-sökning.
42112. Användare utan avancerade datorkunskaper ska kunna utföra sökningar.
42113. En administratör ska kunna administrera sökmotorlösningen utan någon akademisk utbildning.
42114. Systemutbildning för administratörer ska högst ta en arbetsdag.
42115. Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. "", +, och booleska operander så att användare med medel och höga datakunskaper ska kunna använda den.
42116. Det ska finnas aktuell dokumentation om API.
42117. Det ska finnas aktuell och fungerande exempelkod.
42118. Det ska finnas C#- eller Java-bibliotek.
42119. Sökmotorlösningen ska stödja faceted search (faceted navigation).
421110. Sökmotorlösningen vara självlärande gällande synonymer och produktkategorier.
421110. Oindexeringstiden ska vara proportionerlig mot antal produkter.
421112. Prismodellen ska vara proportionerlig mot antal produkter och språk.

Utvidgbarhet

42211. Sökmotorleverantören ska erbjuda möjligheten att uppgradera sökmotorns kapacitet om antalet produkter ökar.

42212. Sökmotorlösningen ska erbjuda möjligheten att aktivera installationen av automatiska uppdateringar.

42213. Skall framgå i dokumentationen om hårdvarukrav som gäller.

42214. Skall framgå i dokumentationen om sökmotorlösningen stödjer redundans.

42215. Skall framgå i dokumentation om sökmotorlösningen stödjer klustring(gruppering)

Tillförlitlighet

42311. Systemet skall vara tillgängligt 99,7% av året.

Prestanda

Kategori	Volym (Artiklar)	Belastning (sökfrågor)	Användare	Uppdatering	Frekvens
A	Ca 35000			1 ggr	10 min
B1	Ca 2400			2 ggr	1 timme
B1	Ca 7000	9000 ~10000/timme	9000/timme		
C1	Ca 10000			1ggr	10 min
C2	Ca 2000			1ggr	10 min

Table 17: Projektkategorier för utförda projekt hos uppdragsgivaren.

Användaren skall erhålla sökresultat inom högst 800 millisekunder.

Användaren bör erhålla sökresultat inom ca 250 millisekunder för små projekt.

Sökmotorn skall klara minst 9000 sökningar från 9000 användare per timme.

Sökmotorlösningen ska klara av minst 125 förfrågningar per sekund.

Svarstiden ska vara mindre än maximum svarstiden för en av klasserna i tabell ovan.

Kostnader

Det ska framgå tydligt i licensavtalet vilken produkt det avser.

Det ska framgå tydligt för hur många användare licensavtalet gäller.

Det ska framgå tydligt i vilken utvecklingsmiljö produkten kan användas.

Det ska framgå tydligt i licensavtalet med vilka kostnader leverantören erbjuder sina tjänster.

Det ska framgå tydligt i licensavtalet hur lång tid leveranstiden är.

Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för att installera sökmotorlösningen.

Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för underhåll av sökmotorlösningen.

Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för att byta från en sökmotor till denna sökmotorlösning när det redan finns ett fungerande system.

Appendix C. Tidsplan

Aktivitet	Arbetstimmar	Beskrivning
Morgonmöte	60	Dagliga statusmöten med handledare
Tidsplan	5	Skriva en tidsplan för projektet samt bestämma milstolpar
Problembeskrivning	10	Reda ut vad problemet innebär.
Förstudie	40	Bestämma bakgrund, syfte och metodologi för projektet.
Fördjupning	121	Studera tekniska detaljer för att utvärdera och undersöka en lämplig lösning på problemet.
Planering	16	Planera och bestämma arbetsmetodik.
Skriva kravlista	16	Fastställa en kravlista för problembeskrivningen
Undersöka kriterier för utvärdering	80	Komma fram till en lämplig metodologi som ligger till grund för utvärderingen.
Utvärdering av tio sökmotorer	1000	Analysera och utvärdera lämpliga sökmotorer.
Sprintredovisning 1	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 2	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 3	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 4	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 5	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 6	6	Redovisning av utfört arbete i pågående sprint.
Sprintredovisning 7	6	Redovisning av utfört arbete i pågående sprint.

Sprintredovisning 8	6	Redovisning av utfört arbete i pågående sprint.
Skriva rapport	140	Skriftlig redovisning av resultatet.
Korrigera rapport	40	Eventuella korrigeringar efter granskning.
Redovisning	30	Muntlig redovisning av arbetet.
Summa antal arbetstimmar	1606	Vilket stämmer bra med antal arbetstimmar för 20 veckors arbete. $20 \times 5 \times 8 \times 2 = 1600 \text{ h}$

Appendix D. Prioritering av kraven

Prioritet	Kravnr.	Krav
1	Krav 41211	Sökmotorlösningen ska klara sökning på svenska.
1	Krav 41215	Sökmotorlösningen ska klara sökning på engelska.
1	Krav 41219	Sökmotorlösningen ska klara sökning på ett av språken i krav 41211~41219
1	Krav 412114	Sökmotorlösningen ska stödja "stemming" på samtliga språk.
1	Krav 41371	Sökmotorlösningen ska kunna ranka sökresultat.
1	Krav 41372	Sökmotorlösningen ska ranka utifrån vilket fält attributet förekommer i.
1	Krav 41462	Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.
1	Krav 41463	Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att information indexeras upp från externa källor.
1	Krav 41462	Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.
1	Krav 41463	Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att information indexeras upp från externa källor.
1	Krav 41641	Sökmotorlösningen ska kunna lagra statistik om alla sökfrågor och de användare som har skickat

		dem till sökmotorn.
1	Krav 41643	Sökmotorlösningen ska kunna lagra statistik om olika sökfrågor.
1	Krav 41644	Sökmotorlösningen ska kunna lagra historik för varje användare.
1	Krav 41711	Sökmotorlösningen ska indexera upp information från externa källor.
1	Krav 42111	Sökmotorlösningen ska erbjuda möjligheten för live-sökning.
1	Krav 42115	Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. "" ,+, och booleska operatorer så att användare med medel och höga datakunskaper ska kunna använda den.
1	Krav 42211	Sökmotorlösningen ska erbjuda möjligheten att uppgradera sökmotorns kapacitet om antalet produkter ökar.
1	Krav 42531	Det ska framgå tydligt i licensavtalet vilken produkt det avser.
2	Krav 41212	Sökmotorlösningen ska klara sökning på danska.
2	Krav 41213	Sökmotorlösningen ska klara sökning på finska.
2	Krav 41214	Sökmotorlösningen ska klara sökning på norska.
2	Krav 412115	Sökmotorlösningen ska kunna söka/indexera flera filformat html, text, PDF.
2	Krav 41373	Sökmotorlösningen ska ranka utifrån kontext så att produkter i en icke-relevant kontext ska rankas lågt.

