

REKOMMENDATIONSMOTOR: – MED FOKUS INOM E-LÄRANDE

Examensarbete Systemarkitekturutbildningen

Lennart Jakobsson (S133058)
Thires Nilsson (S143979)

2018KSAI01



HÖGSKOLAN
I BORÅS

Systemarkitekturutbildningen är en kandidatutbildning med fokus på programutveckling. Utbildningen ger studenterna god bredd inom traditionell program- och systemutveckling, samt en spets mot modern utveckling för webben, mobila enheter och spel. Systemarkitekten blir en tekniskt skicklig och mycket bred programutvecklare. Typiska roller är därför programmerare och lösningsarkitekt. Styrkan hos utbildningen är främst bredden på de mjukvaruprojekt den färdige studenten är förberedd för. Efter examen skall systemarkitekter fungera dels som självständiga programutvecklare och dels som medarbetare i en större utvecklingsgrupp, vilket innebär förtroendet med olika arbetssätt inom programutveckling.

I utbildningen läggs stor vikt vid användning av de senaste teknikerna, miljöerna, verktygen och metoderna. Tillsammans med ovanstående teoretiska grund innebär detta att systemarkitekter skall vara anställningsbara som programutvecklare direkt efter examen. Det är lika naturligt för en nyutexaminerad systemarkitekt att arbeta som programutvecklare på ett stort företags IT-avdelning, som en konsultfirma. Systemarkitekten är också lämpad att arbeta inom teknik- och idédrivna verksamheter, vilka till exempel kan vara spelutveckling, webbapplikationer eller mobila tjänster.

Syftet med examensarbetet på systemarkitekturutbildningen är att studenten skall visa förmåga att delta i forsknings- eller utvecklingsarbete och därigenom bidra till kunskapsutvecklingen inom ämnet och avrapportera detta på ett vetenskapligt sätt. Således måste de projekt som utförs ha tillräcklig vetenskaplig och/eller innovativ höjd för att generera ny och generellt intressant kunskap.

Examensarbetet genomförs vanligen i samarbete med en extern uppdragsgivare eller forskningsgrupp. Det huvudsakliga resultatet utgörs av en skriftlig rapport på engelska eller svenska, samt eventuell produkt (t.ex. programvara eller rapport) levererad till extern uppdragsgivare. I examinationen ingår även presentation av arbetet, samt muntlig och skriftlig opposition på ett annat examensarbete vid ett examinationsseminarium. Examensarbetet bedöms och betygssätts baserat på delarna ovan, specifikt tas även hänsyn till kvaliteten på eventuell framtagen mjukvara. Examinator rådfrågar handledare och eventuell extern kontaktperson vid betygssättning.



HÖGSKOLAN
I BORÅS

BESÖKSADRESS: JÄRNVÄGSGATAN 5 · POSTADRESS: ALLÉGATAN 1, 501 90 BORÅS
TFN: 033-435 40 00 · E-POST: INST.HIT@HB.SE · WEBB: WWW.HB.SE/HIT

Svensk titel: Rekommendationsmotor: med fokus inom E-lärande

Engelsk titel: Recommendation engine: focus within E-learning

Utgivningsår: 2017

Författare: Lennart Jakobsson & Thires Nilsson

Handledare: Tuwe Löfström

Abstract

Studies regarding recommendation engines have gained greater importance in our reality of the digital community. With regards to the continuously growing amount of digital information it has become harder to find information that's of importance to the individual. Some specific domains with enforcement of recommendation engines are more studied than others, domains that distribute services or items usually end up in this category. Other domains that are in need of recommendation engines, that's not as well explored is business which enables learning through the internet. One of these business is called Nomp and provides a learning tool for kids and young teenagers in mathematics. The goal with this study is therefore to implement a recommendation engine for a business that is within this lesser explored domain. The goal is also to explore the advantages a recommendation engine would provide for its users.

The study is based on a framework within design science research, which included various kinds of experiments and a survey. The results from these activities represented the empirics for the analysis that was conducted. The results show some signs that it's possible to implement an artifact for this domain. However, it does not clearly show to what extent it's valuable for the end user. For some part, the objectives for this study was met. Although, the advantages for the users could have been explored in greater depth.

The overall prospects by conducting this study is that it will have some practical consequences, that the user can or will spend lesser time to search for important information. Differences between this study and other similar studies is that the recommendation engine is implemented to fit the needs of a real business. Also, compared to others, this study is based on data collected directly from the end users. Some similar systems have been implemented but the artefact is often more general or might have used data that's not relevant the domain. It's also more common that similar recommendation engines are using direct user feedback to make recommendations, which is not used in this study.

Keywords: Recommendation engine, implicit data, collaborative filtering, E-learning, recommendations.

Sammanfattning

Studier kring rekommendationsmotorer är ett område med större signifikans i en växande digital verklighet. Mängden med information ökar och med mer information blir det svårare att hitta det som för individen är av intresse. Vissa specifika områden med tillämpning av rekommendationsmotorer är mer välstuderade än andra, domäner som sysslar med försäljning hamnar i den mer studerade kategorin. Andra domäner som är i behov av rekommendationsmotorer, som inte är lika välstuderade är verksamheter som tillhandahåller möjlighet för lärande via internet. En av dessa verksamheter heter Nomp och erbjuder ett läroverktyg för barn och ungdomar inom matematik. Målet med denna studie är därför att implementera en rekommendationsmotor inom denna mindre utforskade domän. Målet är även att undersöka nyttan med rekommendationsmotorn för applikationens användare.

Studien har baserats på ett ramverk inom designforskning, vilket inkluderar olika typer av experiment samt en undersökning. Resultaten från dessa aktiviteter utgjorde empirin för den analys som sedan genomfördes. Resultatet ger visst stöd för att det är möjligt att implementera en rekommendationsmotor för denna domän. De visade däremot inget entydigt svar i vilken omfattning dess nytta har för slutanvändaren. Studiens målsättning uppfylldes till viss del, däremot kunde nyttan för slutanvändaren utforskats i större omfattning.

Förhoppningen är att denna studie ska ha effekter i form av praktiska konsekvenser, där användare kan spendera mindre tid på att leta efter information som kan vara till nytta. Det som skiljer sig i denna studie från tidigare liknande studier är att rekommendationsmotorn är implementerad för att passa en verklig verksamhet. I jämförelse med andra studier är denna studie även baserad på data direkt från verksamhetens användare. Vissa liknande artefakter har blivit implementerade, men då är de ofta mer generella eller har använt sig av data som inte är relevant för domänen. Det är också vanligare att liknande rekommendationsmotorer använder sig av direkt användarfeedback för att göra rekommendationer, vilket inte används i denna studie.

Nyckelord: Rekommendationsmotor, implicita data, samarbetsfiltrering, E-lärande, rekommendationer.

FÖRORD

Resan på Systemarkitekturutbildningen i Borås har varit minst sagt slitsam, men också lärorik och spännande. Utan tvekan hade få personer klarat av att inspirera så många unga framtida systemarkitekter som Cecilia Sönströd och Ulf Johansson. Dessa två lärare motiverade oss genom vårt första år på utbildningen och satte ribban för vad som komma skulle. Så här, tre år senare, är vår resa över och detta arbete är en produkt av vårt slitsamma arbete. Vi vill också passa på att tillägna några ord åt de lärare som, på ett eller annat vis, varit inblandade i produktionen av denna studie. Framförallt vill vi tacka Tuwe Löfström för sitt otroliga engagemang och vägledning genomgående i denna process. Men utan de kunskaper som vi fått inom artificiell intelligens, maskininlärning och programspråket Python, hade detta arbetet förmodligen inte varit möjligt. Därför vill vi också passa på att tacka de lärare som ligger bakom vår kunskap i detta ämne vilket är: Patrik Gabrielsson och Henrik Linusson.

Sist, men framförallt största tack till Marianne Norberg på Nomp som gav oss äran och möjlighet att genomföra denna studie. Tack för ditt förtroende och engagemang!

Innehållsförteckning

1	Inledning.....	- 1 -
2	Bakgrund.....	- 2 -
2.1	Problemdiskussion: Verksamhetsanpassa en rekommendationsmotor	- 3 -
2.2	Avgränsning.....	- 4 -
2.3	Syfte och frågeställning	- 4 -
3	Teori	- 5 -
3.1	Indirekt och direkt feedback	- 5 -
3.2	Algoritmer.....	- 6 -
3.2.1	Samarbetsfiltrering.....	- 6 -
3.2.2	Innehållsbaserad filtrering.....	- 8 -
3.2.3	Kunskapsbaserad filtrering.....	- 9 -
3.2.4	Hybrida metoder.....	- 9 -
3.3	Kallstartsproblemet.....	- 10 -
3.4	Överanpassning.....	- 10 -
3.5	Evaluering.....	- 12 -
3.6	Matematik i grundskolan	- 13 -
4	Relaterat arbete.....	- 15 -
4.1	ALAS-KA.....	- 15 -
4.2	Övriga studier	- 16 -
5	Metodval.....	- 18 -
5.1	Design Science Research.....	- 18 -
5.2	Ramverket.....	- 18 -
5.2.1	Datainsamlingsmetod	- 19 -
5.3	Strategier	- 19 -
5.3.1	Undersökning	- 19 -
5.3.2	Inledande experiment	- 19 -
5.3.3	Överraskning, Träffsäkerhet, Variation etc	- 20 -
5.3.4	Parameteroptimering	- 20 -
5.4	Etiska aspekter	- 20 -
6	Metodtillämpning	- 21 -
6.1	Definiera krav	- 21 -
6.2	Implementera artefakten	- 22 -
6.2.1	Skissa och bygga	- 22 -
6.3	Demonstrera Artefakten.....	- 27 -
6.4	Evaluera artefakten	- 28 -
6.4.1	Undersökning	- 28 -
6.4.2	Experiment	- 31 -
6.4.3	Parameteroptimering	- 31 -
7	Resultat.....	- 32 -
7.1	Resultat ifrån undersökningen	- 32 -
7.2	Evaluering.....	- 33 -
7.3	Resultat från parameteroptimeringen	- 34 -
7.4	Rekommendationer.....	- 35 -
8	Slutsatser och diskussion.....	- 37 -
8.1	Rekommendationer.....	- 38 -
8.2	Framtida studier	- 39 -
9	Referenser.....	- 40 -
10	Appendix A	- 42 -
11	Appendix B.....	- 46 -

Figurförteckning

Figur 1: Exempel användarbaserad samarbetsfiltrering	- 6 -
Figur 2: Exempel produktbaserad samarbetsfiltrering	- 7 -
Figur 3: Exempel på matrisfaktorisering, M' en approximation av M	- 8 -
Figur 4: Träning utan Regularization	- 11 -
Figur 5: Träning med Regularization	- 11 -
Figur 6: Flödesdiagram arbetsprocess	- 18 -
Figur 7: Respons från undersökningen.....	- 32 -
Figur 8: Träffsäkerhet / MAE.....	- 33 -
Figur 9: Mae efter parameteroptimering	- 35 -

Tabellförteckning

Tabell 1: Exempel på mål för talluppfattning och tals användning (Skolverket, U.å.).....	- 14 -
Tabell 2: Initial parameterkombination	- 24 -
Tabell 3: Frågor från undersökningen, elev A	- 28 -
Tabell 4: Frågor från undersökningen, elev B	- 29 -
Tabell 5: Frågor från undersökningen, elev C	- 29 -
Tabell 6: Frågor från undersökningen, elev D	- 30 -
Tabell 7: Intervall för parameteroptimeringen	- 31 -
Tabell 8: Frågor tillhörande respektive elev och användargrupp.....	- 32 -
Tabell 9: Intra- och inter användarvariation, täckningsgrad samt förvåning	- 34 -
Tabell 10: För lösningen optimal parameterkombination	- 35 -
Tabell 11: Initiala rekommendationer	- 36 -
Tabell 12: Rekommendationer efter parameteroptimering	- 36 -

1 Inledning

Information som finns tillgänglig på internet växer ständigt, detta leder till att det blir svårare och svårare för användare att snabbt hitta relevant information. Som resultat växer behovet för intelligenta och pålitliga rekommendationsmotorer. Inte enbart för tillämpning i försäljningssyfte utan även för tillämpning inom läroverktyg, det vill säga *undervisning via internet* (E-lärande). Intresset för verktyg inom E-lärande har under senare tid vuxit rejält, exempelvis har flertalet verksamheter börjat att erbjuda distanskurser för olika typer av målgrupper. Nomp är ett av dessa företag vars verksamhet bygger på att erbjuda ett smidigt läroverktyg, riktat mot barn och ungdomar som hjälpmedel för inläring av matematik. Eftersom fokus för denna typ av verksamhet inte i första hand ligger på merförsäljning utan snarare användarens lärande, leder detta till ökat behov av rekommendationsmotorer som tar detta i beaktning.

Tidigare studier bygger bland annat på hur man kan ge pålitliga prediktioner och hur olika typer av rekommendationssystem fungerar (Herlocker, Konstan, Borchers, & Riedl, 1999; Adomavicius & Tuzhilin, 2005). Målsättningen med denna studie är därför att undersöka möjligheter att med hjälp av verksamhetens information kring sina användare, implementera en rekommendationsmotor. Även att undersöka hur den kan anpassas för en verksamhet vars huvudsakliga sysselsättning är E-lärande.

2 Bakgrund

I Nomp's verksamhet ingår både elever, lärare och även föräldrar som stöd för elevernas lärande. Lärare kan tilldela elever uppgifter och föräldrar kan ha en bättre översikt över barnets framgång. Lärarens intresse är att se eleverna klara av de uppgifter som delats ut, dock har lärare ibland riktlinjer att ta hänsyn till liksom läroplaner och dylikt (Skolverket, u.å). Det bör också ligga i elevernas intresse att klara de uppgifter som blivit tilldelade av lärare, möjligtvis även att lära sig mer, alternativt öva tidigare erhållen kunskap. Det finns dock barn och ungdomar vars lärare inte använder detta verktyg, som av denna anledning saknar samma typ av vägledning.

Rekommendationssystem finns i dagsläget i stort utbud på många olika typer av webbplatser. Även om de har liknande syfte så kan den bakomliggande implementeringen skilja betydligt, eftersom att det finns flera olika algoritmer för att lösa detta problem. Val av algoritm baseras på och kan även anpassas utefter verksamhetens produkter och/eller tjänster, vilket alltså resulterar i dessa skillnader. En del av de algoritmer som studerats mycket inom området redogörs av Gorakala och Usuelli (2015) dessa är: *samarbetsfiltrering* (Collaborative filtering), *innehållsbaserad filtrering* (Content-based filtering), *kunskapsbaserad filtrering* (Knowledge-based filtering) samt *hybrida metoder* (Hybrid approaches).

Samarbetsfiltrering innebär, som namnet antyder, att genom samarbete mellan användare kunna rekommendera produkter till andra användare som delar samma intresse. Gorakala och Usuelli (2015) förklarar vidare att om person A och person B exempelvis gillar samma bok så finns det en stor sannolikhet att dessa delar samma intresse för böcker i framtiden. Om person A skulle köpa en ny bok skulle därför en rekommendation på denna bok kunna göras för person B.

Innehållsbaserade rekommendationssystem skiljer sig genom att ta likheter mellan produkter i beaktning. För att åstadkomma detta behövs en relation mellan produkterna, till exempel olika kännetecken (features) för produkterna, i bokexemplet kan detta förslagsvis vara bokens genre. Då en användare visat intresse för en specifik genre kan man rekommendera andra böcker av liknande genre.

Kunskapsbaserade rekommendationssystem används enligt Gorakala och Usuelli (2015) i mer specifika domäner där användarens historiska data är något mer begränsad. Ett exempel är verksamheter där många användare är helt nya användare och det därför inte finns någon tidigare data, alltså bristfällig information angående preferenser. Metoden löser alltså problemet med bristfällig information angående preferenser genom att inhämta ytterligare, kompletterande information från användaren. Likt innehållsbaserade rekommendationssystem krävs även här kännetecknen för olika produkter, vilket jämförs med användarens inhämtade preferens och därefter rekommenderas.

Utöver dessa algoritmer så finns även hybrida metoder vilket ofta resulterar i ett mer robust system (Gorakala & Usuelli, 2015). Tanken med hybrida metoder är att kombinera olika egenskaper av de tidigare förklarade metoderna. Svagheter hos en metod kan därför minimeras eller helt elimineras genom att kombinera funktionalitet hos en annan metod.

Bortsett från metod behöver man även fastställa vad, utifrån användarens perspektiv, som utgör en bra rekommendationsmotor. Enligt Aggarwal's (2015) förklaring så behöver man uppnå vissa kriterier, vilka han namnger enligt följande: *relevans*, *nymodighet* (novelty), *lyckoträff* (serendipity) samt *variation i rekommendationerna*.

Relevans innebär att rekommendationsmotorn ska rekommendera sådant som för användaren är relevant. Endast icke-relevanta rekommendationer ger användaren ingen nytta, då fyller rekommendationsmotorn knappast sitt syfte. Nymodighet enligt Aggarwal's (2015) beskrivning är att föreslå produkter och/eller tjänster för användaren som den inte sett eller varit medveten om tidigare. Lyckoträff handlar om att rekommendera sådant som för användaren kommer lite som en överraskning. Till sist nämner Aggarwal (2015) även att det bör finnas skillnad i rekommendationerna. Detta eftersom att ett antal liknande rekommendationer för en användare kan resultera i att antingen alla rekommendationer kan vara av nytta, eller ingen alls. Har man en viss skillnad på de förslag som ges för användaren, så ökar sannolikheten att åtminstone en av rekommendationerna kan vara relevant.

Sammanfattningsvis vad som lämpligast passar en verksamhet som säljer filmer via internet kanske inte passar en applikation som rekommenderar restauranger för potentiella kunder. Applikationen som rekommenderar restauranger kan möjligtvis behöva ta avstånd i beaktning, då en potentiell kund inte färdas orimligt långt för en måltid enligt preferenser (Aggarwal, 2016). För att kunna anpassa en rekommendationsmotor till en verksamhet så behöver man framförallt fastställa vad det verkliga syftet med rekommendationerna är. I de fall där verksamheter sysselsätter sig med olika typer av försäljning är syftet troligtvis ganska ofta merförsäljning. Den verksamhet som ligger till grund för studien är en verksamhet som fokuserar på barn och ungdomars lärande inom matematik. Merförsäljning är nödvändigtvis inte det huvudsakliga fokuset som driver denna typ av verksamhet. Lösningen kan därför behöva anpassas därefter, istället för att enbart använda en av de vanligare lösningarna för rekommendationsmotorer.

2.1 Problemdiskussion: Verksamhetsanpassa en rekommendationsmotor

Utifrån tidigare förklaring kring vad som utgör en bra rekommendationsmotor så bör rekommendationsmotorn uppfylla dessa ovan nämnda egenskaper, det vill säga: relevans, nymodighet, lyckoträff samt variation. För att uppnå relevans kan man utgå ifrån elevens tidigare slutförda uppgifter. Däremot skiljer sig behov från elev till elev, om tidigare uppgifter slutförts utan några som helst svårigheter kan det därför vara relevant att rekommendera en något svårare uppgift. Å andra sidan om eleven har haft svårigheter med dessa uppgifter kan det vara relevant att rekommendera en något lättare uppgift. Vidare uppstår dock problemet med att avgöra vilken information som kan vara indikationer på att en elev upplevt en uppgift som lätt eller svår. Förutom rekommendationer för lättare eller svårare uppgifter, kan relevans vara att rekommendera liknande uppgifter som tillsammans tidigare har blivit utdelade av lärare.

