HØGSKOLEN I OSLO
OG AKERSHUS

Karoline Kornkveen Hoff

Annoteringsverktøyet TORCH

Brukskvalitetstesting av et annoteringsverktøy

Masteroppgave 2017
Master i bibliotek- og informasjonsvitenskap
Høgskolen i Oslo og Akershus, Institutt for arkiv- bibliotek- og informasjonsfag

Sammendrag

I denne masteroppgaven presenteres resultatene fra en brukskvalitetsevaluering av et annoteringsverktøy utviklet i forbindelse med TORCH-prosjektet ved Høgskolen i Oslo og Akershus. Det ble gjennomført to brukertester med formål å legge til rette for crowdsourcing av annotering av navneentiteter og relasjoner. Interessante problemer og bruksmønstre fra eksisterende litteratur manifesterte seg i undersøkelsene. Til tross for svakheter funnet i grensesnittet til versjonene av verktøyet som ble brukt i forsøkene, viste annotørene en rimelig suksessrate. I tillegg til brukskvalitetsproblemene som ble avdekket, ble det konkludert med to ting: brukerne med rimelige datakunnskaper presterte godt, og det ble ikke funnet noen klar sammenheng mellom hvor vanskelig brukerne opplevde en oppgave og deres interesse i å delta i senere prosjekter med lignende aktiviteter.

Abstract

This master thesis presents the result of two usability tests of an annotation tool developed in connection with The TORCH project at the Oslo and Akershus University College of Applied Science. The experiments employed non-experts with the intention of facilitating for crowdsourcing of named entity annotation. Interesting problems and usability patterns from existing literature manifest in the experiments. Despite some weaknesses in the interface of the tool versions used for the experiments, the annotators showed a reasonable rate of success. In addition to the usability issues discovered, it was found that users with reasonable computer skills performed well, and there was no clear connection between how hard the users found a task and their interest in participating in further projects with similar activities.

Høgskolen i Oslo og Akershus, Institutt for arkiv- bibliotek- og informasjonsfag
Oslo 2017

Forord

Denne masteroppgaven er gjennomført ved institutt for arkiv-, bibliotek- og informasjonsfag innen studieretningen Kunnskapsorganisasjon og -gjenfinning ved Høgskolen i Oslo og Akershus. Oppgaven er uført i løst samarbeid med forskningsgruppen METAINFO ved fakultet for samfunnsvitenskap. De har stilt med ressurser i form av programvare og dokumentposter.

Jeg ønsker først og fremst å takke veilederen min, Michael Preminger, for god rettledning og verdifull hjelp i arbeidet med denne oppgaven. Jeg har satt stor pris på gode råd og uendelig tålmodighet gjennom hele prosessen.

Takk til mamma og pappa, som trodde på meg når jeg ikke gjorde det.

En takk går også til Anton, Modou, Karen, og Koseklubben, som stilte opp som korrekturlesere, diskusjonspartnere, distraksjon og inspirasjon når selvdisiplin og motivasjon var på sitt laveste.

Jeg vil også takke kollegaene mine på Høyskolen Kristiania, som uten å protestere lot meg flytte inn på kontoret de siste ukene før innlevering.

Til slutt vil jeg rette en stor takk til alle som tok seg tid til å stille opp på brukertestene.

Oslo, 25. juni 2017
Karoline Kornkveen Hoff

Innhold

1. Innledning

Det finnes enorme mengder informasjon på internett i dag, og i et forsøk på å gjøre denne informasjonen søkbar har det i senere tid blitt lagt mye arbeid i semantiske webapplikasjoner og forskning på automatisk konvertering av data lagret i eldre eller utdaterte formater til semantisk merkede data. For å gjøre fremskritt her kreves det først såkalte grunnsannheter som kan brukes til å «lære opp» slike konverteringsprogrammer. For å etablere disse grunnsannhetene behøver vi store mengder data annotert av mennesker, hvor de har identifisert entiteter og roller i disse dataene. Slik annotering er intellektuelt arbeid og utføres ved hjelp av annoteringsverktøy.

Formålet med denne masteroppgaven har vært å evaluere det grafiske brukergrensesnittet til et manuelt annoteringsverktøy utviklet i forbindelse med TORCH[1]-prosjektet ved Høgskolen i Oslo og Akershus. En del av dette prosjektet er et konverteringsprosjekt, hvor vi forsøker å konvertere programbeskrivelser fra NRK-arkivet lagret i fritekst til et semantisk format. Formålet med verktøyet, heretter kalt TORCH, er såkalt *named-entity annotation* av disse programbeskrivelsene. Her identifiserer brukeren navneentiteter, for eksempel person og verk, og relasjonen dem imellom i tråd med en gitt ontologi. Annotasjonsverktøyet beskrevet i denne oppgaven er utviklet for å skape grunnsannheter for TORCH-prosjektet. For å legge til rette for innsamling av nok data, jobber prosjektet ut fra en antakelse om at crowdsourcing med annotering av ikke-eksperter er veien å gå. Verktøyet er utviklet for bruk i en kulturarv-kontekst, hvor begrepet ikke-eksperter refererer til et stort publikum. Vi tror også at andre prosjekter som annoterer tekstmateriale kan få et større antall annotører og mer annotert materiale av høy kvalitet hvis annoteringsprosessen blir gjort mer tilgjengelig gjennom verktøy med høy brukskvalitet.

Oppgaven startet med undersøkelser jeg gjennomførte høsten 2015 i forbindelse med the 9th Metadata and Semantic Research Conference, MTSR 2015. Jeg foretok da en begrenset brukertest av verktøyet, og skrev sammen med Michael Preminger et konferansebidrag som ble akseptert og publisert i *Metadata and Semantics Research* (Hoff & Preminger, 2015). Samme året presenterte jeg artikkelen på den tilhørende konferansen i Manchester. Grensesnittet ble omarbeidet etter resultatene fra 2015, og jeg gjennomførte en ny brukertest av den oppdaterte versjonen av verktøyet.

[1] Transforming Organization and Retrieval in Cultural Heritage

Det er flere metoder og teknikker man kan bruke for å evaluere brukergrensesnitt. Jeg har valgt brukskvalitetstesting: en kvalitativ, brukerbasert tilnærming som kombinerer «think aloud»-observasjon og intervju.

I resten av del 1 forklarer hva jeg ønsker å oppnå med dette prosjektet, og jeg vil presentere TORCH-prosjektet og selve annoteringsverktøyet. I del 2 vil jeg gjøre rede for tidligere forskning og lignende prosjekter på feltet. Jeg vil også forklare hva begrepet brukskvalitet innebærer, og forklare de forskjellige måtene man kan evaluere det på. Del 3 tar for seg metoden jeg har valgt, og beskriver hvordan undersøkelsen min er lagt opp, mens jeg i del 4 presenterer gjennomføringen av de to testene og resultatet. I del 5 runder jeg av med konklusjon og forslag til videre arbeid.

1.1 Problemstilling

Hensikten med undersøkelsen min har vært å få et innblikk i brukernes opplevelse av verktøyet og identifisere brukskvalitetsproblemer. Undersøkelsene har vært ledd i en prosess hvor det endelige målet er at verktøyets grensesnitt skal ha tilstrekkelig høy brukskvalitet og være intuitivt nok til at hele eller deler av annoteringsprosessen kan crowdsources og fremdeles resultere i annoteringer av god kvalitet.

Undersøkelsen min tok derfor sikte på å besvare disse spørsmålene:

1. Hvordan oppleves verktøyet av brukerne?
    a. Hvor fornøyde er brukerne av verktøyet?
    b. Hva oppleves som problematisk for brukerne av verktøyet?
2. Hvilke brukskvalitetsproblemer finnes i annoteringsverktøyets grensesnitt?
    a. Hvordan kan verktøyet endres for å minimere problemene jeg fant?
3. Hvilke aktiviteter i annoteringsprosessen indikerer brukertestingen er det vil være mest aktuelt å crowdsource?

1.2 Sentrale begreper

Mye av litteraturen som finnes om annotering og brukskvalitet er skrevet på engelsk, noe som gjør at mange sentrale begreper på området er engelske. Enkelte begrep er forholdsvis innarbeidet i norsk språk, men i enkelte tilfeller har jeg i samarbeid med veileder oversatt på egenhånd, og satt det engelske begrepet i parentes første gangen det omtales.

1.2.1 Hva er annotering?

Formålet med verktøyet er som sagt manuell annotering av navneentiteter og relasjonene dem imellom. Det er derfor hensiktsmessig å definere begrepet *annotering* før noe annet.

Store norske leksikon definerer annotasjon som «(…) notat, bemerkning, anmerkning. Verb: annotere» (Annotasjon, 2009). I «Introduction to Manual Annotation», et veiledningsskriv brukt i forbindelse med GATE, et bredere annoteringsverktøy som blant annet inneholder funksjoner som ligner på de vi finner TORCH, defineres aktiviteten annotering som «a methodology for adding information to a document at some level», og annoteringene selv som «meta-data added to a specific span of text» (Petrillo & Baycroft, 2010, s. 1, 6).

Hva «a specific span of text» innebærer er avhengig av annoteringsprosjektets hensikt. Det kan være alt fra bokmerker og personlige notater til standardiserte metadata koblet til elementer som person- og stedsnavn. En forskjell mellom annotering og andre måter å registrere metadata på er at annoteringer er ankret til helt spesifikke deler av teksten. (2010, s. 1) En annen er at man ved annotering kan si at man ikke legger til noen *ny* informasjon i teksten, man beriker kun teksten ved å gjøre implisitt informasjon eksplisitt. I *Corpus Linguistics* beskriver McEnery og Hardie det på denne måten:

> When a corpus is lingustic annotation, it is important to note what can and cannot be said about those annotations. Firstly, and perhaps most importantly, we cannot say that the corpus contains *new* information. It clearly does not. What a linguistic analysis of this sort does is to make explicit information that is there implicitly in the data. In other words, identifying a word as a noun does not mean that we transform it into a noun in so doing. In corpus annotation we engage in a process of labeling, not creation or transformation. To that extent, we can say that the corpus is *enriched*, from the point of view of a program or user, but we cannot say that the corpus has had new information added to it.
>
> (McEnery & Hardie, 2011, s. 31)

1.2.2 Hva annoteres?

Begrepet annotering sier altså ikke nødvendigvis noe om hvilke elementer som skal annoteres, eller hvilken informasjon annoteringen skal bestå av. Hvilke elementer som annoteres er avhengig av prosjektets formål, og i vårt prosjekt er det snakk om annotering av såkalte *named entities*. Jeg har valgt å oversette det til *navneentiteter* eller *entiteter* i denne oppgaven.

I litteraturen sees begrepet ofte i sammenheng med *Named Entity Recognition*, eller navnegjenkjenning. Delen av TORCH-prosjektet vi jobber med er et slikt navnegjenkjenningsprosjekt.

I «Named Entity Recognition: Fallacies, Challenges and Opportunities» defineres navnegjenkjenning som «[…] a task in Information Extraction consisting in identifying and classifying just some types of information elements, called Named Entities» (Marrero, Urbano, Sánchez-Cuadrado, Morato, & Gómez-Berbís, 2013, s. 482).

I *ARNER, what kind of name is that?: an automatic rule-based entity recognizer for Norwegian,* beskriver Andrea Björk Jónsdóttir navnegjenkjenning som aktiviteten å identifisere og kategorisere egennavn i semantiske kategorier. (2003) Det handler altså om å identifisere og annotere det en gitt prosjekt har definert som entiteter.

Det skilles gjerne mellom *lingvistisk* og *semantisk annotering*. I semantisk annotering dreier det seg om å koble en del av en tekst til en semantisk database eller ontologi hvor det ligger ytterligere informasjon lagret. Petrillo og Baycroft beskriver semantisk annotering som en prosess hvor man omformer en måltekst til en *entitet*, et spesifikt dataelement i et univers av dataelementer (2010, s. 6). De definerer en entitet som noe som har en distinkt, separat eksistens uavhengig av teksten, og det er disse som annoteres. Entiteter kan være *implisitte* ved at dets eksistens kan utledes av teksten, eller *eksplisitte* ved at det nevnes direkte. Ved å annotere disse entitetene vil en semantisk database kodifisere tilleggsinformasjon i et format som kan behandles av datamaskiner. TORCH-prosjektet arbeider med denne typen annotering.

Videre tar man i lingvistisk annotering i større grad hensyn til de språklige egenskapene til et ord eller en setning enn ved semantisk annotering, for eksempel ved å annotere noe som «del av tale» (part of speech), ta hensyn til om noe er i en- eller flertallsform, eller om det refererer til et annet ord i den omkringliggende teksten. (2010, s. 6–7)

I forbindelse med annotering refereres det ofte til begrepet *korpus*. Dette er et definert sett med dokumenter som annoteres i et gitt prosjekt. Korpus med høy grad av konsistens regnes som et kvalitativt godt korpus, og sies å ha *gullstandard* (gold standard). Korpus med gullstandard kan brukes som fasit for å «trene» et nytt, eller evaluere et allerede eksisterende, automatisk annoteringsverktøy. (2010, s. 3–4)

Konsistens betyr at noe blir gjort likt hver gang det gjøres, og med tanke på annotering kan vi snakke om *intraannotør konsistens* og *interannotør konsistens*. Interannotør konsistens sier noe om hvor likt de forskjellige annotørene annoterer en gitt tekst, mens intraannotør konsistens sier noe om hvor likt en gitt annotør annoterer forskjellige tekster.

I sin doktorgradsavhandling *Named Entity Recognition – challenges in document annotation, gazetteer construction and disambiguation* hevder Ziqu Zhang at fullstendig interannotør konsistens ikke er mulig, blant annet fordi hver annotør har forskjellig kunnskap og erfaringer, og derfor vil forstå og tolke teksten som annoteres ulikt. Han oppgir usikkerhet rundt tre ting som hovedgrunnene til at interannotør inkonsistens oppstår: valg av riktig

kategori, forståelsen av hva som er navneentiteter, og hvordan man skal forholde seg til homonymi. Intraannotør inkonsistens oppstår som regel når annotørene blir slitne eller går lei underveis i arbeidet og dermed gjør feil. (2013. s. 55-56) Gode retningslinjer er nøkkelen til høy interannotør konsistens. (Tallerås, Massey, Husevåg, Preminger, & Pharo, 2014)

Petrillo og Baycroft oppsummerer sin artikkel ved å legge vekt på at annotering ikke handler om å markere alle ord i en tekst, men konsistent å identifisere det som er beskrevet i retningslinjene til prosjektet man arbeider i, og annotere dem etter retningslinjenes instruksjoner. (2010, s. 5–6)

Hva slags korpus det arbeides med, og hvilke elementer det er aktuelt å annotere i dette prosjektet blir beskrevet i 1.3 TORCH-prosjektet.

1.2.3 Hvem annoterer?

Annotering kan gjøres automatisk av en datamaskin, manuelt av en person, eller semi-automatisk i en kombinasjon av disse. *Automatisk annotering* er mindre presis, men kan behandle dokumenter i større kvanta enn hva som er rimelig å forvente av enn menneske. *Manuell annotering* er rom regel mer presis, men veldig ressurskrevende. Ved *semi-automatisk annotering* vil en datamaskin foreslå annoteringer, mens en person manuelt kvalitetssjekker og utfyller resultatet. (Petrillo & Baycroft, 2010, s. 3)

Det er viktig å vurdere de ulike brukergruppene av annoteringsverktøyer, da krav og behov vil variere med tanke på brukerens individuelle domenekunnskap og teknologiske kunnskaper. Slike forskjellige brukergrupper av digitale verktøy har blitt identifisert og beskrevet i mye litteratur. Tidlig ble de grunnleggende brukergruppene identifisert som kodere (the coder), de kodende forbrukerne (the coding consumer) og de kodende utviklerne (the coding developer) (Carletta & Isard, 1999). Senere tilpasninger i litteraturen for annoteringsverktøy har identifisert dem som annotører (annotators), korpusforbrukere (corpus consumers) og korpusutviklere (corpus developers). (Reidsma, Hofs, & Jovanović, 2005)

En person som annoterer vil heretter bli referert til som *annotør*. Dette er brukergruppen som bruker verktøyet for å annotere et korpus, og som ikke nødvendigvis vil inneha kunnskap om verken utvikling av verktøyet selv eller informasjonsstrukturene som ligger i bakgrunnen. I faglig tunge annoteringsprosjekter kreves det gjerne at annotørene er *domeneeksperter* (domain experts) i tillegg til at de behersker de tekniske aspektene ved å bruke annoteringsverktøyet. Domene i annotasjonssammenheng betyr prosjektets emne eller fagfelt. I prosjekter som ikke er knyttet til et spesifikt fag vil det ikke nødvendigvis være behov for

domeneeksperter, og i enkelte tilfeller kan det være hensiktsmessig å bruke en kombinasjon av domeneeksperter og lekfolk. (Carletta & Isard, 1999, s. 13; Petrillo & Baycroft, 2010, s. 7; Pustejovsky & Stubbs, 2013, s. 26; Reidsma et al., 2005, s. 1)

> [These users] are typically the cheapest labour source available. They do not wish to know anything about how the coding interface works or even how different sets of tags relate to each other. Their needs are fairly simple: an intuitive coding interface so that they can concentrate on the code distinctions, documentation of how to use the interface (…) and the coding instructions nearby.
>
> (Carletta & Isard, 1999, s. 13)

Korpusforbrukere er brukerne som er interessert i å bruke det ferdig annoterte korpuset. De er domeneeksperter, og har i tillegg en viss forståelse av annoteringsskjemaet eller ontologien som ligger i bunn i verktøyet. Disse brukerne er ute etter å kunne dra nytte av dataene effektivt, for eksempel ved søking og spørringer. (Carletta & Isard, 1999, s. 14; Reidsma et al., 2005, s. 1)

> Users who want to use annotated data for all kinds of reasons, e.g. theory testing, evaluation and training of models, finding relations between phenomena. They have needs for querying and browsing annotated data.
>
> (Reidsma et al., 2005, s. 1)

Korpusutviklere er brukerne som bygger videre på eksisterende korpus og annoteringsskjema for å tilpasse dem til deres prosjekter og behov. Disse er både domene- og tekniske eksperter, og innehar kunnskap om både fagfeltet, annoteringsskjemaer, og markerings- og programmeringsspråk brukt i forbindelse med verktøyet. (Carletta & Isard, 1999, s. 15; Reidsma et al., 2005, s. 1)

> [These users may wish] to design their own coding schemes, either to improve on the reliability or suitability of an existing scheme or in order to test a particular research question.
>
> (Carletta & Isard, 1999, s. 15)

I denne oppgaven er manuell annotering mest relevant, og ordet «annotering» vil heretter henvise til manuell annotering dersom ikke annet er nevnt. I tillegg vil «bruker» være synonymt med «annotør» som beskrevet over.

Jeg skriver mer om dette prosjektets domene i 1.3 TORCH-prosjektet, og mer om domeneeksperter kontra lekfolk i 2.2 Crowdsourcing og annotering.

1.2.4 Typer annoteringsprosjekter

Det utføres en rekke typer annoteringsprosjekter. Petrillo og Baycroft beskriver det de kaller de mest vanlige typene: gullstandardannotering, kvalitetssikring (quality assurance), prosesseringsprosjekter (processing projects) og sosial annotering. (2010, s. 4–5)

Gullstandardannotering er prosessen hvor man kommer frem til en gullstandard som beskrevet i 1.2.2. Dette foregår gjerne i begynnelsen av et prosjekt for å se om to eller flere personer har tolket retningslinjene for annoteringsprosjektet likt og dermed får et lignende resultat. Når resultatene til de uavhengige annotørene når en viss grad av overenstemmelse kan resultatene brukes som grunnlag for automatisk annotering.

I kvalitetssikringsprosjekter søker man etter feil i allerede annoterte dokumenter. Avhengig av prosjektets formål kan man rette feilene man finner, eller bare samle og analysere dem. I slike prosjekter er man som regel ute etter fullstendig nøyaktighet, og retningslinjene vil derfor være strengere enn ved gullstandardannotering.

Prosesseringsprosjekter kjennetegnes ved at annoteringene er midlertidige, som et ledd i en prosess. Et eksempel er prosjekter hvor man har uthevet feil i et dokument og annotert korrigeringer som kan hjelpe datamaskinen til å rette dokumentet automatisk, og deretter fjerner annoteringene fra originalen. I slike tilfeller er retningslinjene være spesifikke, uten mulighet for variasjon.

Sosial annotering er ofte et forsøk på å utarbeide en ny nomenklatur, altså regelverk for bruk av navn eller fagord, eller imøtekomme en folksonomi med et mer formelt system. Her er det generelt sett ikke noe særlig forsøk på å styre eller begrense hvordan annotørene jobber med retningslinjer, men heller høste resultatene av annoteringene og analysere dem. (Petrillo & Baycroft, 2010, s. 4–5)

1.2.5 Crowdsourcing

Mine undersøkelser er som sagt ledd i en prosess for å heve verktøyets brukskvalitet til et nivå der hele eller deler av annoteringsprosessen kan *crowdsources* og fremdeles resultere i annoteringer av god kvalitet. Crowdsourcing er en problemløsning- og produksjonsmodell for samarbeid, og uttrykket stammer fra *outsourcing*. Outsourcing er en metode hvor foretak setter deler av sin produksjon ut til underleverandører, for eksempel ved hjelp av billig arbeidskraft andre steder, eller for å la noen andre ta seg av arbeidsoppgaver som er blitt nedprioritert. (Dvergsdal, 2015; Vikøren, 2012)

Oxford Dictionary definerer crowdsourcing slik:

> The practice of obtaining information or input into a task or project by enlisting the services of a large number of people, either paid or unpaid, typically via the Internet.
>
> (Crowdsourcing, udatert)

Crowdsourcing er altså en måte å benytte en større gruppe mennesker for å løse en felles oppgave. Deltakerne donerer sin fritid, kunnskap og erfaring til noe som kan kalles for dugnad via internett. Disse menneskene krever ofte ikke belønning i form av penger, men jobber heller ut fra en motivasjon om samhold med gruppen de jobber med, anerkjennelse for jobben de gjør, eller selvtilfredshet. (Dvergsdal, 2015)

Siden «crowdsourcing» er et relativt godt innarbeidet uttrykk på norsk, og det ikke finnes en god oversettelse som vil passe bedre i denne konteksten, vil jeg bruke begrepet på engelsk i min oppgave.

Jeg skriver mer om crowdsourcing i annoteringskontekst i delkapittel 2.2.

1.2.6 Usability, eller brukskvalitet

Brukskvalitet, eller det mer dekkende *usability* på engelsk, er kort fortalt et mål for hvor raskt og enkelt en bruker av et system kan utføre de oppgavene som ønskes med et gitt grensesnittdesign. Begrepet kan brukes om alt som blir tatt i bruk av mennesker, for eksempel bygninger, maskiner, eller som i vårt tilfelle en webapplikasjon.

Ifølge Norsk Språkråd bør vi på bruke begrepet brukskvalitet på norsk, men i praksis brukes begrepene brukskvalitet, brukbarhet og brukervennlighet litt om hverandre. Det kan argumenteres for at brukskvalitet ikke favner like vidt som usability, og at begrepet på norsk tradisjonelt sett er brukt i forbindelse med blant annet møbel-, og bygningsdesign, men jeg har likevel valgt å følge språkrådets anbefaling. (Den Norske Dataforening, 2005)

For begrepet *usability test* har jeg for valgt å i hovedsak bruke *brukertest* i stedet for brukbarhets- eller brukskvalitetstest, i likhet med Toftøy-Andersen og Wold (2011, s. 20). I enkelte tilfeller vil jeg likevel bruke *brukskvalitetstest* for å understreke brukskvalitetsaspektet.

Jeg utdyper hva som ligger i begrepet brukskvalitet i 2.1.1 Brukskvalitet.

1.3 TORCH-prosjektet

Ved institutt for arkiv-, bibliotek- og informasjonsvitenskap (ABI) ved Høgskolen i Oslo og Akershus jobber en forskningsgruppe kalt METAINFO med en rekke prosjekter knyttet til produksjon, modellering og utnyttelse av metadata med formål om å skape bedre systemer for informasjonsgjenfinning. Ett av disse prosjektene er TORCH-prosjektet, et samarbeidsprosjekt med NRK. Et av formålene ved TORCH-prosjektet er å støtte

tilgjengeliggjøring av NRKs eldre arkivmateriale ved å gjøre det i stand til å samsnakke med andre kulturarvdata.

TORCH er et stort prosjekt som rommer mange forskningsaktiviteter. Jeg vil her beskrive bakgrunnen for oppgaven min, og de mest aktuelle forskningsaktivitetene i relasjon til mitt prosjekt.