2	Krav 41374	Sökmotorlösningen ska klara av att ranka utifrån säljstatistik.
2	Krav 41375	Sökmotorlösningen ska klara av att ranka utifrån användarkategori.
2	Krav 42116	Det ska finnas aktuell dokumentation om API.
2	Krav 42118	Det ska finnas C# eller Java bibliotek.
2	Krav 42213	Skall framgå i dokumentationen om hårdvarukrav.
2	Krav 42214	Skall framgå i dokumentationen om sökmotorlösningen stödjer redundans.
2	Krav 42215	Skall framgå i dokumentation om sökmotorlösningen stödjer klustring(gruppering)
2	Krav 42331	Systemet skall vara tillgängligt 99,7% av året.
2	Krav 42534	Det ska framgå tydligt i licensavtalet med vilka kostnader leverantören erbjuder sina tjänster.
2	Krav 425312	Det skall framgå tydligt hur priset ökar i förhållande till antal produkter.
3	Krav 41218	Sökmotorlösningen ska klara sökning på europeiska språk.
3	Krav 41376	Sökmotorlösningen ska klara av att ranka efter vald konfiguration.
3	Krav 41377	Sökmotorlösningen ska klara av att ranka efter användarhistorik.
3	Krav 41461	Sökmotorlösningen ska erbjuda möjligheten att konfigurera sökresultat efter specifika sökkriterier.
3	Krav 41464	Sökmotorlösningen ska erbjuda möjligheten att

		konfigurera facktermer och begrepp.
3	Krav 41461	Sökmotorlösningen ska erbjuda möjligheten att konfigurera sökresultat efter specifika sökkriterier.
3	Krav 41464	Sökmotorlösningen ska erbjuda möjligheten att konfigurera facktermer och begrepp.
3	Krav 41645	Sökmotorlösningen ska kunna utföra E-handelsspårning.
3	Krav 42119	Sökmotorlösningen ska stödja faceted search (faceted navigation).
3	Krav 421110	Sökmotorlösningen vara självlärande gällande synonymer och produktkategorier.
4	Krav 412111	Sökmotorlösningen ska filtrera bort stoppord för samtliga språk (Se Krav 4121 till 4128).
4	Krav 412113	Sökmotorlösningen ska klara sökning av synonymer för samtliga språk (Se Krav 4121 till 4128).
4	Krav 41378	Sökmotorlösningen ska klara av att ranka efter antal träffar hos Google.
4	Krav 41465	Sökmotorlösningen ska erbjuda möjligheten att konfigurera en identifiering av populära produkter m.h.a. Google
4	Krav 41465	Sökmotorlösningen ska erbjuda möjligheten att konfigurera en identifiering av populära produkter m.h.a. Google
4	Krav 41642	Sökmotorlösningen ska kunna lagra användarbeteende i samband med varje sökfråga.
4	Krav 41811	Administratören ska kunna uppdatera sökmotorlösningens index manuellt.

4	Krav 42117	Det ska finnas aktuell och fungerande exempelkod.
4	Krav 421111	Tiden för hur oindexeringen ökar ska vara proportionerlig mot antal produkter.
4	Krav 421112	Prismodellen ska vara proportionerlig mot antal produkter och språk.
4	Krav 42212	Sökmotorlösningen ska erbjuda möjligheten att aktivera installationen av automatiska uppdateringar.
4	Krav 42532	Det ska framgå tydligt för hur många användare licensavtalet gäller.
4	Krav 42533	Det ska framgå tydligt i vilken utvecklingsmiljö produkten kan användas.
4	Krav 42535	Det ska framgå tydligt i licensavtalet hur lång tid det tar för leverantören att leverera sökmotorlösningen.
5	Krav 41216	Sökmotorlösningen ska klara sökning på estniska.
5	Krav 41217	Sökmotorlösningen ska klara sökning på isländska.
5	Krav 412110	Sökmotorlösningen ska klara sökning där sökfrågan är formulerad med ett tangentbord med vilket språket inte är samma som hos själva sökfrågan.
5	Krav 412112	Sökmotorlösningen ska ha en rättstavningsfunktion så att när användaren stavar fel får användaren flera alternativ.
5	Krav 42112	Användare utan avancerade datorkunskaper ska kunna utföra sökningar.
5	Krav 42113	En administratör ska kunna administrera sökmotorlösningen utan någon akademisk

		utbildning.
5	Krav 42114	Systemutbildning för administratörer ska högst ta en arbetsdag.
5	Krav 42536	Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för att installera sökmotorlösningen.
5	Krav 42537	Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för underhåll av sökmotorlösningen.
5	Krav 42538	Det ska framgå tydligt licensavtalet hur lång tid och vilka kompetenser som behövs för att byta från en sökmotor till denna sökmotorlösning när det redan finns ett fungerande system.
	Krav 42539	Leverantören ska erbjuda information om hur lång tid och vilka kompetenser som behövs för att implementera scenario 31.
	Krav 425310	Leverantören ska erbjuda information om hur lång tid och vilka kompetenser som behövs för att implementera scenario 32.
	Krav 425311	Leverantören ska erbjuda information om hur lång tid och vilka kompetenser som behövs för att implementera scenario 33.

Appendix E. Tester

Testspecifikation

Revisionsinformation

Utfärdat av	Datum	Version	Ändringar
Siddhartha Pattni	2010-10-26	1.0	Baseline

Inledning

Detta dokument anger rutinerna för att försäkra att sökmotorlösningen som ska levereras uppfyller de krav specificerade i kravspecifikationen, se referens 1.

Dokumentet beskriver de tester som ska genomföras för att säkerhetsställa sökmotorlösningens kvalitet.

Referensdokument

[1] Kravspecifikation Version 1.0

Test

Testen ska täcka samtliga krav specificerade i kravspecifikationen.

Funktionstestspecifikation

Kravspecifikationens namn och version:

Funktionens namn: Sökning på olika språk

Detta funktionstest består av följande testfall:

- 1) En sökfråga formulerad med ett tangentbord på ett annat språk [41210]
- 2) En sökfråga formulerad med ett stoppord [41211]
- 3) En sökfråga formulerad med felstavat namn [41212]
- 4) En sökfråga formulerad med en synonym [41213, 4121, 4122, 4123, 4124, 4125, 4126, 4127, 4128]
- 5) Stöd för ”stemming” på samtliga språk [41214]
- 6) Sökning/indexering i flera filformat [41215]

Kravspecifikationens namn och version:

Funktionens namn: Rankning av sökresultat

Detta funktionstest består av följande testfall:

- 7) Rankning enligt fält [4131,4132]
- 8) Rankning enligt kontext [4133]
- 9) Rankning enligt säljstatistik[4134]
- 10) Rankning enligt användarkategori [4135]
- 11) Rankning enligt konfiguration [4136]

- 12) Rankning enligt användarhistorik [4137]
- 13) Rankning enligt antal träffar hos Google [4138]

Kravspecifikationens namn och version:

Funktionens namn: Konfiguration

Detta funktionstest består av följande testfall:

- 14) Konfigurering av sökkriterier [4141]
- 15) Konfigurering av manuell rankning [4142]
- 16) Konfigurering av indexering från externa källor [4143]
- 17) Konfigurering av facktermer och begrepp [4144]
- 18) Konfigurering av identifiering av populära produkter enligt Googles sökresultat [4145]

Kravspecifikationens namn och version:

Funktionens namn: Användarbeteende/statistik

Detta funktionstest består av följande testfall:

- 19) Statistik om sambandet mellan sökfrågor och användare [4161]
- 20) Användarbeteende i samband med sökfrågor [4162]
- 21) Statistik om olika sökfrågor [4163]

- 22) Användarhistorik [4164]
- 23) E-handelsspårning [4165]

Kravspecifikationens namn och version:
Funktionens namn: Externa källor

Detta funktionstest består av följande testfall:

- 24) Indexering från externa källor [4171]

Kravspecifikationens namn och version:
Funktionens namn: Uppdatering av index

Detta funktionstest består av följande testfall:

- 25) Manuell uppdatering av sökmotorns index [4181]

Kravspecifikationens namn och version:
Funktionens namn: Användarvänlighet

Detta funktionstest består av följande testfall:

- 26) Live-sökning [4211]
- 27) Användarvänlig sökfunktion för vanliga användare [4212]
- 28) Användarvänlig administration av sökmotorn [4213]
- 29) Kort systemutbildning för administratörer [4214]