Både lyckoträff och nymodighet kan i detta fall vara svårare att åstadkomma då domänen är ganska begränsad. I en annan domän, som exempelvis sysslar med försäljning av produkter kan man ibland ha hundratusentals olika produkter. Mängden produkter gör att sannolikheten för att användaren sett eller hört talas om en specifik produkt kan vara relativt liten, därför också lättare överraska användaren. I Noms fall har vi ett begränsat antal uppgifter och en naturlig ordning till uppgifterna, vilket gör det svårare att överraska en användare.

Variation i rekommendationerna kan åstadkommas genom att, i de fall det är möjligt, rekommendera uppgifter från olika typer av matematiska områden. Rekommendationer kan

exempelvis ges för en viss uppgift inom algebra och en viss uppgift inom geometri, inom elevens aktuella årskurs.

Den största utmaningen är att på något sätt säga att de prediktioner som görs av systemet faktiskt är bra, detta är inte helt trivialt. En rekommendationsmotor som rekommenderar produkter som helt saknar relevans för användaren har ingen som helst betydelse. Det finns även andra faktorer som kan vara avgörande för pålitliga prediktioner, faktorer som exempelvis kan vara tid. Om en användare inte har varit aktiv inom systemet under en längre period så kan prediktionerna för användaren vara bra utifrån historiken, men på grund av den tid som gått inte längre vara relevant för eleven. Om det har gått väldigt lång tid kan eleven möjligtvis vara i flera klasser högre än senast.

Problemet ligger också i att identifiera den data som är relevant att basera rekommendationerna på, även mängden data kan vara ett problem. Vissa åtgärder kan behöva vidtas för att begränsa den data som systemet använder för att göra prediktioner, beroende på exempelvis exekveringstid och relevans.

2.2 Avgränsning

Då intresset hos eleverna skiljer sig betydligt från lärarna behöver rekommendationsmotorn antingen ta detta i beaktning eller bara hantera en specifik användargrupp. På grund av lärarens olika förhållningssätt att lära vad gäller metodik och/eller planering, men kanske framförallt läroplanerna (Skolverket, u.å) så bör därför en rekommendationsmotor ha mer syfte och betydelse förslagsvis hos eleverna.

Att anpassa rekommendationsmotorn utefter lärarnas läroplaner kan möjligtvis leda till ett bekvämt verktyg för lärarna, men enligt vår uppfattning bör rekommendationsmotorn uppfylla mer nytta om den anpassas för eleverna. Den användargrupp som särskilt hade haft stor nytta av en rekommendationsmotor är de elever som saknar vägledning av lärare. Men troligtvis blir även rekommendationer från systemet ett extra verktyg för alla elever att lära sig mer, därför avgränsas rekommendationsmotorns hantering till endast eleverna.

2.3 Syfte och frågeställning

Som tidigare nämnt är ett av de största problemen att faktiskt säkerställa att de prediktioner en rekommendationsmotor gör är bra. Förutom detta så är inte rekommendationsmotorer inom E-lärande lika utforskade som inom andra domäner. Dessa två problem är alltså framförallt motiveringen till denna studie. För att kunna lösa dessa problem har vi som avsikt att implementera en rekommendationsmotor för en verksamhet vars huvudsakliga fokus är inom E-lärande. Rekommendationsmotorn kommer att vara underlag för att genomföra experiment som stöd till studien. Vår förhoppning är att experimenten kommer att kunna undergöra underlag för att besvara studiens övergripande frågeställning, vilket är:

Kan en rekommendationsmotor anpassas för en verksamhet vars huvudsakliga fokus är E-lärande inom matematik, så att dess rekommendationer har nytta för användarna?

3 Teori

I detta kapitel kommer vi att redogöra för vissa viktiga begrepp och teorier som är relevanta för fortsatt förståelse för studien. Teorin är också en del av den kunskap som erhållits inom området och har på så sätt varit avgörande för de beslut som tagits för studien.

3.1 Indirekt och direkt feedback

För att en rekommendationsmotor ska ha förmåga att kunna ge rekommendationer till en användare, behövs någon form av kunskap kring respektive användares åsikt kring observerade objekt (Claypool, Brown, Le & Waseda, 2001). Objekt används i detta fall som en mer generell benämning, beroende på domän kan objekt vara den produkt, artikel, tjänst eller liknande som är tänkt att rekommenderas för användaren. Användares åsikter brukar beskrivas genom två vanligt förekommande begrepp, vilket är *implicit* och *explicit* feedback. Explicit feedback är med andra ord, direkt feedback, vilket är också det vanligaste tillvägagångssättet när det kommer till rekommendationsmotorer (ibid.). Direkt feedback får man tillgång till genom att låta användaren uttryckligen informera systemet om dess åsikter. Åsikterna kan exempelvis ges genom att låta användaren rangordna ett objekt på en skala från ett till fem. Claypool et al. (2001) förklarar vidare att explicit feedback är det mest självklara indikatorerna på användares intressen.

Implicit feedback är den direkta motsatsen till explicit feedback, det vill säga indirekt feedback. Claypool et al. (2001) hävdar att implicit feedback kan vara ett bra alternativ, eftersom att det finns vissa nackdelar med explicit feedback. En av dessa nackdelar är då användarna inte ser någon anledning till att ge feedback, då får systemet ingen information att basera rekommendationerna på (ibid). Vidare förklarar de däremot att implicit feedback inte ger samma säkerhet som explicit feedback. Med säkerhet menar Claypool et al. (2001) att indirekt feedback kan betyda olika beroende på situation. Vi kan inte med all säkerhet säga att ett visst beteende tyder på intresse för produkten eller vice versa. Däremot poängterar de att implicit feedback kan skapas eller samlas in utan direkt kommunikation mellan systemet och användaren, vilket utgör en fördel. Exempel på beteendemönster som kan användas som implicit feedback är: totala tiden en användare spenderar på en sida eller hur mycket användaren bläddrar på en sida (Claypool et al. 2001). Det som enligt Claypool et al. (2001) bör vara det bästa alternativet för tydligast indikation av intresse, är att kombinera båda typerna av feedback från användaren.

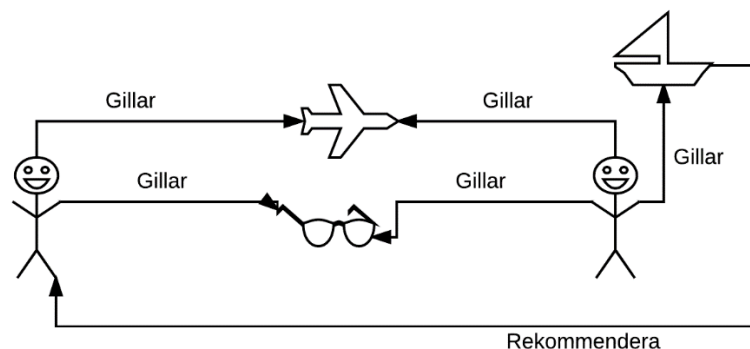
3.2 Algoritmer

I studiens inledande kapitel nämndes flera angreppssätt som kan användas vid implementering av rekommendationsmotorer, vilka var: samarbetsfiltrering, innehållsbaserad filtrering, kunskapsbaserad filtrering samt hybrida metoder (Gorakala, 2016). Angreppssätten beskrivs här mer detaljerat än tidigare. Begrepp och metoder som däremot inte påverkar läsarens fortsatta förståelse för studien kommer att beskrivas mer översiktligt.

3.2.1 Samarbetsfiltrering

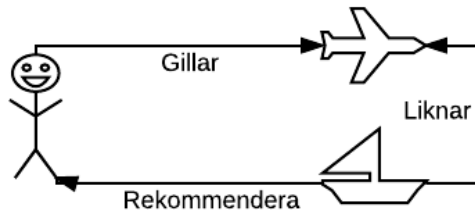
Samarbetsfiltrering är ett samlingsnamn för ett antal olika algoritmer inom området för rekommendationsmotorer. Vidare kan dessa generellt fördelas i två grupper varav den ena brukar kallas för minnesbaserade metoder och den andra för modellbaserade metoder (Gorakala, 2016). Inom gruppen för minnesbaserade modeller hamnar de två mer omtalade angreppssätten för samarbetsfiltrering, vilka är: *användarbaserad samarbetsfiltrering* (User-based collaborative filtering) och *produktbaserad samarbetsfiltrering* (Item-based collaborative filtering) (ibid). Det som i huvudsak skiljer dessa angreppssätt åt är hur beräkningen genomförs för objekten som bör rekommenderas för användaren. Däremot är den gemensamma grundtanken, att liknande användare gillar liknande saker. För att underlätta förståelse kommer fortsatta exempel att utgå ifrån användandet av explicit feedback.

I användarbaserad samarbetsfiltrering utgår man ifrån användarnas åsikter för domänspecifika objekt, exempelvis produkter. En användare kan gilla en produkt genom att ge ett högre värde, alternativt visa sitt misstykke genom att betygssätta en produkt med ett lägre värde. Oavsett hur användaren väljer att betygssätta olika produkter letar man, i användarbaserad samarbetsfiltrering, efter användare som gett likartade bedömningar (ibid). För en specifik användare, identifieras därefter produkter som användaren inte har uttryckt sin åsikt kring, där liknande användare uttryckt sin åsikt. Gillar liknande personer produkten då är den möjligtvis av intresse och kan därför rekommenderas för användaren (ibid). Detta exempel illustreras också visuellt med hjälp av figur 1.



Figur 1: Exempel användarbaserad samarbetsfiltrering

Även i produktbaserad samarbetsfiltrering utgår man ifrån de produkter en användare tidigare uttryckt sin åsikt kring. Skillnaden här är att istället för att jämföra användare med varandra, så jämförs produkter. För de produkter användaren tidigare har gett positiv feedback för, kan liknande produkter möjligtvis vara av intresse (ibid). Visuellt exempel för produktbaserad samarbetsfiltrering ges i figur 2. En fördel med detta angreppssätt över användarbaserad samarbetsfiltrering som nämns av Gorakala (2016) är att i de flesta domäner förändras inte produkterna lika frekvent som användares åsikter, och kräver därför inte samma mängd omberäkningar.



Figur 2: Exempel produktbaserad samarbetsfiltrering

Båda dessa angreppssätt har blivit populära verktyg då de är tämligen lätta att förstå samt att implementera (ibid). Däremot finns det även väsentliga begränsningar, exempelvis att båda dessa metoder kräver bakgrundsdata kring användare eller produkter för att utföra sina likhetsberäkningar. I en större domän där antalet användare eller produkter är stort, blir utförandet av samtliga likhetsberäkningarna en förhållandevis tung operation. Framförallt på grund av denna begränsning uppkom behovet för andra angreppssätt, där en av lösningarna var modellbaserade metoder (ibid).

3.2.1.1 Modellbaserade metoder

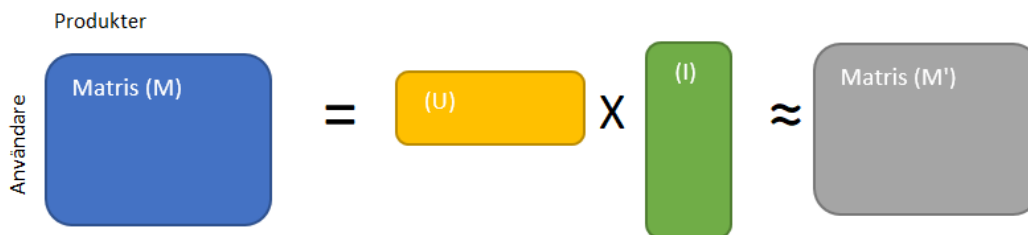
Förutom att modellbaserade metoder löser problemet med likhetsberäkningarna som nämndes tidigare så har dessa även visat sig vara mer träffsäkra än minnesbaserade modeller. Inom området för rekommendationssystem anses dessa tekniker därför att vara överlägsna (Herlocker, Konstan, Borchers, & Riedl, 1999).

Det finns en hel del olika tillvägagångssätt för dessa typer av metoder, vilka bland annat kan inkludera användandet av probabilistiska modeller, maskininlärningsmodeller eller matriser (Gorakala, 2016). Probabilistiska modeller innebär att, utifrån tillgänglig data för sina användare, basera en modell på sannolikheter för att respektive användare gillar eller ogillar en produkt (ibid). I sammanhang för maskininläring kan tillgänglig data användas för att bygga en modell genom användandet av en rad olika maskininlärningsalgoritmer. Algoritmer som exempelvis kan användas är: SVM, klustring, Knn-klassificerare, beslutsträd med flera (ibid). Modellerna används i sin tur för att genomföra prediktioner för om en användare kommer att gilla eller ogilla en produkt.

Användning av matriser i dessa sammanhang är något som Gorakala (2016) valt att kalla för ett matematiskt tillvägagångssätt. Angreppssättet innebär att den data som representerar användare och exempelvis produkter representeras av en matris. Varje cell representerar alltså en användarens åsikt (om någon) för en specifik produkt. Ett potentiellt problem med detta är dock storleken på matrisen (Symeonidis & Zioupos, 2016). En domän med exempelvis flera miljoner användare och flera tusentals produkter resulterar alltså i en matris med uppemot flera miljarder celler. Celler som dessutom till större del kan vara tomma, då troligtvis långt ifrån alla användare gett feedback kring alla produkter. Storleken på matrisen har alltså en betydande roll för hur tidskrävande operationerna för att generera prediktionerna blir. Symeonidis och Zioupos (2016) förklarar vidare att detta är ett problem som generellt löses med hjälp av matrisfaktorisering.

3.2.1.1.1 Matrisfaktorisering

Det finns en mängd olika sätt att tillämpa matrisfaktorisering inom rekommendationssystem och även inom andra områden (Ocepek, Rugelj, & Bosnić, 2015; Symeonidis & Zioupos, 2016; Yin, Chang, & Wang, 2009). Tillämpningen av matrisfaktorisering brukar generellt beskrivas som att den ursprungliga matrisen (M) bryts ned till två mindre matriser U och I . Produkten av U och I genererar en ny matris (M') som representerar approximation av M (Ocepek, Rugelj, & Bosnić, 2015). Illustreras även med hjälp av figur 3.



Figur 3: Exempel på matrisfaktorisering, M' en approximation av M

En skillnad mellan M och M' är att M , som tidigare nämnt, innehåller tomma celler, det gör inte M' . De fält som alltså blivit ifyllda representerar rekommendationssystemets prediktioner. Beroende på val av metod för matrisfaktorisering kan hanteringen av matriserna variera (Celma, 2010). Celma (2010) nämner exempelvis minsta kvadratmetoder som kräver att M' ska avvika så lite som möjligt från M . En minsta kvadratmetod som används i detta sammanhang kallas för alternerande minsta kvadratmetoden eller ALS. ALS beskrivs av Kampffmeyer (2015) som en iterativ metod där U och I växelvis fixeras för att minimera resultatet av en kostnadsfunktion, det vill säga minimera skillnaden mellan M och M' .

Användandet av matrisfaktorisering har blivit ett populärt verktyg inom området för rekommendationssystem. Inte enbart för komprimeringen av data utan även för att dess träffsäkerhet gällande prediktioner (Bell & Koren, 2011).

3.2.2 Innehållsbaserad filtrering

Den bakomliggande grundtanken med innehållsbaserad filtrering att rekommendera produkter baserat på kännetecken (Gorakala, 2016). Alltså försöker man identifiera orsaker till varför en användare kan tänkas föredra en specifik produkt. Baserat på orsakerna rekommenderas därav produkter som fyller behovet, vilka identifieras via produkternas kännetecken.

Gorakala (2016) förklarar att innehållsbaserad filtrering vanligtvis involverar tre steg, generera någon form av användarprofil, en produktprofil samt modellen. Produktprofilen består vanligtvis av kännetecken som kan kopplas till en specifik produkt, vilket varierar beroende på domän. Det viktigaste i detta sammanhang är att dessa kännetecken kan användas för att identifiera liknande, samt särskilja olika produkter (ibid).

Användarprofilen byggs upp genom att identifiera kännetecken som är vanligast förekommande hos en användare (ibid). Ju oftare ett kännetecken återkommer i en användarprofil desto större är sannolikheten att användaren uppskattar produkter med liknande kännetecken. Alltså kan vi utifrån användarprofilen rekommendera produkter med liknande

kännetecken som inte ännu konsumerats av användaren. Till skillnad från samarbetsfiltrering utgår man alltså endast från individens egna preferenser utan vidare jämförelse med andra användare (ibid). Gorakala (2016) nämner detta som både fördel och nackdel då detta kan leda till mer individuella rekommendationer, men att individen även kan ha intresse av trender vilket därav missas.

3.2.3 Kunskapsbaserad filtrering

De tidigare nämnda angreppssätten bygger på att tillräcklig information kring användares preferenser och/eller produkter finns tillgänglig, för att kunna ge bra rekommendationer. Kunskapsbaserad filtrering å andra sidan är ett alternativ för de domäner som har bristfällig, alternativt ingen tidigare information angående användares preferenser. Vanliga förekomster av kunskapsbaserade rekommendationssystem som nämns av Mandl, Felfernig, Teppan och Schubert (2010) är exempelvis domäner där produkter är av något dyrare slag och där inköp görs mer sporadiskt. Exempel på produkter av denna typ är fastigheter, lägenheter och fordon där inköp normalt sett inte görs särskilt ofta. Problemet med bristfällig information löses i denna metod, oftast genom att explicit inhämta ytterligare information från användaren.

Kunskapsbaserade rekommendationssystem kräver i regel en mer omfattande kunskap kring domänens produkter än i exempelvis innehållsbaserad filtrering. Mandle et al. (2010) nämner detta som en betydande nackdel då kunskapsbasen för produkterna i regel kräver mer underhåll än för andra typer av rekommendationssystem.

Ofta guidas användaren igenom en process vars målsättning är att generera rekommendationer ifrån systemet. Mandle et al (2010) förklarar att generellt sker detta i tre steg, varav det första är att inhämta specifikationer ifrån användaren. Specifikationer för hus- eller lägenhets köp kan exempelvis vara: hur många rum, hur stort, prisintervall, område och dylikt. Andra steget i processen innebär att, i de fall där inga produkter kunde hittas inom ramen för användarens specifikationer, korrigeras eller ge förslag på förändringar som kan leda till fler resultat. Tredje steget innebär att resultaten presenteras för användaren, vilket även sker direkt efter steg ett i de fall då resultat identifieras utan behov för ytterligare korrigering av specifikationer. Resultaten kan även då presenteras med argumentationer för hur användarens specifikationer relaterar till produkternas egenskaper (ibid).

3.2.4 Hybrida metoder

Som nämnts i studiens inledande kapitel så är motiveringen bakom hybrida metoder att svagheter hos en metod kan vägas upp med styrkor hos en annan. Detta kan åstadkommas genom att kombinera två eller flera av de tidigare nämnda angreppssätten. Gorakala (2016) nämner exempelvis ett välkänt problem hos samarbetsfiltrering, vilket noggrannare kommer att redogöras i 3.3 Kallstartsproblemet. Kortfattat innebär detta att rekommendationssystemet kan ha svårigheter att rekommendera objekt utan tidigare historik, exempelvis, nya produkter. Samarbetsfiltrering kan på så sätt kombineras med exempelvis innehållsbaserad filtrering för att hantera detta problem. Innehållsbaserad filtrering utgår ifrån kännetecken hos produkterna, vilket innebär att produkter utan historik kan rekommenderas baserat på kännetecken. Alltså undviks problemet genom att kombinera funktionalitet från två olika angreppssätt för att skapa en hybrid rekommendationsmotor (ibid).