Som Michael Preminger skriver i *Bok og bibliotek*, har kulturarvdata blitt lagret digitalt i lang tid, men fordi mye av informasjonen er «lukket inn» i eldre eller utdaterte formater eller systemer er det er vanskelig å utnytte disse dataene til sitt fulle potensiale. Vi kaller dette *informasjons-* eller *datasiloer*. For å motarbeide dette har flere institusjoner, blant annet kulturarvinstitusjoner som bibliotek, begynt å legge ut sine data som lenkede åpne data som en del av den semantiske weben. (Preminger, 2015; Tallerås et al., 2014)

Formålet med dette er å kunne lenke disse dataene til andre typer data, og dermed øke samfunnsnytten av hvert datasett. Man kan for eksempel se for seg at et radioprogram som handler om *Sult* av Knut Hamsun automatisk skal kunne henvise en sluttbruker til Deichmanns katalogpost for boka, og at lenkingen mellom dem skjer maskinelt. For at dette skal være mulig må datamaskinene kunne «forstå» semantikken bak dataene, ikke bare lese tegnene. Det blir satt på spissen i et eksempel som *Sult*; uten kontekst kan ikke en datamaskin vite om vi mener romanen eller følelsen.

NRKs arkivdata er opprettet i henhold til internt utviklede metadataskjema og regler, og ligger lagret i såkalt SIFT[2]-format, hvor den viktigste delen av metadataene er fritekstlige beskrivelser av programmene. Det er gjerne journalister og programskapere som har skrevet disse beskrivelsene, og det er ofte gjort uten å ty til konvensjoner og regler som kunne gjort disse dataene lettere tolkbare for en datamaskin. Verdifulle navneentiteter, for eksempel personnavn, steder og hendelser, blir på denne måten «gjemt» i tvetydigheten som følger naturlige språkbeskrivelser. Dette hindrer maskinbearbeiding, og dermed gjenfinning og lenking. (Preminger, 2015; Tallerås et al., 2014)

For at disse fritekstbeskrivelsene i NRK-arkivene skal kunne tolkes semantisk maskinelt må man definere akkurat *hva* man er ute etter, altså hvilke navneentiteter, roller og relasjoner det er man ønsker å ekstrahere eller gjenkjenne. En del av prosjektet går derfor ut på å utvikle en

[2] Searching In Free Text/Søking I Fri Tekst

ontologi som tar hensyn til NRKs behov og våre forskningsinteresser, men også sikrer semantisk interoperabilitet med andre datasett. (Preminger, 2015; Tallerås et al., 2014)

Preminger (2015) bruker Knut Hamsun og romanen hans *Sult* som eksempel: gjenkjenningsprogrammet må forstå at «Knut Hamsun» er en person, at «Sult» er en roman, og at Knut Hamsun skrev «Sult» - dermed blir Hamsun også gjenkjent som forfatter.

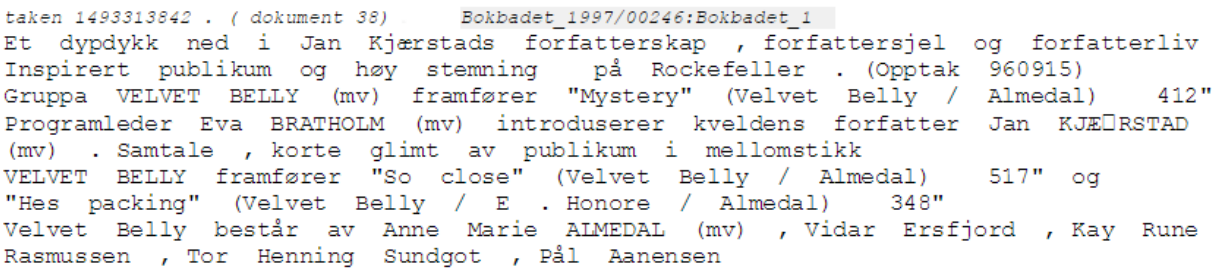
Før man kan kommer til et punkt hvor man kan snakke om *automatisk* navnegjenkjenning eller –ekstraksjon behøver man som sagt et sett med dokumenter som er manuelt annotert, som kan «trene» et dataverktøy til å automatisk annotere teksten, eller evaluere eksisterende annoteringer. Slik «opplæring» innebærer å mate et program med en mengde tekst som «ligner på» den teksten man jobber med i et gitt prosjekt, hvor man på forhånd vet hva som er ønsket resultat. Slik kan man fininnstille et program til å håndtere ukjente, men lignende tekster. For at et annotert korpus skal være brukbart til dette må det være annotert av flere personer og det må være en høy grad av konsistens i annoteringene. (Preminger, 2015)

> Mens en gjenfinningsfasit inneholder stort sett lister av dokumenter relevante til et søkespørsmål, vil fasiten til en ekstraksjonsmetode inneholde kategorien til enhver navneentitet og eventuelle relasjoner mellom entitetene. De som lager fasiten må for hvert dokument markere så mange som mulig av de relevante entitetene, og tilordne relasjoner mellom entitetene. Denne prosessen kalles «annotering». For å teste en ekstraksjonsmetode, lar man ekstraksjonsmetoden «kverne» de samme dokumentene fasiten ble laget for, og sammenligner så de automatiske resultatene med annoteringsresultatene (sistnevnte kalles også en gullstandard).
>
> (Preminger, 2015, s. 60)

Annoteringsverktøyet beskrevet i denne oppgaven brukes til slik manuell annotering, og vårt domene, altså fagområde, er kulturarv i forbindelse med eldre arkivdata fra NRK. Verktøyet er utviklet av TORCH-gruppen selv. Etter å ha vurdert en rekke tilgjengelige verktøy og ikke funnet et som passet deres behov i høy nok grad, så de det som hensiktsmessig å utvikle et eget, blant annet for å få bedre kontroll over de ulike aspektene ved annotasjonen som var spesifikke for NRK-prosjektet. Brukergrensesnittet er programmert i PHP, støttet av en relasjonsdatabasestruktur for både parametersetting og persistens. (Tallerås et al., 2014)

1.3.1 Annotering av entiteter i TORCH-prosjektet

I vårt prosjekt er det altså snakk om annotasjon av navneentiteter i et korpus bestående av SIFT-poster fra NRKs arkiver. Disse postene er programposter i fritekst, beskrivende tekstsnutter med flytende språk uten særlig struktur. Figur 1 viser et eksempel på en tekstpost av typen det vil være aktuelt å annotere. I dette tilfellet beskrives en episode av TV-programmet Bokbadet fra 1997.


taken 1493313842 . (dokument 38) Bokbadet_1997/00246:Bokbadet_1
Et dypdykk ned i Jan Kjærstads forfatterskap , forfattersjel og forfatterliv
Inspirert publikum og høy stemning på Rockefeller . (Opptak 960915)
Gruppa VELVET BELLY (mv) framfører "Mystery" (Velvet Belly / Almedal) 412"
Programleder Eva BRATHOLM (mv) introduserer kveldens forfatter Jan KJÆ□RSTAD
(mv) . Samtale , korte glimt av publikum i mellomstikk
VELVET BELLY framfører "So close" (Velvet Belly / Almedal) 517" og
"Hes packing" (Velvet Belly / E . Honore / Almedal) 348"
Velvet Belly består av Anne Marie ALMEDAL (mv) , Vidar Ersfjord , Kay Rune
Rasmussen , Tor Henning Sundgot , Pål Aanensen

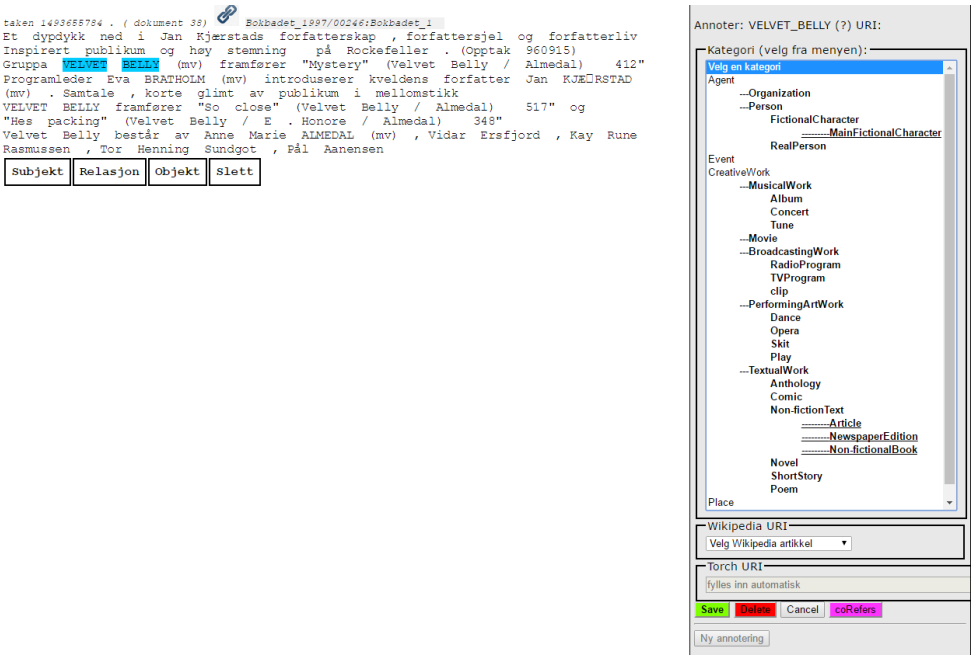
Figur 1: SIFT-post fra NRK som beskriver en episode av Bokbadet

Retningslinjene til annoteringsprosjektet beskriver hva som skal annoteres, og hvordan. Interannotør konsistens er som sagt avhengig av gode retningslinjer, og våre er basert på blant annet doktorgradsavhandlingen til Jónsdóttir nevnt tidligere. Det er snakk om navneentiteter, noe som betyr at alle fiktive og ekte personnavn, organisasjoner, stedsnavn, verk og arrangementer/begivenheter (events). Deler av navn og initialer skal også annoteres, og man skal ikke ta hensyn til skrivefeil. Pronomen som henviser til en entitet, og uttrykk for tid skal *ikke* annoteres. (*Guidelines for annotating NRK SIFT content fields*, 2014)

I retningslinjene beskrives den grunnleggende arbeidsflyten for å annotere en entitet i tre steg:

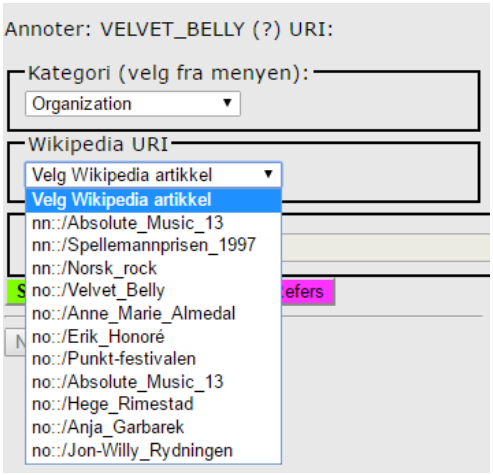
1. Marker den aktuelle entiteten
2. Kategoriser entiteten – velg kategori basert på kontekst og hva du vet, man skal ikke konsultere andre kilder enn retningslinjene
3. Velg en passende Wikipedia-URI dersom det finnes

Man annoterer en valgt entitet ved å markere ordet eller frasen på samme måte som man markerer tekst i et tekstbehandlingsprogram. Dette kalles også å skape et anker. Når man har gjort dette blir teksten man valgte markert i blått, og det dukker opp et panel til høyre, som vist i figur 2. Her velger man først korrekt kategori for entiteten, i dette tilfellet skal bandet «VELVET BELLY» etter retningslinjene kategoriseres som en organisasjon. Listen med kategorier kommer fra TORCH-ontologien som ligger «bak» verktøyet. I kategorihierarkiet skal man til enhver tid velge den «lavest mulig korrekte typen», her kan vi trekke en parallell til spesifisitetsprinsippet i indeksering. Det betyr at dersom «music band» var en kategori under «organization», ville «music band» vært riktig type å velge for entiteten «Velvet belly». I tilfeller der en entitet skal annoteres, men ikke hører til i en passende kategori, skal man velge «other», for eksempel produktnavn. (*Guidelines for annotating NRK SIFT content fields*, 2014)



Figur 2: Skjermbilde fra TORCH-verktøyet med "VELVET BELLY" markert som en entitet

Når man har valgt riktig kategori velger man en passende Wikipedia-URI. Denne listen er de øverste treffene på nynorsk, bokmål og engelsk fra en spørring med frasen mot Wikipedia API-en. Om denne Wikipediaspørringen ikke fører til et relevant resultat i listen, lar man den bare stå tom. I figur 3 kan du se resultatet av spørringen for «Velvet Belly» til Wikipdia. I dette tilfellet vil Wikipediaartikkelen om Velvet Belly på bokmål være det korrekte valget. Entiteten blir tildelt en TORCH-URI automatisk, så når man klikker på «save» er entiteten ferdig annotert.



Figur 3: Skjermbilde av listen med URI-er fra Wikipedia

Retningslinjene beskriver også hvordan man som annotør må se på konteksten til entiteten man annoterer, fordi egennavn kan høre til i forskjellige kategorier basert på konteksten den

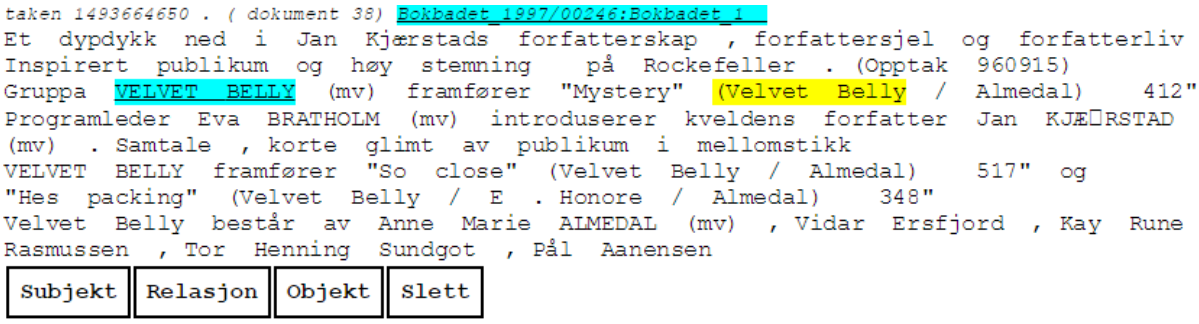
står i. For eksempel er aviser *verk* når vi snakker om produktet, men *organisasjoner* dersom vi omtaler dem som arbeidsplasser

```
Hun jobber i <annotation type="Organization">Aftenposten</annotation>.
Hun fikk <annotation type="Work">Aftenposten</annotation> på døra.
```

Retningslinjene inneholder en rekke instruksjoner om hvordan man skal behandle spesialtilfeller av de fleste typer entiteter.

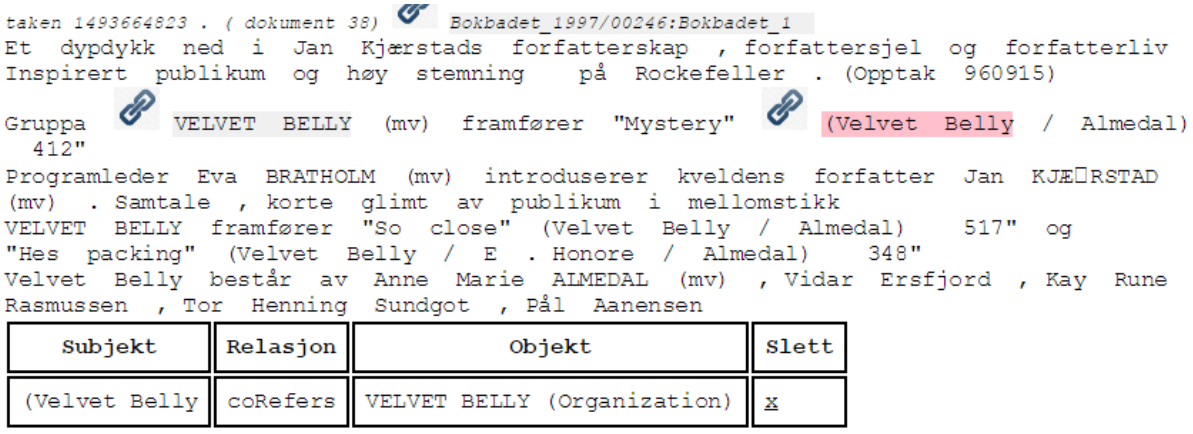
1.3.2 Ekvivalensrelasjoner

Alle navneentiteter i en gitt tekst skal annoteres, men det er slik at når man først har annotert en entitet, vil det ikke være nødvendig å gå gjennom hele annoteringsprosessen på nytt neste gang samme entitet er nevnt i teksten. Da linker man den med en såkalt ekvivalensrelasjon. Dette gjøres ved at man markerer entiteten man vil koble opp til en allerede annotert entitet og klikker på «coRefers». Entitetene du har valgt blir markert i gult, og entitetene det er mulig å linke opp mot blir markert i blått. Man velger entiteten man vil linke til ved å klikke på den. Dette er illustrert i figur 4 under.



Figur 4: Skjermbilde av opprettelsen av en ekvivalensrelasjon.

Når ekvivalensrelasjonen er opprettet, blir henvisningen som er lenket opp mot en annotert entitet markert i rødt, og relasjonen dukker opp i listen under tekstposten, som vist i figur 5.



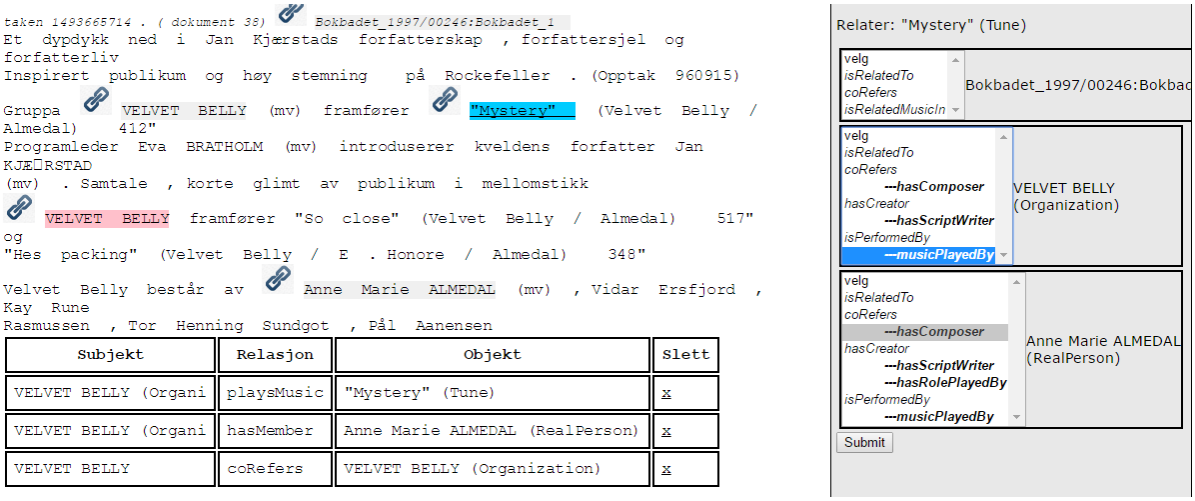
Figur 5: Skjermbilde etter at en ekvivalensrelasjon er opprettet

Retningslinjene beskriver i hvilke tilfeller det vil være aktuelt med ekvivalensrelasjoner og ikke. For eksempel vil pseudonymer, skrivefeil, initialer og språkvariasjoner være ekvivalente, men som i eksempelet i over vil ikke Aftenposten omtalt som verk være ekvivalent til Aftenposten omtalt som organisasjon eller sted.

I tilfeller av navneendringer vil man sette en ekvivalensrelasjon mellom entitetene dersom begge navnene peker til samme Wikipediaartikkel, men ikke dersom de har hver sin. For eksempel vil Høgskolen i Oslo (HiO) og Høgskolen i Oslo og Akershus (HiOA) *ikke* være ekvivalente, da det finnes en Wikipediaartikkel for HiO og en for HiOA. Bynavnene Oslo, Kristiania og Christiania vil være ekvivalente i tilfeller der alle tre er omtalt som sted fordi Wikipedia samler dem i én artikkel. (*Guidelines for annotating NRK SIFT content fields*, 2014)

1.3.3 Annotering av relasjoner mellom entiteter i TORCH-prosjektet

For å annotere relasjoner mellom entiteter dobbeltklikker man på lenkeikonet ved entiteten (🔗) og velger riktig relasjon til riktig entitet i menyen til høyre. I figur 6 illustreres hvordan relasjonsannoteringen til låten «Mystery» av Velvet Belly ser ut. Velvet Belly dukker automatisk opp som en potensiell fremfører av låten, og Anne Marie Almedal dukker opp som en potensiell komponist. Dette er hentet fra ontologien, og er basert på hvilken kategori de forskjellige entitetene har. Det betyr derfor ingenting om du går via Velvet Belly eller «Mystery» for å opprette relasjonen mellom de to. Når relasjonen er lagret dukker den opp i listen over relasjoner under tekstposten.



Figur 6: Skjermbilde av relasjonsannotering av låten "Mystery"

Det finnes per i dag ingen nedskrevne retningslinjer for annotering av relasjoner i TORCH-prosjektet.

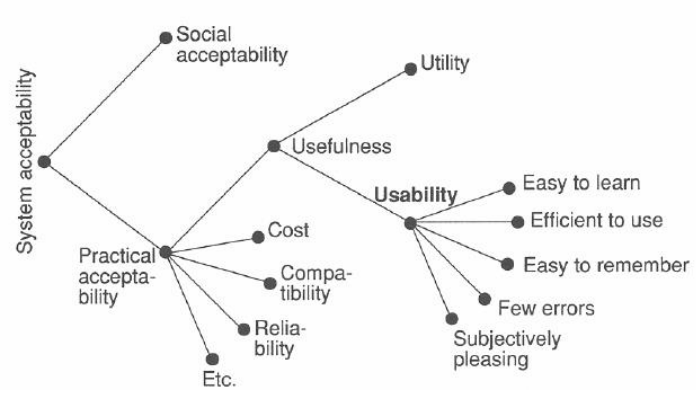
2. Teoretisk perspektiv og tidligere forskning

I dette kapittelet vil jeg presentere tidligere forskning knyttet til evaluering av brukskvalitet i annoteringsverktøy, og bruken av crowdsourcing i annoteringsprosjekter. Først avklares begrepet brukskvalitet og de mest sentrale prinsippene som ligger til grunn for evaluering av brukskvalitet.

2.1 Brukskvalitet i kontekst

Brukskvalitet er et begrep som favner vidt, og det er definert og forklart på en rekke forskjellige måter av mange mennesker gjennom tidene. Fagfeltet er kjent under mange navn, for eksempel CHI (computer-human interaction), HCI (human-computer interaction, som symbolsk «setter mennesket først»), MMI (man-machine interface), UID (user interface design), UCD (user-centered design) og så videre. De henviser til mer eller mindre det samme: studien rundt interaksjonen mellom mennesker og datamaskiner. (Nielsen, 1993, s. 23) På norsk snakker vi som oftest om brukersentrert utvikling eller interaksjonsdesign.

Jakob Nielsen, en dansk spesialist på brukervennlighet, sier at brukskvalitet er et kjennemerke som sier noe om hvor enkelt et grensesnitt er å bruke. Brukskvalitet går utover begrepet *brukervennlighet*, og sier noe om den fullstendige *brukeropplevelsen* av et produkt. For å sette brukskvalitet i kontekst, beskriver Nielsen et hierarki, eller forgrening, hvor begrepet *brukskvalitet* har sitt utspring i systemaksept (system acceptabiltiy). Se figur 7. Han beskriver brukbarhet som ett av flere attributter som bestemmer hvor godt eller dårlig et system blir akseptert av brukerne og interessentene (stakeholders). (1993, s. 25)



Figur 7: Modell av attributtene til systemaksept (Nielsen, 1993, s. 25)

Sosial aksept speiler brukerens oppfatning av produktets intensjon og innvirkning på brukeren. Som eksempel bruker Nielsen et tenkt system som undersøker om personer har rett til arbeidsledighetstrygd. Enkelte brukere kan for eksempel oppfatte spørsmålene som stilles som uakseptabelt påtrengende, eller føle et uakseptabelt ubehag ved at programmet forsøker å

«ta dem» i å lure systemet. Andre brukere være av den oppfatning at dette er en rettferdig, og dermed akseptabel, måte å gjøre det på.

Praktisk aksept går på de helt praktiske aspektene ved et system, for eksempel pris, at det er kompatibelt med andre systemer, og at det fungerer som det skal. Et punkt under praktisk aksept er *nytteverdi* (usefulness), og dette speiler brukernes oppfatning av om systemet kan hjelpe brukeren til å oppnå det de ønsker med det. Nytteverdi er sammensatt av funksjonalitet (utility) og brukskvalitet (usability). Funksjonalitet sier noe om hvilke operasjoner systemet muliggjør og i hvilken grad systemet oppfyller sin hensikt, mens brukskvaliteten er et spørsmål om hvor enkelt det er for brukeren å benytte seg av disse funksjonene. (Nielsen, 1993, s. 24–26)

I et tilfeller hvor målgruppen ikke gir sosial aksept til et system vil det være vanskelig å få aksept for systemet totalt sett selv om det tilfredsstiller de praktiske aspektene, og motsatt. Det er med andre ord mange aspekter å forholde seg til i et utviklingsprosjekt, og man må hele tiden veie dem opp mot hverandre.