- 30) En sökfråga formulerad med en boolesk operator [4215]
- 31) Aktuell dokumentation om API [4216]
- 32) Aktuell och fungerande exempelkod [4217]
- 33) C# eller java bibliotek [4218]
- 34) Faceted search [4219]
- 35) Sjävlärande [42110]
- 36) Proportionerlig omindexeringstid [42111]
- 37) Proportionell prismodell [42112]

Kravspecifikationens namn och version:

Funktionens namn: Utvidgbarhet

Detta funktionstest består av följande testfall:

- 38) Möjlighet för uppgradering av sökmotorns kapacitet [4221]
- 39) Möjlighet att aktivera automatiska uppdateringar [4222]
- 40) Hårdvarukrav [4223]
- 41) Stödja redundans [4224]
- 42) Stödja klustring [4225]

Kravspecifikationens namn och version:

Funktionens namn: Tillförlitlighet

Detta funktionstest består av följande testfall:

- 43) Stabil leverantör [4231]
- 44) Antal per license [4232]
- 45) Utvecklingsmiljö [4233]
- 46) Kostnader [4234]
- 47) Tillförlitlighet [4235]

Kravspecifikationens namn och version:

Funktionens namn: Prestanda

Detta funktionstest består av följande testfall:

- 48) Medelsvarstiden [4245]
- 49) Medelantal sökningar från x antal användare[4246]
- 50) Minsta indexeringstid [4247]
- 51) Minst 125 förfrågningar per sekund [42414]

52) Svarstid mindre än max enligt tabell[42415]

Kravspecifikationens namn och version:

Funktionens namn: Kostnader

Detta funktionstest består av följande testfall:

I licensavtalet framgår:

53) vilken produkt det avser [4251]

54) antal användare [4252]

55) utvecklingsmiljö 4253]

56) kostnader för leverantörens tjänster [4254]

57) leveranstid [4255]

58) kompetenskrav vid installation [4256]

59) kompetenskrav för underhåll [4257]

60) kompetenskrav vid sökmotorbyte [4258]

Kravspecifikationens namn och version:

Funktionens namn: Leverans

Detta funktionstest består av följande testfall:

- 61) Kompatibilitet med Microsoft Commerce Server [4261]
- 62) Dokumentation [4262]

Kravspecifikationens namn och version:

Funktionens namn: Support

Detta funktionstest består av följande testfall:

- 63) Supportens tillgänglighet [4271, 4274, 4275]
- 64) Sökmotorns backsystem [4272]
- 65) Stödwebbplats med dokumentation, FAQ och forum [4273]
- 66) Leverantören ska erbjuda support[4275]
- 67) Leverantören ska besvara frågor inom 24h [4276]

Testinstruktioner

1. Testinstruktioner för Lucene

Utfärdat av	Datum	Version	Ändringar
	2010-10-15	1.0	Baseline

Inledning

Detta dokument innehåller instruktioner för de testfall som specificerats i testspecifikationen.

Referensdokument

[1] Testspecifikationen Version1.0

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER: 1

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-22

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning och stemming på svenska.

Sökfråga: Hörlurarna

Svar:

Lucene stödjer sökning på svenska och stemming på svenska genom en klass Swedish Snowball Stemmer som dock har en del brister, söker man t.ex. på "kurs" stämms det felaktigt till "kur", medan t.ex. "kursen" eller "kurserna" stämms rätt till kurs.

<https://issues.apache.org/jira/browse/LUCENE-1515>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-01-19

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 2

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41371], [41372]

Sökmotorlösningen ska kunna ranka sökresultat.

Sökmotorlösningen ska ranka utifrån vilket fält attributet förekommer i.

Sökfråga: Silver

Svar:

Lucene: Enligt Find All funktionen i Excel som används för att hitta alla förekomster av ett specifikt ord finns det 256 celler i datamängden som innehåller ordet silver. Funktionen tar dock inte hänsyn till om "Silver" skrivs med stort eller litet s utan ger samma svar, vilket innebär att det finns en överhängande risk för felkällor.

Kolumnen Global_Color valdes att prioritera så att produkter som har ordet silver i denna kolumn kommer att rangordnas högre.

Prioritetsordning

Kategori

Antal
förekomster av

		silver
1	DisplayName	90
2	Global_ColorName	127
3	Variant Description	3
4	Global Specification	14
5	Global_MovieUnboxURL	12

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-20

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 3

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41711], [41463]

Sökmotorlösningen ska indexera upp information från externa källor.

Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att information indexeras upp från externa källor.

Testfallet har utgått.

TESTFALLET GODKÄNDES

GODKÄNT DEN

MJUKA TEST

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 4

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. """, +, och booleska operatorer så att användare med medel och höga datakunskaper ska kunna använda den.

Sökfråga: Nokia+7230

Svar:

Lucene stödjer AND, "+", OR, NOT och "-"

http://lucene.apache.org/java/1_4_3/queryparsersyntax.html

TESTFALLET GODKÄNDES GODKÄNT DEN 2010-12-20

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 5

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. """, +, och booleska operatorer så att användare med medel och höga datakunskaper ska kunna använda den.

Sökfråga: Nokia AND 7230

Svar:

Lucene stödjer AND, "+", OR, NOT och "-"

http://lucene.apache.org/java/1_4_3/queryparsersyntax.html

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-20

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 6

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. """
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: iPhone -4

Svar:

Lucene stödjer AND, "+", OR, NOT och "-"

http://lucene.apache.org/java/1_4_3/queryparsersyntax.html

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-20

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 7

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ””
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: iPhone NOT 4

Svar:

Lucene stödjer AND, "+", OR, NOT och "-"

http://lucene.apache.org/java/1_4_3/queryparsersyntax.html

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-20

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 8

UTFÖRDES AV

PÅBÖRJADES DEN 2010-
12-21

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning, stemming och särskrivning på svenska.

Sökfråga: Hör lur

Svar:

Klassen NGramTokenizer kan användas vid sökning på särskrivna ord. Genom att dela upp söksträngen i olika kombinationer t.ex. hö, ör, lu, ur.

http://lucene.apache.org/java/2_4_1/api/contrib-analyzers/org/apache/lucene/analysis/ngram/NGramTokenizer.html

Enligt examensarbete 30 hp januari 2010 Lingvistik och användbarhet i sökmotorer

Sven Vidén från Uppsala Universitet. Tillgängligt på: uu.diva-portal.org/smash/get/diva2:291812/FULLTEXT01

”Lucene av N-grams för att indexera hela denna lista. Nackdelen med just

denna variant är att den är något anpassad för det engelska spraket, vilket gör att den inte kan hantera sarskrivningar.”

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-01-17

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 9

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [41462]

Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.

Lucene har en metod `setBoost(float boost)` i klassen `Field`

Enligt javadoc för java API:

”Just as you can boost Documents, you can also boost individual Fields.”

[http://lucene.apache.org/java/2_0_0/api/org/apache/lucene/document/
Field.html](http://lucene.apache.org/java/2_0_0/api/org/apache/lucene/document/Field.html)

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-
21

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 10

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41641]

Sökmotorlösningen ska kunna lagra användarbeteende i samband med varje sökfråga.

Genom att spara undan tidpunkten då en sökfråga ställs, användarnamnet, tiden för att exekvera en sökfråga m.fl. kan slutsatser om användarbeteende dras.

I klassen Query Query.toString() kan man få reda på hur queryParsern har tolkat en sökfråga. Med metoden getSimpleName() i klassen Query kan man få reda på vilken typ sökfrågan är, t.ex. om det är en boolesk sökfråga.

Query.getClass().getSimpleName()

Query Query.toString()

<http://download.oracle.com/javase/1.4.2/docs/api/java/util/Calendar.html>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2010-12-21

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 11

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41643]

Sökmotorlösningen ska kunna lagra statistik om olika sökfrågor.