3.3 Kallstartsproblemet

Ett välkänt problem (Formoso, Fernández, Cacheda, & Carneiro, 2013; Ocepek, Rugelj, & Bosnić, 2015; Barjasteh, Forsati, Ross, Esfahanian, & Radha, 2016; Yin, Chang, & Wang, 2009) som är vanligt förekommande inom samarbetsfiltrering är kallstartsproblemet. Problemet kan uppstå både hos nya användare eller nya produkter. Möjliga symptom kan vara att systemet inte har möjlighet att rekommendera nya produkter för användare, eller passande produkter för nya användare.

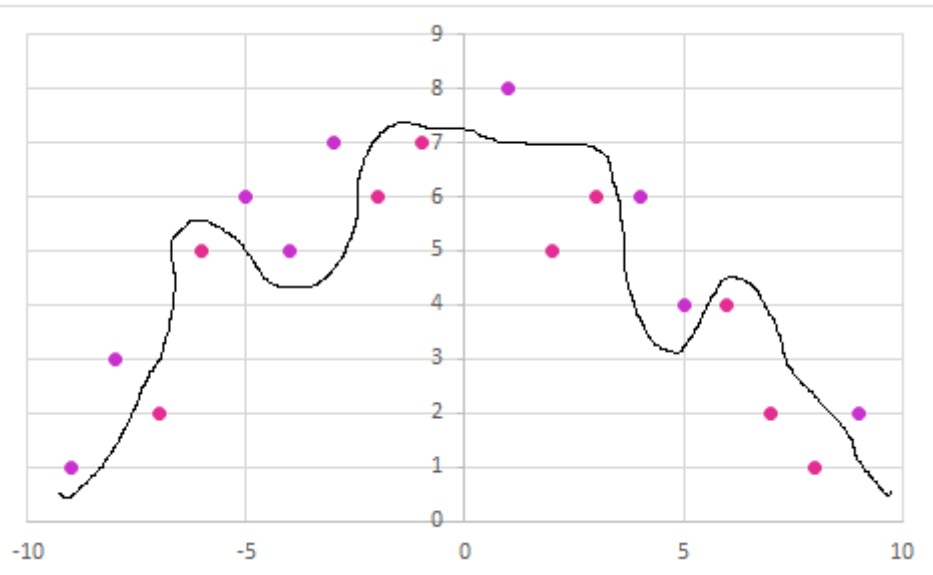
Det finns flera olika typer av lösningar på detta problem och Barjasteh et al. (2016) förklarar att dessa kan grupperas till fyra olika ansatser. Dessa olika grupperingar är: naiva metoder, varmstartsmetoder, känneteckenskombination och modellkombination. Den första metoden är ganska trivial då angreppssättet baseras på det enklast möjliga. För en användare där tidigare historik saknas, kan exempelvis ett antal av de mest populära produkterna och/eller helt slumpmässiga produkter rekommenderas. Barjasteh et al. (2016) påpekar också att detta angreppssätt hanterar alla användare på samma sätt, vilket också resulterar i att träffsäkerheten för prediktionerna påverkas.

Tanken med varmstartsmetoder är att mer direkt tillföra den information som systemet saknar för att kunna utföra prediktioner för en användare (ibid.). Detta kan, till exempel, uppnås genom att explicit be användaren att betygsätta ett antal produkter och sedan utföra prediktioner. Det tredje angreppssättet innebär att kombinera annan typ av information för att få den kunskap systemet behöver (ibid.). Barjasteh et al. (2016) förklarar att exempel på sådan information kan vara kännetecknen för användare så som profiler eller metadata för produkter. Kombinationer av modeller är det fjärde angreppssättet som Barjasteh et al. (2016) redogör för, vilket kan uppnås genom att resultat av olika rekommendationsmotorer kombineras för att utgöra en lösning på problemet. Däremot påpekar de som en uppenbar nackdel, att detta kräver flera separata rekommendationsmotorer.

3.4 Överanpassning

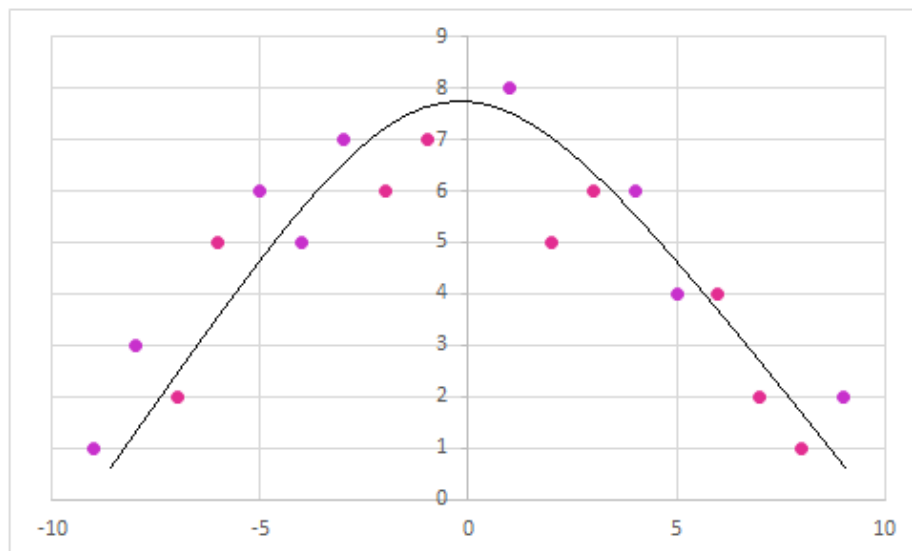
Ett vanligt och välstuderat problem inom området för prediktionsmodeller är överanpassning, vilket alltså även blir ett problem inom samarbetsfiltrering (Dietterich, 1995). Generellt kan man beskriva överanpassning som ett resultat av att modellen är så pass väl bekant med data så att den slutar att generalisera (Symeonidis & Zioupos, 2016). En modell tränas ofta på en viss mängd data där utfallet av intresset redan är känt (Dietterich, 1995). Ett exempel skulle kunna vara då syftet för modellen är att predicera sjukdomar baserat på symptom. Utfallet av intresse är då personens diagnos, vilket i träningsdata redan har fastställts. Problemet med överanpassning uppstår då modellen memorerar alla symptom för en viss diagnos och på så sätt predicerar bra på träningsdata. Prövas modellen sedan på annorlunda data, där symptomen inte stämmer exakt med de modellen memorerat, blir resultaten förmodligen sämre. Bra prediktioner utifrån träningsdata men sämre resultat för olika typer av testdata, kan alltså vara symptom på en överanpassad modell.

En vanlig teknik för att bevara en generell modell kallas för Regularization. Beskrivningen av tekniken sker utifrån en fiktiv modell för ett klassificeringsproblem. Syftet är att försöka predicera utifrån respektive uppgifter för varje objekt, om objektet tillhör klass A eller klass B. Efter träning kan funktionen som modellen baserar sina beslut på illustreras som linjen i följande figur:



Figur 4: Träning utan Regularization

Figuren illustrerar även en överanpassad modell (McCaffrey , 2015), då funktionen anpassats exakt efter datan modellen tränats på. Det finns ett flertal tekniker för Regularization, oavsett vilken teknik så är syftet mer eller mindre att tvinga modellen till mer generella beslut. Vid applicering av Regularization skulle funktionen istället kunna beskrivas som följande illustration:



Figur 5: Träning med Regularization

Resultat av en modell likt denna är att den bör prestera bättre på osedd data, i jämförelse med tidigare modell (ibid).

3.5 Evaluering

I rapportens bakgrund redogjorde vi för en rad olika viktiga egenskaper hos en rekommendationsmotor. Egenskaper så som relevans, nymodighet, lyckoträff samt variation. Utöver dessa egenskaper så är även träffsäkerhet ett viktigt begrepp inom prediktionsmodeller.

Lü et al. (2012) redogör för en hel del olika tillvägagångssätt för hur de ovan nämnda egenskaperna kan evalueras. Två vanliga utvärderingsmått som används för att evaluera träffsäkerheten hos en modell är: *Mean Absolute Error* (MAE) och *Root Mean Squared Error* (RMSE) (ibid.). Utvärderingen sker generellt genom att låta modellen göra prediktioner för data där utfallet redan är känt. Anledningen är att MAE och RMSE används som mätvärden för hur nära modellen är i sina prediktioner jämfört med de faktiska värdena. Ju lägre MAE/RMSE desto högre träffsäkerhet, det vill säga prediktionerna är nära eller lika med de faktiska värdena. Lü et al. (2012) förklarar vidare att den väsentliga skillnaden mellan dessa två, är att RMSE har tendensen att straffa större fel betydligt mer.

När det kommer till variation så finns det två olika områden där variation kan vara av intresse. Variation mellan produkterna i en lista av rekommendationer samt variation mellan olika sådana listor (ibid.). Variation mellan listor kan generellt beskrivas som rekommendationsmotorns möjlighet att rekommendera olika produkter till olika användare. Utvärderingsmättet som mäter variation mellan listor kallas även för *inter- användarvariation* (inter- user diversity) och kan räknas ut med hjälp av följande formel:

$$H_{ij}(L) = \frac{1 - Q_{ij}(L)}{L} \quad (1)$$

I denna formel beskrivs $Q_{ij}(L)$ som antalet objekt som återfinns i både användare i och j :s rekommendationslistor. För alla par av användare kan därefter medelvärdet av $H_{ij}(L)$ räknas ut, desto högre detta värde är, desto mer individuella rekommendationer får användarna.

Måttet som mäter variation mellan produkterna kallas för *intra- användarvariation* (intra- user diversity). Detta görs genom att jämföra skillnaden mellan alla par av objekt som rekommenderats för användare och räknas ut enligt följande:

$$I_i(L) = \frac{1}{L(L-1)} \sum_{\alpha \neq \beta} s(o\alpha, o\beta) \quad (2)$$

$S(O\alpha, O\beta)$ utgör likheten mellan par av objekt rekommenderade för användaren. Likhet mellan produkter kan man enligt Lü et al. (2012) antingen få genom explicit feedback eller genom metadata. För alla användares $I_i(L)$ kan ett medelvärde räknas ut, hur varierade rekommendationerna är i genomsnitt för alla användare. Ett lägre värde tyder på större variation i listorna för respektive användare, att rekommendationerna alltså i regel skiljer sig från varandra.

Ett annat viktigt begrepp som vi tidigare nämnt är nymodighet, rekommendationsmotorns förmåga att förvåna användaren. Lü et al. (2012) förklarar nymodighet som ett mått på hur

annorlunda en rekommendation är i relation till de objekt användaren har sett tidigare. För att mäta till vilken grad en användare överraskas av en rekommendation kan man utgå ifrån sannolikheten att användaren har sett objektet tidigare (ibid.). Ett objekt som är mindre populärt hos systemets användare har större sannolikhet att överraska användaren. Att ett slumpmässigt utvalt objekt redan har påträffats av en användare kan därför beskrivas enligt: K_α/M där K_α är populariteten för produkten, exempelvis hur många gånger en produkt har blivit köpt. Medan M är antalet totala användare (ibid.). Därför enligt Lü et al. (2012) kan oförutsägbarheten mätas enligt nedanstående formel:

$$U_\alpha = \log_2(M/K_\alpha) \quad (3)$$

Lü et al. (2012) redogör även för ett annat begrepp, som inte tidigare nämnts, vilket är täckningsgrad. Detta mått mäter rekommendationsmotorns möjlighet att rekommendera varje enskilt objekt. Detta görs genom att summera alla unika objekt i alla användares rekommendationslistor, som i nedanstående formel representeras av N_d . Detta divideras med antalet totala objekt, vilket i formeln representeras av N . Resultatet av detta är då rekommendationsmotorns täckningsgrad i procent.

$$COV(L) = N_d/N \quad (4)$$

3.6 Matematik i grundskolan

I Sverige bygger undervisning inom matematik på läroplaner som utfärdas av Skolverket. Skolverket är en myndighet vars syfte är att bidra till god kvalitet i svensk utbildning (Skolverket, 2016). Läroplanerna innehåller alltså riktlinjer för den utbildning som en elev bör erhålla under en viss period i sin skolgång. Grundskolan innefattar årskurserna 1 till och med 9, där dessa delats upp i tre nivåer. Dessa nivåer är: lågstadiet, mellanstadiet och högstadiet. Skolverket har utformat en läroplan för vardera nivån, det vill säga en sammanfattande läroplan för exempelvis lågstadiet. Däremot innebär sista årskurs i varje nivå ett kunskapskrav på eleverna. Dessa har explicit utformats för att säkerställa att elevernas kunskaper motsvarar läroplanerna för varje nivå.

Kunskapskraven på lågstadienivå innefattar ett visst antal övergripande mål för de kunskaper som eleverna alltså bör erhållit i årskurserna 1–3. Till skillnad från lågstadiet så fördelas kunskapskraven för mellanstadie- och högstadienivå i ett antal betygskriterier. Vardera betygskriterier innebär varierande krav på förståelse för eleven. Ett högre betyg kräver djupare förståelse medans lägre krav betyder mer övergripande förståelse (Skolverket, U.å).

Gemensamt för läroplanerna inom matematik är att målen för eleverna fördelas i sex stycken områden. Dessa områden är följande: Taluppfattning och tals användning, Algebra, Geometri, Sannolikhet och statistik, Samband och förändringar samt Problemlösning (ibid.).

Läroplanerna avspeglar även den hierarki som naturligt kan identifieras inom matematik, det vill säga att vissa krav kräver att tidigare krav har uppfyllts. Ett exempel går att identifiera i taluppfattning och tals användning, för de tre olika nivåerna förekommer följande krav:

Krav nr:	Lågstadiet/Åk 1–3	Mellanstadiet/ Åk 4–6	Högstadiet/ Åk 7–9
1	<ul style="list-style-type: none"> ”Den del av helhet och del av antal. Hur delarna kan benämnas och uttryckas som enkla bråk samt hur enkla bråk förhåller sig till naturliga tal.” 	<ul style="list-style-type: none"> ”Tal i bråk- och decimalform och deras användning i vardagliga situationer” 	<ul style="list-style-type: none"> ”Centrala metoder för beräkningar med tal i bråk- och decimalform vid överslagsräkning, huvudräkning samt vid beräkningar med skriftliga metoder och digital teknik”
2		<ul style="list-style-type: none"> ”Tal i procentform och deras samband med tal i bråk och decimalform” 	
		<ul style="list-style-type: none"> ”Centrala metoder för beräkningar med naturliga tal och enkla tal i decimalform vid överslagsräkning, huvudräkning samt vid beräkningar med skriftliga metoder och miniräknare. Metodernas användning i olika situationer.” 	

Tabell 1: Exempel på mål för talluppfattning och tals användning (Skolverket, U.å.)

Enligt målen från lågstadienivå bör eleven, till exempel, ha kunskap om enkla bråk och hur de förhåller sig till naturliga tal. Utan förkunskap och kännedom kring enkla bråk är det betydligt svårare, om ens möjligt, att uppnå krav 1–3 för mellanstadienivå. Vidare så är det betydligt svårare utan förkunskap kring dessa mål på mellanstadienivå, att uppnå krav 1 på högstadienivå.

4 Relaterat arbete

Inledningen av denna rapport presenterade olika angreppssätt vad gäller tillämpningar av olika algoritmer för rekommendationsmotorer. Adomavicius och Tuzhilin (2005) poängterar att fastän dessa algoritmer har nått stor framgång, krävs vidare utveckling för att möjliggöra tillämpningar inom flertalet andra applikationer. Algoritmerna ligger däremot till grund för mycket av den forskning som hittills tillämpat rekommendationsmotorer inom E-lärande. Ett av de kanske bästa exemplen som nära relaterar till denna studie är Khan Academys forskning (Ruipérez-Valiente, Muñoz-Merino, & Kloos, 2013) som ligger bakom applikationens analysverktyg för inläring. Analysverktyget som Khan Academy kallar ALAS-KA, analyserar bland mycket annat mönster och beteende hos användarna. Fåtal domäner inom området för E-lärande tillämpar rekommendationsmotorer där egen forskning ligger till grund för applikationen. Khan Academy skiljer sig till den grad att deras analysverktyg är ett direkt resultat av tidigare studier. Att verktyget idag används som hjälpmedel för elever och lärare i den faktiska domänen, gör studien högst intressant för denna rapport.

4.1 ALAS-KA

Khan Academy är en applikation som i stora delar bygger på den nya populära utlärningsmetodiken ”det flippade klassrummet” (the flipped classroom) (Ruipérez-Valiente, Muñoz-Merino, Leony, & Delgado, 2015). Metodiken innebär i stora delar att lärare delar med sig av utbildningsmaterial till sina elever online. Khan Accademy delar med sig av kunskap genom egna kurser och material som finns tillgängligt för alla användare. Men som stöd för metodiken: det flippade klassrummet, finns även möjlighet för lärare att själva dela med sig av eget material till sina elever. Utöver detta är Khan Accademy också unikt för deras användning och illustrationer av insamlade data (ibid).

I Ruipérez-Valiente et al. (2015) studie förklarar de bland annat att det finns två olika angreppssätt vad gäller den beslutstagande processen i elevens inlärande. En beslutstagande process kan till exempel vara, vilket material eleven ska studera härnäst för att uppnå sitt mål i inlärningsprocessen. Ett angreppssätt för beslutstagande är att visualisera data som representerar elevens tidigare prestation. Här menar Ruipérez-Valiente et al. att eleven genom självreflektion får möjlighet att ta beslut om nästa steg i inlärningsprocessen, alternativt att en lärare kan hjälpa eleven i rätt riktning. En annan möjlighet är att använda automatiska verktyg för att hjälpa eleven att ta nästa steg i inlärningsprocessen. I detta sammanhang kan ett sådant verktyg vara exempelvis en rekommendationsmotor (ibid). De påpekar däremot att det finns vissa restriktioner i automatiska verktyg, exempelvis variablerna för vilka besluten grundas på. De hävdar även att dessa typer av verktyg har en tendens till att göra fler misstag än människor.

Alas-ka är ett verktyg som framförallt riktar in sig på elevens möjlighet till eget beslutstagande. I några av de studier som berör Alas-ka (Ruipérez-Valiente, Muñoz-Merino, & Delgado, 2013; Ruipérez-Valiente et al, 2015; Leony, Muñoz-Merino, Ruipérez-Valiente, Arellano, & Delgado, 2014) redogör de bland annat för hur verktyget underlättar för både elever och lärare i inlärningsprocessen. För lärare underlättar verktyget genom att ge en enklare översikt över hur eleverna processerat diverse inlärningsmaterial, representerat som olika typer av illustrationer. Lärare har även möjlighet att ställa kompletterande frågor angående inlärningsmaterialet via plattformen, vilket utgör en ytterligare möjlighet att se om eleven har förstått. Eleven å andra sidan har möjlighet att själv studera framgångar genom diverse illustrationer, så som olika typer av diagram. Den data diagrammen representerar varierar, ett

exempel kan vara för de uppgifter som slutförts hur många som genomförts korrekt gentemot felaktigt.

Det finns flertal faktorer som gör studierna högst intressanta inom ramen för E-lärande. Leony et al (2014) studie redogör bland annat för en modell som tillämpats i Alas-ka för att identifiera humöret hos eleverna. Framförallt intressant är deras motivering för att vissa humör tyder på en negativ eller positiv korrelation till inläring (ibid). Generellt så förklaras fyra olika sinnestämningar hos eleverna vilka är: frustration, förvirring, tristess samt glädje. Glädje och förvirring tyder på en mer positiv korrelation till elevernas inläring medans frustration och tristess tyder på en negativ korrelation (ibid). Identifieringen av dessa olika sinnestämningar sker på lite olika sätt, exempelvis antas frustration vara obefintlig då en elev påbörjat en uppgift för första gången. Elevens frustration ökar därefter med tid, ju längre tid desto mer frustration. En ytterligare faktor är om eleven felaktigt besvarat samma fråga, då antas frustrationen vara högre då eleven påbörjar uppgiften (ibid).