I mitt tilfelle har vi et allerede sosialt og praktisk akseptert verktøy, hvor vi i stor grad har stadfestet nytteverdien med tanke på funksjonalitet og hensikt. På dette punktet er det derfor hensiktsmessig for oss å legge arbeid i å øke brukskvaliteten på verktøyet.

2.1.1 Brukskvalitet

Brooke poengterer at «usability» ikke er en egenskap som eksisterer i en ekte eller absolutt forstand. Han oppsummerer det som «a general quality of the appropriateness to a purpose of any particular artefact», og henviser til forfatteren Terry Pratchett:

> ‘Well, at least he keeps himself fit,’ said the Archchancellor nastily. ‘Not like the rest of you fellows. I went into the Uncommon Room this morning, and it was full of chaps snoring!’
>
> ‘That would be the senior masters, Master,’ said the Bursar. ‘I would say they are supremely fit, myself.’
>
> ‘Fit? The Dean looks like a man who's swallered a bed!’
>
> ‘Ah, but Master," said the Bursar, smiling indulgently, ‘the word "fit," as I understand it, means "appropriate to a purpose," and I would say the body of the Dean is supremely appropriate to the purpose of sitting around all day and eating big heavy meals.’
>
> (Pratchett, 1990, s. 38)

På samme måte må brukskvaliteten til et hvilket som helst verktøy eller system betraktes i konteksten der det brukes, og dets hensiktsmessighet i den konteksten. (Brooke, 1996)

Nielsen definisjon hvor han bruker fem *kvalitetskomponenter* for å definere brukskvalitet er en av de mest kjente og brukte. Jeg har valgt å oversette dem slik:

- Lett å lære (learnability)
- Effektivt (efficiency)
- Lett å huske (memorability)
- Feilhåndtering (errors)
- Tilfredshet (satisfaction)

(Nielsen, 1993, s. 26)

At noe er lett å lære sier noe om hvor enkelt det er for en bruker å utføre grunnleggende oppgaver første gang de møter designet. Effektivitet sier noe om hvor raskt og nøyaktig brukeren kan utføre oppgavene når de har lært designet å kjenne. At det er lett å huske sier noe om hvor lett det er å huske hvordan man bruker programmet etter en periode uten bruk. Feilhåndtering innebærer hvor mange feil som gjøres, hvor alvorlige feilene er, og hvor enkelt brukeren kommer videre etter å ha gjort en feil. Tilfredshet sier noe hvor tilfreds brukeren selv er med å bruke designet, med andre ord at de aksepterer det. Nielsen presiserer at brukskvalitet ikke er én egenskap, men at egenskapene nevnt ovenfor må ses i sammenheng. (Nielsen, 1993, kap. 2.2)

Standardiseringsorganisasjonen International Organization for Standardization (ISO) har utarbeidet en egen standard som omhandler interaksjon mellom menneske og maskin, *ISO 9241 Ergonomics of Human System Interaction*. Dette er en svært omfattende og stadig voksende serie som tar for seg forskjellige aspekter ved interaksjonsdesign. Jeg har sett spesielt på del 11 «Usability: Definitions and concepts» og del 210 «Human-centred design for interactive systems». Del 11 ble publisert i 1998 og het da «Guidance on usability». Den ble sist gjennomgått og bekreftet i 2008, men en revidert versjon er i disse dager i det siste stadiet av utviklingen før endelig godkjenning og publisering, såkalt DIS-registrert (Draft International Standard). Den er blant annet oppdatert for å samsvare med del 210 som ble publisert i 2010, så mens jeg har konsultert begge versjonene, har jeg valgt å i hovedsak forholde meg til den nyeste. (ISO, 1998, 2010, udatert)

ISO-standard 9241:210 definerer brukskvalitet på denne måten: «the extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use» (ISO, 2010, del 2.13). Som vi ser så er det to ord for effektivitet i det engelske språket; *effectiveness* og *efficiency*. Disse er definert som henholdsvis: «accuracy and completeness with which users achieve specified goals» og «resources expended in relation to the accuracy and completeness with

which users achieve goals» (ISO, 2010, del 2.3, 2.4). Veldig forenklet kan vi si at effectiveness er å gjøre de rette tingene, mens efficiency er å gjøre ting raskt.

I *Research methods in information* skriver Pickard at vi i bibliotek- og informasjonsvitenskap går ut over de typiske forskningsmetodene innen brukskvalitet som kun fokuserer på forholdet mellom bruker og system, ved at vi i tillegg assosierer det med brukeratferd, informasjonssøkeadferd og informasjonsbehov. Det er en kombinasjon av alle disse aspektene som sier noe om et produkts brukskvalitet. (2007, s. 127)

I «Technical Review: Current Issues of Usability Testing» sammenlignet Alshamari og Mayhew seks av de mest brukte referanseverkene for brukskvalitet, blant annet Nielsen og ISO-standarden jeg beskrev tidligere. De konkluderte med at de fleste definisjonene vektlegger «efficiency, effectiveness and user satisfaction», mens variasjonene i definisjonene ofte er avhengig av hva slags systemer det er snakk om. (2009, s. 402)

For å understreke og illustrere at brukskvalitet er et mangefasettert konsept det kan være vanskelig å definere, spesielt på norsk, har jeg valgt å inkludere Dix, Finlay, Abowd og Beale sin liste med prinsipper som støtter brukskvalitet. Prinsippene er hentet fra *Human-computer interaction,* og har en rekke likhetstrekk med Nielsens kvalitetskomponenter, men er mye mer spesifikke. Prinsippene kan deles inn i tre hovedkategorier på samme abstraksjonsnivå: *learnability, flexibility* og *robustness.* Hver av disse hovedkategoriene inneholder en rekke faktorer som påvirker brukskvaliteten til et produkt. (Dix, Finlay, Abowd & Beale, 2003, kap. 7.2) Tabellene 1-3 under viser sammendrag av de tre kategoriene og hva de inneholder.

Principle	Definition	Related principles
Predictability	Support for the user to determine the effect of future action base don past interaction history	Operation visibility
Synthesizability	Support for the user to assess the effect of past operations on the current state	Immediate/eventual honesty
Familiarity	The extent to which a users' knowledge and experience on other real-world or computer-based domains can be can be applied when interacting with a new system	Guessability, affordance
Generalizability	Support for the user to extend knowledge of specific interaction within and across applications to other similar situations	
Consistency	Likeness in input-output behaviour arising from similar situations or similar task objectives	

Tabell 1: Sammendrag av prinsippene som påvirker "learnability" (Dix et al., 2003, s. 261)

Principle	Definition	Related principles
Dialog initiative	Allowing the user freedom from artificial constraints on the unput dialog imposed by the system	System/user pre-emptiveness
Multi-threading	Ability of the system to support user interaction pertaining to more than one task at a time	Concurrent vs. Interleaving, multi-modality
Task migratability	The ability to pass control for the exevution of a given task so that is becomes either internalized by the user of the system or shared between them	
Substitutivity	Allowing equicalent values of input output to be arbitrarily substituted for each other	Representation multiplicity, equal opportunity
Customizability	Modifiability of the user interface by the user or the system	Adaptivity, adaptability

Tabell 2: Sammendrag av prinsippene som påvirker "flexibility" (Dix et al., 2003, s. 266)

Principle	Definition	Related principles
Observability	Ability of the user to evaluate the internal state of the system from its perceivable representation	Browsability. static/dynamic defaults, reachability, persistence, operation visibility
Recoverability	Ability of the user to take corrective action once an error has been recognized	Reachability, forward/backward recovery, commensurate effort
Responsiveness	How the user percives the rate of communication with the system	Stability
Task conformance	The degree to which the system services support all of the tasks the user wishes to perform and in the way that the user understands them	Task completeness, task adequacy

Tabell 3: Sammendrag av prinsippene som påvirker "robustness" (Dix et al., 2003, s. 266)

Learnability sier, i likhet med Nielsens brukskvalitetskomponent med samme navn, noe om hvor enkelt det er for nye brukere å utføre grunnleggende oppgaver, men inkluderer her hvor raskt de kan oppnå effektiv samhandling med systemet. *Flexibility* sier noe om mangfoldet av måter som bruker og system utveksler informasjon på. *Robustness* sier noe om i hvilken grad brukeren støttes i prosessen for å oppnå ønsket resultat.

2.1.2 Brukskvalitet og annoteringsverktøy

Hvis crowdsourcing skal bli en realitet, må det være et mål å skape et verktøy som legger til rette for at ikke-eksperter skal kunne skape meningsfulle og konsekvente annoteringer.

> Corpus search tools have over the years developed to become very user-friendly. By contrast, corpus annotation programs – while widely available – typically require so much advanced computer expertise to install and use that they are, effectively, not accessible to most linguists.
>
> (McEnery & Hardie, 2011, s. 33)

I «Semantic Enrichment by Non-Experts: Usability of Manual Annotation Tools» identifiserte forfatterne følgende nøkkelkrav for manuelle annoteringsverktøy for ikke-eksperter: etablerte interaksjonsmønstre (established interaction patterns), enkle vokabularer (simple vocabularies), kontekstuell semantisk identitet (contextual semantic identity) og fokus på brukerens oppgave (focus on the user's task). (Hinze, Heese, Luczak-Rösch, & Paschke, 2012, del 3)

Etablerte interaksjonsmønstre innebærer at semantiske webapplikasjoner ser ut som tradisjonelle applikasjoner og bruker kjente interaksjonsparadigmer. Enkle vokabularer er viktig fordi forskning tyder på at komplekse kategoristrukturer gjør at det oppstår forvirring hos brukeren. Kontekstuell semantisk identitet betyr å bygge bro mellom objektiv kunnskap kodet i datamodeller og subjektiv kunnskap fra menneskelig kognisjon. For eksempel hvordan datamaskiner identifiserer ressurser ved hjelp av URI-er, i motsetning til mennesker som identifiserer entiteter ved navn og fjerner tvetydighet ved hjelp av kontekst. Fokus på brukerens oppgave innebærer at semantisk forfatterskap og semantisk annotering er integrert på en god måte. (2012, del 3) Sistnevnte er ikke relevant for denne oppgaven, da TORCH-verktøyet er et rent annoteringsverktøy.

Hinze et al. hevder at forskningen som er gjort på ikke-eksperters bruk av semantiske annoteringsverktøy har fokusert på resultatene av annoteringene, ikke sluttbrukernes opplevelse av verktøyet. For eksempel finnes det en rekke undersøkelser som tar for seg inter- og intraannotørkonsistens, eller som sammenligner korpus annotert av ikke-eksperter mot gullstandarder annotert av eksperter. De hevder at fokuset bør flyttes, fordi man ved å undersøke brukskvaliteten til slike verktøy, og dermed skape systemer som er enkle å forstå og enkle å bruke, vil ende opp med annoteringer av bedre kvalitet. (Hinze et al., 2012)

> Few studies involving end-users have been executed in the context of semantic annotations. In particular, manual annotation tools have so far not been systematically evaluated for appropriate interaction design and semantic understanding. System evaluations that incorporated human participants did not seek their feedback on interaction issues nor did they evaluate the participants' mental models of the system interaction. So far, issues of understanding of semantic annotations by (non-expert) users have not been studied in a systematic manner.
>
> (Hinze et al., 2012)

I artikkelen beskriver de en brukertest av et egenutviklet annoteringsverktøy, *loomp One Click Annotator*, med tolv brukere på en papirprototype av systemet. De tolv brukerne hadde variert bakgrunn, og basert på deres vurdering av egen kjennskap til tekstbehandling, annotering/tagging, informasjonsteknologi og semantisk web kategoriserte forskerne dem

som henholdsvis 6 tekniske eksperter og 6 ikke-eksperter. Basert på resultatene av testen ble brukerne kategorisert i tre grupper: akseptable annotører (acceptable annotators), annotører med forbedringspotensiale (annotators with room for improvement) og uakseptable annotører (failed annotators). Alle i kategorien akseptable annotører var også kategorisert som tekniske eksperter, altså at de hadde kjennskap til semantisk web og lignende, men ikke alle de tekniske ekspertene ble kategorisert som akseptable annotører. Noe overraskende havnet enkelte tekniske eksperter fra yrkesgrupper man ville antatt at gjorde det godt i denne settingen, for eksempel bibliotekarer, i kategorien uakseptable annotører. Resultatene indikerte altså at domeneeksperter ikke nødvendigvis er gode annotører, fordi det i tillegg krever en viss forståelse av konseptet semantisk annotering. For eksempel ønsket enkelte deltakere å lage sammendrag eller synonymer i stedet for annoteringer, og noen oppga at de var usikre på om de var ferdige med en annotering eller om annoteringen var hensiktsmessig. Artikkelforfatterne argumenterer for at årsakene til disse vanskelighetene var mangel på konseptuell forståelse hos brukerne og dårlig brukskvalitet på systemet. Forfatterne mener at resultatene til de to første gruppene kan forbedres ved hjelp av bedre instruksjoner og annoteringsverktøy, mens den siste gruppen ikke er egnet som annotører. Jeg har brukt samme fremgangsmåte som Hinze et al. for å kategorisere deltakerne og resultatene i mine forsøk.

I *Engineering Annotation Usability - Toward Usability Patterns for Linguistic Annotation Tools*, doktorgradsavhandlingen til tyske Manuel Burghardt, presenteres det han kaller brukskvalitetsmønstre (usability patterns), som han har utledet ved å evaluere ulike verktøy for manuell lingvistisk annotering. Et designmønster er en formell måte å dokumentere en gjenbrukbar løsning på et vanlig designproblem. Han evaluerte elleve verktøy, og definerte tjueseks slike mønstre som skal hjelpe utviklere med å unngå vanlige feil og fallgruver i utformingen av grensesnitt til annoteringsverktøy spesielt. De kan også brukes som veiledere for å velge passende annoteringsverktøy til gitte prosjekter med hensyn til bruksbehov. Med utgangspunkt i teori om håndskreven annotering og lingvistiske annotasjonsstandarder og -praksis, bygget undersøkelsene hans på etablerte konsepter og metoder for evaluering av brukskvalitet. Selve prosessen ved å identifisere slike mønstre er ifølge Burghardt ofte en noe vag prosess, men han beskriver sin avhandling som en systematisk tilnærming som integrerer kvalitative resultater fra en rekke heuristiske gjennomganger i mønsteridentifikasjon for brukskvalitet. (Burghardt, 2014, kap. 5)

Han kategoriserer de tjueseks mønstrene i ulike kategorier: generelt grensesnitt (general UI), installering (installation), primærdata (primary data), annoteringsskjema (annotation scheme), annoteringsprosessen (annotation process) og annoteringspresentasjonen (annotation visualization). I denne undersøkelsen er problemer ved det generelle grensesnittet og annoteringsprosessen de viktigste kategoriene, da de andre omhandler ting som våre brukere ikke må forholde seg til, som installering av programvare og endring av primærdata eller annoteringsskjema. Av trinnene Burghardt beskriver som en del av annoteringsprosessen er disse trinnene aktuelle for oss: velge anker, velge eksisterende annotering, lagre annotering, rediger annotering og slett annotering. Jeg brukte dette som utgangspunktet da jeg konstruerte oppgavene i testene mine.

Burghardt presenterer også en rekke unike brukskvalitetsbehov og funksjonskrav til annoteringsverktøy. Ved å utelate rene funksjonskrav som ikke er aktuelle i vår kontekst, og de blandede kravene til funksjoner som TORCH-verktøyet ikke innehar, er fem av kravene relevante for TORCH-verktøyet. Det er to rene brukskvalitetsbehov: dokumentasjon (documentation), altså tilgjengeligheten av en brukerhåndbok eller retningslinjer, og et generelt behov for et grensesnitt som er enkelt å bruke (easy-to-use interface). De tre blandede kravene som er relevante for oss beskriver brukskvalitetsaspekter som er kategorisert som relevante for vår brukergruppe: presentasjon av primærdata (visualization of primary data), presentasjon av annoteringer (visualization of annotation) og markering av anker (marking of anchors).

Presentasjon av primærdata og annoteringer betyr at den opprinnelige teksten skal vises riktig i annoteringsverktøyet, og at verktøyet tydelig skal skille mellom original og annotert tekst. Markering av anker sier at grensesnittet skal hjelpe annotøren med å identifisere og markere potensielle ankre på en intuitiv og effektivt måte.

Burghardt arbeidet mot lingvistiske annoteringsverktøy med et bredere nedslagsfelt enn annotering av navneentiteter, men siden annoteringsprosjekter innebærer lignende aktiviteter uansett type, er funnene hans relevante og gyldige også i denne konteksten. Jeg brukte derfor hans mønstre og krav for å identifisere og kategorisere problemene jeg fant i mine undersøkelser.

2.2 Crowdsourcing i annoteringsprosjekter

Som nevnt tidligere har det blitt gjort en rekke undersøkelser hvor forskere har diskutert fordeler og ulemper ved crowdsourcing i en rekke typer annoteringsprosjekter, ofte ved å

sammenligne korpus annotert av ikke-eksperter mot gullstandarder annotert av eksperter. Det er gjort undersøkelser i forbindelse med mange typer annoteringsprosjekter, for eksempel innen biomedisin og sosiale medier, og det er bred enighet i at man kan spare mye penger og tid på å atomisere annoteringsprosessen og crowdsource enkelte annoteringsaktiviteter, men at man risikerer at det kan gå på bekostning av kvaliteten til en viss grad.

Et verktøy som er mye brukt i sammenheng med crowdsourcing er Amazon Mechanical Turk (AMT), en internettbasert «markedsplass» hvor man kan legge ut «human intelligence tasks» som andre mennesker kan utføre for en liten sum penger. Her kan man enkelt gjennomføre crowdsourcing av oppgaver, som for eksempel fylle ut spørreundersøkelser, skrive produktbeskrivelser, stemme i kåringer, eller annotere data. I «Cheap and Fast – But is it Good? Evaluation Non-Expert Annotations for Natural Language Task» gjennomførte forfatterne en rekke forsøk med AMT for å vurdere dataene ved annotering av ikke-eksperter og eksperter i flere forskjellige typer annoteringsoppgaver. De konkluderte med at det for de fleste annoteringsoppgavene ikke var nødvendig med mer enn 10 ikke-ekspertannoteringer for å likestille resultatet med gullstandarden satt av en ekspert. De hevder at man absolutt kan spare tid og penger på å crowdsource enkelte annoteringsaktiviteter, for eksempel regnet de ut at ved gjenkjenning og annotering av følelser formidlet i tekst, krevdes det i gjennomsnitt kun fire annoteringer fra ikke-eksperter for å emulere en annotering av ekspertkvalitet. (Snow, O'Connor, Jurafsky, & Ng, 2008)

Innen biomedisin har det vært prosjekter hvor det har blitt eksperimentert med crowdsourcing i forbindelse med gjenfinning i store artikkeldatabaser. «Microtask crowdsourcing for Disease Mention Annotation in PubMed Abstracts» er en av de tidligere forskningsprosjektene som brukte AMT for å crowdsource annotering, i dette tilfellet av sykdommer i artikkelsammendrag fra PubMed. Hvert sammendrag ble annotert av 15 personer, og totalt annoterte 145 personer 593 dokumenter i løpet av 9 dager til en samlet pris på under $600. Kvaliteten på annoteringene økte med antall personer som ble tildelt hvert dokument, men gevinsten blir liten i forhold til økt forbruk av tid og ressurser når antallet annotører oversteg 8. Crowdsourcing i annoteringsprosjekter gjør det altså mulig å skape langt flere og langt større treningskorpus enn hva man kan se for seg med ekspertbaserte tilnærminger. Samtidig anerkjenner de at deres prosjekt har en tydelig gullstandard: enten nevner en tekst sykdommer eller ikke, det finnes ingen gråsoner. Dette er en «luksus» man ikke vil ha i alle annoteringsprosjekter, da dette binære premisset ikke reflekterer språk i virkeligheten. De mener likevel at crowdsourcet data i disse tilfellene vil kunne brukes til å identifisere

tvetydigheter i språk og annotasjonsoppgaver på en måte som kan føre til viktige fremskritt på feltet. (Good, Nanis, Wu, & Su, 2015)

I «Web 2.0-based crowdsourcing for high-quality gold standard development in clinical natural language processing» kom de til samme konklusjon, men la i tillegg vekt på viktigheten av et godt annoteringsverktøy.

> Based upon the results of our experiments, we conclude that crowdsourcing is a feasible, inexpensive, fast, and practical approach to annotate clinical text (when protected health information is not included) on large scale for medical named-entities. We believe that well-designed user interfaces for entity annotation and linking were critical to the success of this work.
>
> (Zhai et al., 2013)

I dette prosjektet var det snakk om annotering av medisinnavn og -typer, og deres egenskaper. I det første forsøket var fokuset på å annotere medisinske navneentiteter i et stort korpus. I det andre handlet det om å knytte attributter til allerede annoterte entiteter. Det tredje forsøket ble gjennomført etter en iterativ modell i et forsøk på å skape en mer robust, manuelt opprettet gullstandard ved å la annotørene korrekturlese andres annoteringer i flere omganger. De presenterte også en strategi for kvalitetskontroll av annotører som inkluderte en innledende quiz, geografiske avgrensinger, opplæring og kontinuerlig prestasjonsovervåking. Ved først å opprette en intern gullstandard for en del av korpuset kunne de bruke det til opplæring av ikke-eksperter, gjennomføre kvalitetssikring kontinuerlig underveis og kvalitetssjekk til slutt. For eksempel måtte alle annotører gjennom en innledende test, og om resultatene deres samsvarte med gullstandarden med minst 50% ville personen bli kategorisert som klarert (trusted). Etter å ha silt ut uegnede annotører på denne måten, målte de kontinuerlig resultatene på 20% av det faktiske korpuset mot gullstandarden, og kunne på denne måten gi annotørene advarsler dersom kvaliteten på annoteringene deres sank underveis. Dersom troverdigheten til annotøren sank under 50% og den ikke forbedret seg på de to neste gullstandardtestene etter advarselen, ville bli annotøren hindret fra å delta videre, og resultatene deres ble ekskludert fullstendig fra forsøket. Til sammen fullførte 156 personer det første forsøket, 86 andre og 46 tredje. Ved å måle fullstendighet og presisjon på de forskjellige typene annoteringsaktivitetene mot gullstandarden, regnet de ut F-scoren[3] (det harmoniske gjennomsnittet for presisjon og fullstendighet) for resultatet av hver av dem. F-scoren ble brukt til å måle samsvaret mellom det crowdsourcede resultatet og gullstandarden annotert av interne eksperter. De rapporterte «høyt» samsvar i entitetsannotering (0,87 i F-

[3] $F - measure = 2 * (precision * recall)/(precision + recall)$

score for medisinnavn, 0,73 for medisintyper) og retting av andres resultater (0,90 for medisinnavn, 0,76 for medisintyper), og «utmerket» samsvar i kobling mellom entiteter og attributter (0,96). De konkluderer med at crowdsourcing er en gjennomførbar, billig, rask og praktisk tilnærming til å annotere medisinske navneentiteter i kliniske studier i stor skala. (Zhai et al., 2013)

Det er også gjort studier hvor man har sett på crowdsourcing av annotering av navneentiteter i sosiale medier, blant annet twitter. Twitter er en mikrobloggingtjeneste som lar brukerne sende og lese tekstbaserte meldinger bestående av inntil 140 tegn. Twitter er en uformell kommunikasjonsplattform, og i kombinasjon med begrensingen på 140 tegn er det veldig vanlig å bryte normale skriveregler, for eksempel med utradisjonelle forkortninger og akronymer. Det finnes også en rekke elementer som man vanligvis ikke finner i vanlig tekst, men som er vanlig i tweets, for eksempel emneknagger (hashtags), emotikons og brukernavn.

I «Annotating named entities in Twitter data with crowdsourcing» ble AMT og crowdsourcingplattform kalt CrowdFlower brukt for å crowdsource annotering av navneentiteter i tweets (Finin et al., 2010). Brukerne fikk instruksjoner om å annotere personer, organisasjoner og steder, og i likhet med studien til Zhai et al. ble brukernes resultater evaluert mot en gullstandard av ekspertannoterte tweets underveis i annoteringsprosessen. I AMT hadde 1 av 5 tweets en tilhørende gullstandard for å luke ut personer som annoterte tilfeldig for profitt. Resultatene deres indikerte at en bruker kan annotere cirka 400 tweets i timen, og at man ved å kombinere resultatene fra kun to brukere kan få et brukbart resultat til en mye lavere pris enn ved ekspertannotering.

> Also of interest is the raw effectiveness of MTurk workers; did they manage to tag tweets as well as our experts? After investigating the data, our verdict is that the answer is not quite—but by carefully combining the tags that two people give the same tweet it is possible to get good answers nevertheless, at much lower cost than employing a single expert.
>
> (Finin et al., 2010)

De konkluderer med at crowdsourcing er en effektiv måte å skaffe annoteringer til bruk i forskning på naturlig språk og informasjonssøkning.