Sökfråga:

Svar:

Lucene:

Konstruktorn för klassen `QueryParser(String query, String field, Analyzer analyzer)` som tar en sökfråga, ett fält och en analyser vars syfte är att dela upp en sökfråga i termer som inparametrar.

används för att komma åt metoden `parse(query)` som får in en sökfråga som argument och returnerar ett `Query` objekt som `Searcher` klassens `search(query)` metod tar som inparameter.

Har man tillgång till sökfrågor kan de sparas undan och på så sätt erhålla statistik.

http://lucene.apache.org/java/2_4_0/api/org/apache/lucene/search/Sea

rcher.html#search%28org.apache.lucene.search.Query%29

http://lucene.apache.org/java/2_4_0/api/org/apache/lucene/queryParser/QueryParser.html

TESTFALLET GODKÄNDES GODKÄNT DEN 2010-12-21

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 12

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41644]

Sökmotorlösningen ska kunna lagra historik för varje användare.

Genom att spara undan tidpunkten en sökfråga ställs, användarnamnet, sökfrågan innan och efter tolkning, tiden för att exekvera en sökfråga m.fl. kan slutsatser om historik dras.

TESTFALLET GODKÄNDES

GODKÄNT DEN 010-12-21

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 13

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42531]

Det ska framgå tydligt i licensavtalet vilken produkt det avser.

Lucene: Available as Open Source software under the Apache

License

<http://lucene.apache.org/java/docs/features.html>

<http://www.apache.org/licenses/>

TESTFALLET GODKÄNDES GODKÄNT DEN 2010-12-20

2. Testinstruktioner för Sphinx

Utfärdat av	Datum	Version	Ändringar
	2011-02-01	1.0	Baseline

Inledning

Detta dokument innehåller instruktioner för de testfall som specificerats i testspecifikationen.

Referensdokument

[1] Testspezifikationen Version 1.0

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER: 1

UTFÖRDES AV PÅBÖRJADES DEN

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning och stemming på svenska.

Sökfråga: Hörlurarna

Svar:

Sphinx indexering har fullt stöd för SBCS (Single Byte Character Set) och UTF-8 kodning vilket gör att den har stöd för alla världens språk. Stemming på svenska är tillgängligt via snowballstemmer som är ett tredjepartsbibliotek.

<http://sphinxsearch.com/docs/1.10/features.html>

<http://sphinxsearch.com/about/sphinx/>

<http://www.linux.com/archive/feed/118721>

TESTFALLET GODKÄNDES GODKÄNT DEN 2011-02-01

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 2

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41371], [41372]

Sökmotorlösningen ska kunna ranka sökresultat.

Sökmotorlösningen ska ranka utifrån vilket fält attributet förekommer i.

Sökfråga: Silver

Svar:

Återstår att testas.

TESTFALLET GODKÄNDES GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 3

UTFÖRDES AV PÅBÖRJADES DEN

Testar krav: [41711], [41463]

Sökmotorlösningen ska indexera upp information från externa källor.

Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att information indexeras upp från externa källor.

Testfallet har utgått.

TESTFALLET GODKÄNDES GODKÄNT DEN

MJUKA TEST

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 4

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: Nokia+7230

Svar:

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 5

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-
20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: Nokia AND 7230

Svar:

Sphinx stödjer operatörn AND enligt:

Nokia & 7230

<http://sphinxsearch.com/docs/1.10/boolean-syntax.html>

TESTFALLET GODKÄNDES	GODKÄNT DEN 20 2011-02-01
----------------------	---------------------------

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 6

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. """
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: iPhone -4

Svar:

Sphinx stödjer operatören NOT enligt:

iPhone -4

iPhone !4

<http://sphinxsearch.com/docs/1.10/boolean-syntax.html>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-01

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 7

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: iPhone NOT 4

Svar:

Sphinx stödjer operatörn NOT enligt:

iPhone -4

iPhone !4

<http://sphinxsearch.com/docs/1.10/boolean-syntax.html>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-01

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 8

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning och stemming på svenska.

Sökfråga: Hör lur

Svar:

För att hantera särskrivningar kan man använda NGRAM som splittar ordet hörlur i flera bokstavskombinationer tex Hö, ör, lu, och ur. Däremot stödjer Sphinx bara Kinesiska, Japanska och Koreanska. I skrivande stund är endast ngram_len = 1 är implementerat vilket innebär att ”hör lur” hade indexerats som ”H ö r l u r”. Bara de tecken (chars), sekvenserna som är listade i ngram_chars tabellen tas med.

<http://sphinxsearch.com/docs/current.html#conf-ngram-len>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 9

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41462]

Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 10

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41641]

Sökmotorlösningen ska kunna lagra användarbeteende i samband med varje sökfråga.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 11

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41643]

Sökmotorlösningen ska kunna lagra statistik om olika sökfrågor.

Sökfråga:

Svar:

Metoden Query (\$query, \$index="*", \$comment="") returnerar en mängd resultat som innehåller de matchade resultaten och en del statistic om sökfrågor. Mängden är en PHP specifik hash med följande nycklar och värden.

"matches":

Hash som matchar document ID mot en hash innehållande vikt (weight) och andra valda attribut.

"total":

Totalt antal matchade document som skall hämtas från servern för en sökfråga.

"total_found":

Totalt antal matchade document i index.

"words":

Hash som mappar nyckelord för sökmotorer till en hash medstatistik ("docs", "hits").

"error":

Feldmeddelande från searchd enligt (string, feldmeddelande).
Tomt om det inte finns någon fel.

"warning":

Varningsmeddelande från searchd enligt (string, varningsmeddelande). Tomt om det inte finns någon varning.

Om SphinxQL som är en så kallad SQL dialect som har all funktionallitet som searchd använder genom att använda standard SQL syntax med några Sphinx specifika tillägg används kan

SHOW META shows additional meta-information about the latest query such as query time and keyword statistics:

```
mysql> SELECT * FROM test1 WHERE MATCH('test|one|two');
```

```
mysql> SHOW META;
```

<http://sphinxsearch.com/docs/current.html#sphinxql-show-meta>

<http://sphinxsearch.com/docs/current.html>

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 12

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41644]

Sökmotorlösningen ska kunna lagra historik för varje användare.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN 2011-02-04

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 13

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42531]

Det ska framgå tydligt i licensavtalet vilken produkt det avser.

Sphinx är distribuerad under GPLv2 (The GNU General Public License version 2) vilket tillåter både kommersiell och icke-kommersiell användning.

<http://sphinxsearch.com/about/sphinx/>

TESTFALLET GODKÄNDES GODKÄNT DEN 2011-02-01

3. Testinstruktioner för SQL Full Text Search

Revisionsinformation

Utfärdat av	Datum	Version	Ändringar
	2010-10-15	1.0	Baseline

Inledning

Detta dokument innehåller instruktioner för de testfall som specificerats i testspecifikationen.

Referensdokument

[1] Testspezifikationen Version 1.0

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER: 1

UTFÖRDES AV PÅBÖRJADES DEN

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning och stemming på svenska.

Sökfråga: Hörlurarna

Svar:

Det finns en inbyggd funktion i MS SQL Server som söker på grundformen av ett ord. Funktionen FREETEXT kan användas i samband med SELECT då SQL Server söker på grundformen av ett ord i ett specifikt språk. Följande SQL-kommandot söker på grundformen av ordet "hörlurarna" och ger träff på kolumner i databasen som innehåller "Sony Ericsson HPM-88 hörlur med bullerreducering".

```
select * from Test_Table_2 where  
freetext(Title,'hörlurarna', LANGUAGE 1053)
```

Källor:

How to disable word stemming in SQL Server Full Text Search:
<http://stackoverflow.com/questions/545322/how-to-disable-word-stemming-in-sql-server-full-text-search>

sys.syslanguages (Transact-SQL): <http://msdn.microsoft.com/en-us/library/ms190303.aspx>

FREETEXT (Transact-SQL): <http://msdn.microsoft.com/en-us/library/ms176078.aspx>

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 2

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41371], [41372]

Sökmotorlösningen ska kunna ranka sökresultat.