För att ytterligare komplettera Alas-ka poängterar däremot Ruipérez-Valiente, Muñoz-Merino, & Delgado (2013) möjligheterna att ge rekommendationer baserat på data från Alas-ka. De poängterar att för vissa elever kan diagramen vara överväldigande och på så sätt inte lika effektiva. Alltså finns motiveringen till att erbjuda ett alternativ till visualiseringar. Rekommendationerna bygger övergripande på att kommunicera slutsatser ifrån den data som finns tillgänglig i Alas-ka. Ruipérez-Valiente, Muñoz-Merino, & Delgado (2013) ger ett exempel för data som visar hur mycket tid elever lägger på att lösa uppgifter kontra titta på inlärningsvideo. Vidare förklarar de att rekommendationer i detta fall kan skickas till lärare för att ytterligare uppmärksamma dessa beteenden. Lärare kan därefter ta beslut kring lämplig åtgärd för att underlätta för eleven.

4.2 Övriga studier

Till skillnad från bland annat Khan Academys rekommendationsmotor (Ruipelez-Valiente, Muñoz-Merino, & Delgado, 2013) finns det även mer generell forskning (Recker, Walker, & Lawless, 2003; Tang & McCalla, 2005; Chen, Niu, Zhao, & Li, 2012) som fokuserat på hur man kan tillämpa de olika rekommendationsalgoritmerna för att stödja elever i sitt lärande.

Studien som genomfördes av Chen et al. (2012) redogör för en hybrid ansats där rekommendationer baseras på användarens rangordnade preferenser inom ett visst antal områden. Områden som gavs som exempel var: matematik, psykologi, musik, programmering, magi och ekonomi (Chen et al. 2012). Bakgrunden till studien var att algoritmen skulle kunna användas i en E-lärande miljö där plattformen erbjuder ett antal olika kurser.

Ytterligare exempel är en studie som utfördes av Tang och McCalla (2005). Syftet här var att hjälpa elever att hitta relevant information på internet genom en sökrobot. Tang och McCalla (2005) menar att en sådan applikation, kan vara användbart då elever försöker hitta relevant information för exempelvis större uppsatser. Genom feedback från användaren angående artiklar som redan lästs, så kan rekommendationerna ytterligare förfinas. Resultatet av detta är att rekommendationsmotorn anpassar sig efter eleven. Användandet är däremot något begränsad då det kräver kontinuerlig feedback från användaren. Däremot är den generell i den bemärkelsen att individen kan använda applikationen inom flertal olika områden.

Ett potentiellt problem med mer generella studier, så som tidigare exempel, redogörs av Manouselis, Vuorikari och Van Assche (2010). De menar att det endast finns ett fåtal studier där rekommendationsmotorer implementerats, testats och evaluerats på ett representativt för en specifik domän. Resultaten från sådana studier anser de därav vara något missvisande (ibid). Ett exempel på en sådan studie är den genomförd av Bobadilla, Serradilla och Hernando (2009). Kortfattat handlar deras studie om att möjliggöra för en rekommendationsmotor att särskilja användares kunskapsnivå. Evalueringen av rekommendationsmotorn utgick däremot ifrån ett omtalat dataset från ett företag vid namn MovieLens. Datasetet representerar användares betyg av diverse filmer, alltså är datasetet av helt annan kontext i jämförelse med önskvärt användningsområde.

5 Metodval

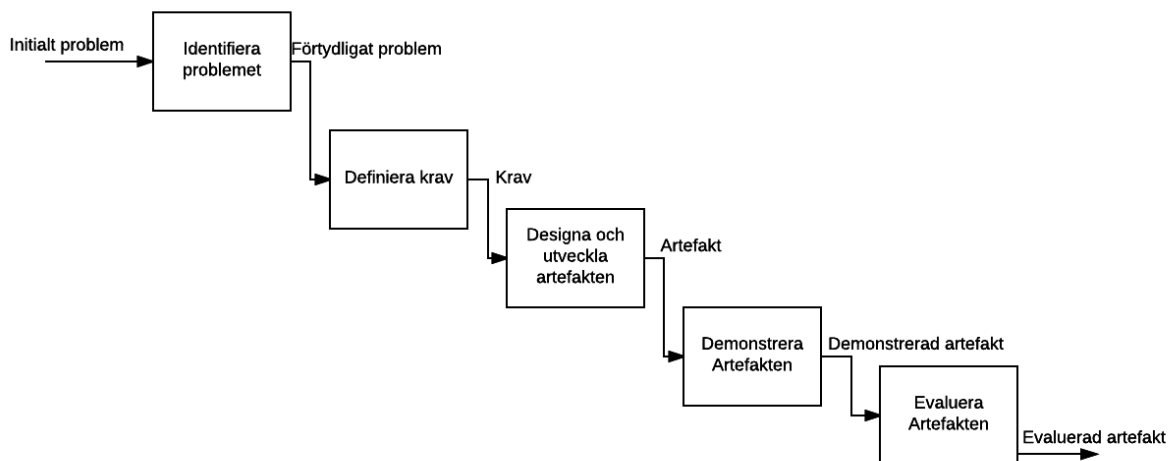
I detta kapitel kommer vi att redogöra för de forskningsstrategier och metodval som gjorts, för att strukturera arbetet genom studien.

5.1 Design Science Research

Vi har valt att arbeta enligt ett ramverk inom *Design science research* (DSR). Ramverket vi valt att arbeta med introducerades av Johannesson och Perjons (2014). En förutsättning för att kunna besvara studiens frågeställning är att implementera en applikation som löser problemet beskrivet i föregående kapitel. Därav begränsades möjliga arbetsmetoder betydligt. Ett alternativ till DSR som också bygger på implementering av en artefakt är *Action Research*. Men enligt Järvinen (2007) är båda dessa metoder väldigt lika. Det som till största del skiljer metoderna åt är hur metoderna förhåller sig till verkligheten. DSR utgår ifrån verkligheten och försöker att lösa problem som existerar i den. Inom Action Research modifieras istället verkligheten, alternativt så sker utveckling av ett helt nytt system. Nomp är en mindre verksamhet med få tillgängliga resurser, alltså fanns det lite utrymme och motivering till förändringar i allt för stor omfattning. Inom tidsramen för studien var det också mer rimligt att inte utveckla ett helt nytt system. Baserat på dessa anledningar ansågs DSR vara bättre passande för studien.

5.2 Ramverket

Ramverket är i huvudsak uppdelad i fem stycken aktiviteter, vilka upprepas iterativt under arbetsprocessen. Aktiviteterna för respektive iteration beskriver Johannesson och Perjons (2014) enligt följande flödesdiagram:



Figur 6: Flödesdiagram arbetsprocess

Eftersom att arbetsprocessen är iterativ påpekar Johannesson och Perjons (2014) att pilarna för respektive aktivitet snarare ska ses som in- och output till aktiviteterna. Inte nödvändigtvis att aktiviteterna ska ske sekventiellt. Vid evaluering av en artefakt är det, till exempel, tänkbart att upptäcka nya problem eller brister. Då finns eventuellt ett behov att gå tillbaka till första aktiviteten, identifiera problemet (ibid).

5.2.1 Datainsamlingsmetod

Artefakten och resultat från evalueringar kommer att utgöra en förutsättning för att besvara studiens forskningsfråga. För att kunna evaluera artefakten behövs någon typ av data för att simulera hur artefakten skulle prestera i ett verkligt scenario. Vi nämnde i kapitlet för relaterat arbete att Manouselis, Vuorikari och Van Assche (2010) poängterat ett potentiellt problem med att inte använda data relaterat till kontexten. Den data som kommer att användas i denna studie är därför verklig, anonymiserad data från Nomp's användare. Förhoppningen är att vi därav ska kunna undvika missvisande resultat genom att använda data direkt från domänen.

Eftersom datan kommer att behöva presenteras som dataset kommer viss information om individernas data att plockas ut för att generera dessa dataset. Syftet är alltså att datasetet kommer att användas av artefakten för diverse evalueringar.

5.3 Strategier

Johannesson och Perjons (2014) poängterar att inom DSR är det vanligt att ibland använda olika typer av forskningsstrategier, då vissa aktiviteter ha behov av olika angreppssätt. Vidare förklarar de även att alla aktiviteter inte nödvändigtvis utförs på samma djup. Vanligt är att huvudfokus brukar vara på en eller två av aktiviteterna, beroende på studien och/eller problemet i fråga (ibid). Vidare kommer vi att redogöra för de strategier som valts för studien. Studien har varit av en mer utvecklings- och evalueringsfokuserad karaktär.

5.3.1 Undersökning

För att försöka öka studiens validitet genomfördes även en enkätundersökning. Beslutet för undersökningens utförande baserades på det som för studien verkade mest relevant. Även på grund av praktiska skäl och/eller bristande resurser uteslöts vissa alternativ. Bristande resurser i detta fall var tidsramen för studien men också inom verksamheten. Ett önskvärt alternativ var att distribuera en enkätundersökning i samband med användande av applikationen. Alltså att en eller flera rekommendationer ges för användaren och att direkt få feedback från användaren. Detta alternativ diskuterades men valdes bort på grund av tidigare nämnda skäl.

Ett mer rimligt alternativ som diskuterades var att ta kontakt med en eller flera mattelärare för att få bekräftelse eller påpeka svagheter hos rekommendationerna. Vidare efterforskning ledde till insikten att det är svårt att inom tidsramen för studien, få tag i tillräckligt många lärare för att göra resultatet intressant. Utöver detta är det svårare att översätta rekommendationerna till en eller flera personer som inte alls är insatta i applikationen. Däremot finns det lärare som använder applikationen och som besitter expertkunskap inom matematik. Dessa lärare har förmodligen även ett genuint intresse för att uppgifter som rekommenderas, för potentiellt deras egna elever, håller en viss standard. Respondenterna bestod till slut av sju stycken kvalificerade lärare inom matematik. Analysens utförande påverkades som resultat av urvalsgruppens storlek, däremot ansågs fördelarna med experter inom området väga något tyngre. En betydande fördel är att faktiskt få återkoppling från några användare av applikationen och som bonus dra nytta av deras kompetens. Enkäten utformades på ett kvantitativt angreppssätt, men där vi ändå som slutfråga gav möjlighet för ytterligare kommentarer. Ett absolut tänkbart alternativ hade varit att göra en mer kvalitativ studie med enbart öppna frågor. Men eftersom att resterande data i studien är av något mer kvantitativt slag så togs beslutet att göra undersökningen enligt samma slag.

5.3.2 Inledande experiment

Eftersom att ramverket stödjer ett iterativt arbetssätt utfördes olika typer av experiment i efterföljande iterationer. Det inledande experimentet för studien hade som syfte att utvärdera

artefaktens möjlighet att identifiera olika typer av användargrupper. Intresset baserades på det faktum att rekommendationsmotorn bör kunna ge rekommendationer oavsett typ av användare. För studien valdes fyra elever ut där vissa indikationer tydde på att dessa var olika typer av användare. Ett alternativt angreppssätt hade varit att slumpmässigt välja ut ett antal elever. Detta beslut uteslöts, även om sannolikheten är liten så fanns ändå möjligheten att liknande elever hade valts.

5.3.3 Överraskning, Träffsäkerhet, Variation etc

För att återkoppla till teorikapitlet (3.5 Evaluering) så förklarades ett antal utvärderingsmått som är ganska vanligt återkommande i studier för rekommendationsmotorer. Alltså ansågs det passande att även i denna studie utvärdera rekommendationsmotorn med dessa utvärderingsmått. Utvärderingen skedde genom tillämpning av tidigare nämnda utvärderingsmått, det vill säga: MAE, intra- och inter-användarvariation, nivå av överraskning samt täckningsgrad.

5.3.4 Parameteroptimering

För att ytterligare evaluera rekommendationsmotorn så genomfördes även en parameteroptimering. Val av optimala parametrar betyder inte nödvändigtvis den optimala lösningen för problemet. Däremot ger det möjlighet för ett så bra utfall som möjligt för lösningen som presenteras i studien.

5.4 Etiska aspekter

Under studiens fortgång finns det även ett antal viktiga etiska aspekter som tagits i beaktning. Denna studie görs på uppdrag av en arbetsgivare, det är därför av stor vikt att belysa vår målsättning att utföra den på ett så objektiva arbetsätt som möjligt. Framförallt för att inte påverka de resultat som studien möjligtvis kan tillföra. Wright (2007) redogör för vissa viktiga perspektiv, bland annat, det faktum att system är gjorda av människan. Det finns därför stora möjligheter för accidentell eller medveten subjektivitet. Förhoppningarna är att resultaten även har nytta för uppdragsgivaren. Däremot har arbetsprocessen skett i åtanke att försöka upprätthålla objektivitet. Det som ytterligare styrker vår objektivitet är att vi haft fria tyglar att använda de verktyg, metoder och tillämpningar vi själva önskat. Förfrågan om uppdraget gjordes även den på vårt bevåg, vilket gör att uppdragsgivaren inte pressat oss för ett förväntat resultat. Vad gäller datan vi har fått tillgång till är anonymiserad data från verksamhetens användare vilket också är en viktig etisk aspekt för användarna.

Viktiga etiska aspekter inom DSR är även att ifrågasätta vilken etisk påverkan artefakten kan ha på sin omgivning. Aspekter så som, tänkbara resultat av att rekommendera en för lätt eller en för svår uppgift för en elev. Alla möjliga utfall är svårt, om inte omöjligt att lista eftersom att alla individer reagerar olika. Däremot är tänkbara scenarion av att rekommendera en för svår uppgift till en elev känslor av uppgivenhet och/eller frustration. Beroende på ålder kan det möjligtvis vara lättare eller svårare att skaka av sig, eftersom att rekommendationerna görs av ett system. Resultatet blir oavsett en ganska negativ uppfattning och användaren kan således tappa förtroendet för systemet. Likaväl kan en användare möjligtvis tappa förtroende för ett system som rekommenderar för lätta uppgifter. Förmodligen blir det däremot inte samma emotionella effekt som att rekommendera för svåra uppgifter. Sammanfattningsvis borde en rekommendation för användaren som är något lättare föredras i jämförelse med en svårare.

6 Metodtillämpning

Detta kapitel redogör för hur vi har arbetat med de olika faserna enligt det ramverk som framtagits av Johannesson och Perjons (2014). Ramverkets första aktivitet handlar om att identifiera och definiera problemet vilket vi redogjort för i rapportens inledande kapitel som resulterar i en problemformulering. Vidare kommer detta kapitlet alltså handla om resterande fyra aktiviteter vilket är: definiera krav, implementering av artefakten, demonstration av artefakten, evaluering av artefakten.

6.1 Definiera krav

Utifrån Johannesson och Perjons (2014) ramverk handlar denna aktivitet om att definiera de krav artefakten har som avsikt att uppfylla. De krav som gavs från uppdragsgivaren formulerades ifrån början som ett önskemål angående en rekommendationsmotor som fungerar mot uppdragsgivarens verksamhet. I dagsläget har verksamheten ingen möjlighet att rekommendera uppgifter för deras användare vilket utgör huvudproblemet. Uppdragsgivaren hade inga specifika åsikter kring val av representation, programspråk eller övriga icke-funktionella krav. Ett krav var däremot att systemet skulle kunna hantera en stor mängd data eftersom att informationen verksamheten hanterar är ganska omfattande.

Erhållna data fanns tillgänglig i en SQL databas där viss information var av intresse för applikationen. Verksamheten saknar möjlighet för användaren att ge explicit feedback, vilket gör att datan endast informerar om individernas användande. Alltså finns information om exempelvis vilka uppgifter som gjorts, inom vilka områden, hur många gånger etc. Med andra ord finns ingen direkt koppling till hur en elev upplevt en uppgift, till exempel, svårighetsgrad. Alltså finns behov för att konstruera implicita dataset, som representation för användarnas åsikter.

Ytterligare information som saknas är information om användarnas ålder eller årskurs. Alltså ingen direkt information om över eller underrepresenterade användare. En hypotes var att uppgifterna som gjorts minst, borde representera den minst frekventa typen av användare. Av totala antalet uppgifter kontrollerades därför den tredjedel av uppgifterna som gjorts minst, för att identifiera vilken årskurs dessa tillhörde. Av dessa var 64% av uppgifterna på högstadienivå, vilket ger stöd för två generella användargrupper. Användargrupperna utgörs alltså av: låg- och mellanstadieelever samt högstadieelever, där högstadieelever är underrepresenterade.

Vidare siktar vi mot att utgå ifrån bästa praxis inom systemutveckling. Utifall verksamheten önskar att lägga till/ändra i implementationen i framtiden ska modulerna inte vara för hårt kopplade till varandra. Andra praxis inom systemutveckling som artefakten ska uppfylla är namnstandarder, vettigt dokumenterad kod samt obefintlig redundant kod. Dessa skäl förekommer framförallt då syftet är att verksamheten ska ha nytta av artefakten. Användning av goda principer inom systemutveckling underlättar dessutom för förvaltning av produkten.

Baserat på problemformulering, de krav vi tillhandahållits av uppdragsgivare, valet av implementation samt teoretisk bakgrund kan vi sammanfatta kraven för systemet enligt följande:

Icke funktionella krav:

- *Artefakten ska vara utvecklad i programspråket Python med hjälp av Apache Spark's ramverk, Pyspark*
- *Artefakten ska byggas enligt bästa praxis inom systemutveckling vad gäller löst kopplade moduler, namnstandarder, vettigt dokumenterad kod samt upprepning av kod.*

Funktionella krav:

- *Artefakten ska kunna ge rekommendationer för systemets olika användargrupper.*
 - *Inhämtning av användarinformation ska ske från verksamhetens databas.*
 - *Informationen hämtad från verksamhetens databas ska användas av artefakten för att skapa implicita dataset.*
 - *Dataseten ska användas för att generera rekommendationer för verksamhetens användare.*
-

6.2 Implementera artefakten

Implementeringen av artefakten har skett i ett antal iterationer där vissa av verktygen Johannesson och Perjons (2014) förslår för aktiviteten har tillämpats mer eller mindre regelbundet. De föreslår, exempelvis, att man bör ha brainstormingsessioner i samband med utvecklingen av artefakten. Vi har istället för brainstormingsessioner arbetat med att fånga idéer under hela arbetet med studien, inte enbart med utveckling av artefakten. Därav ansåg vi inte behovet att ha utsatta tidpunkter för detta. De idéer som fångades upp utreddes för att ta beslut om att gå vidare med idén eller inte.

6.2.1 Skissa och bygga

För att få en översiktlig bild på hur artefakten skulle kunna struktureras så konstruerades från början en översiktlig systemskiss. Systemskissen illustrerade de övergripande moduler som ansågs vitala för att uppnå de mål som identifierats för artefakten. Implementeringen påbörjades genom att konstruera de identifierade modulerna med lättare funktionalitet. Succesivt under iterationernas fortgång utökades modulernas funktionalitet. De mest centrala delarna i artefakten kommer att redogöras i resterande del av detta kapitel.

6.2.1.1 Inhämta information från verksamhetens databas

För att kunna ge rekommendationer för verksamhetens användare behövde först dataseten skapas. Som tidigare nämnt har inte verksamheten idag något stöd för explicit feedback, vilket lämnar alternativet att skapa dataset utifrån implicit feedback. Däremot behövde informationen hämtas ifrån verksamhetens databas. Alltså fick en modul i artefakten ansvaret för att inhämta den information som ansågs intressant. Informationen var vilka uppgifter som slutförts för respektive elev. För respektive uppgift ansågs dessutom följande information vara intressant: antalet fel eleven hade på uppgiften, starttid, sluttid samt pausad tid. Starttid, sluttid och pausad tid användes för att räkna ut den aktiva tiden eleven spenderat för att slutföra uppgiften.