I «Crowdsourcing Named Entity Recognition and Entity Linking Corpora» beskriver forfatterne et prosjekt hvor de brukte CrowdFlower og annoteringsverktøyet GATE til å annotere tweets på to forskjellige måter: den ene typen var annotering av navneentiteter (annotation with named entities), den andre var entitetslenking mot en ekstern kilde av navneautoriteter (entity linking). Den første delen av prosjektet var entitetsannotering, og

korpuset bestod av 100 tweets annotert av eksperter og 475 utvalgte tweets for crowdsourcing. De fikk eksperter til å annotere en gullstandard for en del av korpuset for kvalitetskontroll, og i likhet med Finin et al.. ble minst hver femte tweet en deltaker annoterte sammenlignet med gullstandarden. I tillegg måtte deltakerne fullføre en test hvor resultatene deres måtte samsvare med gullstandarden i en viss grad før de fikk annotere tekster uten tilhørende fasit. I annoteringen av navneentiteter bestod 76 % av deltakerne denne testen, og resultatene til disse brukerne oppnådde 97 % interannotør konsistens med materialet som var annotert av eksperter. I entitetslenkingen skulle deltakerne identifisere entiteter og tildele dem en passende URI fra DBbpedia. I likhet med vårt prosjekt ble brukeren presentert med et begrenset antall URI-er å velge mellom, innholdet var avhengig av hva systemet anså som mest relevant fra et oppslag på det aktuelle ankeret. Dette er spesielt risikabelt i korpus fra sosiale medier, da det som sagt er normalt å ikke følge formelle skriveregler, og entiteter gjerne vil bli referert til med akronymer, kallenavn eller forkortelser. De mappet derfor ontologien deres til ontologien i DBpedia, og kombinerte oppslaget med informasjonen i de allerede annoterte entitetene fra den første delen av prosjektet. På denne måten ville for eksempel et oppslag på ankeret «Paris» ha sted eller person som avgrensing i DBpedia, avhengig av hva som var annotert fra før. I denne delen bestod korpuset av 400 tweets med 577 forekomster av entiteter, 177 av disse entitetene hadde en tilhørende gullstandard. Her var det kun 11 deltakere, men alle deltakerne bestod den innledende testen. Interannotør konsistens ble rapportert å være 80 % mellom ikke-ekspertene, noe som er på høyde med interannotør konsistensen mellom ekspertene, som ble rapportert som 89 %. De fant også ut at ved å presentere eksperter med resultatene til ikke-ekspertene, og la dem markere det som korrekt eller ukorrekt, var en raskere og enklere måte å oppnå et godt resultat enn å la eksperter annotere alle postene fra bunn av. (Bontcheva, Derczynski, & Roberts, under utgivelse)

Denne studien er ekstra interessant med tanke på at entitetsannoteringen i TORCH-prosjektet kombinerer de to typene annotering som ble beskrevet. Som nevnt i 1.3.1 Annotering av entiteter i TORCH-prosjektet identifiserer brukeren entiteter og annoterer entitetens rolle i en gitt ontologi, og entiteten kan tildeles en URI fra Wikipedia.

3. Brukskvalitetsevaluering

Jeg har valgt å bruke brukskvalitetstesting, eller brukertesting, som metode for å evaluere brukskvaliteten på annoteringsverktøyet i min oppgave. I likhet med Toftøy-Andersen og Wold bruker jeg i hovedsak uttrykket brukertstesting, men de betyr det samme i denne teksten. En brukertest er en kvalitativ, brukerbasert tilnærming som kombinerer observasjon og intervju. I dette kapittelet redegjør jeg for valg av metode, og jeg diskuterer svakheter og styrker ved brukertesting og andre måter å evaluere brukskvalitet på.

> «[Brukertesting] er det nærmeste du kommer til et trylleformular for å forbedre brukervennlighet.» (Arlov, 1999, s. 271)

Målet med brukertesting er å få innsikten man trenger for å evaluere brukervennligheten til et produkt, og dermed identifisere eventuelle feil og mangler. I tradisjonell brukertesting observerer man brukere i samhandling med det man ønsker å teste, i mitt tilfelle en nettbasert applikasjon. Man benytter testpersoner som er representative for brukergruppen, og ber disse gjennomføre oppgaver som er konstruert for å representere ulike funksjoner i designet som testes. Det simuleres en reell situasjon hvor man observerer brukeren utføre oppgaver knyttet til systemets viktigste funksjonalitet mens brukeren tenker høyt. Ved å observere brukeren mens hun løser oppgaver får man se mer av ubevisste forventninger til designet. Brukertesting er ifølge Toftøy-Andersen og Wold den eneste evalueringsmetoden som tar for seg *interaksjonen* mellom bruker og system (Arlov, 1999, s. 173; Toftøy-Andersen & Wold, 2011, s. 20).

Rubin og Chisnell diskuterer i *Handbook of Usability Testing : How to Plan, Design, and Conduct Effective Tests* hvordan den grunnleggende metodikken for å gjennomføre en brukskvalitetstest ikke kan følge den klassiske tilnærmingen for å gjennomføre et kontrollert eksperiment, til tross for at den har sitt utspring i metoden. Et kontrollert eksperiment innebærer at det formuleres en bestemt hypotese, og at den deretter testes ved å isolere og manipulere variabler under kontrollerte forhold. Forhold som årsak og virkning blir deretter nøye undersøkt, og hypotesen enten bekreftet eller avvist ved hjelp av kontrollgrupper. Den klassiske metoden med kontrollerte eksperimenter er utformet for å skaffe seg kvantitative bevis på forskningshypoteser, for eksempel ved å bevise at et design presterer bedre enn et annet. Å gjennomføre en brukertest på denne strengt formelle måten er ikke umulig, men lite hensiktsmessig. Den er ikke egnet for å skaffe seg kvalitativ informasjon om hvordan du løser problemer og redesigner produkter, som er det vi ønsker å oppnå med brukertesting. (2008, s. 23–24)

Pickard skriver at brukertesting i formell forstand heller bør knyttes til kvasi-eksperimentet, hvor oppførsel blir observert, registrert og analysert i en strukturert kontekst. Konteksten kan variere, for eksempel kan en gruppe testbrukere bli bedt om å utføre et sett med forhåndsbestemte oppgaver på et system i deres naturlige omgivelser, eller inviteres inn i en testlab for å observeres mens de interagerer med systemet. (2007, s. 128)

Arlov mener at der man med intervjuer og diskusjoner ender opp med meninger, får man med brukertesting noe som nærmer seg data. Hun sier «nærmer seg» fordi man må «jukse litt» i forhold til forskningsmessige standarder ved at en må vurdere hvordan testbrukerne er representative for brukergruppen, og hvordan de avviker. Det er heller ikke mulig å teste så mange brukere at man kan påstå at noe er «bevist». Det er heller ikke ønskelig, da bredde i testgruppen er viktigere enn antall testbrukere for å finne problemer. Jeg skriver mer om dette i 3.3.2 Utvalg av testbrukere. Man har heller ikke anledning til å teste i autentiske arbeidsomgivelser med reelle data, svartider og oppgaver. Det blir derfor en utfordring å vurdere hvordan testsituasjonen avviker fra virkeligheten, og hvordan det påvirker resultatene. I tillegg kan oppgavens utforming og testleders oppførsel påvirke testbrukerens oppførsel og reaksjoner. (1999, s. 275)

Mehlenbacher presiserer at dataene som produseres ved brukertesting ikke i seg selv løser designproblemer, men at man ved å tolke disse kan identifisere systemmangler og foreslå mulige designløsninger. I litteraturen har det vært en del forvirring rundt dette, og han siterer Flower som sier det på en god måte:

> All data can do is provide the foundation for interpretation… To say that data is only data, is also a statement about epistemology. In taking an observation-based approach to theory-building one cannot treat data as if it were a source of immutable, objective facts or transparent proofs, even when that data comes from personal experience. When data is used to build an interpretive theory, it cannot be "read" directly without reference to the rules of evidence that constitute the discourse of research.
>
> L. Flower i artikkelen *Cognition, Context and Theory Building* (sitert i Mehlenbacher, 1993, s. 211)

Data samlet under brukervennlighetstesten *er* med andre ord en del av designprosessen. Testing produserer data som utviklerne kan tolke for å peke på problemer i brukskvaliteten, og bruke det for å finne mulige løsninger på disse problemene. (Mehlenbacher, 1993, s. 211)

3.1 Hvorfor evaluere brukskvalitet?

Ifølge forfatterne av *Human-Computer Interaction* (2003), har brukskvalitetsevaluering tre hovedmål: å vurdere omfanget av og tilgjengeligheten til programmets funksjonalitet, å

evaluere brukernes opplevelse av programmet, og å identifisere spesifikke problemer ved designet. (Dix et al., 2003, s. 319)

Å kartlegge systemets funksjonalitet er viktig fordi det vil være enklere for brukerne å utføre de tenkte oppgavene dersom programmets funksjonalitet stemmer overens med brukernes krav og forventninger. Dette innebærer mer enn at de aktuelle funksjonene *eksisterer* i programmet, de må også være lett tilgjengelige og forståelige for de tiltenkte brukerne. Det kan også være aktuelt å måle brukerens ytelse for å se hvor effektivt systemet støtter utførelsen av diverse oppgaver.

I tillegg til å evaluere systemets design med tanke på funksjonalitet, er det viktig å kartlegge brukernes opplevelser med å bruke programmet. Hvor lett føler brukerne det er å lære seg å bruke programmet? Hvor fornøyd er brukerne med programmets funksjonalitet? (Dix et al., 2003, s. 319–320)

Formålet med brukskvalitetsevalueringer er altså å luke ut elementer som kan føre til uforutsette eller uønskede resultater under bruk, eller som gjøre at det oppstår forvirring og usikkerhet hos brukerne. Å identifisere problemene er første steg mot å rette dem opp.

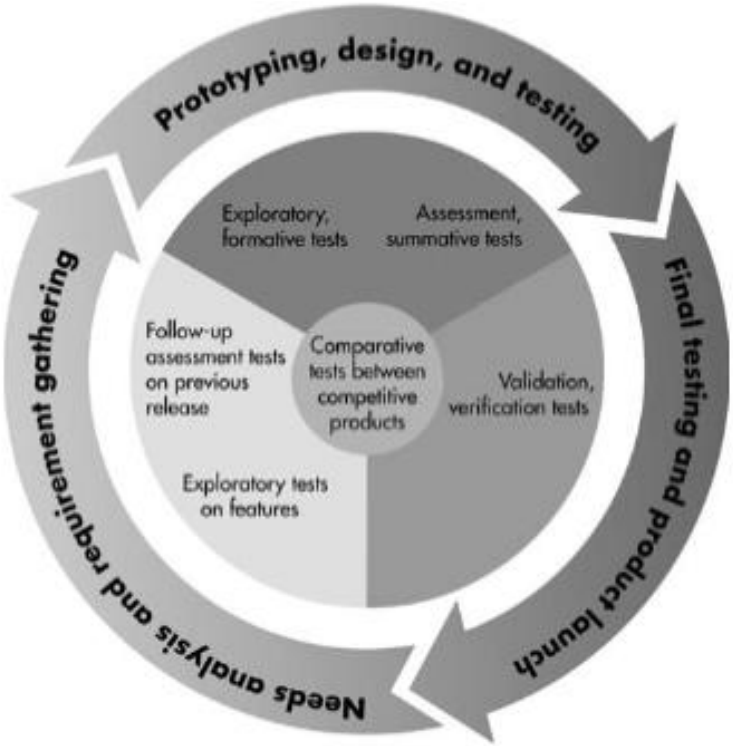
3.2 Hvordan evaluere brukskvalitet?

I *Human-Computer Interatction* hevder Dix et al. at de mest diskuterte metodene for å evaluere brukskvalitet kan deles inn i to hovedkategorier basert på hva de fokuserer på: ekspertanalyse og brukermedvirkning. I brukerbaserte metoder får man som regel et utvalg fra målgruppen til å bruke programmet, mens man i ekspertbaserte metoder får en eller flere brukskvalitetseksperter til å foreta en vurdering av programmet. Hva man velger å bruke er avhengig av ressurser som tid, tilgang på hensiktsmessige eksperter og hvor i utviklingsprosessen produktet er. (2003, s. 319) Scholtz (2004) introduserer en tredje kategori: modellbaserte metoder. I modellbaserte metoder benytter en ekspert formelle modeller for å si noe om brukervennligheten. Dix et al. har inkludert disse i kategorien ekspertbaserte metoder fordi de nevnte modellene gjerne krever at en ekspert bruker dem, men jeg har, i likhet med Scholtz, valgt å skille ut modellbaserte metoder i et eget underkapittel.

Det er ikke uvanlig at et system blir evaluert ved hjelp av flere metoder i løpet av utviklingen.

3.2.1 Brukerbaserte metoder

Å teste et program med en gruppe representative brukere som utfører et sett med forhåndsbestemte oppgaver, anses generelt for å gi det mest pålitelige og gyldige estimatet av et produkts realistiske brukervennlighet. Målet med denne typen evaluering er å undersøke i hvilken grad programmet støtter de tiltenkte brukerne i arbeidet de ønsker å utføre. Scholtz skriver at evalueringer som involverer brukerne kan kategoriseres på to måter: formativt og summativt. Rubin og Chisnell inkluderer to ytterligere kategorier, validerende og komparativt. Man oppnår mest med brukskvalitetstester når de inngår i en iterativ utviklingsprosess, altså når det testes i flere omganger i forskjellige stadier i prosessen. På denne måten vil problemer som kanskje ble oversett i tidlige tester, eller som oppstod som følge av endringer man gjorde underveis, fanges opp i senere tester. Brukertestene i de fire kategoriene ligner mye på hverandre, forskjellen ligger i hovedsak i *når* i et produkts livssyklus de blir gjennomført, hva de legger vekt på og dermed hva man ønsker å få ut av resultatene. (Rubin & Chisnell, 2008, s. 27–28; Scholtz, 2004, s. 751–753) I figur 8 kan du se Rubin og Chisnells illustrasjon av utviklingsforløpet til et produkt og når det aktuelt med hvilken type testing.



Figur 8: Rubin og Chisnells modell for brukertesting gjennom et produkts livssyklus (2008, s. 28)

Formative, eller utforskende, evalueringer blir gjennomført veldig tidlig i utviklingsfasen. Resultatet legger grunnlaget for utviklingen av produktet, og er med på å forme produktets design i stor grad, derav navnet. En formativ evaluering kan til og med gjennomføres før det

finnes et grensesnitt å teste, for eksempel ved hjelp av papirprototyper som består av «mock-ups» i papir av alle systemets vinduer, menyer og dialogbokser. Dette er en uformell måte å teste på. Rubin og Chisnell sier det kan ligne mer på et samarbeid mellom testbruker og testmoderator, der moderator for eksempel ved å stille spørsmål om brukerens tankegang påvirker brukeren mye mer enn ved de andre typene evalueringer. I motsetning til senere tester hvor det legges mer vekt på å måle *hvor effektivt* brukeren kan utføre oppgaver ved å samle kvantitative data, forsøker du ved formative evalueringer å forstå *hvorfor* brukeren ville gått frem for å utføre oppgavene som hun gjør ved å samle kvalitative data. I tillegg er du her gjerne ute etter å bekrefte antagelser du har om målgruppen din i større grad enn ved de andre typene evalueringer. (Rubin & Chisnell, 2008, s. 29–31; Scholtz, 2004, s. 752–752)

Summative evalueringer gjennomføres tidlig eller i midtveis i utviklingsfasen, og er mer formelle enn formative evalueringer i både gjennomføringen av undersøkelsene og dokumenteringen av resultatene. De utføres på et nært ferdig eller ferdig system, og sier noe om designets effektivitet og brukernes opplevelse av det. I stedet for bare å utforske hvordan en bruker teoretisk sett ville gått frem for å løse en oppgave, vil man undersøke hvor enkelt og effektivt en bruker *utfører* faktiske oppgaver, og identifisere spesifikke bruksproblemer. De bygger gjerne videre på resultater fra formative evalueringer, men her er det mye mindre innblanding fra testmoderator, og større fokus på bruk av representative testpersoner og realistiske oppgaver. (Rubin & Chisnell, 2008, s. 34–35; 2004, s. 753)

Validerende, eller verifiserende, evalueringer kommer mot slutten av utviklingsfasen til et produkt, og utføres for å vurdere hvordan produktet sammenligner seg med en forhåndsbestemt brukskvalitetsstandard eller etablerte «benchmarks», altså lignende produkter som ansees som gode. De gjennomføres også for å bekrefte at problemer oppdaget ved tidligere tester har blitt løst, og at nye ikke har blitt innført underveis. I motsetning til de to første typene, som foregår underveis i utviklingen, gjennomfører man valideringstester med brukere langt nærmere utgivelsen av produktet. Hensikten er å fastslå at produktet oppfyller en gitt standard før utgivelsen, eller å fastslå årsakene til hvorfor det ikke gjør det. Det er mer fokus på kvantitative ytelsesdata enn kvalitative brukeropplevelser. (Rubin & Chisnell, 2008, s. 35–36)

Komparative evalueringer, eller sammenligningstester, går ut på å måle to eller flere design opp mot hverandre. De er, i motsetning til de tre andre typene brukertester, ikke assosiert med et spesifikt stadium i livssyklusen til et produkt. Tidlig i utviklingen kan de brukes til å sammenligne forskjellige grensesnittdesign ved å gjennomføre flere formative tester for å se

hvilket design som har størst potensiale hos målgruppen. Mot midten av livssyklusen kan en sammenligningstest for eksempel brukes til å måle effektiviteten til et spesifikt element, for eksempel om bildeknapper eller tekstknapper foretrekkes av brukerne. Mot slutten av livssyklusen kan en sammenligningstest brukes til å se hvordan det ferdige produktet presterer i forhold til et konkurrerende produkt. Sammenligningstester brukes ofte til å fastslå hvilke design som er lettest å bruke eller lære, eller for å bedre forstå fordeler og ulemper ved forskjellige design. Resultatet er ifølge Rubin og Chisnell gjerne at man kombinerer det man finner ut at fungerer bra i de forskjellige alternativene, og utformer en hybridløsning. (2008, s. 37–38)

Eksempler på datainnsamling i brukertester er notater eller opptak fra observasjon, gjerne med «think/talk aloud»-notater, hvor brukeren selv forteller testmoderatoren hva hun tenker mens hun gjennomfører oppgavene hun er gitt, intervju med brukeren, fokusgrupper og spørreundersøkelser. Beta-testing, tilbakemeldinger fra brukere etter installasjon, og analyse av brukerlogger er eksempler på metoder for å samle data om brukskvalitetsaspekter ved produktet i en ekte brukskontekst. Man kombinerer gjerne flere av disse måtene å samle data på når man evaluerer et produkt brukerorientert.

3.2.2 Ekspertbaserte metoder

Ekspertbaserte metoder utføres, som navnet tilsier, av en ekspert i brukskvalitet. Ved bruk av ekspertbaserte metoder er brukerne av et gitt produkt utelatt, og grunnlaget for evalueringen ligger i evaluatorens tolkning og vurdering. Generelt sett sparer ekspertbaserte metoder mye tid, og det kan argumenteres for at man får både raskere og billigere resultater enn ved brukerbaserte metoder. Imidlertid vurderes ikke den faktiske bruken av systemet, man kan bare stadfeste om et system tar hensyn til aksepterte brukskvalitetsprinsipper (Dix et al., 2003, s. 320; Scholtz, 2004, s. 754–755)

Det finnes mange måter en ekspert kan vurdere grensesnitt på, men to av de vanligste formene for formelle ekspertbaserte evalueringsmetoder er *heuristisk evaluering* og *kognitiv gjennomgang*. Begge teknikkene innebærer at eksperter gjennom å sette seg inn i brukers sted forsøker å forutse hvilke brukervennlighetsproblemer brukeren kan komme til å støte på – såkalt *prediktiv* evaluering. Disse metodene kan brukes i alle stadier av utviklingsprosessen, fra designspesifikasjon til full implementering, noe som gjør dem til svært fleksible evalueringsmetoder. (Dix et al., 2003, s. 320; Scholtz, 2004, s. 754–755)

Heuristisk evaluering innebærer å sjekke et grensesnitt opp mot visse designprinsipper – også kalt *heuristikker*. Dix et al. beskriver en heuristikk som en retningslinje eller et generelt prinsipp som kan brukes for å ta korrekte designbeslutninger eller brukes for å kritisere en beslutning som allerede er tatt. Det er vanlig at flere eksperter gjør sine evalueringer på egenhånd, før man sammenligner resultatene og utformer en liste med problemer basert på det de har til felles. Erfaring tilsier at tre til fem eksperter er et tilstrekkelig antall, og at ved å bruke fem stykker blir rundt 75 % av de totale bruksproblemene som regel oppdaget. (2003, s. 324–325)

Det finnes mange etablerte heuristikker å gå ut i fra, og Jacob Nielsens ti heuristikker er mye brukt som utgangspunkt. Disse er gode å ha i bakhodet også når man driver med tradisjonell brukertesting, så jeg har valgt å oversette dem i sin helhet her. De lyder som følger:

1. Visning av systemstatus
    - Systemet skal alltid holde bruker informert om hva som skjer gjennom passende feedback innen rimelig tid.
2. Samsvar mellom virkelighet og system
    - Systemet skal kommunisere på et språk som gir forståelig og konstruktive tilbakemeldinger til brukeren.
3. Brukerkontroll og frihet
    - Når brukere gjør feil må de lett kunne avbryte og komme tilbake til utgangspunktet. Muligheter for å angre og omgjøre bør vektlegges.
4. Konsistens og standarder
    - Brukere skal ikke behøve å bekymre seg om hvorvidt ulike ord, situasjoner og handlinger betyr det samme i ulike settinger. Følg plattformkonvensjoner.
5. Forebygge feil
    - Eliminer så godt det lar seg gjøre brukers mulighet til å gjøre feil. Forklarende feilmeldinger og bekreftelsesdialogbokser er viktig.
6. Gjenkjennelse er bedre enn hukommelse
    - Minimer belastningen av brukerens hukommelse ved å gjøre objekter, handlinger og alternativer synlige. Brukeren skal ikke behøve å huske informasjon fra tidligere sesjoner. Instruksjoner for hvordan systemet fungerer burde til enhver tid være lett tilgjengelig.
7. Fleksibilitet og effektiv bruk
    - Systemet skal imøtekomme behovet til både nye og erfarne brukere. For erfarne brukere skal det være mulighet for å bruke snarveier og skreddersy ofte brukte handlinger, som er "usynlige" for nye brukere.
8. Estetisk og minimalistisk design

- Dialoger bør ikke inneholde irrelevant eller lite brukt informasjon. All overflødig informasjon vil konkurrere med det relevante, og dermed gjøre det mindre synlig.

9. Hjelp brukere å gjenkjenne, tolke og reparere feilsituasjoner
   - Feilmeldinger bør uttrykkes på et forståelig språk, med en presis forklaring på problemet og forslag til mulige løsninger.
10. Hjelp og dokumentasjon
    - Optimalt sett bør et system kunne brukes uten retningslinjer, men det kan være nødvendig å ha hjelp og dokumentasjon tilgjengelig. All slik informasjon bør være enkel å søke i, være fokusert på brukerens oppgaver, liste konkrete steg for å gjennomføre en gitt oppgave, og ikke være for omfattende.

Oversatt fra Nielsen (1995) og (1993, s. 20).

Ved en heuristisk evaluering vil hver enkelt ekspert vurdere systemet mot de gitte heuristikkene. Alle brudd hun finner på disse noteres, da de kan indikere potensielle brukbarhetsproblemer. Det er også vanlig at eksperten vurderer alvorlighetsgraden på hvert problem basert på fire faktorer: hvor ofte oppstår problemet, hvor lett er det for brukeren å komme rundt det, vil det være et engangstilfelle eller et vedvarende problem, og hvor alvorlig vil problemet oppfattes av brukeren?

Det anbefales ofte å gjennomføre en heuristisk evaluering i forkant av en brukertest, da en ekspertvurdering av grensesnittet kan gi en pekepinn på hvilke aspekter det vil være hensiktsmessig å teste på ekte brukere, og hvilke områder man ikke nødvendigvis behøver å legge like mye vekt på. (Dix et al., 2003, s. 325; Scholtz, 2004, s. 754)

Ved kognitiv gjennomgang forsøker ekspertene å se ting fra brukernes perspektiv. Ved å gå gjennom hvert steg av oppgavene som skal utføres i systemet simulerer de brukerens problemløsningsprosess, og på denne måten finne eventuelle feil i grensesnittet. Dette gir grunnlag for en liste over brukskvalitetsproblemer som ekspertene enten har støtt på eller sett for seg.

For å gjennomføre en kognitiv gjennomgang behøver man ifølge Dix et al. fire ting:

1. En spesifikasjon eller prototype av systemet. Den trenger ikke å være fullstendig, men den bør være ganske detaljert. Detaljer som ordlyd i en meny kan utgjøre en stor forskjell.
2. En beskrivelse av oppgaven brukeren skal utføre i systemet. Dette bør være en representativ oppgave som de fleste brukere vil ønske å gjennomføre.
3. En komplett, skriftlig liste over stegene som må gjennomføres for å fullføre oppgaven i systemet som evalueres.
4. En beskrivelse av hvem brukerne er, og hvilken erfaring og kunnskap ekspertene kan anta at målgruppen innehar.