Sökmotorlösningen ska ranka utifrån vilket fält attributet förekommer i.

Sökfråga: Silver

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 3

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41711], [41463]

Sökmotorlösningen ska indexera upp information från externa källor.

Sökmotorlösningen ska erbjuda möjligheten att konfigurera så att

information indexeras upp från externa källor.

Testfallet har utgått.

TESTFALLET GODKÄNDES GODKÄNT DEN

MJUKA TEST

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 4

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””
,+, och booleska operatorer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: Nokia+7230

Svar:

Det finns stöd för bolska termer i SQL Server. Man kan använda
AND, OR och NOT i samband med SELECT-kommandot. Följande
koden ger träff på ” Nokia 7230 - prisvärd designmobil”.

```
select * from Test_Table_2 where contains(Title, 'Nokia')  
and contains(Title, '7230')
```

Man kan impletera en metod som gör om en boolsk sökfråga ett
boolskt SQL-kommando.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 5

UTFÖRDES AV

PÅBÖRJADES DEN 2010-12-20

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””
,+, och booleska operatörer så att användare med medel och höga
datakunskaper ska kunna använda den.

Sökfråga: Nokia AND 7230

Svar:

Det finns stöd för bolska termer i SQL Server. Man kan använda
AND, OR och NOT i samband med SELECT-kommandot. Följande
koden ger träff på ” Nokia 7230 - prisvärd designmobil ”.

```
select * from Test_Table_2 where contains(Title,'Nokia')  
and contains(Title,'7230')
```

Man kan impletera en metod som gör om en boolsk sökfråga ett
boolskt SQL-kommando.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 6

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””, +, och booleska operatörer så att användare med medel och höga datakunskaper ska kunna använda den.

Sökfråga: iPhone -3Gs

Svar:

Det finns stöd för booleska termer i SQL Server. Man kan använda NOT i samband med SELECT-kommandot. Följande koden ger träff på produkter som heter ”iPhone” och inte innehåller 4 i kolumnen Title: ” Apple iPhone 4 16GB Black Telia-låst (3G) ”

```
select * from Test_Table_2 where  
contains(Title,'iPhone') AND NOT contains(Title,'3gs')
```

Man kan impletera en metod som gör om en boolsk sökfråga ett boolskt SQL-kommando.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 7

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42115]

Sökmotorlösningen ska stödja etablerade sökfunktioner som t.ex. ”””, +, och booleska operatörer så att användare med medel och höga datakunskaper ska kunna använda den.

Sökfråga: iPhone NOT 3GS

Svar:

Det finns stöd för booleska termer i SQL Server. Man kan använda NOT i samband med SELECT-kommandot. Följande kod ger träff på produkter som heter ”iPhone” och inte innehåller 4 i kolumnen Title: ” Apple iPhone 4 16GB Black Telia-låst (3G) ”

```
select * from Test_Table_2 where  
contains(Title,'iPhone') AND NOT contains(Title,'3gs')
```

Man kan impletera en metod som gör om en boolsk sökfråga ett boolskt SQL-kommando.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 8

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41211], [412114, svenska]

Sökmotorlösningen ska klara sökning och stemming på svenska.

Sökfråga: Hör lur

Svar:

Det finns ingen information om en metod som kan klara av att söka på särskrivningar men det finns möjligheter för vidare utveckling av dessa metoder.

<http://www.sqlservercentral.com/Forums/Topic476575-149-1.aspx>

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 9

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41462]

Sökmotorlösningen ska erbjuda möjligheten att konfigurera manuell rankning (force position) för specifika produkter.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 10

UTFÖRDES AV PÅBÖRJADES DEN

Testar krav: [41641]

Sökmotorlösningen ska kunna lagra användarbeteende i samband med varje sökfråga.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 11

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41643]

Sökmotorlösningen ska kunna lagra statistik om olika sökfrågor.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 12

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [41644]

Sökmotorlösningen ska kunna lagra historik för varje användare.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN

SYSTEMTESTINSTRUKTIONER

TESTFALLETS NUMMER 13

UTFÖRDES AV

PÅBÖRJADES DEN

Testar krav: [42531]

Det ska framgå tydligt i licensavtalet vilken produkt det avser.

Framgår inte i dokumentationen.

TESTFALLET GODKÄNDES

GODKÄNT DEN

Appendix F. Enkät svar

Vilka sökmotorlösningar känner du till?	Vilka sökmotorer har du använt i projekten?	Vad var bra med de sökmotorerna?	Vad var minde bra?	Någon funktion du saknade hos sökmotorn/sökmotorerna?	Övriga kommentarer?
Google, Bing, ESales, Lucene, Microsoft Fast, Google Fast Theca FactFinder Lucene Avail	ESales Lucene FactFinder Lucene Inga SQL Server Fulltext Search Avail Google Mini	ESales: prestanda och flexibilitet Lucene: gratis och anpassningsbart FactFinder: bra backoffice för tuning extra funktioner tex predications m.m Google Mini: är billig och lite snabbare att konfigurera SSFTS: följer med SQL Server, integrerad med Commerce	Lucene: mycket men svår dokumentation och referensmateria Svårt att veta om de är rätt uppsätta SSFTS: endast grundläggande sök-funktionalitet Avail: dyr Google Mini klarar inte av att sortera mer än på datum (senast ändrad) och relevans. För Minin är allting "dokument", den förstår inte objekt med properties	Nej nej, bra täckning i båda "menade du" ... Ajax förslag av i sökrutan, som i Google ja de jag nämner ovan. Commerce Servers egen kan jag inte tillräckligt bra.	Jag har inte utvecklat mot Lucene mer än att det används i våra nuvarande projekt. Endast jobbat med det som projektledare, inte tekniskt ansvarig

Apptus		Server	såsom produkter, varumärken eller kategorier. Pagineringen blir också konstig med Minin eftersom man inte får ett tydligt svar från Minin på max antal träffar. man får bara veta ungefär hur många träffar man fick på sökningen. Sen är det ett problem att det är en maskin från Google med begränsad admin- möjligheter. Man kan bara komma åt Minin via ett webbgränssnitt , så felsökning kan ibland vara lite svårt.		
SQL Server Fulltext Search (SSFTS)		Avail: extra funktioner t ex predications m.m. Google Mini: är billig och lite snabbare att konfigurera			

Appendix G. Första urvalet

Sökmotorer med öppen källkod

[illegible]