Resonemang angående val av indikationerna baserades på följande hypotes. Upplever en elev en uppgift som lätt bör hen ha genomsnittligt mindre fel, än en elev som upplever samma uppgift som svår. Dessutom bör eleven i genomsnitt lägga kortare tid på uppgiften. Vi kan även utesluta inaktivitet som resultat till varför en uppgift har tagit lång tid. Eftersom pausad tid

indikerar den tid eleven har varit inaktiv och subtraheras från den totala tiden mellan start- och sluttid.

Vidare togs beslutet att utesluta information som var mer än ett år gammal. Som tidigare nämnts saknades information kring användarnas ålder. Det ansågs för stor risk för att genomföra rekommendationer på allt för gammal data, det vill säga inaktiva användare. En användare som har varit inaktiv i över ett år kan potentiellt vara i en årskurs högre upp. Rekommendationerna kan möjligtvis vara bra utifrån användarens historik, men inte längre stämma överens med elevens faktiska kunskaper. För att inte riskera att användaren tappar förtroendet för rekommendationsmotorn hanteras elever som varit inaktiva i ett år eller mer som helt nya användare. Då eleven på nytt slutfört en eller flera uppgifter kommer nya rekommendationer att ges baserat på denna data. Hur detta hanteras inom artefakten kommer att beskrivas senare i detta kapitel.

6.2.1.2 Skapa implicita dataset

När informationen hämtats från verksamhetens databas tar en annan modul över för att generera dataseten. För att konstruera de implicita dataseten krävdes dock att representera dessa som ett intervall av möjliga värden. Intervallet beskriver alltså elevernas implicita åsikt angående respektive uppgift. Antalet fel ansågs däremot vara en tydligare indikation på att eleven haft problem med en uppgift. Därav togs beslutet att antalet fel bör väga tyngre än tiden. För att åstadkomma denna representation så multiplicerades elevernas prestation med vikter. Alltså kan värdet som beskriver en elevs implicita åsikt för en uppgift beskrivas enligt följande formel:

$$\text{Implicit åsikt} = \text{Antalet fel} * V_1 + \text{Tid erlagd i minuter} * V_2$$

V_1 och V_2 motsvarar alltså vikterna för indikationerna, vilket fick värdena 0,2 samt 0,1.

Vissa åtgärder vidtogs för att hantera eventuella extremvärden. Extremvärden i detta sammanhang är, till exempel, då eleven gjort orimligt många fel. Ett ytterligare exempel på extremvärden är då eleven har ägnat orimligt lång tid på en uppgift. Här valdes två gränsvärden för vad som ansågs rimligt i tid samt fel per uppgift. För tid ansågs mer än en timma vara orimligt, alternativt mer än 100 fel. För att hantera dessa extremvärden i samband med konstruktionen av dataseten ersattes dessa istället till värdet noll.

På grund av dessa åtgärder blir maximala värdet för en implicit åsikt följande:

$$\text{Maximalt värde för åsikt} = 100 * 0,2 + 60 * 0,1$$

Alltså hamnar intervallet som representerar användarnas implicita åsikter mellan 0 till 26.

6.2.1.3 Rekommendationer för systemets användare

Rekommendationsmotorn hämtar de dataset som skapats från tidigare beskrivna modul för att först och främst träna modellen. Datasetet fördelades till tränings- och valideringsset inför träning för att möjliggöra validering på, för modellen, okända data. Det finns flera olika angreppssätt för att utföra fördelning av dataset inom maskininlärning. Angreppssättet som valdes för modellen var slumpmässigt utvald fördelning. Slumpmässigt utvald fördelning innebär att en post med en viss sannolikhet har möjlighet att bli vald till träningsdata alternativt valideringsdata. Sannolikheten för att bli utvald till valideringsdata valdes till 0,2.

För att kontrollera de prediktioner som rekommendationsmotorn gjorde för träningsdata så behövdes kännedom om vad som är rätt eller fel. Därför maskerades data i träningssetet så att prediktionerna kunde jämföras med faktiska värden, undansparade i valideringssetet. Processen för slumpmässig fördelning kan därför beskrivas enligt följande pseudokod:

```
def randomSplit():
    data = allDataForStudents
    trainingset = []
    testset = []
    foreach student in allDataForStudents:
        if random > 0.2:
            trainingset.Add(student)
        else:
            testset.add(student)
            student.ImplicitValues = 0
            trainingset.Add(student)
```

Vidare så skapas modellen med hjälp av matrisfaktorisering som beskrevs i studiens teorikapitel. Eftersom matrisfaktorisering är ett välstuderat verktyg inom området för rekommendationsmotorer, som också nämndes i studiens teorikapitel, så användes Apache Sparks implementation för matrisfaktorisering (Apache Software Foundation, u.å). Alltså kan skapandet av modellen beskrivas enligt följande pseudokod:

```
def create_model():
    # Build the recommendation model using Matrix Factorization using
    # Alternating Least Squares
    model = ALS.trainImplicit(trainingImplicitData, rank,
                             numberOfIterations, lambda, alpha, seed)
```

Parametrarna för ALS är till stor del självbeskrivande där lamda, alfa och rank behöver ytterligare förtydligande. Alfa är parametern för ALS som avgör modellens konfidensnivå, det vill säga den grad av säkerhet modellen påvisar. Lamda är den parameter som används för Regularization, för att bibehålla en mer generell modell. Denna teknik beskrevs noggrannare i studiens teorikapitel. Rank i sin tur bestämmer hur många kännetecken för dataseten som ska användas för att bygga modellen.

Initialt användes följande parametrar för att träna modellen:

Tabell 2: Initial parameterkombination

<i>Alpha</i>	<i>Lamda</i>	<i>Rank</i>	<i>Iterations</i>
0,01	0,01	16	5

Målsättningen är att alla resultat för studien ska vara reproducerbara. Eftersom slumpmässigt val av träningsdata- och valideringsdata sker, valdes därför ett seed till värdet 11 för att möjliggöra reproducerbarhet.

Under iterationernas fortgång genomfördes flertal aktiviteter för att utvärdera artefakten. Vid de tillfällen evalueringarna påvisade brister i artefakten förändrades ibland parametrarna för att motverka bristerna. Förändringar i parametrarna kommer att redogöras i de delar av studien som förklarar hur evalueringarna utförts.

6.2.1.3.1 Kallstart

Eftersom kravet för artefakten innebär att rekommendationsmotorn ska kunna ge rekommendationer för verksamhetens alla användare. Innebär detta även att artefakten behöver kunna hantera kallstartsproblemet. Kallstartsproblemet för artefakten uppstod till exempel, som vi tidigare nämnt, när eleven varit inaktiv i över ett år. Det finns då ingen tillgänglig data att basera rekommendationer på eftersom att vi valt att utesluta för gammal data. För de angreppssätt som nämntes i studiens teori så uteslöts direkt vissa alternativ. Det fanns exempelvis ingen möjlighet att tillföra extra information om användarna. Om möjligt hade man kunnat göra detta genom verksamhetens applikation. Lösningen blev alltså att använda sig av en naiv metod. För en naiv metod kan, exempelvis, de populäraste uppgifterna rekommenderas för en elev. Dessa har möjligtvis störst sannolikhet att vara av intresse för eleven. Däremot är detta en något ytlig metod och kan möjligtvis påverka användarens förtroende i systemet. Att rekommendera de uppgifter som gjort minst uteslöts också, på grund av den analys av data som förklarades tidigare. Analysen påvisade tydligt att de uppgifter som gjorts minst tillhör i majoritet högstadienivån. Sannolikheten är då stor att användaren får rekommenderat en för svår uppgift. Vikten i att inte rekommendera en allt för svår uppgift relaterar även till de etiska aspekter som diskuterades i 5.4 Etiska Aspekter. Hanteringen för att rekommendera uppgifter för en kallstartsanvändare kan därför beskrivas enligt följande pseudokod:

```
def cold_start_recs():  
  
    studentTasks = allRecordsOfTasksSortedByPopularity  
    count = studentTasks.count()  
    aThirdOfTasks = int(count / 3)  
    top = studentTasks.take(aThirdOfTasks+aThirdOfTasks)  
  
    middle = top.Delete(0 - aThirdOfTasks)  
  
    recommendForStudent = []  
  
    for(i = 0; i < 10; i++):  
        randomTask = middle.pop(random(0, len(middle) - 1))  
        recommendForStudent.Add(randomTask)  
  
    return recommendForStudent
```

Sammanfattningsvis så utesluts alltså först de minst populära tredjedelen av uppgifterna. För att sedan utesluta den mest populära delen av uppgifterna. Av de uppgifter som sedan kvarstår väljs tio stycken slumpmässigt utvalda för att rekommendera för användaren.

6.2.1.4 Olika användargrupper

För att uppnå detta mål i kravställningen krävdes i implementeringen stöd för olika typer av evalueringar. Kan rekommendationsmotorn ge bra rekommendationer för verksamhetens alla användare ger detta i sin tur stöd för att den kan hantera olika typer av användare.

De olika utvärderingsmått som beskrevs i teorikapitlet, implementerades för utvärderingen av rekommendationsmotorn. Implementeringarna skedde enligt de formler som presenterades med undantag för MAE, då denna uträkning fanns som stöd i Apache Sparks ramverk. För samtliga utvärderingsmått räknades ett medelvärde ut, för att få ett mer representativt värde på

rekommendationsmotorns prestation. Nedan redogörs implementeringarna för respektive utvärderingsmått.

Inter användarvariation:

```
all_users = getUserRecommendations(nbr_of_recs_per_users)
number_of_users = all_users.count()
total inter user diversity = 0

for each userI in all_users
  for each userJ in all_users
    if userI != userJ
      Qij = GetNumberOfCommonItems(userI, userJ)
      inter user diversity = 1 - (Qij / nbr_of_recs_per_users)
      total inter user diversity += inter user diversity

mean inter user diversity = total inter user diversity /
                             (nbr_of_users * (nbr_of_users - 1))
```

För uträkningen av inter- användarvariation så användes hälften av datan, på grund av tidsbrist. Uträkningen för all data skulle ta orimligt lång tid att utföra. Beslutet för halva datan grundades på en kompromiss mellan tidsåtgång och tillräcklig mängd data för ett representativt resultat.

Intra användarvariation:

```
nbr_of_users = GetNumberOfUniqueUsers()
all_user_recommendations = getUserRecommendations(nbr_of_recs_per_users)
total intra user diversity = 0

for each recommendations in all_user_recommendations
  similarity = 0
  for itemA in recommendations
    for itemB in recommendations
      if itemA != itemB
        similarity = getSimilarity(itemA, itemB)

  intra diversity = 1 / (self.nbr_of_recs *
                       (self.nbr_of_recs - 1)) * similarity

  total intra user diversity += intra diversity

mMan intra user diversity = total intra user diversity / nbr_of_users
```

Metadatan för att beräkna likheten mellan par av matematiska uppgifter i intra-användarvariation, representerar det matematiska område respektive uppgift hör hemma.

Täckningsgrad:

```
all_user_recommendations = getUserRecommendations(nbr_of_recs_per_users)
unique_products = getUniqueproducts(all_user_recommendations)
nd = unique_products.count()

products_recs = recommendUsersForProducts(1)
numberOfProducts = products_recs.count()
```

```
coverage = nd / numberOfProducts
```

Som nämnt i studiens teorikapitel används täckningsgrad för att mäta artefaktens möjlighet att rekommendera varje enskilt objekt. För implementeringen beräknades alltså varje unik uppgift som rekommenderats, dividerat med samtliga tillgängliga uppgifter.

Nivå av överraskning:

```
number_of_users = getUsers().count()
products = GetProducts()

total_u = 0
for each product in products
    popularity = getPopularity(product)
    surprisal = math.log2(number_of_users / popularity)
    total_u += surprisal

mean surprisal = total_u / number_of_products
```

Nivå av överraskning representerar alltså sannolikheten för att användaren redan är bekant med en uppgift. Populariteten i implementeringen representerade antalet gånger en uppgift blivit slutförd av samtliga användare.

6.2.1.4.1 Parameteroptimering

Utöver utvärderingsmått implementerades även i modulen stöd för parameteroptimering för att säkerställa ett, för denna lösning, optimalt utfall. Istället för tidigare nämnda angreppssätt vad gäller tränings- och valideringsdata, användes här istället 10-faldig-korsvalidering. För varje korsvalidering mättes MAE för att på så sätt ge ett genomsnittligt MAE för varje parameterintervall.

6.3 Demonstrera Artefakten

Enligt Johannesson och Perjons (2014) aktivitet för demonstration av artefakten innebär denna konceptvalidering (Proof of concept) för artefakten. Meningen är att testa artefakten på ett troligt scenario för att påvisa genomförbarhet. Syftet är alltså att visa att artefakten kan användas för att lösa en del eller delar av ett problem. Vidare förklarar Johannesson och Perjons (2014) att en demonstration kan ses som en svag variation av evaluering.

För att demonstrera artefakten utformades ett verkligt scenario genom ett urval av elever från verksamhetens potentiella användargrupper. Scenariot innebär alltså att artefakten bör kunna ge rekommendationer för dessa olika typer av användare. Baserat på hypotesen angående användargrupper som redogjordes i samband med kravframtagningen. Identifierades med större sannolikhet två typer av användargrupper. Låg- och mellanstadieelever utgör en majoritetsgrupp medans högstadieelever utgör en minoritet.

Semi slumpmässigt utvaldes fyra stycken elever utifrån verkliga data för att användas i demonstrationen. Med semi slumpmässigt menar vi att elever slumpmässigt valdes ut för att noggrannare analyseras. Vidare togs slutsatser, baserat på användarnas gjorda uppgifter vilken klass respektive elev sannolikt hör hemma. För respektive användargrupp valdes dessa även utifrån en elev som generellt verkade göra fler fel än en motsvarande elev i samma grupp. Sammanfattningsvis valdes alltså två elever från låg- och mellanstadiet samt två elever från högstadiet.

Artefakten demonstrerades sedan genom ett experiment, vars syfte var att rekommendera tio stycken uppgifter för respektive användare. För att alltså påvisa att rekommendationsmotorn kan användas för att delvis eller helt lösa detta problem.

6.4 Evaluera artefakten

Johannesson och Perjons (2014) förklarar att meningen med evalueringen är att undersöka till vilken grad artefakten uppfyller de krav som sattes i 6.1 Definiera krav. Målsättningen är också att evalueringen ska undersöka till vilken grad artefakten kan lösa problemet som studerats. För evaluering av artefakten tillämpades i olika iterationer diverse strategier. I följande delar av rapporten kommer vi att redogöra för dessa.

6.4.1 Undersökning

En av strategierna som tillämpades var undersökningen som nämndes i tidigare kapitel för metodval. Syftet med undersökningen var att inhämta feedback för den demonstration som gjorts för artefakten, därav upptäcka eventuella styrkor eller brister. Framförallt låg fokus på att identifiera skillnader för rekommendationer mellan användargrupperna. Det vill säga om rekommendationsmotorn gör bättre eller sämre rekommendationer beroende på grupp.

Frågorna för undersökningen konstruerades alltså baserat på resultaten ifrån demonstrationen. Eftersom tio stycken rekommendationer gjordes för respektive elev ansågs det inte rimligt att begära bedömning av lärarna för samtliga rekommendationer. Därför summerades de rekommenderade uppgifterna till stor del rekommenderade matematiska områden, dessa områden representerar respektive fråga i undersökningen. Med till stor del menar vi med undantag för vissa frågor, som istället representerar områden där eleven inte tidigare hade gjort några uppgifter. För elev A är fråga 3 en sådan rekommendation. Dessa valdes för att noggrannare undersöka om detta var en önskvärd egenskap hos artefakten.

Det fullständiga frågeformuläret finns att studera i Appendix B för denna rapport. Vidare redogörs däremot en sammanfattning av de frågor som ställdes i undersökningen.

Tabell 3: Frågor från undersökningen, elev A

Elev A har tidigare gjort uppgifter inom följande områden:	<ul style="list-style-type: none"> • Enheter klass 4–5 • Geometri klass 6 • Division klass 5 • Multiplikation klass 4 • Bråktal klass 4–5
Elev A har haft svårigheter inom följande områden:	<ul style="list-style-type: none"> • Enheter • Bråktal
Fråga 1:	Division för klass 5, svårare uppgift än de eleven tidigare gjort.
Fråga 2:	Enheter för 5:e klass, samma nivå som eleven tidigare haft svårigheter för.
Fråga 3:	Statistik för klass 6

Tabell 4: Frågor från undersökningen, elev B

Elev B har tidigare gjort uppgifter inom följande områden:	<ul style="list-style-type: none"> • Bråktal klass 8 • Geometri klass 7–8 • Algebra klass 8 • Aritmetik klass 6–8 • Enheter klass 3–8 • Multiplikation klass 8 • Procent klass 8 • Division klass 5–8
Elev B har haft svårigheter inom följande områden:	<ul style="list-style-type: none"> • Enheter • Aritmetik
Fråga 4:	Bråktal för klass 5–6, lättare uppgifter än de eleven tidigare gjort.
Fråga 5:	Bråktal för klass 8, på samma nivå som eleven gjort tidigare.
Fråga 6:	Algebra för klass 6, lättare uppgift än eleven tidigare gjort.

Tabell 5: Frågor från undersökningen, elev C

Elev C har tidigare gjort uppgifter inom följande områden:	<ul style="list-style-type: none"> • Division klass 8 • Procent klass 8 • Enheter klass 8 • Multiplikation klass 8 • Bråktal klass 8
Elev C har haft svårigheter inom följande områden:	<ul style="list-style-type: none"> • Enheter • Bråktal • Division
Fråga 7:	Bråktal för klass 5–6, lättare uppgifter än de eleven tidigare gjort.
Fråga 8:	Enheter för klass 8, på samma nivå som eleven gjort innan.
Fråga 9:	Enheter för klass 9, svårare uppgift än eleven gjort innan.

Tabell 6: Frågor från undersökningen, elev D

Elev D har tidigare gjort uppgifter inom följande områden:	<ul style="list-style-type: none"> • Multiplikation klass 3–4 • Geometri klass 4
Elev C har delvis haft svårigheter inom:	<ul style="list-style-type: none"> • Multiplikation
Fråga 10:	Multiplikation för klass 3, på samma nivå som eleven tidigare har gjort.
Fråga 11:	Multiplikation för klass 2, lättare nivå än de uppgifter eleven tidigare gjort.
Fråga 12:	Multiplikation för klass 4, högre nivå än de uppgifter eleven tidigare har gjort.

På grund av mängden data för respektive elev redovisas inte all data för respektive elev i denna rapport, däremot finns datasetet för elev A att studera i Appendix A. Baserat på samma anledning var det inte rimligt att presentera hela datasetet för lärarna. För att ge lärarna bakgrundsinformation angående respektive elev sammanställdes alltså de matematiska områden respektive användare gjort flest uppgifter. Angående användarens påvisade svårighet, baserades detta på en ökning i antalet fel eleven tidigare påvisat inom andra områden. För datasetet som finns tillgänglig i Appendix A kan exempel på detta identifieras. För eleven är det inte ovanligt med ett par fel på diverse olika uppgifter. Däremot kan man identifiera en betydlig ökning framförallt inom bråktalet i fjärde- och femteklass.

Respektive fråga i undersökningen representerar en av de tio rekommendationer som artefakten gjorde i samband med demonstrationen. Lärarna fick sedan möjlighet att rangordna dessa rekommendationer utifrån en skala mellan 0 och 10. De fall där lärarna överensstämmer till stor grad med rekommendationen ombads läraren att ange 10, respektive 0 om lärarna misstyccker helt och fullt.