Oversatt fra Dix et al., 2003, s. 321.

Utrustet med denne informasjonen går eksperten gjennom stegene beskrevet i punkt 3 over, mens hun vurderer systemets brukskvalitet ved å stille seg selv en rekke spørsmål koblet til informasjonen hun er gitt om målgruppen. Dix et al. beskriver fire typiske spørsmål:

1. Er virkningen av handlingen jeg utfører nå den samme som brukerens mål på dette punktet?
2. Vil brukerne se at handlingen er tilgjengelig/mulig å gjennomføre?
3. Når brukerne har funnet riktig handling, vil de vite at det er det de trenger?
4. Etter at handlingen er utført, vil brukerne forstå tilbakemeldingen de får?

Oversatt fra Dix et al., 2003, s. 321-322.

Det er anerkjent at denne metoden først og fremst tar for seg *learnability*-aspektet ved brukskvalitet, og det anbefales derfor å kombinere den med andre metoder for å få et mer helhetlig bilde av et produkts brukskvalitet. (Dix et al., 2003, s. 322; Scholtz, 2004, s. 754)

3.2.3 Modellbaserte metoder

En tredje tilnærming er bruken av modeller. Her benyttes forhåndsdefinerte strukturer, vurderingsprosedyrer og lignende for å evaluere et gitt design basert på modellens forutsigelser om designets ytelse ved bruk. I likhet med de to andre kategoriene, finnes det også modeller som kan brukes på forskjellige stadier i utviklingen av et produkt.

Den mest brukte modellen brukt i forbindelse med evaulering av brukskvalitet er GOMS. GOMS står for «Goals, Operators, Methods, and Selection rules», og er en modell som bryter ned brukeratferd til sekvenser av grunnleggende bestanddeler, for eksempel å flytte musepekeren til et gitt sted på skjermen, som gis et tidsestimat for ferdigstillelse basert på funn i undersøkelser av menneskelig prestasjon fra kognitiv psykologi. Her er en metode (methods) et sett med steg (operators) som vil fullbyrde et gitt mål (goal), og i gitt grensesnitt kan flere metoder oppnå det samme resultatet. Man velger deretter metode basert på gitte utvalgsregler (selection rules). GOMS-modellen er begrenset til evaluering av *efficiency*-aspektet av brukskvalitet i oppgaver som krever lite eller ingen beslutningstaking av brukeren. En utvidelse av modellen, Natural GOMS Language, kan også si noe om *learnability.* For begge teknikkene må en oppgaveanalyse (task analysis) gjennomføres for å konstruere og definere målene, operatørene, metodene og utvalgsreglene i prosjektet. GOMS brukes altså til å forutse brukerens prestasjoner med et gitt grensesnitt, og kan på denne måten brukes til å filtrere designalternativer. (Dillon, 2001; Dix et al., 2003, s. 324; Scholtz, 2004, s. 755)

3.2.4 Valg av metode

I «Usability evaluation» argumenterer Dillon med at siden det endelige målet ved enhver brukskvalitetsevaluering er å vurdere i hvilken grad *virkelige* brukere kan utnytte systemet, vil

alltid korrekt utførte brukertester gi det sanneste resultatet. (Dillon, 2001) På samme måte hevder Scholtz at den største fordelen ved brukerbaserte metoder er at man involverer faktiske brukere. Resultatene er basert på å observere hvilke aspekter av brukergrensesnittet som forårsaker problemer for representative brukere. Ulempen er at brukerevalueringer ofte er dyre og tidkrevende. Det er vanskelig å finne og sette opp avtaler med et passende antall representanter. Et passende sted og riktig teknisk utstyr er også nødvendig for å gjennomføre evalueringene og analysere resultatene. Realismen i testsituasjonen er også noe man må ta spesielt hensyn til; Har man valgt de riktige oppgavene? Har man fått tak i riktige brukere? Hvordan vil produktet fungere i en ekte brukskontekst? (Scholtz, 2004, s. 753)

Når det kommer til ressursbruk er heuristiske vurderinger billigere og mindre tidkrevende å utføre enn brukerbaserte metoder. Kognitive gjennomganger kan gjennomføres med en tekstbeskrivelse av brukergrensesnittet og kan derfor gjøres svært tidlig i programvareutviklingsprosessen. Slike inspeksjonsteknikker git imidlertid ikke mulige løsninger på problemene man finner. I tillegg kan det være en utfordring å oppsummere funnene fra ekspertbaserte metoder hvor man involverer flere eksperter fordi de kan rapportere problemer på forskjellige måter, og kan ha forskjellige oppfatninger av problemenes alvorlighetsgrad. (2004, s. 754)

Modellbaserte metoder er definitivt de billigste å gjennomføre isolert sett, men kostnadene øker betraktelig når man tar hensyn til tiden og arbeidet det tar å utføre de nødvendige oppgaveanalysene, og det faktum at man tester en veldig liten del av systemet med hver modell. (2004, s. 756)

I «Tracking the effectiveness of usability evaluation methods» sammenlignet John og Marks en rekke evalueringsmetoder, og konkluderte med at ingen metode kan ansees som objektivt best, og alle metodene har begrenset verdi. I deres undersøkelse fant de flere problemer i de heuristiske evalueringene enn ved brukertesten, men de fant de like mange problemer ved brukertesten av den originale versjonen av systemet og versjonen som var oppdatert i tråd med anbefalingene etter ekspertvurderingene. Det var altså få problemer av problemene brukerne reelt støtte på som ble fjernet som følge av de heuristiske evalueringene. (1997) Samtidig beskriver Scholtz en studie som sammenlignet heuristiske vurderinger med resultatene fra en brukertest. Denne konkluderte med at heuristiske evalueringer er bedre enn kognitive gjennomganger og retningslinjebaserte evalueringer til å forutse problemer som ekte brukere møter, men at ingen av disse metodene fant mer enn 50 prosent av problemene som ble oppdaget ved en brukertest. (Scholtz, 2004, s. 754–755)

> However, while individual methods have limitations and can be flawed in their implementation, it is certain that performing some evaluation is better than doing nothing.
> (Scholtz, 2004, s. 756)

Det kan altså diskuteres om det er hensiktsmessig å sammenligne metodene, da man ikke bare må se det i sammenheng med ressursbruken, men også hvor realistiske resultatene er. Er det for eksempel hensiktsmessig å avdekke så mange potensielle feil som mulig, dersom det er feil en bruker aldri realistisk vil støte på? Tabellen under er hentet og oversatt fra Dillon (2001) og viser en oppsummering av fordelene og ulempene ved hver av metodene.

Metode	Fordeler	Ulemper
Brukerbasert	Gir det mest realistiske vurderingen av brukskvaliteten Kan gi tydelig oversikt over viktige problemer	Tidkrevende Dyrt for stor utvalg av brukere Krever prototype
Ekspertbasert	Billig Rask	Ekspertens kvalifikasjoner og kunnskap påvirker utfallet Utfordrende å finne egnede eksperter Kan overvurdere antall reelle problemer
Modellbasert	Gir et grundig estimat av brukskvalitetsprinsippet som undersøkes Kan utføres på kun grensesnittspesifikasjon	Måler kun ett brukskvalitetsprinsipp Svært begrenset når det kommer til varierte oppgaver

Tabell 4: «Relative advantages and disadvantages of each usability evaluation method», oversatt og tilpasset fra Dillon (2001)

Valg av metoder er altså i stor grad et spørsmål om hvilken informasjon man er ute etter, og på hvilket stadium i utviklingssyklusen evalueringen skal foregå. Det vil for eksempel være vanskelig for en ekspert å undersøke en prototype for å forutsi brukertilfredshet, eller for en bruker å vurdere egen effektivitet pålitelig fra en grensesnittspesifikasjon. Generelt sett er det konsensus om at det er fordeler og ulemper ved alle metodene, og en kombinasjon av evalueringsmetodene er mest produktivt. Den ultimate testen er når ekte brukere opptrer under normale arbeidsforhold, og alle evalueringsmetodene gjør forsøk på å forutsi problemene som vil oppstå i denne typen reell bruk. Jeg har valgt brukertesting fordi det er metoden som i min situasjon med mine ressurser, gir den mest realistiske vurderingen av brukskvaliteten.

3.3 Brukertesting

Undersøkelsen min ble altså gjennomført ved praktisk brukertesting, og jeg har i hovedsak brukt boka *Praktisk brukertesting* av Eli Toftøy-Andersen og Jon Gunnar Wold (2011) som veileder i utforming og gjennomføring av testen.

Det finnes mange måter å gjennomføre brukertester på, og hvilken man velger avhenger av testens hensikt og ressursene man har tilgjengelig, som budsjett, tid og tilgang på eksperter. En «vanlig» brukertest er en såkalt lavbudsjettesting som ikke gir statistisk gyldige resultater, men som likevel er en anerkjent metode fordi man får mye innsikt og lærdom som kan brukes til å rette opp problemer i brukskvaliteten til produktet man tester. I en vanlig brukertest sitter som regel testleder ved siden av brukeren under hele testen, og stiller spørsmål underveis for å klargjøre observasjoner og hjelpe brukeren til å tenke høyt. Ulempen ved å gjennomføre en test på denne måten er at man som testleder risikerer å påvirke brukerens adferd med kommentarer og kroppsspråk. Fordelene er at brukeren ofte vil stille testleder spørsmål underveis, og man vil på denne måten få innsikt i brukerens tankegang, selv om testleder skal forholde seg nøytral og ikke hjelpe brukeren. I tillegg vil man kunne oppklare tvetydige observasjoner underveis. I likhet med Toftøy-Andersen og Wold mener jeg at fordelene veier opp for ulempene, og har derfor valgt å gjennomføre den vanlige typen brukertest i mine undersøkelser. (2011, s. 128–129)

En typisk brukertest er todelt. I den første delen er det vanlig at brukeren får en rekke praktiske oppgaver som skal utføres. Hensikten her er å studere hvor *intuitiv* løsningen virker. I andre del er det vanlig å stille brukeren oppfølgingsspørsmål relatert til svakheter og problemer som ble avdekket under observasjonen av første del. Kort sagt avdekker de praktiske oppgavene svakheter og problemer ved applikasjonen, og oppfølgingsspørsmålene gir oss kunnskap om hva som oppleves vanskelig eller uklart, og hvorfor. (2011, Kapittel 1)

I brukertesting er det i hovedsak tre typer oppgaver: konkrete oppgaver, delvis åpne oppgaver og helt åpne oppgaver. Konkrete oppgaver består av et startpunkt i systemet for brukeren, et «riktig svar» som sier hva som kreves for at brukeren skal ha fullført oppgaven og ting testleder skal se etter mens brukeren løser oppgaven. Konkrete oppgaver er gode når man har konkrete ting man vil teste, både fordi det er enklere å sammenligne resultatene når man har et «riktig svar» og fordi man guider brukeren inn på tingene man vil teste. Ulempen ligger i at man kan styre brukeren mot å gjøre ting han/hun normalt ikke ville gjort. Delvis åpne oppgaver er åpne i den forstand at man lar brukeren selv bestemme hvordan han/hun vil løse den, men man stiller oppfølgingsspørsmål basert på brukerens handlinger underveis. Delvis

åpne oppgaver er gode dersom det er flere innganger til løsningen på oppgaven, og man vil styre brukeren i mindre grad enn ved konkrete oppgaver. Ulempen er at man må være mye mer nøyaktig når man noterer for at det skal være mulig å sammenligne resultatene etterpå. Helt åpne oppgaver er oppgaver med svært åpne instruksjoner, og kan gi veldig forskjellige resultater avhengig av hva brukeren gjør. Disse er hensiktsmessige når man vil vite mer om brukerens vaner og tankesett, og man ønsker å la dem gjøre det som er naturlig for dem. Dette er mest aktuelt for brukertesting av nettsteder, og der kan åpne oppgaver kan være risikable, da man ikke har en garanti for at brukeren ender opp i ditt system i det hele tatt. Det viktigste er å gjøre alle oppgavene så realistiske som mulig, fordi det gjør at brukeren reflekterer på en realistisk måte. (2011, s. 49– 51)

Til tross for at verktøyet allerede støttet opprettelse og behandling av relasjoner, valgte jeg i første testrunde å utelate relasjonsannotering. Tidligere forskning indikerer som sagt at å dele opp annoteringsprosessen i mindre oppgaver er hensiktsmessig ved crowdsourcing (Bontcheva et al., under publisering; Finin et al., 2010; Good et al., 2015; Zhai et al., 2013), og entitetsannotering er på mange måter en enklere og mindre omfattende prosess enn relasjonsannotering. Vi anså derfor entitetsannotering som en mer aktuell aktivitet å crowdsource. Oppgavene i den første testen gikk derfor kun ut på å behandle entiteter og ekvivalensrelasjoner.

I den andre, utvidede brukertesten anså jeg det som det hensiktsmessig at oppgavene lignet på de i den første testen for at jeg skulle kunne sammenligne resultatene og se om problemer funnet i den første testen var løst på en hensiktsmessig måte. I pilottesten for den utvidede testen ble jeg fort klar over at jeg med det gamle oppsettet av oppgavene risikerte at brukeren annoterte entiteter uten relasjoner seg imellom, og at de dermed måtte annotere unødvendig mange entiteter for å fullføre testen. Jeg endte derfor opp med å konstruere fire nye oppgaver som tilrettela for annotering av relasjoner, og gav deltakerne beskjed om at de kunne ta utgangspunkt i de allerede annoterte entitene eller lage helt nye. Oppgavene beskrives nærmere i deres respektive delkapitler i del 4, og kan leses i sin helhet i vedlegg 1 og 2.

Deltakerne fikk realistiske oppgaver å utføre ved å samhandle direkte med verktøyet. Verktøyet forklarer ikke hvordan grensesnittet skal fungere, noe som gjør det mulig å identifisere hvilke deler av grensesnittet som er selvforklarende og hvilke deler som er forvirrende. Utfordringen i utformingen av oppgavene lå i å gi brukerne nok frihet til å kunne gjøre egne feil og dermed la meg se hvor forvirringen i annoteringsprosessen oppstod, men samtidig styre dem nok til at de fullførte oppgavene og endte opp med brukbare data.

Jeg har valgt å dele inn resultatene av oppgavene i kategoriene «korrekt», «feil (semantisk meningsbærende)» og «feil (semantisk meningsløs)». En korrekt annotering samsvarer med det jeg som testansvarlig har etablert som fasit eller gullstandard. En semantisk meningsbærende feil refererer til en annotert navngitt entitet eller reell relasjon som ikke samsvarer med grunnsannheten. Et eksempel vil være om en bruker valgte feil relasjon mellom to relaterte entiteter. En semantisk meningsløs feil vil ikke referere til en navngitt entitet, for eksempel «filmen».

3.3.1 Datainnsamling

> Data er bedre beslutningsgrunnlag enn erfaringer. Erfaringer er bedre beslutningsgrunnlag enn meninger. Brukertesting kan produsere noe som nærmer seg data.
>
> (Arlov, 1999, s. 270)

Som nevnt tidligere mener Arlov at der man med intervjuer og diskusjoner ender opp med meninger, får man med brukertesting noe som nærmer seg data. Det man samler i en brukertest er i hovedsak notater fra observasjonen og resultatene fra selve testen, i tillegg er det vanlig å gjennomføre et intervju i forbindelse med testen. Det er naturlig å samle inn informasjon med et lite spørreskjema. I tester med store budsjetter er det også mulig å investere i programvare som sporer brukerens øyebevegelser og lignende.

Ved å be brukeren om å «tenke høyt» (think aloud) mens hun løser oppgavene får man som testleder mye verdifull informasjon som sier noe om hvordan brukeren forstår, eller ikke forstår, systemet. Ved at brukerne verbaliserer tankene sine underveis i testforløpet kan man få innsikt i hvor eventuell forvirring oppstår og få et bedre grep om hva de forventer av systemet. Denne metoden har sitt utspring i psykologisk forskningsmetode, og brukes mye i evalueringer av menneske-maskin-interaksjon. Det innebærer at man som testleder må påvirke brukeren til en viss grad, for eksempel ved å stille spørsmål som «hva tenker du nå?» om brukeren glemmer å snakke, men samtidig passe på å ikke lede brukeren og svare på en avvergende måte dersom brukeren spør om hjelp, for eksempel svare «hva tror du at du skal gjøre?» dersom brukeren ber om bekreftelse på en handling i stedet for å utføre den. Resultatene fra å bruke «tenke høyt»-metoden i en brukertest kan ikke brukes som noen form for ytelsesmåling, men innsikten man får kan støtte opp om de kvalitative dataene fra observasjon og intervju. (Toftøy-Andersen & Wold, 2011, Kapittel 7; Tullis & Albert, 2013, s. 102–103)

Intervjuet gjennomføres etter at testen er gjennomført, og i tillegg til de forhåndsbestemte spørsmålene er det vanlig at man tar opp ting som kom underveis i brukertesten. Man kan for

eksempel spørre brukeren om hva hun synes ville vært en god løsning på et problem hun støtte på. De forhåndsdefinerte spørsmålene blir stilt for å forsikre seg at man får med informasjon om de punktene man er ute etter. (Arlov, 1999, s. 130–135)

I tillegg kan man bruke spørreskjemaer for å samle målbare tilbakemeldinger om brukskvaliteten til systemet. Et skjema som ofte brukes er System Usability Scale (SUS). Her presenteres brukeren for ti påstander hvor hun må krysse av hvor enig hun er med påstanden på en skala fra 1 til 5. For eksempel «jeg synes systemet var unødvendig komplisert» eller «jeg tror folk flest vil lære å bruke dette systemet svært raskt». Skjemaet fylles ut etter at deltakeren har hatt anledning til å bruke systemet, men før eventuell diskusjon eller intervju. Hvert svar er vektet, og samlet regnes scoren ut på en slik måte at man sitter igjen med en deltakers tilfredsstillelse på en skala fra 1 til 100, hvor en høy SUS-score indikerer et brukervennlig program. En verdi over 68 anses for å være over gjennomsnittlig bra, og en verdi under 68 anses for å være under gjennomsnittlig bra. (Brooke, 1996; Toftøy-Andersen & Wold, 2011, s. 147–148) For å trekke ut ytterligere informasjon fra den samlede SUS-scoren er det ifølge Lewis & Sauro (2009) mulig å bryte ned poengsummen til å si noe om brukskvaliteten (usability) og hvor lett det er å lære (learnability) hver for seg. Spørsmål 4 og 10 omhandler hvor lett det er å lære, mens de resterende åtte spørsmålene tar for seg brukskvaliteten mer generelt. Vedlegg 4 inneholder det oversatte SUS-skjemaet jeg brukte.

I mine undersøkelser ble data samlet inn på flere måter. Før selve testen ble brukeren bedt om å fylle ut et enkelt demografisk spørreskjema. Dette skjemaet kan sees i vedlegg 1 for den første testen og vedlegg 2 for den utvidede testen. Jeg satt som sagt sammen med brukeren gjennom hele testseansen og noterte ned mine observasjoner, og deltakerne ble bedt om å «tenke høyt» og forklare hva de gjorde underveis. For å kunne måle brukernes forventninger opp mot deres opplevelser i henhold til Albert og Dixon (sitert i Tullis & Albert, 2013, s. 132–133) ble deltakeren før hver oppgave bedt om å rangere deres forventninger om oppgavens vanskelighetsgrad på en skala fra 1 til 5, og etter at oppgaven var fullført eller når deltakeren gav opp, ble de bedt om å rangere hvor vanskelig de opplevde oppgaven på samme skala. Dette skjemaet kan også sees i vedlegg 1 for den første testen og vedlegg 2 for den utvidede testen. I tillegg ba jeg deltakerne om å fylle ut et SUS-skjema når de var ferdige med alle oppgavene. Etter testen gjennomførte jeg et lite semi-strukturert intervju basert på hvordan testen ble gjennomført, og noen forhåndsdefinerte spørsmål. Disse spørsmålene kan sees i vedlegg 3.

Oppsummert samlet jeg inn følgende data:

- Demografiske data
- Notater fra observasjon og intervju
- Forventninger til oppgavenes vanskelighetsgrad
- Tilbakemeldinger på oppgavenes vanskelighetsgrad
- System Usability Scale
- Resultatet av annoteringene

3.3.2 Utvalg av testbrukere

I følge Jakob Nielsen vil fem testpersoner i mange tilfeller være nok for å oppdage 80 % av områdene som skaper problemer for brukerne, ved ytterligere testpersoner vil gevinsten bli liten i forhold til økt forbruk av tid og ressurser, og det vil derfor være mer hensiktsmessig å teste flere ganger i stedet for flere personer. (Nielsen, 2000)

Det største argumentet mot denne modellen er at man ikke kan få et representativt utvalg med så få testpersoner. Det blir derfor en avveining man må gjøre med tanke på hva det er man skal teste og hvilke resultater man ønsker å oppnå. De fleste som jobber med brukertesting i dag er enige om at fem testpersoner er et absolutt minimum, og at man gjerne tester med flere hvis man har mulighet til det. I «Beyond the five-user assumption: Benefits of increased sample sized in usability testing» gjennomførte Laura Faulkner (2003) 60 brukskvalitetstester og analyserte resultatene. I hennes undersøkelse varierte prosentandelen av problemer funnet av et gitt sett på fem brukere fra 55 % til nesten 100 %. Det var altså stor variasjon mellom forsøkene med få testpersoner, og man risikerte å kun finne litt over halvparten av problemene. Med ti brukere økte den minste prosentandelen til 95 %, og med femten brukere var den 97 %. Hun konkluderer med at man bare ved å øke fra fem brukere til ti brukere får en dramatisk forbedring, men at det beste er å teste så mange brukere som mulig, og mange ulike brukertyper som mulig.

Når man skal rekruttere testpersoner er det viktig at man finner representative brukere, og man må ta utgangspunkt i en klart definert målgruppe for systemet. I mitt tilfelle er det snakk om annotering i et allment emne og et mål om å legge til rette for crowdsourcing, noe som betyr at «alle» er i målgruppen.

> We believe that the success of the Semantic Web depends on reaching a critical mass of users creating and consuming semantic content. This would require tools that hide the complexity of semantic technologies and match the compelling simplicity of Web 2.0 applications: light-weight, easy-to-use, and easy-to-understand. Very little research has been done on supporting non-expert end-users in the creation of semantically-enriched content.
>
> (Hinze et al., 2012)

Siden målet er å gjøre verktøyet brukbart i crowdsourcing, og derfor for ikke-eksperter, var fokus på å velge deltakere med variert bakgrunn og ulik grad av teknisk kunnskap. Hver deltaker ble bedt om å rangere sine kunnskaper på en skala fra 1 til 5, der 1 tilsvarte «ingen kunnskap» og 5 var «svært kunnskapsrik». Jeg brukte ferdigheter og kjennskap til tekstbehandlingsprogrammer som et mål på datakyndighet, kjennskap til tagging som mål på erfaring med annotering, og kjennskap til brukervennlighet og semantisk web som teknisk kompetanse. Basert på deres egne vurderinger kategoriserte jeg dem som henholdsvis eksperter og ikke-eksperter. Ordet «ekspert» brukes i denne konteksten veldig løst, og vil i denne konteksten altså henvise til en person med kunnskap om eller erfaring med annoteringsprosessen og semantisk web, og ikke korpusets domene.

I den første testen valgte jeg å teste med syv brukere. Som et ledd i en iterativ prosess var det logisk å velge et lignende antall testpersoner i andre runde, så på grunn av Laura Faulkners funn beskrevet over, og fordi det var mindre tidspress enn ved første test, valgte jeg å teste med ti brukere i den andre undersøkelsen.

3.3.3 Testforløp

I begge testene ble deltakere testet én om gangen. Ved ankomst presenterte jeg meg som testleder, og forklarte litt om hva som skulle skje. De ble fortalt at det er *systemet* som testes, og ikke deltakernes ferdigheter på noen måte. Dersom deltakeren skulle føle ubehag eller skulle ønske å avbryte testen, ble de fortalt at de kunne avbryte når som helst uten å måtte oppgi noen grunn. Deltakerne ble oppfordret til å tenke høyt under testingen. Det ble presisert at jeg som testleder ikke kunne hjelpe deltakerne underveis, men at jeg kunne komme til å stille spørsmål som «Hva tenker du nå?» og «Hva ser du etter?». Deretter fylte deltakerne ut samtykkeerklæring og et skjema med demografisk informasjon om deltakeren, disse kan leses i vedlegg 1 og 2. Jeg passet på igjen å presisere at dersom deltakeren skulle gjøre noe feil, så er det ikke deltakerens skyld, men verktøyets, og at det hjelper oss ved å indikere hva som kan forbedres. Jeg repeterte også at det var viktig at deltakeren tenkte høyt.