Kommersiella sökmotorer

Criteria	Oracle Secure Enterprise Search	MS FAST	Dissepoint	MS_Search_Server_2010_Express	MS_Search_Server_2010	Google Search Appliance	Autonomy IDOL Server	eComm Search	Apptus	MS FULL TEXT SEARCH	Adobe SiteSearch	Thunderstone Webinator	Google Site Search	Google Mini	EasyAsk eCommerce Edition	Siderean Seamark
Language	All	40+	40	51	22	27	100+	.	All	Multi	Multi	Multi	Multi	28	.	.
Did you mean	.	☒	☒	☒	☒	☒	.	☒	☒	.	☒	.	.	☒	.	☒
Boolean Search	☒	☒	☒	☒	☒	☒	☒	.	.	.	☒	.	.	☒	.	.
Faceted Search	.	☒	☒	☒	☒	☒	.	.	.	.	.	.	.	☒	.	.
Relevance Ranking	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	.	☒	☒	☒
Platform compability	All	.	Cross	Win	Win	.	Win	.	.	Win	.	.	.	☒	.	EPISer
Data format compatibility	150+	400	6+	31	.	220	1000	.	.	15+	15+	100+	6+	All	☒	.
Database indexing	☒	☒	☒	.	☒	☒	☒	.	.	☒	.	.	.	.	.	.
Categorization	☒	☒	☒	.	manag	.	☒	☒	.	☒	☒	.	☒	.	.	☒
Exact Match	.	☒	☒	☒	.	.	.	.	.	☒	.	.	.	☒	.	.
Fuzzy Matching	☒	☒	☒	.	.	.	☒	☒	.	☒	.	☒	.	No	.	.
Index persistence	Disk	.	.	SQL	SQL	.	.	.	.	SQL	.	.	.	.	.	.
Multi-Site Search	☒	☒	☒	☒	☒	☒	.	.	☒	.	.	☒	☒	☒	.	.
More like this	☒	.	.	☒	☒	.	.	.	.	.	.	☒	.	.	.	.
Soundex	.	☒	.	.	☒	.	.	☒	.	.	☒	.	.	.	.	.
Wildcards	☒	☒	☒	☒	☒	.	☒	.	.	.	.	.	☒	.	.	.
Metaphone	.	.	.	.	.	.	.	☒	.	.	.	.	.	.	.	.
Acronym	.	☒	.	☒	☒	☒	.	.	.	.	☒	.	.	.	.	.
Synonyms	☒	☒	☒	.	.	.	☒	☒	☒	☒	☒	.	☒	☒	.	.
Stemming	☒	☒	☒	.	.	☒	☒	☒	☒	☒	.	.	☒	.	.	☒
Natural Language Processing (NLP)	☒	.	.	.	.	.	☒	.	.	.	.	☒	.	.	☒	.
Ontology	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
Partial Word Matching	.	.	.	.	.	.	.	☒	.	.	.	.	.	.	.	.
Phrase Searching	☒	☒	☒	☒	☒	☒	☒	.	.	☒	.	.	.	☒	.	.
Proximity Searching	☒	☒	.	.	.	.	☒	.	.	☒	.	☒	.	.	.	.
Results List	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	.	☒
Score	☒	☒	.	.	☒	☒	.	.	.	☒	☒	.	☒	.	.	.
Sorting Results	☒	.	☒	☒	.	☒	.	☒	☒	☒	☒	.	☒	.	.	☒
Thesaurus	☒	.	☒	.	.	.	☒	.	.	☒	.	.	.	.	.	.
Vertical Portals	☒	.	.	.	.	☒	.	.	.	.	.	.	.	.	.	.
Search Zones	☒	☒	.	.	.	.	.	.	.	.	.	☒	.	.	.	.
Last release	R. 11.1	.	.	.	.	V. 6	.	.	.	R. 2	V. 6.0	V. 5.1	.	.	.	.
Click tracking	☒	☒	☒	.	.	☒	☒	☒	☒	.	☒	☒	☒	.	☒	☒
Link tracking	☒	☒	.	.	.	☒	.	.	.	.	.	☒	.	.	.	.
Configuration	☒	☒	☒	☒	☒	☒	☒	.	☒	.	.	.	☒	☒	☒	☒
Autocomplete	☒	☒	☒	☒	☒	☒	.	☒	☒	☒	☒	.	☒	.	.	.
API	☒	☒	☒	☒	☒	☒	☒	☒	.	☒	.	.	☒	.	.	☒
Programming Language	.	.	Java	.	.	.	.	.	Java	.	.	.	.	.	.	.
Web crawler	☒	☒	☒	☒	.	.	.	.	.	☒	.	.	.	.	.	.
Simultaneous update and searching	☒	☒	☒	☒	.	.	☒	.	☒	.	☒	☒	.	.	.	.
Available Documentation	Online	.	FAQ	TechN	TechN	Online	.	.	.	☒	.	.	Online	Online	.	.
Technical Support packages	☒	.	.	☒	.	.	.	.	.	.	.	.	☒	☒	.	.
Number of documents	1M	.	.	10M	10M	10M	.	.	.	.	Unlimi	200k	500k	300k	.	.
Fees	.	245,668 kr per processor	.	Free	.	.	.	.	.	.	.	10K docs Free, 20K docs \$700	10K docs 3,500€, 20K docs 6000€, 50K \$250	100K docs 3,500€	.	.
Points	26	24	17	16	14	14	12	12	11	11	10	9	9	9	5	6

Appendix H. Installationsmanualer

Installationsguide för Lucene

Gå in på Lucenes hemsida för nerladdning på adressen:

<http://www.apache.org/dyn/closer.cgi/lucene/java/>

som vidarebefordrar en till

<http://apache.dataphone.se//lucene/java/>

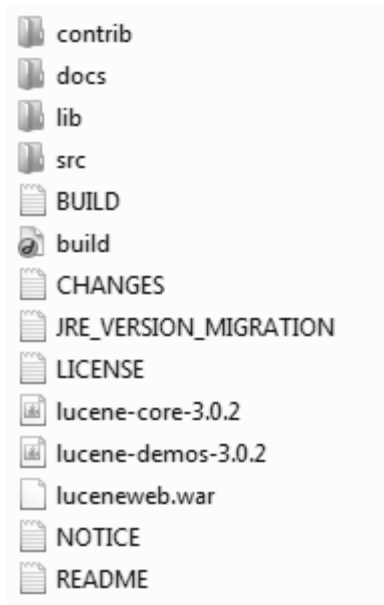
där hittar man de senaste versionerna i både .zip- och .tar format. Där framgår datum, tid och storlek för filerna.

	lucene-3.0.2.tar.gz Java-Apache (old)	17-Jun-2010 15:48	17M
	lucene-3.0.2.tar.gz.asc Java-Apache (old)	17-Jun-2010 15:48	831
	lucene-3.0.2.zip Java-Apache (old)	17-Jun-2010 15:48	25M
	lucene-3.0.2.zip.asc Java-Apache (old)	17-Jun-2010 15:48	831

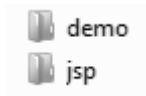
I detta exempel valdes att installera den senaste versionen, v.3.0.2.zip och efter att ha laddat ner filen får man en mapp som visar Lucene och versionsnummer.

 lucene-3.0.2

När mappen packas upp får man en mapp med följande struktur.



I mappen src som står för source finns följande två mappar, demo och jsp. Demomappen innehåller demofiler och jsp-mappen är för Lucenes webb-demo och behandlas inte här.



För att kompilera Lucene demo från utvecklingsmiljön Eclipse behöver man utföra följande steg, förutsatt att man har valfri version av Eclipse och JDK eller SDK installerat.

- Skapa ett nytt projekt i Eclipse genom menyn File->New->Java Project->Finish
- Kopiera över innehållet i demo mappen till src-mappen i det nyskapade projektet.

För att kunna kompilera behöver två jar-filer läggas till projektet lucene-core-3.0.2.jar och lucene-demos-3.0.2 som ligger i lucene mappen.

- Högerklicka på projektmappen->Välj properties->Java Build Path-> Add External Jars
markera lucene-core-3.0.2.jar och lucene-demos-3.0.2 som ligger i lucene mappen och tryck på open. Jarfilerna läggs då till.

För att indexera alla filer i en mapp används filen IndexFiles.java

- Högerklicka på IndexFiles->Run As -> Run Configurations->Arguments-> fyll i den fullständiga sökvägen till mappen med filer som skall indexeras tryck sedan på Apply->Run
En ny mapp med namnet index skapas då automatiskt i ditt projekt och i konsoll (concole) fönstret framgår information om vilka filer som har indexerats.