Under genomförandet av enkätundersökningen vidtogs vissa etiska åtgärder. Respondenternas samtycke att delta i undersökningen är, tillexempel, en viktig aspekt (Turner, 1982). Enkätundersökningen genomfördes genom ett online formulär, alltså erbjöds friheten att delta eller inte. För att däremot väcka intresse hos respondenterna så kontaktades dessa innan undersökningen distribuerades. Undersökningen hölls dessutom anonym för att informationen inte på något sätt skulle bli känslig eller personlig. En anonym undersökning ger alltså större möjlighet för att inte på något sätt påverka lärarnas svar.

Undersökningen var tillgänglig för respondenterna under en period av två veckor, då detta ansågs som rimlig tidsram för att hinna svara på enkäten. Respondenterna meddelades alltså på förhand och även vid tillfället då enkäten blev tillgänglig. Av de sju lärare som bjöds in att delta i undersökningen var det sex stycken som valde att delta.

Som också nämndes i metodvalet påverkas analysen för en undersökning med ett fåtal respondenter. Den feedback som däremot inhämtades är åsikter från användare av systemet och dessutom experter inom området. Feedbacken utgjorde därför en betydelsefull insikt för vidare utforskning och/eller förbättring av artefakten.

6.4.2 Experiment

Både svagheter och styrkor hos artefakten identifierades som resultat av feedbacken från undersökningen. Vissa åtgärder vidtogs för att motverka svagheterna hos artefakten. En förändring som gjordes var att förändra algoritmens alfa värden, den som alltså styr algoritmens konfidensnivå. För Apache Sparks implementering av matrisfaktorisering (Apache Software Foundation, u.å), i de fall inget annat värde anges som alfa-värde, används värdet 0,01 som standardvärde. Detta var, som tidigare nämnts, artefaktens initiala värde då inget annat hade angetts. Förändringen som genomfördes var alltså att ändra detta till värdet 40. Motiveringen bakom just detta värdet baserades på en studie (Hu, Koren, & Volinsky, 2008) där en parameteroptimering av matrisfaktorisering algoritmen genomfördes. Alltså ansågs detta vara en passande utgångspunkt för att vidare utforska svagheterna hos artefakten.

Syftet med detta experiment var förutom ovan nämnda anledning, att kontrollera artefaktens möjlighet att ge rekommendationer för samtliga av verksamhetens användare. För att utforska detta noggrannare användes samtliga utvärderingsmått som implementerades i artefakten. Implementeringen av dessa beskrevs noggrannare i föregående del av rapporten. Samtliga evalueringar upprepades 10 gånger för att styrka reliabiliteten hos resultaten.

6.4.3 Parameteroptimering

Som tidigare nämnts utfördes även ett experiment i form av en parameteroptimering. Syftet med parameteroptimeringen var att säkerställa ett så bra utfall som möjligt för lösningen som presenteras i denna studie. Förutom detta var motiveringen även att undersöka om förändring av parametrarna kunde resultera i bättre resultat.

För att jämföra resultat mellan parametrarna jämfördes dessa baserat på utvärderingsmättet MAE. Parameteroptimeringen skedde genom att testa alla möjliga kombinationer av ett visst intervall av värden. Nedanstående tabell redogör för respektive intervall för samtliga parametrar.

Tabell 7: Intervall för parameteroptimeringen

<i>Lamda</i>	0.001	0.01	0.1	1.0	5.0	10.0		
<i>Rank</i>	1	3	6	10	16	20		
<i>Iterations</i>	1	3	5	7	9	10	20	
<i>Alpha</i>	0.01	1.0	5.0	10.0	20.0	30	40.0	100.0

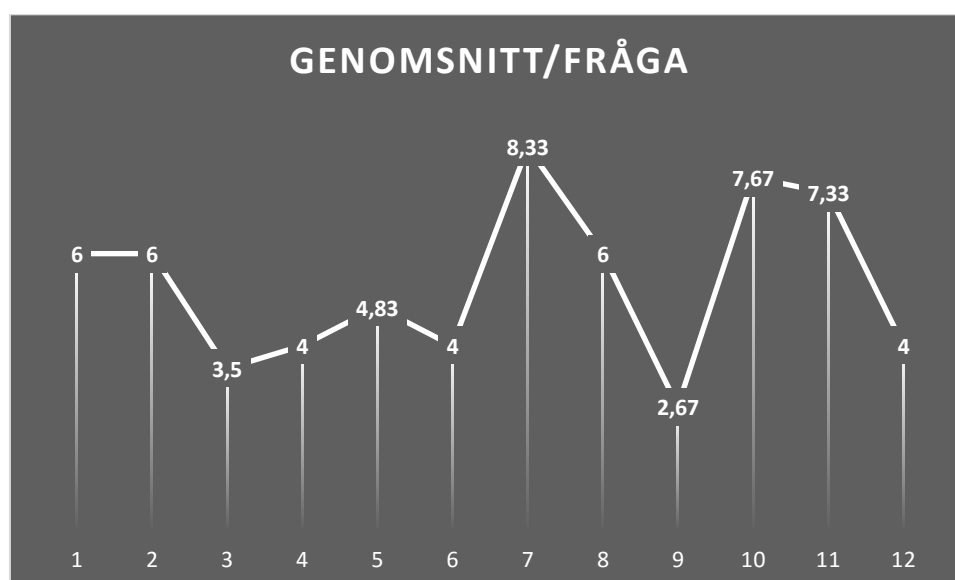
Testning av samtliga kombinationer resulterar i 2016 olika möjliga kombinationer. Detta ansågs vara tillräckligt uttömmande samt en rimlig tidsåtgång.

7 Resultat

I denna del av rapporten kommer samtliga resultat utifrån studien att presenteras. Resultaten kommer att presenteras utifrån aktiviteterna som genomförts. Som även beskrevs i tidigare kapitel för metodtillämpningen. Syftet med aktiviteterna var att undersöka till vilken grad artefakten uppfyller de krav som sattes i 6.4 Definiera krav. Inte minst att undersöka i vilken omfattning artefakten löser problemet som formulerades i rapportens inledande kapitel. Analysen kommer därför ligga till grund för hur väl artefakten anses uppfylla dessa aspekter.

7.1 Resultat ifrån undersökningen

En sammanställning av respondenternas svar representeras i Figur 9. Figuren visar den genomsnittliga summan för de svar som gavs av respondenterna för respektive frågeställning. Siffran för varje stapel motsvarar ordningen för frågorna i undersökningen, det vill säga stapel ett motsvarar fråga ett.



Figur 7: Respons från undersökningen

Samtliga frågor redogjordes för i avsnittet för i metodtillämpningen och finns även att studera i ursprungligt format i Appendix B. För att underlätta läsning visar nedanstående tabell vilka frågor som hör till respektive elev samt användargrupp.

Tabell 8: Frågor tillhörande respektive elev och användargrupp

Elev:	Fråga Nummer:	Användargrupp:
Elev A:	1–3	Låg/Mellanstadiet
Elev B:	4–6	Högstadiet
Elev C:	7–9	Högstadiet
Elev D:	10–12	Låg/Mellanstadiet

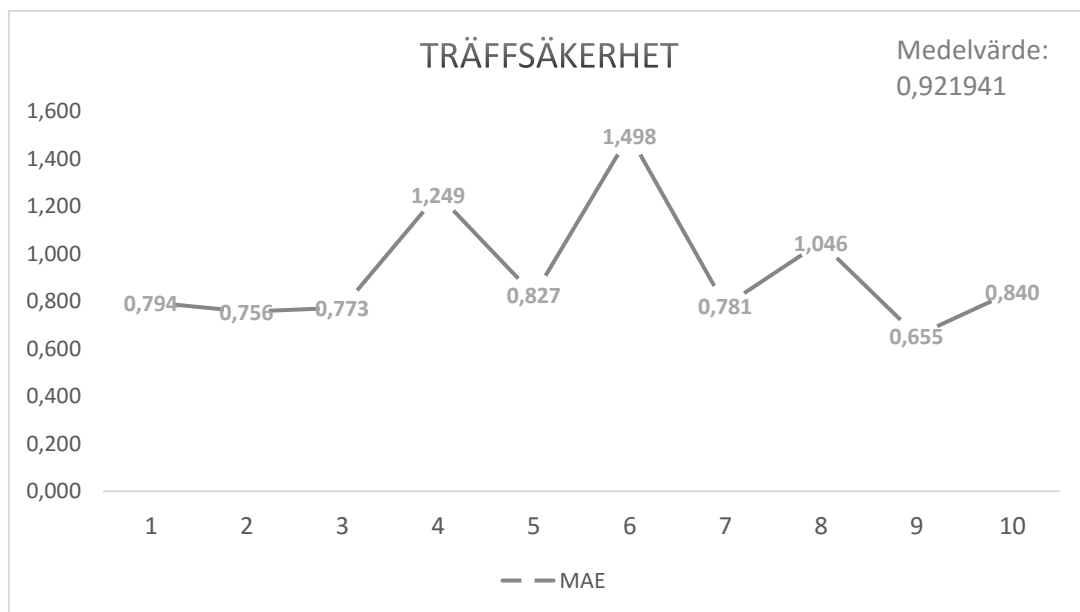
Noggrannare analys av resultaten gav indikationer för vilka rekommendationer lärarna tyckte var bättre respektive sämre. Initialt var förväntningarna att rekommendationsmotorn skulle prestera något sämre för verksamhetens minoritetsgrupp, högstadieelever. Av lärarnas åsikter

att döma har rekommendationerna för högstadieelever både fått sämsta och bästa resultat. Ett genomsnitt för samtliga av lärarnas bedömningar ger en bedömning på cirka 5,36. Detta i sin tur ger en överblick för vilka rekommendationer som hamnar under genomsnittet, det vill säga rekommendation: 3–6, 9 och 12. Av dessa sex uppgifter är fyra av dessa på högstadienivå. Alltså finns det visst stöd för vår initiala förväntning, att rekommendationsmotorn presterar något sämre på verksamhetens minoritetsgrupp. Analysen gav alltså tydliga indikationer på utrymme att förbättras. Samtidigt kan dock de högre presterande rekommendationerna tyda på funktionalitet hos artefakten värd att bevara.

7.2 Evaluering

I kapitlet för metodtillämpning där utförande av respektive aktivitet förklarades, nämndes bland annat att respektive utvärderingsmått upprepades tio gånger. I denna del av rapporten kommer vi att redogöra för resultaten för dessa utvärderingsmått.

Nedanstående figur visar resultat för varje mätning av artefaktens träffsäkerhet (MAE). Varje körning illustreras i x-led och varje punkt motsvarar resultatet för respektive exekvering.



Figur 8: Träffsäkerhet / MAE

Ett medelvärde för samtliga körningar resulterade i ett värde på ungefär 0,92, vilket redovisas i figurens högra hörn. Som också nämndes i studiens teori, är det önskvärt med ett MAE så nära noll som möjligt. Ett värde på exakt noll betyder motsvarar 100% träffsäkerhet. Däremot innebär 0,92 att modellen genomsnittligt var 0,92 fel i sina prediktioner. Med tanke på mängden möjliga värden, det vill säga intervallet 0 – 26 som utgör elevernas potentiella implicita åsikt, är resultatet inte överdrivet illa.

Förutom MAE innebar evalueringen även att mäta följande utvärderingsmått: Täckningsgrad, Intra- och inter användarvariation samt förvåning. Dessa resultat redovisas i nedanstående tabell:

Tabell 9: Intra- och inter användarvariation, täckningsgrad samt förvåning

Täckningsgrad: 79 %	Intra användarvariation: 0,3311	Inter användarvariation: 0,9095	Förvåning: 11,3785
------------------------	------------------------------------	------------------------------------	-----------------------

Dessa resultat gav vissa betydelsefulla indikationer. Som beskrivet i teorikapitlet bör, exempelvis, nivån av förvåning vara så lågt som möjligt. Närmare analys av detta gav däremot insikten att det är svårt i en så pass begränsad domän att påverka nivån av förvåning ytterligare. Förutom att domänen är relativt begränsad så bygger applikationens uppgifter på den naturliga hierarki för studieplanerna inom matematik, som också beskrevs i studiens teori. För en fiktiv elev på lågstadienivå som har gjort två uppgifter inom addition är det därför svårt att skapa stor förvåning med, exempelvis, en uppgift i subtraktion. Man kan inte heller förvåna eleven med en uppgift i algebra på högstadienivå, vilket möjligtvis hade förvånat men möjligtvis gjort mer skada än nytta. Nackdelarna med detta är något som diskuterats tidigare i rapportens, alltså ingenting att överväga. Sammanfattningsvis kan en högre värde här vara att föredra med tanke på domänen som studerats.

Liknande slutsats togs för artefaktens täckningsgrad. Många av de uppgifter som gjorts minst, eller inte alls, är på högstadienivå. Dessutom krävs av applikationen att vissa uppgifter slutförs innan uppgifter på högre nivå kan påbörjas. Därav förväntades inte artefakten ha perfekt täckningsgrad. Baserat på detta bedömdes täckningsgraden vara av acceptabel karaktär.

För inter användarvariation är det önskvärt med ett högre värde, vilket påvisar större variation mellan användares respektive rekommendationslistor. Ett extremt fall av perfekt variation är då alla elever får rekommenderat unika uppgifter. Problemet med detta inom denna domän uppstår, till exempel, då det finns två elever som är ungefär jämförbara. Eleverna har alltså gjort ungefär samma uppgifter, har samma styrkor, svagheter och så vidare. I denna situation är det önskvärt att eleverna får liknande rekommendationer, därav är inte helt perfekt inter användarvariation att föredra. Även här bedöms alltså resultaten för inter användarvariation vara acceptabla.

Intra användarvariation bör däremot, enligt rekommendationer från teorin, ha ett lägre värde för att indikera större variation. Detta utvärderingsmått indikerar istället variation i respektive användares egna rekommendationslista. För domänen ifråga är det däremot tänkbart att användaren får många liknande uppgifter, i jämförelse med de eleven tidigare har gjort. Det kan till och med vara att fördra lägre variation för att säkerställa rekommendationernas relevans. Baserat på dessa resonemang ansågs alltså dessutom resultaten för intra användarvariation vara acceptabla.

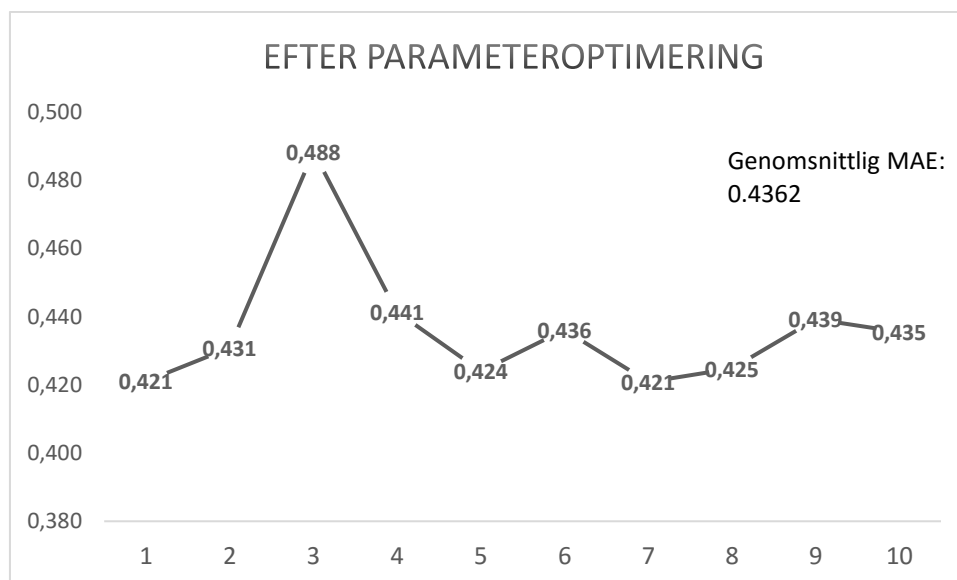
7.3 Resultat från parameteroptimeringen

På grund av mängden kombinationer som testades var det inte rimligt att ta med alla resultat för alla kombinationer i studien. Den parameterkombination som däremot genererade lägst MAE var följande parameterkombination:

Tabell 10: För lösningen optimal parameterkombination

<i>Alpha</i>	<i>Lamda</i>	<i>Rank</i>	<i>Iterations</i>
40	10	6	3

MAE för denna parameterkombination resulterade i 0,41579128378354896, det vill säga ungefär 0.42. För samma typ av evaluering som gjordes tidigare där resultaten var 0,92 resulterade med de nya parameterinställningarna följande resultat:



Figur 9: Mae efter parameteroptimering

Alltså kan man baserat på resultaten från innan och efter parameteroptimeringen se att marginalen för genomsnittlig fel prediktion halverades.

7.4 Rekommendationer

Denna del av resultaten kommer att redogöra för de rekommendationer som har gjorts i samband med evalueringarna för att tydliggöra dess skillnader.

För att förstå resultatens krävs dock ytterligare förtydligande kring formaten på respektive uppgift. Respektive uppgift har ett prefix i form av två siffror, där första siffran står för årskursen uppgiften hör hemma. Andra siffran indikerar vilken nivå inom applikationen uppgiften tillhör. Det vill säga 61.FRACCIONS, innebär en uppgift inom bråktal för 6:e klass, där 62.FRACCIONS hade varit en nivå högre än förstnämnda. För respektive uppgift finns det också ett suffix som anger vilken uppgift bland ett flertal på samma nivå. Det vill säga 61.FRACCIONS.1 och 61.FRACCIONS.2 är uppgifter inom samma årskurs och nivå, men 61.FRACCIONS.2 är en efterföljande uppgift av den förstnämnda.

Nedanstående tabell redogöra för de rekommendationer som undersökningen baserades på. Alltså var dessa de initiala rekommendationerna som artefakten genererade.

Tabell 11: Initiala rekommendationer

Elev A	Elev B	Elev C	Elev D
51.DIVISION.1	61.FRACTIONS.1	61.FRACTIONS.1	33.MULTIPLICATION.3
51.NUMBERS.4	61.NUMBERS.3	61.NUMBERS.3	34.MULTIPLICATION.2
51.UNITS.3	54.FRACTIONS.1	53.FRACTIONS.3	33.MULTIPLICATION.2
51.NUMBERS.1	61.ALGEBRA.1	61.FRACTIONS.2	21.MULTIPLICATION.4
33.MULTIPLICATION.3	31.GEOMETRY.2	11.ADDITION.8	32.MULTIPLICATION.2
53.DIVISION.1	63.NUMBERS.2	51.FRACTIONS.1	34.MULTIPLICATION.1
51.UNITS.1	51.FRACTIONS.1	61.NUMBERS.4	11.TIME.1
43.MULTIPLICATION.4	74.NUMBERS.1	32.MULTIPLICATION.2	43.MULTIPLICATION.4
62.STATISTICS.3	82.FRACTIONS.1	84.UNITS.2	22.MULTIPLICATION.1
64.STATISTICS.2	82.FRACTIONS.2	91.UNITS.2	21.MULTIPLICATION.2

Resultatet från undersökningen med lärarna, visade bland annat att rekommendationen för enheter för klass 9 till Elev C var mest problematisk. Andra rekommendationer som fick sämre resultat var området statistik för Elev A och 4:e klass multiplikation för Elev D.

Vidare kommer vi redogöra för de rekommendationer som gavs som slutresultat efter samtliga evalueringar. Nedanstående tabell representerar dessa rekommendationer.