Før testdeltakerne fikk interagere med verktøyet, forklarte jeg raskt hvordan grensesnittet fungerte, og de ble bedt om å kikke gjennom retningslinjene. Jeg forklarte oppgavenes mål fra brukerens perspektiv, og gav instruksjoner om hvordan de skulle rapporteres som ferdig. Før oppgaven rangerte deltakeren deres forventninger om oppgavens vanskelighetsgrad, og når oppgaven var fullført (eller når deltakeren gav opp), ble de bedt om å rangere hvor vanskelig de opplevde oppgaven på samme skala. Jeg satt sammen med hver bruker gjennom hele testseansen og noterte ned mine observasjoner, for eksempel vanskeligheter brukeren støtte

på, kommentarer de kom med, og om de fullførte oppgaven. Jeg brukte resultatene fra pilottesten som utgangspunkt for rimelig tidsbruk per oppgave, men det var det ingen tidsbegrensning under testen. Nøyaktig tidtaking kan brukes som mål på effektivitet, men når man ber deltakerne tenke høyt under testen vil dette resultatet bli urealistisk. (Tullis & Albert, 2013, s. 81) Det blir altså en vurderingssak om man vil bruke tid som et mål, eller om man vil at deltakerne skal tenke høyt, og jeg vurderte det som mer verdifullt for mine undersøkelser at deltakerne kommuniserte hva de tenkte om og forventet av verktøyet underveis.

Etter siste oppgave fylte deltakerne først ut et SUS-skjema (vedlegg 4), et skjema med deres opplevde vanskelighetsgrad og interesse for å delta på lignende aktiviteter senere (vedlegg 1 for den første testen og vedlegg 2 for den andre), og det ble deretter utført et uformelt avsluttende intervju. De ble spurt om hva som var lett og vanskelig, hva de tenkte om de oppgavene de hadde vanskeligheter med å gjennomføre, og om de hadde noen tilbakemeldinger generelt sett.

I begge testene ble en SIFT-post som omhandlet en episode av radioprogrammet Filmfront brukt. Posten inneholdt en rekke potensielle entiteter og relasjoner det ikke var nødvendig å inneha noen spesiell kunnskap for å gjenkjenne. Med unntak av en feil annotert entitet i den første testen var posten ren tekst og fri for annoteringer. Nevnte feil var annotert av meg på forhånd for å muliggjøre oppgaven som ba brukeren rette en feilaktig annotering. Posten som ble brukt kan leses i sin helhet i vedlegg 5.

Siden verktøyet er nettbasert kunne deltakerne selv velge hvor testen skulle gjennomføres, og såfremt de hadde nettleseren FireFox installert på egen maskin var det ingenting i veien for at de kunne bruke sin egen datamaskin under testen. De fleste valgte likevel å bruke min bærbare datamaskin, og de fleste testene ble gjennomført i lokalene til biblioteket på Høyskolen Kristiania eller i fellesarealene på Høyskolen i Oslo og Akershus.

3.3.4 Beskrivelse av alvorlighetsgrader

For å lettere kunne prioritere hvilke problemet som må løses, er det vanlig å kategorisere feilene man finner under testingen etter alvorlighetsgrad. Nielsen sier at alvorlighetsgraden til et brukskvalitetsproblem er en kombinasjon av tre faktorer: frekvensen på problemet oppstår, innvirkningen det har på brukeren, og hvor vedvarende problemet er.

Det er mange skalaer og tabeller man kan bruke til dette, for eksempel presenterer Jakob Nielsen en skala fra 0 til 4:

0 = Dette er ikke et brukskvalitetsproblem i det hele tatt

1 = Dette er kun et kosmetisk problem, og må ikke løses om det ikke er tid til overs

2 = Mindre brukskvalitetsproblem, lav prioritet

3 = Stort brukskvalitetsproblem, høy prioritet

4 = Brukskvalitetskatastrofe: presserende å rette opp før produktet kan utgis

Hentet og oversatt fra Nielsen (1995)

En annen måte å kategorisere på, som er den jeg har valgt å bruke, er en tabell fra Society for Technical Communications «usability toolkit» som er oversatt til norsk i *Praktisk brukertesting* (Toftøy-Andersen & Wold, 2011, s. 152–153). Den inneholder prioriteringsnivåer fra 1 til 4, hvor 1 er «Kritisk» og 4 er «Lav», og hvert nivå inneholder eksempler. En forkortet versjonen av denne tabellen kan ses i tabell 5 under.

Nivå	Beskrivelse
1 – Kritisk	En kritisk situasjon som forårsaker at systemet krasjer, eller at bruker mister eller får ødelagt sine data. Det finnes ingen vei utenom problemet. En nødvendig funksjon for at bruker skal fullføre sin oppgave mangler eller virker ikke. Eksempel: Data går tapt fordi brukerne «burde vite» hvordan de lagrer eller må utføre kompliserte kommandoer for å lagre
2 – Høy	En alvorlig situasjon som hindrer utførelsen, eller den fortsatte bruken av en eller flere funksjoner i systemet som ikke kan forbigås på en enkel måte. Systemet hindrer ikke brukeren i å begå grove feil. Problemet er påvirker mange brukere hyppig og gjentagende. Et stort avvik fra standard. Eksempel: Manglende tilbakemelding til bruker for viktige operasjoner
3 – Medium	Et ikke-kritisk begrenset problem (ikke tap av data eller systemfeil). Det hindrer ikke fullføring av oppgaven fordi det kan forbigås eller unngås. Problemet forårsaker moderat irritasjon eller forvirring hos brukeren. Eksempel: Et menyvalg eller knapp gjør ikke det som normalt forventes
4 – Lav	Ikke-kritiske problemer eller generelle spørsmål om produktet. Det er mindre avvik fra standarder, små ulikheter eller inkonsistens. Mindre visuelle problemer som skjemafelter og titler som ikke ligger på rett linje. Eksempler: Skrivefeil (kan kategoriseres som 3 -Medium eller 2 - Høy dersom det er skrivefeil på for eksempel forsiden, menyer eller betalingsløsning) eller mindre kosmetiske problemer

Tabell 5: Alvorlighetsgrader som beskrevet i Toftøy-Andersen & Wold, 2011, s. 152–153

4. Undersøkelsene

Det tidlige arbeidet mitt med masteroppgaven sammenfalt med temaet for et call for papers fra MTSR 2015, og vi så en mulighet til å få publisert noe av resultatene mine. Jeg gjennomførte den begrensede brukertesten av verktøyet allerede høsten 2015, og artikkelen vi skrev ble akseptert til konferansen. Etter at jeg presenterte resultatene i Manchester ble artikkelen publisert i årets antologi. Mye av det jeg skriver om den første brukertesten kan derfor også leses i «Usability Testing of an Annotation Tool in a Cultural Heritage Context» av undertegnede og Michael Preminger.

I tillegg til brukskvalitetsproblemene vi avdekket i 2015, konkluderte vi med to ting: brukerne med rimelige datakunnskaper presterte godt, og vi fant ingen klar sammenheng mellom hvor vanskelig brukeren opplevde oppgaven og deres interesse i å delta i flere lignende aktiviteter. (Hoff & Preminger, 2015) Dette var veldig oppmuntrende, spesielt med tanke på crowdsourcing. Vi oppdaterte derfor verktøyet etter funnene fra undersøkelsene i 2015, og jeg gjennomførte en utvidet brukertest høsten 2016.

I dette kapitlet presenterer jeg i de to brukertestene jeg gjennomførte. Først presenterer jeg gjennomføringen av og resultatet fra den begrensede brukertesten. Deretter presenterer jeg gjennomføringen av og resultatet fra den utvidede testen.

4.1 Første test: Annotering av entiteter

Vi vurderte annotering av entiteter som mest aktuelt å crowdsource. Som nevnt tidligere var dette fordi entitetsannotering på mange måter er en enklere og mindre omfattende prosess enn relasjonsannotering, og tidligere forskning indikerer at å atomisere annoteringsprosessen er hensiktsmessig ved crowdsourcing (Bontcheva et al., under publisering; Finin et al., 2010; Good et al., 2015; Zhai et al., 2013). Oppgavene i den første testen gikk derfor kun ut på å behandle entiteter og ekvivalensrelasjoner. Dette ble besluttet tidlig for å begrense undersøkelsens omfang for konferansebidraget, og for gjøre resultatet mer håndterbart med tanke på variabler og årsakssammenhenger.

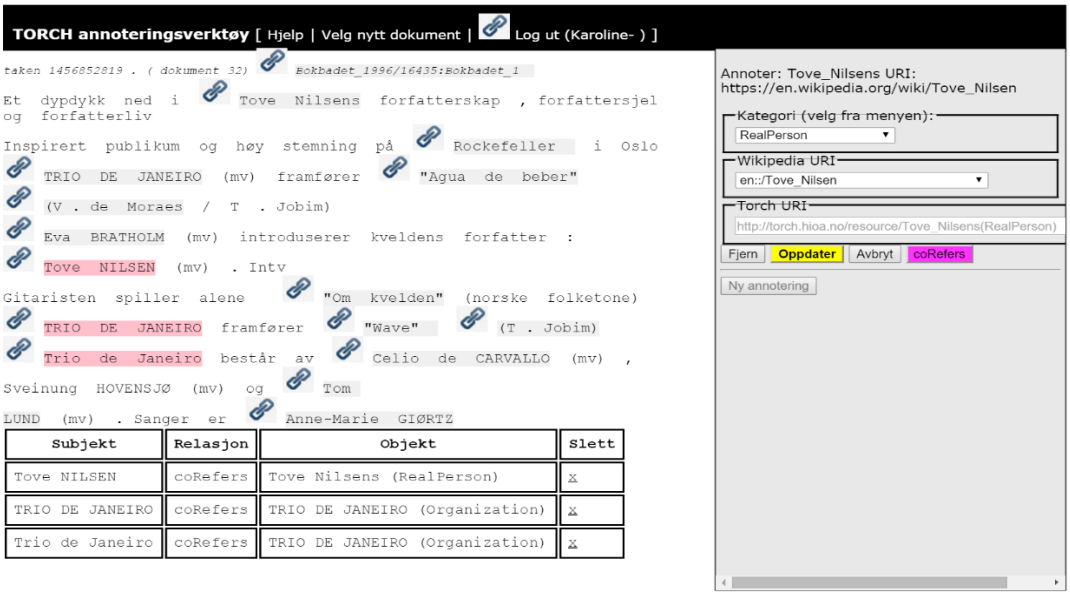
Jeg vurderte først om det var aktuelt å be en bruker om å annotere hele eller deler av en post, men besluttet at det ville bli for tidkrevende.

Vi identifiserte de viktigste oppgavene i forbindelse med annotering av navneentiteter: annotere entiteter, endre annoterte entiteter, slette annoterte entiteter og opprette ekvivalensrelasjoner mellom entiteter. Vedlegg 1 viser det fullstendige oppgaveheftet

brukerne fikk utdelt i den første undersøkelsen. En kort oppsummering av oppgavene er presentert under:

- Oppgave 1: Annoter to valgfrie personnavn
- Oppgave 2: Annoter to valgfrie verk
- Oppgave 3: Annoter ett valgfritt sted
- Oppgave 4: Lag en ekvivalensrelasjon mellom to entiteter
- Oppgave 5: Endre opplysningene i en annotert entitet
- Oppgave 6: Slett en valgfri entitet

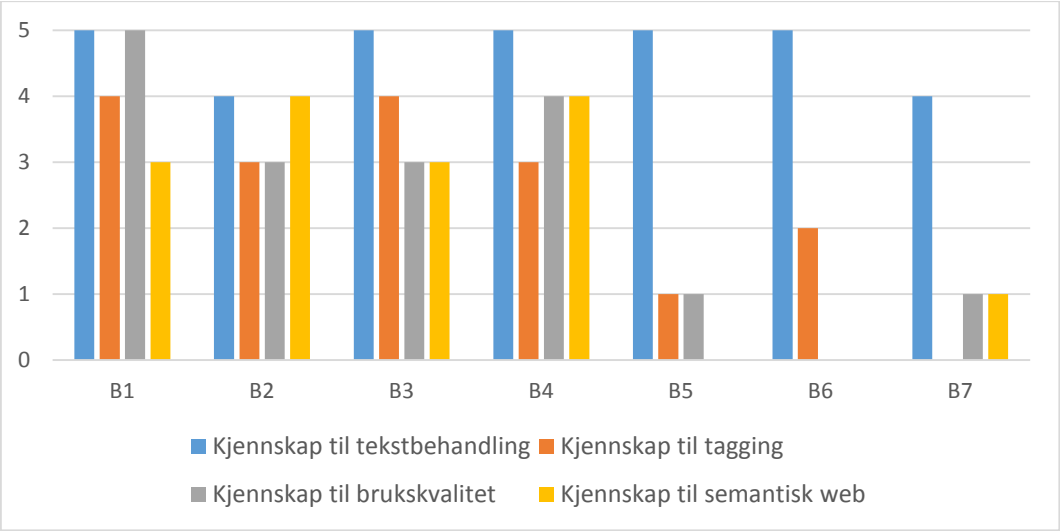
Jeg gjennomførte en vellykket pilottest, og brukte resultatene fra denne testen som et mål på hva som var rimelig tid på hver av oppgavene. Dette varierte mellom 1-4 minutter per oppgave. Testen ble gjennomført som beskrevet i 3.3.3 Testforløp. Versjonen av grensesnittet som ble brukt i den første brukertesten er vist under i figur 9.



Figur 9: Skjermbilde av verktøyet slik det fremstod i 2015

4.1.1 Deltakere

Jeg rekrutterte deltakere ved å be medstudenter på Høgskolen i Oslo og Akershus og tilfeldige studenter ved Høyskolen Kristiania om å delta. Som sagt satte jeg fokus på å få tak i deltakere med variert bakgrunn og ulik grad av teknisk kunnskap. Jeg ba hver deltaker om å rangere sine kunnskaper som beskrevet i 3.3.2. Figur 10 på neste side viser de syv deltakernes egenvurderinger.

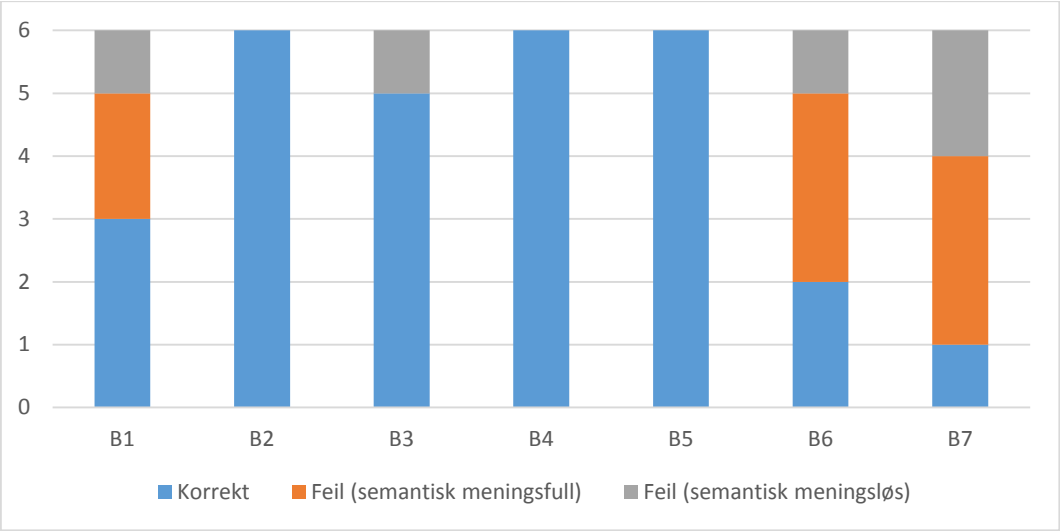


Figur 10: Deltakernes vurdering av egen kunnskap i den første testen

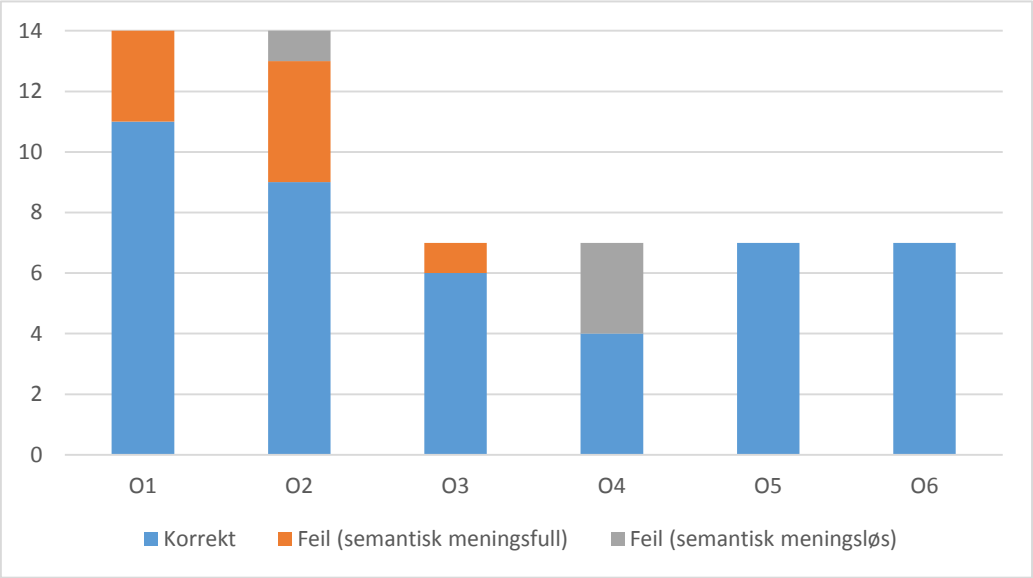
Alle deltakerne regnet seg som datakyndige. Fire av deltakerne rapporterte kjennskap til tagging og semantisk web, mens tre deltakere hadde liten eller ingen kunnskap til det. Jeg tok utgangspunkt i kategoriseringen til Hinze et al. (2012), og basert på deltakernes egne vurderinger kategoriserte jeg B1-B4 som tekniske eksperter og B5-B7 som ikke-eksperter.

4.1.2 Annoteringer

Som nevnt i del 3.3 ble resultatet av hver annoteringsoppgave registrert som korrekt, semantisk meningsbærende feil, eller semantisk meningsløs feil, basert på om annoteringene var i overenstemmelse med det jeg som testleder hadde satt som gullstandard. Alle deltakerne lagde minst én korrekt annotering, og to av deltakerne utførte alt korrekt. Ingen deltakere avbrøt eller ga opp på en oppgave. Slik oppgavene var lagt opp, resulterte hver deltaker i seks annoterte opplysninger: fem navneentiteter og en ekvivalensrelasjon. I oppgave 1 og 2 var resultatet to entitetsannoteringer per deltaker, altså fjorten annoteringer. Oppgave 3 resulterte i syv navneentiteter, og oppgave 4 syv ekvivalensrelasjoner. I oppgave 5 og 6 var suksessraten binær: enten fullførte deltakeren endringen/slettingen eller ikke. Figur 11 viser suksessraten på resultatene fra oppgave 1-4 fordelt på deltaker, mens figur 12 viser deltakernes suksessrate fordelt på alle oppgavene.



Figur 11: Resultatet av annoteringene fra oppgavene 1-4 i den første testen fordelt på deltaker



Figur 12: Resultatet av annoteringene i den første testen fordelt på oppgave.

Det var to typer feil som fremdeles hadde semantisk mening: i enkelte tilfeller fikk entiteten tildelt feil Wikipedia-URI, for eksempel fikk en forekomst av *filmen* Øyenstikkeren tildelt Wikipedia-URI-en for *insektet* Øyenstikker. Den andre typen semantisk meningsbærende feil var noe Bård Eskeland fra Språkrådet kunne fortelle meg at er «apposisjoner med det første substantivet som kjerne og det andre som etterstilt beskriver» (personlig kommunikasjon på e-post, 31.mai, 2017). Det de hadde til felles var at de inkluderte substantivet som henviste til entiteten som skulle annoteres, for eksempel «filmen Øyenstikkeren» eller «programlederen Mikal Olsen Lerøen».

Det var to typer semantiske meningsløse feil. I et par tilfeller ble det annotert ekvivalensrelasjoner mellom et substantiv og egennavnet det refererte til, for eksempel mellom «filmen» og «Øyenstikker». Den andre typen var et tilfelle av en bruker som

annoterte «dogmefilm» som en navneentitet, men denne brukeren ga uttrykk for at han trodde dette var et verk. Det er tydelig forklart i retningslinjene hvordan man håndterer slike forekomster, og ingen av dem skal annoteres. Til tross for at deltakerne hadde retningslinjene tilgjengelig under testingen, *og* at de måtte kikke gjennom dem på forhånd, var det bare én deltaker som faktisk konsulterte dem da hun var i tvil. Dette indikerte at vår antagelse om at brukere ikke aktivt bruker eller leser retningslinjene grundig, stemmer.

Ingen deltakere ga opp uten å fullføre en oppgave, men noen uttrykte at de var usikre på om de var helt ferdige, og om annoteringen var lagret. Flere rapporterte at de fikk for lite tilbakemeldinger fra verktøyet, og at de så etter tegn på at de hadde gjort det riktig. Kommentarer som «jeg *tror* jeg gjorde det» eller «jeg er vel ferdig» var gjengangere. Noen deltakere trodde at skjermbildet hadde fryst når verktøyet brukte litt tid på oppslaget mot Wikipedia.

Basert på resultatene til de forskjellige brukerne vil jeg, på samme måte som Hinze et al. (2012), kategorisere B2, B3, B4 og B5 som akseptable annotører, B1 og B6 som annotører med forbedringspotensiale og B7 som uakseptabel annotør.

I tillegg kunne jeg observere at, i likhet med forsøkene til Hinze et al., byttet flere deltakere på rollene som informasjonsleverandør (information provider) og informasjonsforbruker (information consumer) i løpet av testen. Fire deltakere ønsket å åpne en Wikipedia-URI for å lese mer om noe de hadde markert, eller sørge for at de hadde valgt riktig URI. To deltakere viste interesse for ontologien utover det som ble beskrevet i retningslinjene, for eksempel stilte en deltaker spørsmålet «hva om jeg fant et pseudonym?».

4.1.3 Deltakernes egenvurderinger av opplevd vanskelighetsgrad og interesse

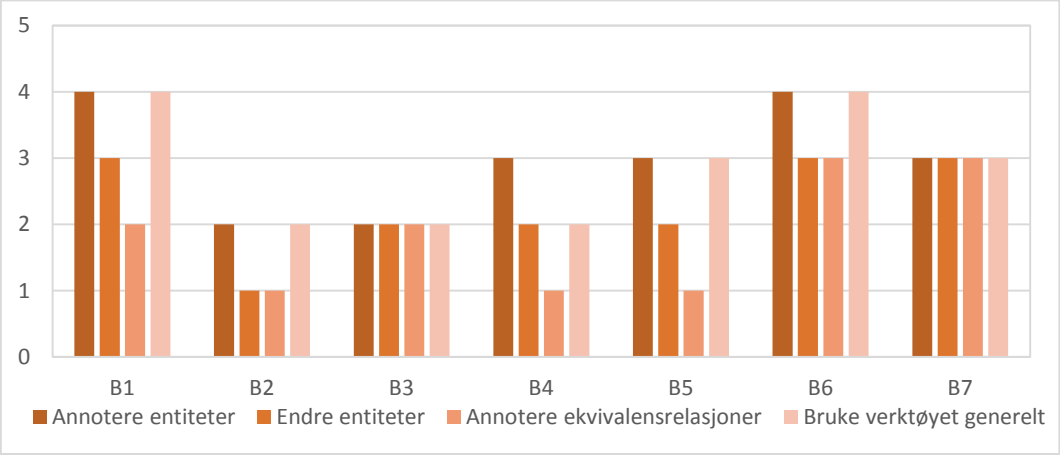
Som nevnt tidligere ba vi alle deltakerne om underveis å rangere hvor vanskelig de *forventet* at en oppgave skulle være før de gjennomførte den, og hvor vanskelig de *opplevde* at oppgaven var, på en skala fra 1 til 5 hvor 1 var «veldig enkel» og 5 var «veldig vanskelig». I forbindelse med intervjuet etter testen ba vi de rangere hvor vanskelig de følte de forskjellige aktivitetene var, hvor mye de følte de hadde lært om aktiviteten og hvor interessert de var i å delta på lignende aktiviteter i fremtiden på samme skala.

Figur 13 viser hvor vanskelig de opplevde de forskjellige aktivitetene: å annotere entiteter (venstre), redigere annoteringer (andre til venstre), opprette ekvivalensrelasjoner (andre til høyre) og deres generelle erfaring med å bruke verktøyet i sin helet (høyre). Figur 14 viser deltakernes vurderinger hvor interessert de er i å delta i prosjekter med lignende aktiviteter

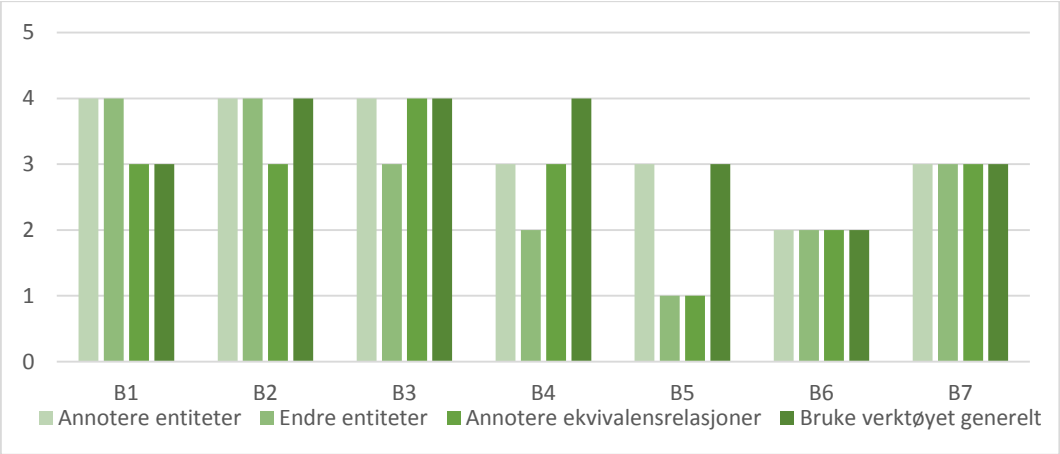
ved en senere anledning. Figur 13 og 14 er interessante i lys av hverandre, figur 15 viser de to grafene overlappende hverandre.