Nu kan filen SearchFiles köras genom att:

- Högerklicka på SearchFiles.java->välj Run As -> Java Application

På webbplatsen youtube finns två användbara videoguider om hur man stegvis går till väga för att installera och börja använda Lucene ihop med Eclipse miljön.

Getting Started with Lucene and Eclipse

<http://www.youtube.com/watch?v=KHvmHhyy-zg>

SettingUp-Lucene-Eclipse-part1

http://www.youtube.com/watch?v=os0-Np_rbGg&feature=related

Installationsguide för Sphinx

Sphinx finns att ladda hem på deras hemsida <http://sphinxsearch.com/downloads/>, väljer man den senaste versionen får man fram en lista där versionsnummer och storlek framgår.

Här valdes sphinx-1.10-beta-win32-id64-full.zip genom att välja Win32 binaries w/MySQL+PostgreSQL+libstemmer+id64 och trycka på download.

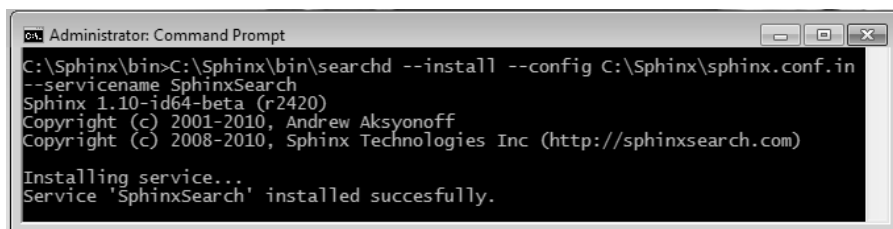
Sphinx 1.10-beta (r2420; Jul 19, 2010)			
Source tarball (tar.gz)	1.10-beta	1.4M	Download
Win32 binaries w/MySQL support	1.10-beta	3.1M	Download
Win32 binaries w/MySQL+PostgreSQL support	1.10-beta	4.6M	Download
Win32 binaries w/MySQL+PgSQL+libstemmer+id64 support	1.10-beta	4.9M	Download
RHEL/CentOS 5.x x86_64 RPM	1.10-beta	3.1M	Download
RHEL/CentOS 5.x i386 RPM	1.10-beta	3.6M	Download
RHEL/CentOS 4.x i386 RPM	1.10-beta	3.5M	Download
RHEL/CentOS 5.x x86_64 binaries	1.10-beta	3.1M	Download
RHEL/CentOS 5.x i386 binaries	1.10-beta	3.6M	Download
RHEL/CentOS 4.x i386 binaries	1.10-beta	3.5M	Download

Öppna en kommandoprompt i Windows genom tex skriva cmd i start -> lokalisera var Sphinxmappen är och lägg dig i /bin genom att skriva C:\addres till Sphinx\bin>

För att installera windows service SphinxSearch.

Skriv C:\Sphinx\bin\searchd --install --config C:\Sphinx\sphinx.conf.i
--servicename SphinxSearch

Vilket resulterar I följande resultat.



```
Administrator: Command Prompt
C:\Sphinx\bin>C:\Sphinx\bin\searchd --install --config C:\Sphinx\sphinx.conf.in
--servicename SphinxSearch
Sphinx 1.10-id64-beta (r2420)
Copyright (c) 2001-2010, Andrew Aksyonoff
Copyright (c) 2008-2010, Sphinx Technologies Inc (http://sphinxsearch.com)
Installing service...
Service 'SphinxSearch' installed succesfully.
```


Appendix J. Antal träffar på Altavista

Name	AltavistaHits
Site Search 2	587000000
FAST Data Search	66100000
ICE	61700000
Java Search Engine	60500000
Google Mini	34200000
Zoom Search Engine	31800000
Google Custom Search and Site Search (Business)	23400000
Subject Search Server	17000000
ic-find	14200000
Microsoft Index Server	13100000
Microsoft Index Server	13100000
Phantom	12400000
ARTS PDF Search	11900000
ARTS PDF Search	11900000
Boolean Search	11500000
Microsoft Search Server 2010 Express	10900000

crawl-it	9430000
Net.Sprint	7580000
Full text search SQL-Server	6560000
Zebra	5840000
The Clever Project	5800000
DarWin SET	5660000
Microsoft Sharepoint Server 2010	4790000
Blossom	4050000
Pan-Enterprise Search	3750000
Fluid Dynamics Search Engine	2790000
Fluid Dynamics Search Engine	2790000
Terrier	2640000
Swish++	2530000
Sphinx	2440000
Fluffy Search	2340000
Simple Site Searcher & Super Site Searcher	2030000
80-20 Document and Records Management Solution	1860000
Lucene	1700000
Sandy Bay Results Engine	1670000
FreeFind	1500000

Java Search Tools Product Listings	1480000
URL Spider Pro	1370000
Universal Knowledge Processor	1370000
Minion	1320000
Sphinx SQL & Full-text Search Engine	1230000
Web Server 4D Site-Search	1050000
Lucene.net	997000
Autonomy Enterprise Search	922000
IMP Database Search Engine	863000
Namaz	834000
Warning: this service has been discontinued	777000
Swish-e	678000
CI Vertical Search	593000
mnoGoSearch	590000
XRS	577000
AltaVista Enterprise Search	546000
ht://Dig	537000
Perl Search Tools Product Listings	446000
MPS Information Server	429000
Alkaline Search Engine Server	419000

WAIS	315000
S.L.I Systems Learning Search	204000
Harvest web crawler	193000
S.L.I. Systems Learning Search System	185000
BizNetic	174000
Isearch	169000
WebWorks Search	165000
Simple Search (NMS)	161000
Simple Search (NMS)	161000
Perlfect	148000
ELISE Smart Search and Match	141000
MS Enterprise Search (FAST ESP)	139000
Thunderstone	136000
WebSTAR Search	132000
JXTA Search	116000
Sinequa	107000
Panoptic Search	93300
Xapian	91600
Isys:web	88500
PicoSearch	88200

yacy	67800
Wumpus	63100
Sphider	52600
Ultraseek	52200
Ultraseek	52200
Albert Searcher	49700
Open Text Livelink Search	48400
Open Text LiveLink	48300
Galago	47900
InteractiveTools Search Engine	46300
WebSearch Perl Script	45900
YourAmigo	44500
Brainware	44200
DataparkSearch	43200
YourAmigo Site Search	42100
Nutch (open source web-scalable search engine)	39800
Searchbutton	38700
Google Search Appliance (GSA) and Mini Search Appliance	37500
InQuira for Search	35600

Atrise Everyfind	33700
Coveo Enterprise Search	31700
suntek	31700
ASPSeek	30300
OpenBridge Search & Navigation Engine (formerly ZNow)	25300
WebGlimpse	24300
Plumtree Portal Search Server	22900
Virage Product Suite	20900
SharePoint (Microsoft Tahoe)	20600
SharePoint (Microsoft Tahoe)	20600
SharePoint (Microsoft Tahoe)	20600
Extropia Site Search	20500
Extropia Site Search	20500
FusionBot	17800
Solr Open Source Search and Faceted Metadata Search Server	17500
KSearch	16800
DMP Scout	12900
Searchengine.com SiteSearch	12600