Tabell 12: Rekommendationer efter parameteroptimering

Elev A	Elev B	Elev C	Elev D
51.BOSS	64.STATISTICS.3	64.STATISTICS.3	24.BOSS
32.ALGEBRA.1	95.ALGEBRA.3	64.STATISTICS.2	34.PROBLEM_SOLVING.1
54.TIME.1	64.ALGEBRA.3	81.UNITS.1	31.ARITHMETIC.1
53.TIME.1	32.ALGEBRA.1	81.ARITHMETIC.1	32.ALGEBRA.1
52.UNITS.1	95.GEOMETRY.1	64.STATISTICS.1	24.ALGEBRA.1
24.BOSS	52.FRACTIONS.1	63.STATISTICS.3	41.BOSS
44.UNITS.3	64.ALGEBRA.4	62.STATISTICS.4	33.SUBTRACTION.1
31.ALGEBRA.1	64.STATISTICS.1	81.FRACTIONS.1	31.ALGEBRA.1
33.FRACTIONS.2	64.STATISTICS.2	81.PRECENTAGE.2	53.TIME.1
34.PROBLEM_SOLVING.1	31.ARITHMETIC.1	82.FRACTIONS.1	31.BOSS

Utifrån de sex uppgifter som lärarna tyckte sämre om utifrån undersökningen har fyra av dessa rekommendationer utgått efter parameteroptimeringen. De uppgifter som utgått kopplade till undersökningen gäller alltså följande:

- Enheter för klass 9, rekommenderad för Elev C (91.UNITS.2)
- Statistik för klass 6, rekommenderad för Elev A (62.STATISTICS.3, 64.STATISTICS.2)
- Bråktal för klass 8, rekommenderad för Elev B (82.FRACTIONS.1, 82.FRACTIONS.2)
- Multiplikation för klass 4, rekommenderad för Elev D (43.MULTIPLICATION.4)

Däremot kan fortfarande vissa likheter identifieras. Lärarna gav, exempelvis, sämre bedömning för 5-6:e klass bråktal för Elev B. Resultaten efter parameteroptimeringen resulterade i en uppgift för bråktal i 5:e klass, däremot är det bara en uppgift istället för tre som tidigare. Vidare kan man dock analysera resultaten baserat på de uppgifter lärarna gett bäst bedömning. Dessa var bland annat 5-6:e klass bråktal för elev C samt 2-3:e klass multiplikation för Elev D. Rekommendationerna efter parameteroptimeringen resulterade även i avsaknad av dessa uppgifter. Kvalité för rekommendationerna kan alltså inte fastställas baserad på resultaten ifrån undersökningen utan endast till delvis grad genom utvärderingsmåten.

8 Slutsatser och diskussion

För de resultat som presenterats kommer vår slutsats grundas i både analysen av resultaten samt kravuppfyllnad. Samtliga icke-funktionella krav som ställdes uppfylldes genom de designval som gjordes för artefakten. För de krav som presenterades i 6.4 är de funktionella kraven till stor relevans för att besvara studiens inledande frågeställning vilket var:

Kan en rekommendationsmotor anpassas för en verksamhet vars huvudsakliga fokus är E-lärande inom matematik, så att dess rekommendationer har nytta för användarna?

De funktionella krav som definierades för artefakten var följande krav:

- *Artefakten ska kunna ge rekommendationer för systemets olika användargrupper.*
- *Inhämtning av användarinformation ska ske från verksamhetens databas.*
- *Informationen hämtad från verksamhetens databas ska användas av artefakten för att skapa implicita dataset.*
- *Dataseten ska användas för att generera rekommendationer för verksamhetens användare.*

Artefakten uppfyller till stor del många av de krav som definierades. Några av dessa krav är, exempelvis, inhämtning av användarinformation, generera dataset samt att dataseten används för att ge rekommendationer. Dessa uppfylls eftersom artefakten har möjlighet att ge rekommendationer baserad på verksamhetens användarinformation. Däremot kan vi inte med större sannolikhet säga att artefakten kan ge rekommendationer för verksamhetens olika användargrupper. Baserat på den analys som gjordes för undersökningen och den effekt parameteroptimeringen hade på rekommendationerna, kan vi inte helt säkert säga att rekommendationerna har nytta för användarna. Utvärderingsmåttan ger visst stöd för att en rekommendationsmotor kan anpassas för en verksamhet inom E-lärande i matematik. Däremot ger inte resultaten stöd för att rekommendationerna skulle ge nytta för användarna. En viss nytta kan möjligtvis vara att användaren slipper att leta efter uppgifter. Men baserat på resultaten finns alltså ingen säkerhet att dessa hade varit till relevans för användaren.

Möjligtvis hade man kunnat besvara forskningsfrågan med större säkerhet genom ytterligare underlag från systemets användare, alternativt ytterligare återkoppling till lärarna. På grund av de skäl som nämnts tidigare i rapporten, exempelvis, tidsramen för studien så var inte detta möjliga alternativ.

Utifrån de resultat som presenterades är det däremot stor vikt att även diskutera studiens reliabilitet samt validitet. Samtliga mätningarna har genomförts upprepade gånger för att säkerställa reliabiliteten, vilka i sin tur resulterade i liknande värden. De skillnader som däremot noterats i mätvärdena beror möjligen på de olika delmängder av datasetet som användes för respektive mätningar. De mätvärden som vi har mätt är relevanta i sitt sammanhang även om de, som tidigare nämnts, hade kunnat kombineras till större grad med ytterligare underlag. Baserat på detta så anser vi att studien bör vara valid samt uppnått reliabilitet.

Angående den undersökning som genomfördes som underlag till experimenten så hade man kunnat öka generaliteten genom en större urvalsgrupp. Dock anser vi inte att detta påverkar studiens resultat då analysen utfördes med medvetenhet hos rapportskrivarna. Användning av feedbacken bör däremot öka tillförlitligheten i de resultat som presenterades. Eftersom respondenterna är både användare i systemet och dessutom behöriga lärare inom matematik. Vidare upplever vi studien så pass generell att resultaten kan vara till nytta, inte enbart för

domäner där verksamheten sysslar med lärande inom matematik. Utan möjligtvis även inom andra områden. Eftersom rekommendationerna har baserats på implicita data kan denna skapas utifrån annan relevant data verksamheten har tillgång till.

Vi bedömer studien som fullt reproducerbar, då tillvägagångssätt samt implementering noggrant finns beskrivna i kapitlets metodtillämpning. Samtliga metoder för, exempelvis, evaluering är noggrant förklarade i 6.2 Implementera Artefakten. Undersökningen är den del som möjligtvis gör studien svår att reproducera. På grund av svårigheter att hitta andra kvalificerade lärare inom matematik som använder sig av systemet.

8.1 Rekommendationer

Då studien gjorts på uppdrag av en arbetsgivare redogör denna del i studien vissa rekommendationer, som kan vara till nytta för verksamheten angående tillämpning av artefakten.

I metodtillämpningskapitlet där implementeringen av artefakten beskrevs, nämndes de extrempunkter vi tagit hänsyn till. Vår uppfattning är att verksamheten har bättre kunskap och åsikter kring vad som kan vara extremt när det kommer till, exempelvis, antalet fel gjorda för en viss uppgift. Vi identifierade uppgifter där elever gjort ibland flera hundratals fel, ibland till och med upp mot tusen fel. Artefakten bör möjligtvis ge ännu bättre resultat ju fler extrema punkter som kan elimineras, men som sagt bör verksamheten bäst avgöra vad som klassas som extremt i detta fall.

Angående den parameteroptimering som genomfördes så innebär värdena för dessa, ett optimalt utfall för lösningen som presenterats i studien. Förändras däremot hanteringen av data, exempelvis, för extremvärden förändras modellen och en ny parameteroptimering kan då behövas.

Som resultat av vår efterforskning kring rekommendationsmotorer följer här vissa mer allmänna rekommendationer. Vissa steg i utvecklingen av rekommendationsmotorn hade underlättats betydligt med ytterligare information kring användarna av systemet. En hel del studier tyder på att ju mer information kring användarna, desto pålitligare rekommendationer. Däremot är det en avvägning kring hur mycket information man faktiskt vill lagra angående sina användare. Det finns fler tillgängliga alternativ för att hantera, exempelvis, kallstartsproblemet genom ytterligare information. Elevens årskurs är framförallt den mest saknade del av information i samband med utveckling av artefakten. Istället för att göra kvalificerade gissningar, finns det möjlighet att artefakten kan prestera bättre med tillämpning av information likt denna.

Ett ytterligare alternativ för att förbättra rekommendationerna, kan vara att ge användarna möjlighet till feedback för respektive slutförd uppgift. Man skulle även vidare kunna utnyttja den kompetens som finns tillgänglig hos lärarna som använder systemet, genom explicit feedback. Studiens teorikapitel redogjorde bland annat för en studie skriven av Claypool et al. (2001). De menar att en kombination av implicit och explicit feedback kan ge framgångsrika resultat. Grunden för detta är eftersom explicit feedback ger större säkerhet, men implicit feedback är tillgängligt då det förstnämnda saknas.

8.2 Framtida studier

Enligt det svar som getts på forskningsfrågan, bedömer vi alltså att frågan möjligtvis hade kunnat besvaras med större säkerhet genom ytterligare information. Vår rekommendation för framtida studier är därför att applicera studien under en längre period. En längre period hade kanske möjliggjort för applicering av artefakten i en användarbaserad miljö, eller att ytterligare involvera slutanvändaren på annat sätt. Vår upplevelse är även att många studier baseras på antingen explicit eller implicit feedback, beroende på vad som finns tillgänglig. För framtida studier rekommenderar vi att noggrannare undersöka resultat av att applicera båda delar i en faktisk domän.

Domäner är även någonting som poängterades i studiens inledande kapitel. Det är många domäner som är betydligt mer välstuderade än andra i området för rekommendationsmotorer. Domäner som sysslar med försäljning är, exempelvis, betydligt vanligare. Intresse kan därför ligga i att noggrannare studera vilka andra typer av domäner som kan tillämpa och dra nytta av dessa tekniker.

9 Referenser

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. (IEEE, Red.) *IEEE Transactions on Knowledge and Data Engineering*, 17(6), ss. 734-749.
- Aggarwal, C. C. (2016). *Recommender systems*. New York: Springer International Publishing.
- Apache Software Foundation. (u.å). *Collaborative Filtering - RDD-based API*. Hämtat från Apache Spark: <http://spark.apache.org/docs/latest/mllib-collaborative-filtering.html> den 10 02 2017
- Barjasteh, I., Forsati, R., Ross, D., Esfahanian, A.-H., & Radha, H. (2016). Cold-Start Recommendation with Provable Guarantees: A Decoupled Approach. *IEEE Transactions on Knowledge and Data Engineering*, 28(6), ss. 1462-1474. doi:10.1109/TKDE.2016.2522422
- Bell, R., & Koren, Y. (2011). Advances in Collaborative Filtering. i F. Ricci, L. Rokach, B. Shapira, F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Red.), *Recommender Systems Handbook* (ss. 145-186). Springer. doi:10.1007/978-0-387-85820-3
- Bobadilla, J., Serradilla, F., & Hernando, A. (2009). Collaborative filtering adapted to recommender systems of e-learning. *Knowledge-Based Systems*, 22(4), 261-265.
- Celma, O. (2010). *Music Recommendation and Discovery*. Barcelona: Springer. doi:10.1007/978-3-642-13287-2
- Chen, W., Niu, Z., Zhao, X., & Li, Y. (2012). A hybrid recommendation algorithm adapted in e-learning environments. *World Wide Web*, 17(2), 271-284. doi:10.1007/s11280-012-0187-z
- Claypool, M., Brown, D., Le, P., & Waseda, M. (2001). Inferring User Interest. *IEEE Internet Computing*, 5(6), ss. 32-39. doi:10.1109/4236.968829
- Formoso, V., Fernández, D., CACHEDA, F., & Carneiro, V. (2013). Using profile expansion techniques to alleviate the new user problem. *Information Processing and Management*, 49(3), 659-672.
- Gorakala, S. K. (2016). *Building Recommendation Engines*. Birmingham : Packt Publishing.
- Gorakala, S. K., & Usielli, M. (2015). *Building a recommendation system with R*. Birmingham: Packt Publishing Ltd.
- Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. *SIGIR '99 Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (ss. 230-237). Berkeley: ACM New York. doi:10.1145/312624.312682
- Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative Filtering for Implicit Feedback Datasets. *2008 Eighth IEEE International Conference on Data Mining* (ss. 263-272). IEEE. doi:10.1109/ICDM.2008.22
- Johannesson, P., & Perjons, E. (2014). *An Introduction to Design Science*. Springer International Publishing. doi:10.1007/978-3-319-10632-8
- Järvinen, P. (2007). Action Research is similar to Design Science. *Quality & Quantity*, 41(1), ss. 37-54. doi:10.1007/s11135-005-5427-1
- Kampffmeyer, M. (2015). *Parallelization of the Alternating-Least-Squares Algorithm With Weighted Regularization for Efficient GPU Execution in Recommender Systems*. Master's Thesis, UiT The arctic university of Norway, Computer Science, Tromsø.
- Leony, D., Muñoz-Merino, P. J., Ruipérez-Valiente, J. A., Arellano, D., & Delgado, C. (2014). Rule-based detection of emotions in the Khan Academy platform. *International Workshop on Massive Open Online Courses*. Antigua Guatemala.
- Lü, L., Medo, M., Yeung, C., Zhang, Y.-C., Zhang, Z.-K., & Zhou, T. (2012). Recommender systems. *Physics Reports*, 519, 1-49. doi:10.1016/j.physrep.2012.02.006

- Mandl, M., Felfernig, A., Teppan, E., & Schubert, M. (2010). Consumer decision making in knowledge-based. *Journal of Intelligent Information Systems*, 37(1), 1-22. doi:DOI 10.1007/s10844-010-0134-3
- Manouselis, N., Vuorikari, R., & Van Assche, F. (2010). Collaborative recommendation of e-learning resources: an experimental investigation. *Journal of Computer Assisted Learning*, 26(4), 227-242. doi:10.1111/j.1365-2729.2010.00362.x
- McCaffrey, J. (02 2015). *Test Run - L1 and L2 Regularization for Machine Learning*. Hämtat från MSDN Magazine Blog: <https://msdn.microsoft.com/en-us/magazine/dn904675.aspx> [blogg] den 15 09 2017
- Ocepek, U., Rugelj, J., & Bosnić, Z. (2015). Improving matrix factorization recommendations for examples in cold start. *Expert Systems with Applications*, 42(19), 6784-6794.
- Recker, M. M., Walker, A., & Lawless, K. (2003). What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education. *Instructional Science*, 31(4), 299-316. doi:10.1023/A:1024686010318
- Ruipérez-Valiente, J. A., Muñoz-Merino, P. J., Leony, D., & Delgado, C. (2015). ALAS-KA: A learning analytics extension for better understanding. *Computers in Human Behavior*(47), 139-148.
- Ruipérez-Valiente, J. A., Muñoz-Merino, P. J., & Delgado, C. (2013). An architecture for extending the learning analytics support in the Khan Academy framework. *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality* (ss. 277-284). Salamanca: ACM Publications. doi:10.1145/2536536.2536578
- Skolverket. (den 15 09 2016). *Det är gör Skolverket*. Hämtat från Skolverket: <https://www.skolverket.se/om-skolverket/det-har-gor-skolverket-1.61165> den 29 08 2017
- Skolverket. (U.å). *Matematik*. Hämtat från Skolverkets Hemsida: <https://www.skolverket.se/laroplaner-amnen-och-kurser/grundskoleutbildning/grundskola/matematik> den 28 08 2017
- Symeonidis, P., & Zioupos, A. (2016). *Matrix and Tensor Factorization Techniques for Recommender Systems*. Springer International Publishing. doi:10.1007/978-3-319-41357-0
- Tang, T. Y., & McCalla, G. (2005). Smart Recommendation for an evolving E-learning system: Architecture and experiment. *International Journal on Elearning*, 4(1), 105-129.
- Turner, G. A. (1982). *The Ethics of Social Research Surveys and Experiments, Part II. Survey Research and Protection of Privacy*. New York: Springer New York.
- Wright, D. R. (2007). *Motivation, Design, and Ubiquity: A Discussion of Research Ethics and Computer Science*. North Carolina State University, Computer Science.
- Yin, H., Chang, G., & Wang, X. (2009). A Cold-start Recommendation Algorithm Based on New User's Implicit Information and Multi-Attribute Rating Matrix. *2009 Ninth International Conference on Hybrid Intelligent Systems*, 2, ss. 353-358. doi:10.1109/HIS.2009.184

10 Appendix A

Tabell: dataset för Elev A

Elev A	Starttid:	Sluttid:	Antal fel:
34.FRACTIONS.1'	2015-11-09 16:29:47'	2015-11-09 16:29:13'	0'
34.FRACTIONS.1'	2015-11-09 16:30:19'	2015-11-09 16:29:51'	1'
34.FRACTIONS.1'	2015-11-09 16:30:58'	2015-11-09 16:30:22'	1'
34.FRACTIONS.1'	2015-11-09 16:31:28'	2015-11-09 16:31:01'	0'
34.FRACTIONS.1'	2015-11-09 16:31:59'	2015-11-09 16:31:31'	0'
34.FRACTIONS.1'	2015-11-09 16:32:36'	2015-11-09 16:32:05'	1'
34.FRACTIONS.1'	2015-11-09 16:33:07'	2015-11-09 16:32:37'	1'
34.FRACTIONS.1'	2015-11-09 16:33:42'	2015-11-09 16:33:12'	0'
34.FRACTIONS.1'	2015-11-09 16:34:31'	2015-11-09 16:33:46'	1'
34.FRACTIONS.1'	2015-11-09 16:35:23'	2015-11-09 16:34:36'	3'
34.FRACTIONS.2'	2015-11-09 16:36:34'	2015-11-09 16:35:31'	0'
34.FRACTIONS.2'	2015-11-09 16:37:47'	2015-11-09 16:36:39'	1'
34.FRACTIONS.2'	2015-11-09 16:40:45'	2015-11-09 16:37:51'	0'
34.FRACTIONS.2'	2015-11-09 16:41:55'	2015-11-09 16:40:49'	3'
34.FRACTIONS.2'	2015-11-09 16:43:00'	2015-11-09 16:42:03'	0'
51.DIVISION.1'	2015-11-09 16:44:04'	2015-11-09 16:43:17'	1'
51.DIVISION.1'	2015-11-09 16:44:45'	2015-11-09 16:44:08'	1'
51.DIVISION.1'	2015-11-09 16:45:16'	2015-11-09 16:44:48'	0'
51.DIVISION.1'	2015-11-09 16:45:45'	2015-11-09 16:45:20'	0'
51.DIVISION.1'	2015-11-09 16:46:13'	2015-11-09 16:45:47'	1'
33.MULTIPLICATION.3'	2015-11-17 13:17:38'	2015-11-17 13:16:55'	2'
33.MULTIPLICATION.3'	2015-11-17 13:18:23'	2015-11-17 13:17:40'	0'
33.MULTIPLICATION.3'	2015-11-17 13:18:53'	2015-11-17 13:18:25'	0'
33.MULTIPLICATION.3'	2015-11-17 13:19:29'	2015-11-17 13:18:59'	0'
33.MULTIPLICATION.3'	2015-11-17 13:20:15'	2015-11-17 13:19:33'	0'
42.FRACTIONS.2'	2015-11-17 13:22:54'	2015-11-17 13:20:20'	0'
42.FRACTIONS.2'	2015-11-17 13:24:05'	2015-11-17 13:23:11'	0'
42.FRACTIONS.2'	2015-11-17 13:25:56'	2015-11-17 13:24:08'	0'
43.FRACTIONS.3'	2015-11-17 13:27:06'	2015-11-17 13:26:00'	0'
43.FRACTIONS.3'	2015-11-17 13:27:52'	2015-11-17 13:27:09'	0'
43.FRACTIONS.3'	2015-11-17 13:28:46'	2015-11-17 13:27:54'	0'
43.MULTIPLICATION.4'	2015-12-10 07:53:19'	2015-12-10 07:52:37'	0'
43.MULTIPLICATION.4'	2015-12-10 18:49:30'	2015-12-10 18:48:52'	1'
43.MULTIPLICATION.4'	2015-12-10 18:50:10'	2015-12-10 18:49:40'	0'
43.MULTIPLICATION.4'	2015-12-10 18:50:53'	2015-12-10 18:50:15'	2'
43.MULTIPLICATION.4'	2015-12-10 18:51:32'	2015-12-10 18:50:55'	0'
52.FRACTIONS.3'	2015-12-10 18:52:20'	2015-12-10 18:51:38'	1'
52.FRACTIONS.3'	2015-12-10 18:52:57'	2015-12-10 18:52:25'	0'
52.FRACTIONS.3'	2015-12-10 18:53:40'	2015-12-10 18:53:01'	4'
52.FRACTIONS.3'	2015-12-10 18:54:20'	2015-12-10 18:53:42'	3'
52.FRACTIONS.3'	2015-12-10 18:54:53'	2015-12-10 18:54:22'	0'
21.ARITHMETIC.1'	2015-12-10 18:55:51'	2015-12-10 18:55:01'	0'
21.ARITHMETIC.1'	2015-12-10 18:56:44'	2015-12-10 18:55:53'	0'
21.ARITHMETIC.1'	2015-12-10 18:57:50'	2015-12-10 18:56:46'	1'
21.ARITHMETIC.1'	2015-12-10 18:58:57'	2015-12-10 18:57:50'	4'
21.ARITHMETIC.1'	2015-12-10 19:00:15'	2015-12-10 18:59:01'	0'