Fem av de syv deltakerne oppgav at å redigere annoteringer og å opprette ekvivalens-relasjoner var lettere enn å lage nye annoteringer og å bruke verktøyet generelt. De to andre deltakerne følte at alle oppgavene var like vanskelige. Dette er interessant med tanke på resultatene deres vist i figur 11: for eksempel følte B5 at verktøyet var relativt vanskelig å bruke (3), men hun var den eneste ikke-eksperten med som utførte alle oppgavene korrekt. Å endre annoterte entiteter ble vurdert som bemerkelsesverdig vanskelig med tanke på at alle deltakerne fullførte oppgave 5 korrekt og relativt fort. Dette kan være forbundet med usikkerheten rundt om de hadde fullført oppgaven, og at problemet derfor ligger i manglende tilbakemelding fra verktøyet.

Med intime brukertester som denne er det en viss fare for at deltakerne vurderer verktøyet som vanskeligere enn hva de egentlig føler på grunn av en underliggende frykt for å fremstå som dumme dersom de rapporterer at oppgavene var enkle, men har løst dem feil. Som sagt presiserte jeg tydelig og flere ganger at det var systemet som ble testet, og at eventuelle feil som ble gjort var en følge av at systemet ikke var brukervennlig, men faren ligger der likevel. Jeg fikk også inntrykk av at dette enkelte ganger var tilfellet, for eksempel mumlet en deltaker «jeg brukte jo ganske lang tid på den, så jeg får vel si den var vanskelig da» mens hun fylte ut skjemaet. Det er derfor en risiko for at verktøyet ble vurdert som vanskeligere enn nødvendig fordi brukerne undervurderer egne evner, og at vi derfor har mye å hente i å utvikle bedre tilbakemeldinger og å få verktøyet til å fremstå som tryggere for brukeren.

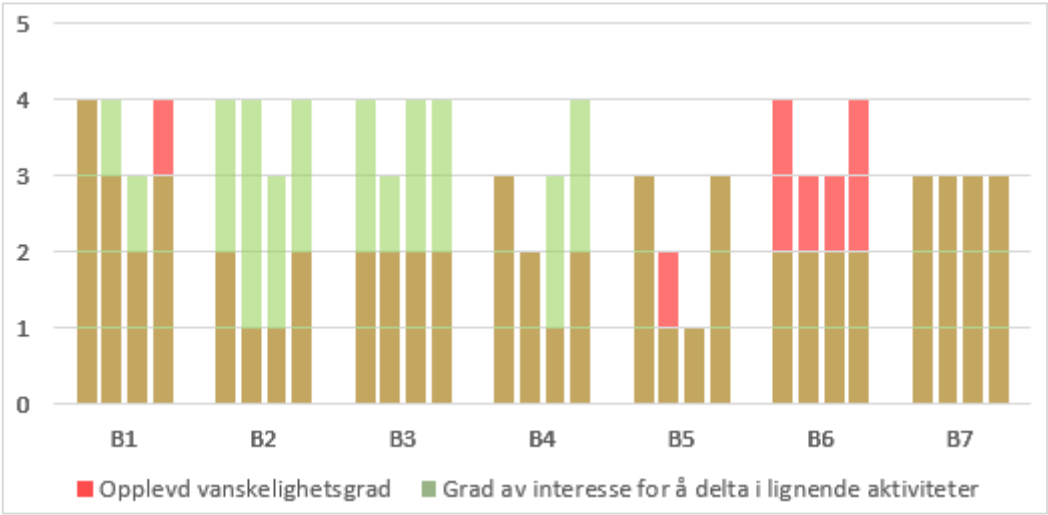


Figur 13: Deltakernes selvrapporterte grad av opplevd vanskelighet



Figur 14: Deltakernes selvrapporterte grad av interesse for å delta i prosjekter med lignende aktiviteter

Vi kan se det generelt sett var høy interesse for å delta på lignende aktiviteter senere, og at det ikke finnes noen tydelig relasjon mellom hvor vanskelig en bruker opplevde verktøyet og hvor interessert de var i å delta i lignende aktiviteter.



Figur 15: Deltakernes selvrapporterte grad av opplevd vanskelighet (rød) og interesse (grønn) for aktivitetene: å annotere entiteter (venstre), redigere annoteringer (andre til venstre), opprette ekvivalensrelasjoner (andre til høyre) og deres generelle erfaring med å bruke verktøyet i sin helet (høyre)

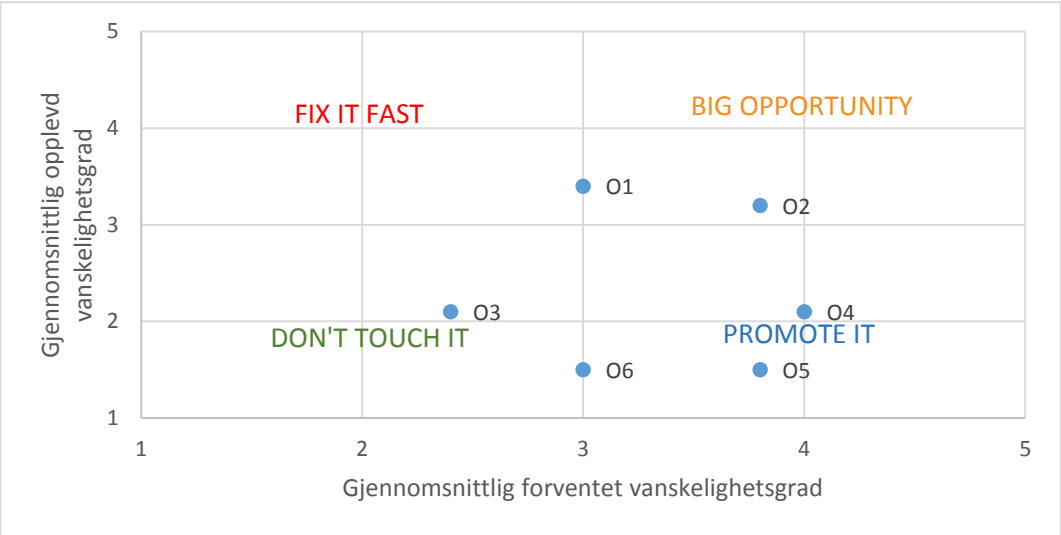
Albert og Dixon (sitert i Tullis & Albert, 2013, s. 132–133) hevder at det viktigste aspektet ved brukskvaliteten til et system er hvor vanskelig eller lette oppgaver er å utføre i forhold til brukerens forventninger. Ved å se på den gjennomsnittlige verdien for deltakernes *forventinger* om en oppgaves vanskelighetsgrad og deres *opplevde* vanskelighetsgrad, kan vi legge dataene inn i et spredningsplott og bruke systemet til Albert og Dixon for å se hvilke aspekter som bør prioriteres. På denne måten får man oppgavene som brukerne forventet at skulle være enkle, men som viste seg å være vanskelige, øverst til høyre. Dette kaller Albert og Dixon for «fix it fast»-sektoren. Nederst til høyre havner oppgavene brukerne både forventet at skulle være enkle og som de opplevde som enkle, oppgavene som med andre ord

fungerer som ønsket. Denne delen av spredningsplottet kaller de «don't touch it»-sektoren. Nederst til venstre havner da oppgavene som var lettere enn brukerne forventet, kalt «promote it»-sektoren, og øverst til venstre ligger oppgavene som både var forventet og opplevd som vanskelige, kalt «big opportunity»-sektoren. Aktiviteter som er mye vanskeligere enn forventet er problemer man bør prioritere å løse fort. Funksjoner som fungerer som forventet og oppleves som enkle bør ikke røres, mens enkle oppgaver brukerne forventer at skal være vanskelige kan ofte bli markedsført bedre, derfor «promote it». I «big opportunity» ligger aktivitetene som både forventes og oppleves som vanskelige, kan man ha mye å hente i å gjøre disse funksjonene enklere for brukerne. (Sitert i Tullis & Albert, 2013, s. 132–133)

Tabell 6 viser de gjennomsnittlige verdiene for forventet og opplevd vanskelighetsgrad fordelt på oppgave, mens figur 16 viser spredningsplottet basert på tabellen.

	Gjennomsnittlig forventet vanskelighetsgrad	Gjennomsnittlig opplevd vanskelighetsgrad
O1	3	3,4
O2	3,8	3,2
O3	2,4	2,1
O4	4	2,1
O5	3,8	1,5
O6	3	1,5

Tabell 6: Gjennomsnittlig forventet og opplevd vanskelighetsgrad fordelt på oppgave



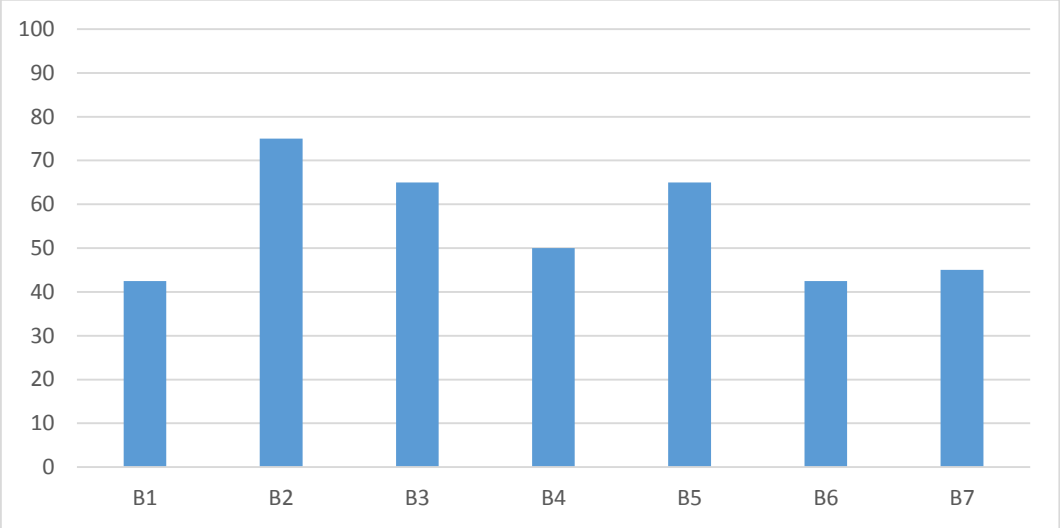
Figur 16: Spredningsplott basert på gjennomsnittlig forventet og opplevd vanskelighetsgrad fordelt på oppgave

Det vi kan lese av spredningsplottet er at mens ingen av oppgavene befinner seg i den kritiske «fix it fast»-sektoren, finnes det rom for forbedringer i de fleste oppgavene. Samtidig kan vi

se at deltakernes opplevde vanskelighetsgrad gikk betraktelig nedover i testen, men at forventningene til at nye typer oppgaver skulle være vanskelige var ganske høy.

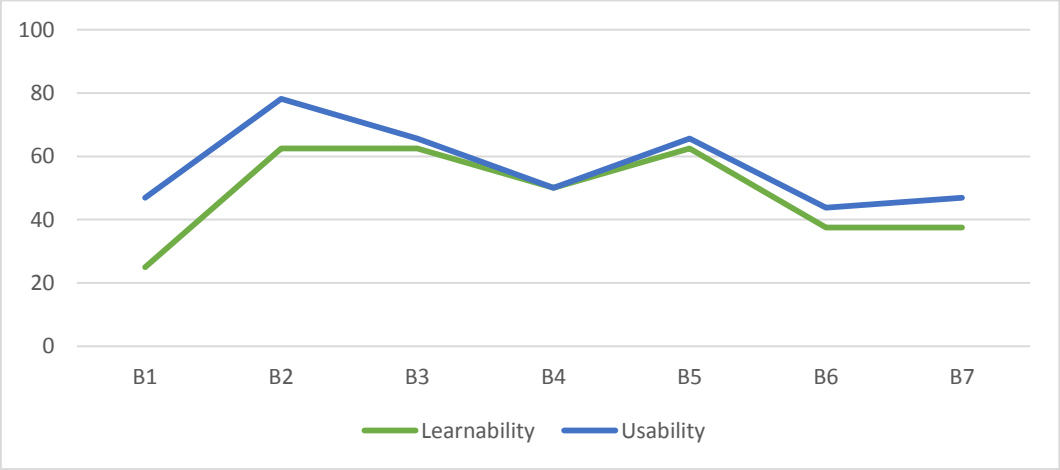
4.1.3.1 SUS-score

Figur 17 viser SUS-scoren fra de forskjellige deltakerne. Som vi ser i figuren var den høyeste scoren 75, og den laveste 42,5, gjennomsnittet er 55. Som nevnt tidligere vil en score på over 68 tilsi at systemet er vurdert som over gjennomsnittlig bra, med andre ord vurderte deltakerne annoteringsverktøyet som noe under gjennomsnittet godt å bruke.



Figur 17: SUS-score fordelt på deltaker

Poengscorene for påstandene i SUS bør ikke vurderes individuelt, siden hvert svar vektes og man kommer frem til en sluttsum som sier noe om brukskvaliteten, men ved å bryte ned poengsummen som beskrevet i 3.3.1 Datainnsamling kan vi se hvordan den generelle brukskvaliteten stiller seg i forhold til hvor lett brukerne føler verktøyet er å lære seg. Figur 18 viser en moderat korrelasjon mellom de to verdiene.



Figur 18: Resultatet av å bryte ned SUS-scoren til learnability- og usability-aspektene

Den gjennomsnittlige verdien for *learnability* var 48,2 og den gjennomsnittlige verdien for brukskvaliteten 56,7. Dette indikerer at deltakerne syntes verktøyet var vanskeligere å lære enn å bruke, noe som stemmer overens med det jeg observerte og fikk vite i intervjuene.

4.1.4 Observasjoner og intervju

Som nevnt valgte jeg å kategorisere problemene jeg observerte etter mønstrene til Burghardt. Hovedvekten av problemene jeg fant tilhørte kategorien *generelt grensesnitt*. I denne kategorien beskriver Burghardt problemer som påvirker den generelle opplevelsen av verktøyet, uten tilknytning til et spesifikt stadium i annoteringsprosessen. Av mønstrene Burghardt beskrev kunne jeg identifisere følgende: trygg utforskning (safe exploration), hjelp for domenespesifikke funksjoner (help for domain-specific functions) eller utilstrekkelig dokumentasjon (insufficient documentation), overflødige kontroller (redundant controls), mangel på en eksplisitt funksjon for å lagre (no explicit save action) og skreddersydd visning av data (tailored display of data). Burghardt hevder det ikke finnes generiske løsninger på problemer i kategorien generelt grensesnitt (2014, s. 148).

Når det kom til trygg utforskning av verktøyet uttrykte flere brukere at de var «redde for å ødelegge noe». En bruker sa «jeg føler at utviklerne stoler på meg…kanskje litt for mye». Vi hadde åpenbart noe å hente i å gjøre verktøyet mindre «skummelt», og sørge for at brukerne forstår at ingen feil er umulige å rette.

Vi identifiserte også et problemer som kunne kategoriseres i Burghardts mønster kalt hjelp for domenespesifikke funksjoner. «Domene» henviser her til semantisk annotasjon som en aktivitet, ikke korpusets eller annoteringsprosjektets emne. Flere brukere uttrykte at de var usikre på hva en ekvivalensrelasjon var, og var som følge usikre på om de hadde annotert dem korrekt. Som vist i figur 12 var også oppgaven hvor brukerne skulle opprette ekvivalensrelasjoner den oppgaven med flest semantisk meningsløse resultater. Dette til tross for at alle deltakerne på forhånd hadde lest gjennom retningslinjene som beskriver hvordan man skal håndtere ekvivalensrelasjoner. Burghardt foreslår her flere løsninger, hvorav en er å integrere korte forklaringer i selve verktøyet, i stedet for å lene seg for mye på frittstående retningslinjer. Dette er et spesielt viktig punkt å ta hensyn til med tanke på tilrettelegging for ikke-ekspertbrukere.

Overflødige kontroller er et problem beskrevet av Burghardt, og to deltakere var usikre på hvilket «sett» med knapper de skulle bruke. Disse to «settene» med knapper har identiske

funksjoner, og var plassert der for at brukeren skulle slippe å scrolle opp for å lagre dersom hun befant seg nederst på siden, men det ekstra settet skapte kun forvirring.

At det ikke var en eksplisitt funksjon for å lagre (no explicit save action) er også et problem Burghardt beskriver. Som du kan se i figur 19 har knappen for å lagre teksten «oppdater», og en av deltakerne utbrøt «men jeg vil ikke *oppdatere* noe, jeg vil *lagre*!» midt i testen. Mens flere deltakere nølte første gangen de så knappen, var hun den eneste som kommenterte ordlyden.

Figur 19: Skjermbilde av knappene for å slette, lagre, avbryte og opprette ekvivalensrelasjon

Et par deltakere uttrykte også forvirring rundt ordlyden for ekvivalensrelasjoner. I oppgaveheftet hadde jeg omtalt det som «ekvivalensrelasjon (coRef)», mens det i retningslinjene er kalt identicalTo og knappen selv leser «coRefers». I tillegg til uoverensstemmelsen i ordlyden, sa flere deltakere at de følte «coRefers» var intetsigende.

En deltaker kommenterte fargen på knappen, og sa at gult fikk han til å tenke på fare. To av deltakerne kommenterte mengden primærdata som ble presentert, og uttrykte av de følte seg litt overveldet. Sistnevnte går under Burghardts punkt om skreddersydd visning av data (tailored display of data), hvor han anbefaler å gi brukerne mulighet til å tilpasse elementer som skriftstørrelse, -type og linjeavstand. Dette gir brukerne mer kontroll og frihet, og noe som legger til rette for mer effektiv annotering.

At TORCH-URI-en tildeles automatisk og at brukeren derfor skal la feltet stå i fred var også en kilde til forvirring. Feltet er en vanlig tekstboks, og det er ikke nevnt hva man (ikke) skal gjøre med det i retningslinjene, så instinktet til flere deltakere var å plassere pekeren i boksen uten noen formening om hva de skulle skrive der.

Figur 20: Skjermbilde av feltet for TORCH-URI

Som nevnt tidligere var det viktig for oss å se hvordan brukerne opplevde de forskjellige trinnene i annoteringsprosessen, altså om de klarte å velge et anker, velge en eksisterende annotering, lagre annoteringer, redigere annoteringer og slette annoteringer. Alle brukerne hadde litt problemer med å markere ankeret første gangen. At man må starte markeringen i midten av en frase eller et ord virket fremmed for dem, men etter å ha gjort det en gang følte

alle foruten en bruker at det var relativt enkelt. En deltaker ønsket å finne alle forekomstene av en entitet med «finn»-funksjonen ctrl+f i Windows. En annen ønsket å annotere i bulker, altså markere alle forekomster av en entitet og annotere dem i ett for å spare tid. Det oppstod ingen problemer ved å velge annoterte entiteter, og som man kan se i figur 12 var alle deltakerne i stand til å endre og slette annoteringer.

I tillegg til aspektene beskrevet over, gikk tilbakemeldingene i intervjuene i hovedsak ut på at verktøyet var lite intuitivt, men enkelt i bruk når de visste hva de skulle gjøre. En deltaker sa for eksempel at «det var vanskelig først, men så skikkelig enkelt, hvis du skjønner?». Flere deltakere kommenterte på at grensesnittet fremstod som litt gammeldags.

Alle deltakerne fullførte oppgavene innen rimelig tid, og det var minimal forvirring rundt valg av kategorier fra ontologien.

4.1.5 Oppsummering av funn og anbefalinger etter første testrunde

Resultatene etter den første testen indikerte at annotører med rimelig datakompetanse kan ha høy suksessrate uten å være domeneeksperter. I tråd med eksisterende litteratur så vi at deltakerne foretrekker og forventer at grensesnittet er utformet og fungerer som tradisjonelle nettapplikasjoner. Jeg fant ingen klar sammenheng mellom deltakernes opplevde vanskelighetsgrad og deres interesse i å delta i videre aktiviteter av samme type, noe som er oppmuntrende med tanke på crowdsourcing. Ingen annoterte fullstendig irrelevante deler av teksten, men dette er sannsynligvis en konsekvens av måten vi utformet oppgavene på.

Deltakernes *opplevde* vanskelighetsgrad gikk betraktelig nedover i testen, men siden de opplevde verktøyet som lite intuitivt, *forventet* de at oppgavene skulle være vanskelige. Med tanke på de fem brukskvalitetskomponentene til Nielsen, kan vi anta at verktøyet har mye å hente når det kommer til hvor lett det er å lære. Men selv om det ble ikke var intuitivt for deltakerne, mente de fleste at det var både enkelt å bruke og effektivt etter å ha gjennomført en oppgave eller to. Vi kan derfor si at verktøyet ble opplevd som litt vanskelig å lære, men også som effektivt og lett å huske. Den generelle tilfredsheten var rimelig.

Det var få semantisk meningsløse feil i resultatet av annoteringene, noe som kan tyde på at å ha tydelige retningslinjer lett tilgjengelig kan bidra til å heve kvaliteten. Ved for eksempel å tydeliggjøre for brukerne hvilke Wikipedia-URI-er som er aktuelle, og presisere at substantivet som tilhører egennavn ikke skal annoteres, vil feilraten kunne gå ned. Vi kan anta at det også er viktig at retningslinjene er lett tilgjengelige, da kun én deltaker tok seg bryet med å konsultere de utskrevne retningslinjene underveis.

Som sagt er det også en fare for at verktøyet ble vurdert som vanskeligere enn nødvendig fordi brukerne undervurderte egne evner, og at vi derfor har mye å hente i å utvikle bedre tilbakemeldinger og å få verktøyet til å fremstå som tryggere for brukeren. På samme måte må man ta høyde for at deltakerne kan ha rangert interessen sin høyere enn reelt for å være snille.

Tabell 7 viser en oppsummering av de konkrete brukskvalitetsproblemene som ble avdekket i den første testen, kategorisert etter Burghardts bruksvalitetsmønstre (2014) og rangert etter alvorlighetsgradene beskrevet i 3.3.1.1.