IBM OmniFind Enterprise Search, Yahoo! Edition	12000
IBM DB2 Intelligent Miner for Text	11900
Q-go	11700
Verity K2 and other Search Products	11400
Verity K2 and other Search Products	11400
Excerpt (Alma Mater Software)	11200
OpenFTS	10400
iPhrase	10300
MondoSearch	10200
MondoSearch	10200
linkSearch	9990
Infoglide	9870
Verticrawl	9630
Oracle Text, Ultra Search and interMedia Search	8020
Oracle Text, Ultra Search and interMedia Search	8020
Oracle Text, Ultra Search and interMedia Search	8020
Egothor	3520
Progress EasyAsk	3490
SiteMiner	2940
MG4J	2800

Zettair	2410
siteLevel (whatUseek)	2190
siteLevel (whatUseek)	2190
siteLevel (whatUseek)	2190
Northern Light (SinglePoint) Search	1780
focuseek searchbox	1750
Hounder	1680
OpenWebSpider	1420
XYZFind	1360
Endeca Faceted Metadata Search and Browse	1220
Amberfish	1210
Apptus	1160
LexiQuest	1070
iHound	1030
Onix Full Text Indexing and Retrieval Toolkit	954
Inxight SmartDiscovery	912
Inxight SmartDiscovery	912
Ontolica Search for Sharepoint/Search Server	910
OpenPipeline	894
SiteSearch Indexer (Windows/JavaScript)	775

SurfMap Search	741
Hummingbird SearchServer	665
Hummingbird SearchServer	665
Methabot	653
WebMerger	603
SeekIt Site Search	593
SeerSuite	590
Magnifi Enterprise Server	577
Extense Station Pro / Extense On Line	575
WideSource Peer Search	569
Indiri	520
ZyIndex	502
NQL (Network Query Language)	445
Visual Net from Antarcti.ca	445
Educe-ASP Search Engine (was Educesoft ASP Search Engine)	419
Xavatoria Index Search	403
ASTAWare SearchKey PRO	398
Webrom	377
Mercado IntuiFind	367

Mercado IntuiFind	367
Intrafind Search+	361
Delphes Knowledge Management System	346
Cybotics Search Engine	339
Blue Angel MetaStar	325
BDDBot	325
Noviforum Search	309
Noviforum Search	309
t.find (Eidetica)	289
WebSideStory Search (formerly Atomz)	268
InMagic DB/Text Spider	257
Nathra	255
Spiderline - Remote Search Service	252
TxSearch	239
Siderean Seamark Faceted Metadata Search	239
Folio siteDirector	234
iPlanet Portal (was Netscape Compass Server)	234
Engenium Semetric	231
SiteFerret	221
orange (empolis)	221

PinPoint (NetCreations)	205
OpenSearchServer	199
RightSearch	197
Microsearch WebSearch	197
RuterSearch	182
(formerly ZNow)	182
WizDoc	179
AskJeeves - JeevesOne Search & Enterprise	166
Intelligenx Discovery Engine (was i411)	129
Intelligenx Discovery Engine (was i411)	129
Fabasoft Mindbreeze	123
FindinSite (formerly Spy-Server)	112
FindinSite (formerly Spy-Server)	112
Juggernautsearch Engine	90
ebhath	89
CGISRCH - AnalogX CGI Search Engine	81
Orangevalley Intranet Search Engine	75
SiteSearchPro	72
Dieselpoint Faceted Metadata Search and Browse	70
Nav4 from Think Tank 23	68

Siteforum DocFather (SFS-Software)	58
Siteforum DocFather (SFS-Software)	58
ReachCast ReachSite	49
EWS (Excite for Web Servers) - obsolete	47
Recommind MindServer IR	46
IXE: Ideare indeXing Engine	40
F3DSearch	33
Intrafind iFinder Enterprise Search	31
Smartlogik Discover (APR)	31
LivePublish NextPage NXT 3 Search Engine	31
MiniSearch (SpaceSurfer)	30
IIC I-Search	30
BBDBot	28
PLWeb Turbo (PLS / AOL)	25
JHLSearch	21
ODARS4Search	19
Sharewire SiteSearch	17
ActiveSearch SiteSearch SDK	14
Datagold Archiving Services	9
TYPENGO N300 Enterprise Search (from FrontLogic)	9

XML Query Engine, aka XQEngine	8
Cogia Provides 360° Technologies	5
ITMS (J-Space) Search	5
IB 2.0 WebCat (InterBasis BSn)	3
Arexera X-IAS Internet Archive Search	1
Everyfind (Atrise) (formerly VirTime Everywhere Search)	1
Arexera X-IAS Internet Archive Search	1
PortalJuice (nextopia) ECommSearch	0

Appendix K. Testresultat

Lucene

num_ret	222	131
num_rel	222	131
num_rel_ret	222	131
map	222	0,7172
ndcg	222	0,6647
ndcg15	222	0,2956
R-prec	222	1
bpref	222	0
recip_rank	222	0,0156
P5	222	0
P10	222	0
P15	222	0
P20	222	0
P30	222	0
P100	222	1,31
P200	222	0,655
P500	222	0,262
P1000	222	0,131
num_ret	all	131
num_rel	all	131
num_rel_ret	all	131
map	all	0,7172
ndcg	all	0,6647

R-prec	all	1
bpref	all	0
recip_rank	all	0,0156
P5	all	0
P10	all	0
P15	all	0
P20	all	0
P30	all	0
P100	all	1,31
P200	all	0,655
P500	all	0,262
P1000	all	0,131

Tabell 19: Testresultat för Lucene från IReval

SQL Full Text Search

num_ret	test_nr.2	500
num_rel	test_nr.2	474
num_rel_ret	test_nr.2	71
map	test_nr.2	0,0688
ndcg	test_nr.2	0,1584
ndcg15	test_nr.2	0
R-prec	test_nr.2	0,1498
bpref	test_nr.2	0,0292
recip_rank	test_nr.2	0,0069
P5	test_nr.2	0
P10	test_nr.2	0
P15	test_nr.2	0
P20	test_nr.2	0
P30	test_nr.2	0
P100	test_nr.2	0
P200	test_nr.2	0,34
P500	test_nr.2	0,142
P1000	test_nr.2	0,071
num_ret	test_nr.2	500
num_rel	test_nr.2	77
num_rel_ret	test_nr.2	50
map	test_nr.2	1,0936
ndcg	test_nr.2	0,6617
ndcg15	test_nr.2	0,1706

R-prec	test_nr.2	0,6494
bpref	test_nr.2	0,4748
recip_rank	test_nr.2	1
P5	test_nr.2	0,2
P10	test_nr.2	0,7
P15	test_nr.2	0,4667
P20	test_nr.2	0,35
P30	test_nr.2	0,3333
P100	test_nr.2	0,5
P200	test_nr.2	0,25
P500	test_nr.2	0,1
P1000	test_nr.2	0,05
num_ret	test_nr.2	500
num_rel	test_nr.2	10
num_rel_ret	test_nr.2	10
map	test_nr.2	0,1041
ndcg	test_nr.2	0,3862
ndcg15	test_nr.2	0
R-prec	test_nr.2	0
bpref	test_nr.2	1
recip_rank	test_nr.2	0,0526
P5	test_nr.2	0
P10	test_nr.2	0
P15	test_nr.2	0
P20	test_nr.2	0,05
P30	test_nr.2	0,0333

P100	test_nr.2	0,1
P200	test_nr.2	0,05
P500	test_nr.2	0,02
P1000	test_nr.2	0,01
num_ret	all	1500
num_rel	all	561
num_rel_ret	all	131
map	all	0,4222
ndcg	all	0,4021
R-prec	all	0,2664
bpref	all	0,5013
recip_rank	all	0,3532
P5	all	0,0667
P10	all	0,2333
P15	all	0,1556
P20	all	0,1333
P30	all	0,1222
P100	all	0,2
P200	all	0,2133
P500	all	0,0873
P1000	all	0,0437

Tabell 4: Testresultat för SQL FTS från IReval