34.UNITS.1',	2016-01-25	20:27:31',	2016-01-25	20:26:48',	1',
34.UNITS.1',	2016-01-25	20:28:15',	2016-01-25	20:27:41',	0',
34.UNITS.1',	2016-01-25	20:28:53',	2016-01-25	20:28:17',	1',
34.UNITS.1',	2016-01-25	20:29:29',	2016-01-25	20:28:58',	0',
34.UNITS.1',	2016-01-25	20:30:09',	2016-01-25	20:29:36',	0',
44.UNITS.2',	2016-01-25	20:32:07',	2016-01-25	20:30:16',	10',
44.UNITS.2',	2016-01-25	20:35:00',	2016-01-25	20:32:12',	18',
44.UNITS.2',	2016-01-25	20:36:48',	2016-01-25	20:35:05',	28',
44.UNITS.2',	2016-01-25	20:37:46',	2016-01-25	20:36:56',	4',
44.UNITS.2',	2016-01-25	20:38:46',	2016-01-25	20:37:49',	2',
54.FRACTIONS.1',	2016-01-25	20:39:58',	2016-01-25	20:38:48',	7',
54.FRACTIONS.1',	2016-01-25	20:40:33',	2016-01-25	20:40:01',	1',
54.FRACTIONS.1',	2016-01-25	20:41:22',	2016-01-25	20:40:36',	4',
54.FRACTIONS.1',	2016-01-25	20:43:19',	2016-01-25	20:41:31',	13',
54.FRACTIONS.1',	2016-01-25	20:44:47',	2016-01-25	20:43:25',	10',
41.UNITS.1',	2016-02-01	16:26:40',	2016-02-01	16:24:05',	1',
41.UNITS.1',	2016-02-01	16:54:27',	2016-02-01	16:53:07',	5',
41.UNITS.1',	2016-02-01	16:55:36',	2016-02-01	16:54:45',	2',
41.UNITS.1',	2016-02-01	16:57:12',	2016-02-01	16:55:56',	4',
41.UNITS.1',	2016-02-01	16:58:25',	2016-02-01	16:57:30',	1',
43.MULTIPLICATION.4',	2016-02-01	16:59:16',	2016-02-01	16:58:31',	2',
43.MULTIPLICATION.4',	2016-02-01	16:59:56',	2016-02-01	16:59:18',	0',
43.MULTIPLICATION.4',	2016-02-01	17:00:48',	2016-02-01	16:59:58',	2',
43.MULTIPLICATION.4',	2016-02-01	17:01:36',	2016-02-01	17:00:56',	1',
43.MULTIPLICATION.4',	2016-02-01	17:02:28',	2016-02-01	17:01:38',	1',
51.UNITS.3',	2016-02-01	17:08:52',	2016-02-01	17:07:34',	13',
51.UNITS.3',	2016-02-01	17:09:42',	2016-02-01	17:08:54',	1',
51.UNITS.3',	2016-02-01	17:10:43',	2016-02-01	17:09:45',	7',
51.UNITS.3',	2016-02-01	17:11:36',	2016-02-01	17:10:45',	5',
51.UNITS.3',	2016-02-01	17:12:24',	2016-02-01	17:11:41',	3',
51.UNITS.3',	2016-02-05	06:18:41',	2016-02-05	06:17:26',	2',
51.UNITS.3',	2016-02-05	06:19:50',	2016-02-05	06:18:49',	1',
51.UNITS.2',	2016-02-05	06:21:16',	2016-02-05	06:20:10',	1',
51.UNITS.2',	2016-02-05	06:22:34',	2016-02-05	06:21:30',	2',
51.UNITS.1',	2016-02-05	06:23:47',	2016-02-05	06:22:59',	2',
51.UNITS.1',	2016-02-05	06:24:46',	2016-02-05	06:23:49',	2',
51.UNITS.4',	2016-02-05	06:25:47',	2016-02-05	06:24:50',	2',
51.UNITS.2',	2016-02-05	07:25:38',	2016-02-05	07:24:39',	2',
51.UNITS.2',	2016-02-05	07:26:39',	2016-02-05	07:25:40',	5',
51.UNITS.2',	2016-02-05	07:27:33',	2016-02-05	07:26:44',	3',
42.MULTIPLICATION.3',	2016-02-10	07:20:41',	2016-02-10	07:19:37',	3',
42.MULTIPLICATION.3',	2016-02-10	07:21:30',	2016-02-10	07:20:43',	1',
42.MULTIPLICATION.3',	2016-02-10	07:22:07',	2016-02-10	07:21:33',	0',
42.MULTIPLICATION.3',	2016-02-10	07:22:53',	2016-02-10	07:22:10',	3',
42.MULTIPLICATION.3',	2016-02-10	07:23:44',	2016-02-10	07:22:55',	0',
44.UNITS.1',	2016-02-10	07:24:51',	2016-02-10	07:23:48',	6',
44.UNITS.1',	2016-02-10	07:25:43',	2016-02-10	07:24:54',	4',
44.UNITS.1',	2016-02-10	07:26:37',	2016-02-10	07:25:46',	4',
44.UNITS.1',	2016-02-10	07:33:04',	2016-02-10	07:31:50',	13',
44.UNITS.1',	2016-02-10	07:34:26',	2016-02-10	07:33:06',	1',
52.DIVISION.1',	2016-02-10	07:35:46',	2016-02-10	07:34:35',	0',
52.DIVISION.1',	2016-02-10	07:36:25',	2016-02-10	07:35:49',	0',

52.DIVISION.1',	2016-02-10	07:36:58',	2016-02-10	07:36:26',	0',
52.DIVISION.1',	2016-02-10	07:37:33',	2016-02-10	07:37:00',	0',
52.DIVISION.1',	2016-02-10	07:38:12',	2016-02-10	07:37:35',	0',
51.UNITS.4',	2016-02-19	15:29:07',	2016-02-19	15:28:15',	1',
51.UNITS.4',	2016-02-19	15:33:19',	2016-02-19	15:32:12',	5',
51.UNITS.4',	2016-02-19	15:38:51',	2016-02-19	15:38:12',	0',
51.UNITS.4',	2016-02-19	15:39:40',	2016-02-19	15:38:56',	1',
51.UNITS.2',	2016-02-19	15:41:05',	2016-02-19	15:39:57',	10',
51.UNITS.2',	2016-02-19	15:42:02',	2016-02-19	15:41:09',	1',
51.UNITS.2',	2016-02-19	15:43:04',	2016-02-19	15:42:06',	3',
51.UNITS.2',	2016-02-19	15:43:57',	2016-02-19	15:43:09',	0',
51.UNITS.2',	2016-02-19	15:46:28',	2016-02-19	15:45:10',	2',
51.UNITS.1',	2016-02-19	15:48:18',	2016-02-19	15:47:18',	9',
51.UNITS.1',	2016-02-19	15:49:23',	2016-02-19	15:48:21',	5',
51.UNITS.1',	2016-02-19	15:50:40',	2016-02-19	15:49:32',	6',
51.UNITS.2',	2016-02-19	15:52:13',	2016-02-19	15:51:05',	5',
51.UNITS.2',	2016-02-19	15:53:19',	2016-02-19	15:52:17',	3',
51.UNITS.2',	2016-02-19	15:54:28',	2016-02-19	15:53:20',	2',
51.UNITS.2',	2016-02-22	20:47:53',	2016-02-22	20:47:10',	0',
51.UNITS.2',	2016-02-22	20:48:29',	2016-02-22	20:47:54',	0',
51.UNITS.2',	2016-02-22	20:49:11',	2016-02-22	20:48:32',	1',
51.GEOMETRY.1',	2016-02-22	20:50:04',	2016-02-22	20:49:16',	4',
51.GEOMETRY.1',	2016-02-22	20:50:55',	2016-02-22	20:50:06',	1',
51.UNITS.2',	2016-03-07	07:26:05',	2016-03-07	07:25:13',	6',
51.UNITS.2',	2016-03-07	07:27:17',	2016-03-07	07:26:31',	0',
51.UNITS.2',	2016-03-11	07:17:10',	2016-03-11	07:16:22',	3',
51.UNITS.3',	2016-03-11	07:18:31',	2016-03-11	07:17:24',	4',
51.UNITS.3',	2016-03-11	07:19:32',	2016-03-11	07:18:34',	6',
51.UNITS.3',	2016-03-20	10:08:12',	2016-03-20	10:07:15',	2',
51.UNITS.3',	2016-03-20	10:10:39',	2016-03-20	10:08:54',	2',
51.TIME.1',	2016-03-20	10:27:29',	2016-03-20	10:26:21',	4',
51.UNITS.1',	2016-03-20	10:29:40',	2016-03-20	10:27:40',	1',
51.UNITS.1',	2016-03-20	10:30:50',	2016-03-20	10:29:51',	3',
51.UNITS.3',	2016-03-20	10:31:45',	2016-03-20	10:30:57',	3',
51.UNITS.2',	2016-03-20	10:34:01',	2016-03-20	10:31:51',	1',
51.UNITS.2',	2016-03-20	10:34:50',	2016-03-20	10:34:03',	1',
51.UNITS.2',	2016-03-20	10:38:28',	2016-03-20	10:37:30',	6',
51.UNITS.3',	2016-03-20	10:39:42',	2016-03-20	10:38:34',	3',
51.UNITS.2',	2016-03-20	10:43:35',	2016-03-20	10:42:04',	2',
51.UNITS.2',	2016-03-21	20:49:38',	2016-03-21	20:48:53',	2',
51.UNITS.2',	2016-03-21	20:50:28',	2016-03-21	20:49:39',	4',
51.UNITS.1',	2016-03-21	20:51:26',	2016-03-21	20:50:38',	0',
51.UNITS.1',	2016-03-21	20:52:12',	2016-03-21	20:51:28',	1',
51.UNITS.3',	2016-03-21	20:53:22',	2016-03-21	20:52:22',	5',
51.UNITS.3',	2016-03-21	20:54:11',	2016-03-21	20:53:23',	1',
51.UNITS.3',	2016-03-22	16:28:16',	2016-03-22	16:27:18',	2',
51.UNITS.3',	2016-03-22	16:31:01',	2016-03-22	16:30:11',	0',
51.UNITS.2',	2016-03-22	16:31:45',	2016-03-22	16:31:08',	0',
51.UNITS.2',	2016-03-22	16:32:40',	2016-03-22	16:31:47',	4',
51.UNITS.2',	2016-03-22	16:33:48',	2016-03-22	16:32:47',	2',
51.UNITS.1',	2016-03-22	16:34:49',	2016-03-22	16:34:03',	2',
51.UNITS.2',	2016-03-31	21:04:17',	2016-03-31	21:03:35',	1',

51.UNITS.3',	2016-03-31	21:05:27',	2016-03-31	21:04:23',	5',
51.UNITS.3',	2016-03-31	21:06:22',	2016-03-31	21:05:29',	0',
51.UNITS.1',	2016-03-31	21:07:31',	2016-03-31	21:06:28',	3',
51.UNITS.3',	2016-04-04	16:11:53',	2016-04-04	16:11:08',	2',
51.UNITS.1',	2016-04-04	16:26:57',	2016-04-04	16:25:56',	0',
52.DIVISION.1',	2016-04-04	16:27:51',	2016-04-04	16:27:13',	0',
52.ADDITION.1',	2016-04-04	16:29:27',	2016-04-04	16:28:03',	2',
52.BOSS',	2016-04-04	16:45:16',	2016-04-04	16:31:50',	22',
61.NUMBERS.3',	2016-04-04	16:47:03',	2016-04-04	16:46:25',	1',
61.GEOMETRY.1',	2016-04-04	16:52:49',	2016-04-04	16:51:07',	2',
61.GEOMETRY.2',	2016-04-04	16:53:39',	2016-04-04	16:52:52',	0',
62.GEOMETRY.2',	2016-04-04	17:13:59',	2016-04-04	17:12:55',	3',
63.NUMBERS.2',	2016-04-04	17:15:30',	2016-04-04	17:14:22',	4',
63.NUMBERS.2',	2016-04-04	17:17:10',	2016-04-04	17:16:28',	0',

11 Appendix B

Undersökning för läroplattformen Nomp

Tack på förhand för att du ställer upp på denna undersökningen! Undersökningen görs i samarbete med elever på högskolan i Borås. Dina svar är anonyma och kommer att vara underlag till en studie vars målsättning är att framställa ett verktyg som kan hjälpa Nomp att fortsätta leverera ett utmärkt hjälpmedel för undervisning inom matematik.

Varje fråga innehåller en hypotetisk elev som har gjort ett antal uppgifter inom olika matematiska områden, baserat på dessa uppgifter önskar vi att du värdesätter våra rekommendationer på uppgifter eleven bör kunna göra.

Elev A har under den senaste tiden bland annat gjort uppgifter inom följande områden:

- Enheter klass 4-5,
- Geometri klass 6,
- Division klass 5,
- Multiplikation klass 4,
- Bråktal klass 4-5.

Områden där eleven behöver mer övning (haft svårighet för) är inom enheter och bråktal. Hur anser du rekommendationerna nedan skulle passa denna elev?

Division för 5:e klass, svårare uppgift än de eleven tidigare gjort. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

Enheter för 5:e klass, samma nivå som eleven tidigare har haft svårigheter för. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

⋮

Statistik klass 6 *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

Elev B har under den senaste tiden bland annat gjort uppgifter inom följande områden:

- Bråktal för 8:e klass,
- Geometri för 7-8:e klass,
- Algebra för 8:e klass,
- Aritmetik för 6-8:e klass,
- Enheter för 3-8:e klass,
- Multiplikation för 8:e klass,
- Procent för 8:e klass,
- Division för 5-8:e klass,

Områden där eleven behöver mer övning (haft svårighet för) är inom enheter och aritmetik. Hur anser du rekommendationerna nedan skulle passa denna elev?

5 - 6:e klass bråktal, lättare uppgifter än de eleven tidigare har gjort. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

8:e klass bråktal, samma nivå som eleven gjort tidigare *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

6:e klass algebra, lättare uppgift än eleven tidigare har gjort *

	1	2	3	4	5	6	7	8	9	10	
inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

Elev C har under den senaste tiden bland annat gjort uppgifter inom följande områden:

- Division 8:e klass,
- Procent 8:e klass,
- Enheter 8:e klass,
- Multiplikation 8:e klass,
- Bråkta 8:e klass.

Områden där eleven behöver mer övning (haft svårighet för) är inom bråkta, enheter och division. Hur anser du rekommendationerna nedan skulle passa denna elev?

5-6:e klass bråkta, lättare uppgifter än de eleven gjort innan. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

8:e klass enheter, på samma nivå som eleven gjort innan *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

9:e klass enheter, svårare uppgift än eleven gjort innan *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

Elev D har under den senaste tiden bland annat gjort uppgifter inom följande områden:

- 3-4:e klass multiplikation,
- 4:e klass geometri.

Områden där eleven behöver mer övning (haft svårighet för) är delvis inom multiplikation. Hur anser du rekommendationerna nedan skulle passa denna elev?

3:e klass multiplikation, på samma nivå eleven tidigare har gjort. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

2:e klass multiplikation, lättare nivå än de uppgifter eleven tidigare har gjort. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

4:e klass multiplikation, högre nivå än de uppgifter eleven tidigare har gjort. *

	1	2	3	4	5	6	7	8	9	10	
Inte alls, dåligt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mycket bra

Nedan kan du om du önskar fylla i om du har några åsikter, kommentarer eller om det är någonting du vill tillägga angående undersökningen. Tack så mycket för din tid!

...

Kommentarer:

Long-answer text

Högskolan i Borås är en modern högskola mitt i city. Vi bedriver utbildningar inom ekonomi och informatik, biblioteks- och informationsvetenskap, mode och textil, beteendevetenskap och lärarutbildning, teknik samt vårdvetenskap.

På **institutionen Handels- och IT-högskolan (HIT)** har vi tagit fasta på studenternas framtida behov. Därför har vi skapat utbildningar där anställningsbarhet är ett nyckelord. Ämnesintegration, helhet och sammanhang är andra viktiga begrepp. På institutionen råder en närhet, såväl mellan studenter och lärare som mellan företag och utbildning.

Våra **ekonomiutbildningar** ger studenterna möjlighet att lära sig mer om olika företag och förvaltningar och hur styrning och organisering av dessa verksamheter sker. De får även lära sig om samhällsutveckling och om organisationers anpassning till omvärlden. De får möjlighet att förbättra sin förmåga att analysera, utveckla och styra verksamheter, oavsett om de vill ägna sig åt revision, administration eller marknadsföring. Bland våra **IT-utbildningar** finns alltid något för dem som vill designa framtidens IT-baserade kommunikationslösningar, som vill analysera behov av och krav på organisationers information för att designa deras innehållsstrukturer, bedriva integrerad IT- och affärsutveckling, utveckla sin förmåga att analysera och designa verksamheter eller inrikta sig mot programmering och utveckling för god IT-användning i företag och organisationer.

Forskningsverksamheten vid institutionen är såväl professions- som design- och utvecklingsinriktad. Den övergripande forskningsprofilen för institutionen är handels- och tjänsteutveckling i vilken kunskaper och kompetenser inom såväl informatik som företagsekonomi utgör viktiga grundstenar. Forskningen är välrenommerad och fokuserar på inriktningarna affärsdesign och Co-design. Forskningen är också professionsorienterad, vilket bland annat tar sig uttryck i att forskningen i många fall bedrivs på aktionsforskningsbaserade grunder med företag och offentliga organisationer på lokal, nationell och internationell arena. Forskningens design och professionsinriktning manifesteras också i InnovationLab, som är institutionens och Högskolans enhet för forskningsstödande systemutveckling.



HÖGSKOLAN
I BORÅS

BESÖKSADRESS: JÄRNVÄGSGATAN 5 · POSTADRESS: ALLÉGATAN 1, 501 90 BORÅS
TFN: 033-435 40 00 · E-POST: INST.HIT@HB.SE · WEBB: WWW.HB.SE/HIT