Problem, kategori og alvorlighetsgrad	**Beskrivelse**	**Forslag til løsninger**
Problem: Inkonsekvent ordlyd rundt ekvivalensrelasjoner **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Knappen i verktøyet leser «coRefer» mens retningslinjene henviser til ekvivalensrelasjoner som «identicalTo»	Oppdater verktøyet eller retningslinjene
Problem: Høy læringskurve **Kategori**: Generelt grensesnitt: trygg utforskning **Alvorlighetsgrad**: 3 - Medium	Verktøyet er ikke intuitivt ved førstegangsbruk, men oppleves som enkelt etterhvert	Innfør en opplærings-/øvelsessekvens (tutorial) i begynnelsen av første økt, hvor brukeren får mulighet til å utforske systemet og opprette en annotering uten konsekvenser
Problem: Redundant knappesett for lagre/slette **Kategori**: Generelt grensesnitt: overflødige kontroller **Alvorlighetsgrad**: 3 -Medium	Det finnet to identiske knappesett for lagring/sletting, dette forvirrer brukerne	Fjern ett av knappesettene
Problem: Ordlyd i knapp for å lagre annotering **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Knappen for å lage/oppdatere leser «oppdatere», brukere forventer «lagre»	Endre ordlyd til «lagre»
Problem: Inntrykk av fryst system under Wikipedia-oppslag **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Bruker tror verktøyet har fryst når det bruker tid på oppslag mot Wikipedia	Tilbakemelding om at systemet arbeider
Problem: Manglende tilbakemelding på om en annotering er lagret korrekt **Kategori**: Generelt grensesnitt: hjelp for domenespesifikke funksjoner **Alvorlighetsgrad**: 3 - Medium	Brukere uttrykte usikkerhet rundt om en annotering var utført korrekt, og om resultatet deres var lagret	Tilbakemelding om gjennomført lagring i systemet

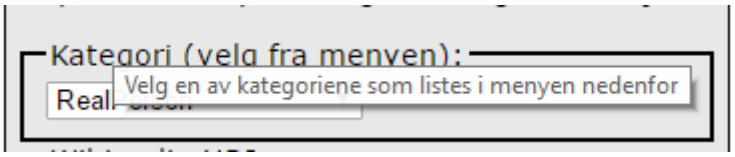
Problem: Forvirring rundt opprettelse av anker **Kategori**: Annoteringsprosessen: opprett anker **Alvorlighetsgrad**: 3 - Medium	Å starte markering av anker midt i ordet/frasen virker fremmed for brukeren ved første gangs bruk	-
Problem: Farge på knapp for å lagre annotering **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 4 - Lav	Knappen for å lage/oppdatere er gul, dette forvirrer brukerne	Endre farge til systemstandard eller grønt
Problem: Overveldende lang post **Kategori**: Visning av primærdata **Alvorlighetsgrad**: 4 - Lav	Brukere føler posten som skal annoteres er lang	Gjør det mulig å endre mengde tekst som vises
Problem: Bruker vil åpne Wikipedia-URI **Kategori**: Ønsket funksjon **Alvorlighetsgrad**: 4 - Lav	Bruker ønsker å åpne Wikipedia-lenker for å 1) forsikre seg om at det er korrekt URI, eller 2) lese mer av ren interesse	Gjør lenken klikkbar og åpne siden i ny fane

Tabell 7: Oppsummering av brukskvalitetsproblemer etter første test

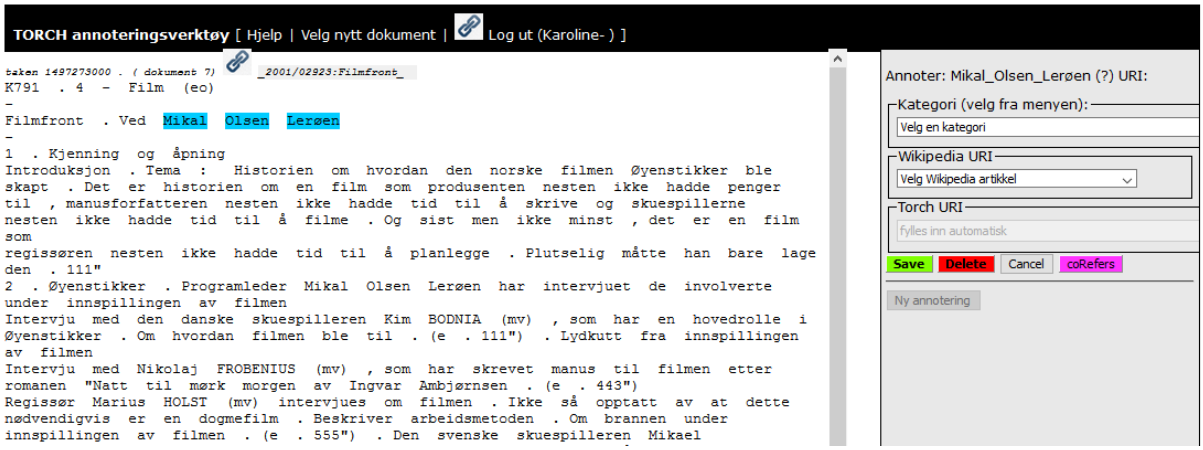
Som videre forskning så vi behovet for å gå dypere i brukernes holdninger til annotering på den ene siden, for eksempel ved dybdeintervju, og behovet for oppdatere grensesnittet og teste en større del av verktøyets funksjonalitet for ytterligere brukervennlighetstesting. (Hoff & Preminger, 2015) Jeg valgte å gå videre med sistnevnte.

4.2 Andre test: Annotering av entiteter og relasjoner

Før jeg gjennomførte den utvidede testen endret vi deler av grensesnittet etter anbefalingene fra den første testen. For eksempel ble farger og ordlyd på enkelte knapper endret, og det ble lagt til korte hjelpetekster ved mouseover (at man holder pekeren over et gitt område). Figur 21 viser et eksempel på en slik hjelpetekst. De redundante knappene ble fjernet, og tekstfeltet for TORCH-URI ble gjort utilgjengelig. Det oppdaterte grensesnittet kan sees i figur 22. Funnene i den første testen påvirket grensesnittet, det ble med andre ord gjort en formativ evaluering.



Figur 21: Eksempel på hjelpetekst ved mouseover



Figur 22: Grensesnittet slik det fremstod under den andre brukertesten

I den utvidede brukertesten var det hensiktsmessig at oppgavene lignet på de i den første testen for at jeg skulle kunne sammenligne resultatene. I pilottesten for den utvidede testen ble jeg fort klar over at jeg med det gamle oppsettet av oppgavene risikerte at brukeren annoterte entiteter uten relasjoner seg imellom, og at de dermed måtte annotere unødvendig mange entiteter for å fullføre testen. Jeg endte derfor opp med å redusere de opprinnelige oppgavene, og legge til fire oppgaver som tilrettela for annotering av relasjoner. I relasjonsdelen av oppgavene fikk deltakerne beskjed om at de kunne ta utgangspunkt i de allerede annoterte entitene eller lage helt nye. Vedlegg 2 viser det fullstendige oppgaveheftet brukerne fikk utdelt i den andre undersøkelsen. En kort oppsummering av oppgavene er presentert under:

Del 1: Entiteter

- Oppgave 1: Annoter et valgfritt personnavn
- Oppgave 2: Annoter et valgfritt verk
- Oppgave 3: Annoter et valgfritt sted
- Oppgave 4: Endre opplysningene i en annotert entitet
- Oppgave 5: Slett en valgfri entitet
- Oppgave 6: Lag en ekvivalensrelasjon mellom to entiteter

Del 2: Relasjoner

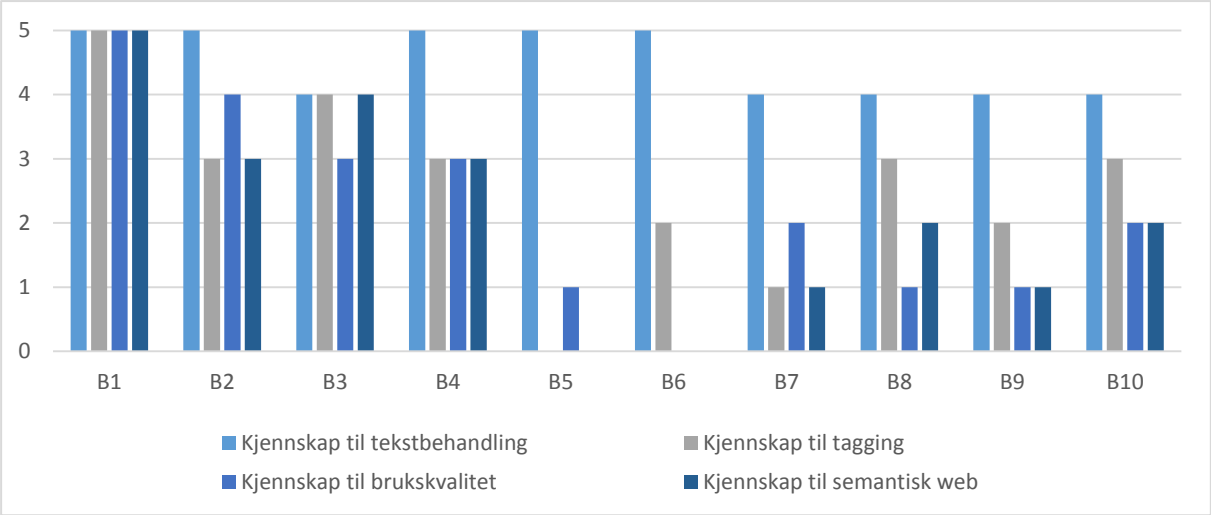
- Oppgave 7: Annoter to relaterte entiteter, og relasjonen dem i mellom
- Oppgave 8: Gjenta oppgave 7, men med to nye entiteter
- Oppgave 9: Endre en relasjon
- Oppgave 10: Slett en relasjon

I likhet med første test gjennomførte jeg først en vellykket pilottest, og brukte resultatene fra denne testen som et mål på hva som var rimelig tid på hver av oppgavene. Dette varierte mellom 2-6 minutter per oppgave. Jeg fant også ut at det her var hensiktsmessige å be alle

deltakerne om å lese fort gjennom oppgavene først, så de var forberedt på relasjonsannoteringen og kunne velge passende entiteter tidlig i testen. Testen ble gjennomført som beskrevet i 3.3.3 Testforløp.

4.2.1 Deltakere

Igjen rekrutterte jeg deltakere ved å be medstudenter på Høgskolen i Oslo og Akershus og tilfeldige studenter ved Høyskolen Kristiania om å delta, og ba hver deltaker om å rangere sine kunnskaper som beskrevet i 3.3.2. Siden jeg i denne omgangen hadde mindre tidspress enn ved første test, valgte jeg å teste med ti brukere. Figur 23 viser de ti deltakernes egenvurderinger.

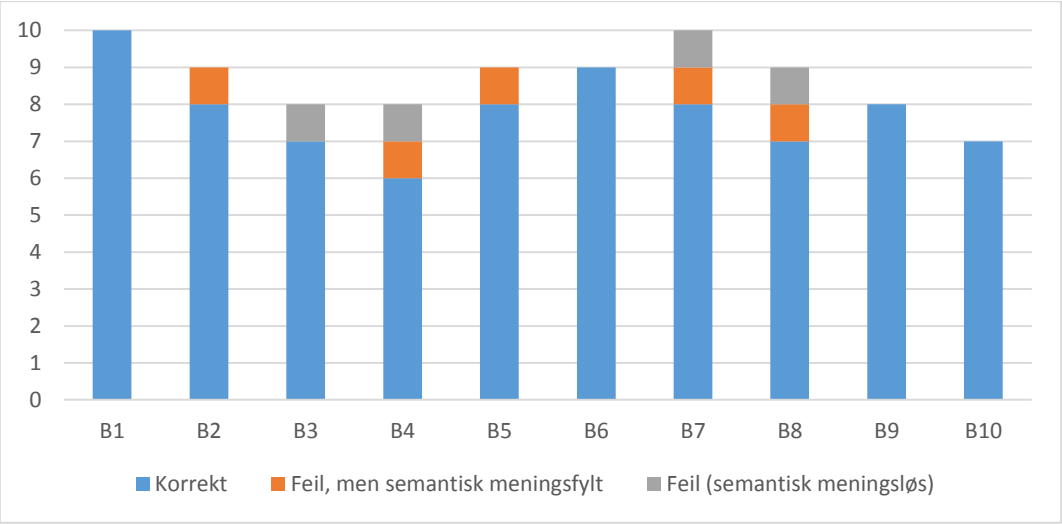


Figur 23: Deltakernes vurdering av egen kunnskap i den første testen ved den andre testen

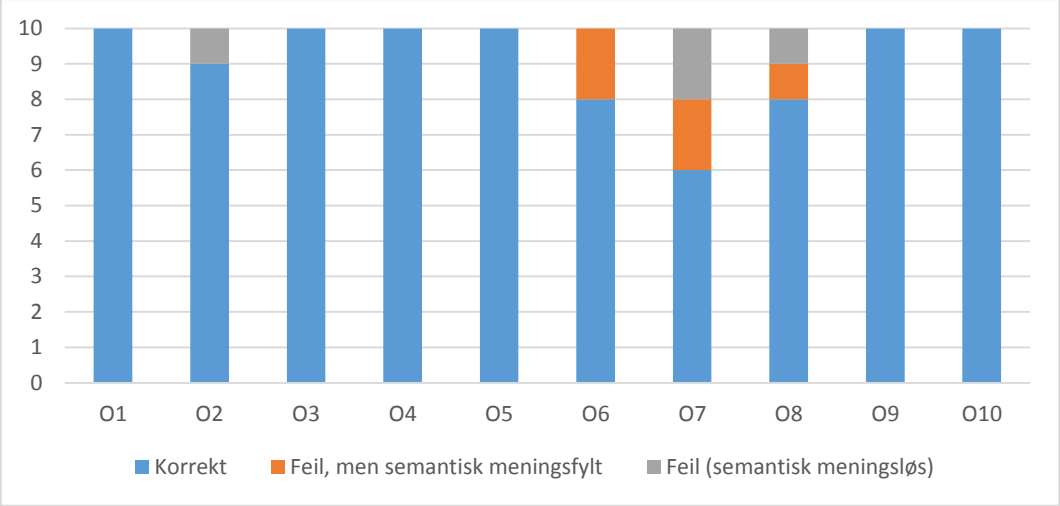
Alle deltakerne kunne regnes som datakyndige. Fire av deltakerne rapporterte kjennskap til tagging og semantisk web, mens seks deltakere følte at de ikke hadde nevneverdig kunnskap på området. Basert på deres egne vurderinger kategoriserte jeg B1-B4 som tekniske eksperter og B5-B10 som ikke-eksperter.

4.2.2 Annoteringer

Som en følge av hvordan oppgavene var lagt opp, kunne deltakerne ende opp med fire til syv annoterte entiteter, avhengig av om hun tok utgangspunkt i entitetene fra oppgave 1-3 i oppgave 7 og 8. I tillegg ville testen resultere i en ekvivalensrelasjon og tre vanlige relasjoner. Antall annoteringer indikerer altså ikke om en bruker gjorde mye feil eller riktig. Alle ti deltakerne opprettet minst seks korrekte annoteringer, og fire deltakere hadde kun korrekte annoteringer. Figur 24 viser suksessraten til resultatene fra oppgave 1-3 og 6-8 fordelt på deltaker, mens figur 25 viser deltakernes suksessrate fordelt på alle oppgavene.



Figur 24: Resultatet av annoteringene i den andre testen (oppgave 1-3 og 6-8) fordelt på bruker



Figur 25: Resultatet av annoteringene i den andre testen fordelt på oppgave

Det dukket opp to typer «feil, men semantisk meningsfulle» annoteringer. I oppgave 6 var det to tilfeller av at deltakeren satte en ekvivalensrelasjon mellom en entitet og substantivet som henviste til entiteten. Dette ønsket om å opprette ekvivalensrelasjoner mellom en navngitt entitet og substantivet som henviste til det ble nevnt flere ganger enn det ble registrert, og kan vise seg å være et problem dersom deltakere skal annotere hele poster. Da en deltaker under oppgave 6 var usikker på om hun skulle velge «Øyenstikker» eller ordet «filmen», sa hun «jeg kan jo like gjerne velge ordet "Øyenstikker", for da er jeg sikker».

Av semantisk meningsfylte feil i forbindelse med annotering av relasjoner hendte det at brukeren valgte korrekte, relaterte entiteter, men valgte feil relasjon dem imellom. For eksempel annoterte en deltaker relasjonen «hasScriptWriter» mellom verket *Natt til mørk morgen* og personen Ingvar Ambjørnsen, men i dette tilfellet ville «hasAuthor» vært korrekt. En deltaker forsøkte også å opprette en relasjon mellom Stockholm og Mikael Persbrandt

fordi han bor der, men ontologien tar ikke høyde for det, så deltakeren forstod at det ikke kunne stemme.

Feil av typen «feil, semantisk meningsløs» forekom på to måter: i ett tilfelle ble et substantiv feiltolket som verk, mens det i annoteringen av relasjoner hendte at deltakeren opprettet en relasjon eller mellom to entiteter i samme kategori, for eksempel mellom Maria Bonnevie og Kim Bodnia fordi begge er skuepillere.

Ingen annoterte feil deler av teksten, men igjen er dette mest sannsynlig en konsekvens av måten oppgavene var lagt opp. Selv om alle brukerne hadde tilgang til retningslinjene, var ingen som konsulterte dem når de var i tvil om hva som var korrekt. Kun én deltaker (B10) klikket på menyknappen «Hjelp» i fanen øverst, og han var den eneste deltakeren som fant hjelpetekstene i programmet ved mouseover. At ni av ti deltakere ikke engang la merke til at disse hjelpetekstene fantes betyr i praksis at de er usynlige, og dermed ubrukelige, for brukerne der de ligger i denne versjonen av verktøyet. Hjelpetekstene må derfor enten fremheves ved hjelp av ikoner, eller formidles til brukerne av programmet på en annen måte.

Basert på resultatene til de forskjellige brukerne vil jeg kategorisere B1, B3, B6, B9 og B10 som akseptable annotører, og resten av brukerne som annotører med forbedringspotensial. Gjennom at hun tenkte høyt underveis i testen fikk jeg vite at feilen til B3 var at hun oppfattet et substantiv som et verk, så feilen hennes lå i feiltolking av selve teksten, ikke at hun misforstod annoteringsprosessen eller hva som skulle annoteres. Ingen brukere hadde så mange eller så grove feil at jeg vil kategorisere dem som uegnede annotører.

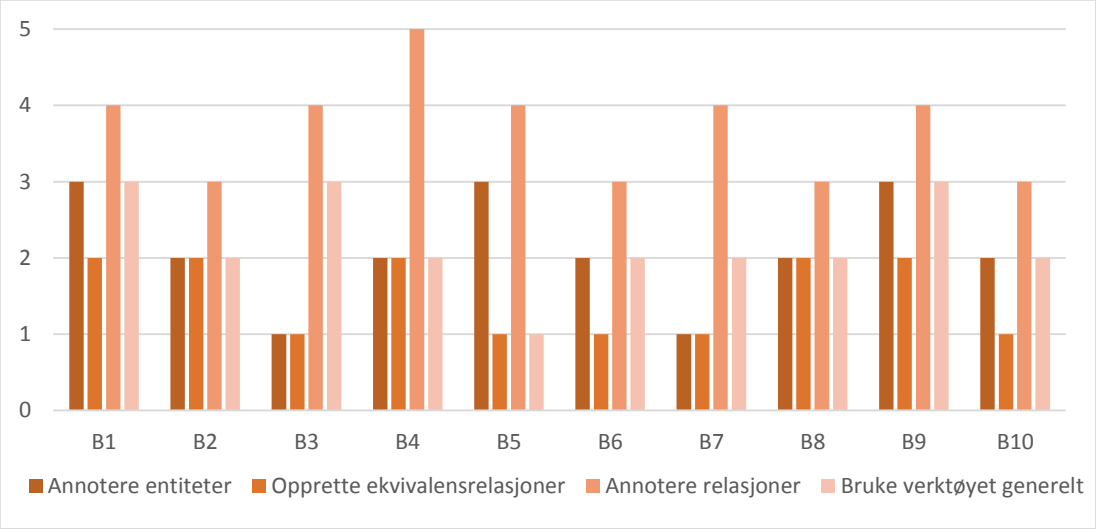
Det var kun én oppgave som noen brukere gav opp og måtte be om hjelp for å komme videre i: den første relasjonsoppgaven. Når det oppstod forvirring i de andre oppgavene klarte alle brukerne å navigere rundt på siden til de fant det de lette etter, men å *dobbeltklikke* på relasjonsikonet slet de med. Kun én deltaker fant ut av det uten hjelp (B10), han fant det ut ved å lese hjelpeteksten i mouseover på ikonet. En bruker fant det riktignok ved en tilfeldighet da hun klikket vilt i frustrasjon, men hun måtte be om hjelp for å gjøre det på nytt i neste oppgave. Jeg regnet ikke denne oppgaven som avbrutt når jeg måtte hjelpe dem ved å «avsløre» at de måtte dobbeltklikke fordi denne oppgaven var flerfasettert, og fordi det hendte med nesten alle deltakerne. Alle de andre oppgavene ble fullført innen rimelig tid.

4.2.3 Deltakernes egenvurderinger av opplevd vanskelighetsgrad og interesse

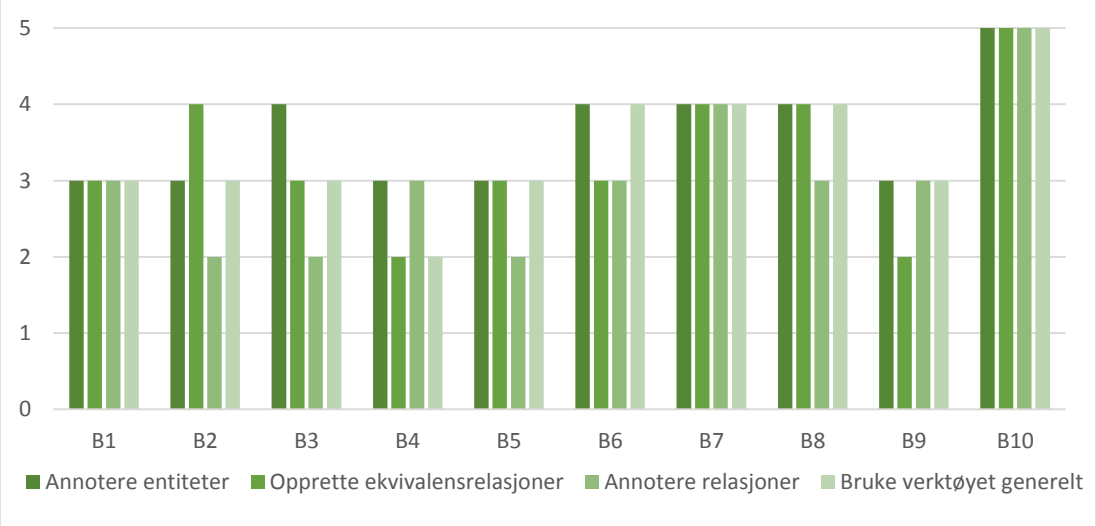
Også i denne testen ble deltakerne etter å ha gjennomført testen bedt om å rangere deres opplevde vanskelighetsgrad og deres interesse for å delta i prosjekter med lignende aktiviteter

i fremtiden på en skala fra 1 til 5, hvor 1 var «ingen vanskeligheter»/«ingen interesse» og 5 var «veldig vanskelig»/«veldig interessert».

Figur 26 viser hvor vanskelig de opplevde de forskjellige aktivitetene: å annotere entiteter (venstre), opprette ekvivalensrelasjoner (andre til venstre), annotere relasjoner (andre til høyre) og å bruke verktøyet generelt (høyre). Figur 27 viser deltakernes vurderinger hvor interessert de er i å delta på lignende aktiviteter.



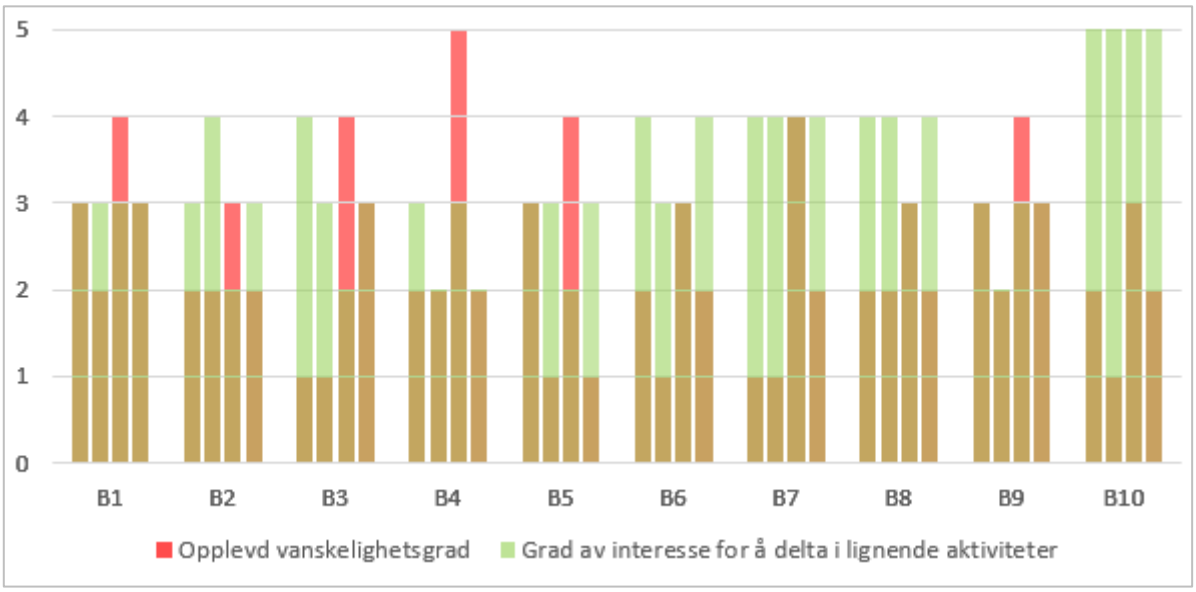
Figur 26: Deltakernes selvrapporterte grad av opplevd vanskelighet



Figur 27: Deltakernes selvrapporterte grad av interesse for å delta i prosjekter med lignende aktiviteter

Igjen er disse to grafene er spesielt interessante i lys av hverandre. Figur 28 viser grafene fra figur 26 og 27 overlappet. Aktiviteten å annotere relasjoner stikker seg ut, og vi kan se at det helt klart er aktiviteten det er minst interesse for, og som helt klart ble rapportert som vanskeligst. I likhet med i den forrige testen er det jevnt over høy interesse for å delta på lignende aktiviteter igjen, selv på aktivitetene som ble rapportert som moderat vanskelige. Vi

kan ikke anta at det er direkte utelukket for deltakerne å delta i aktiviteter de rangerte med høyere vanskelighetsgrad enn grad av interesse, men kan vi se for oss at det bør innebære en form for belønning eller annen motivasjon for å få deltakere til å gjennomføre disse aktivitetene i større skala.



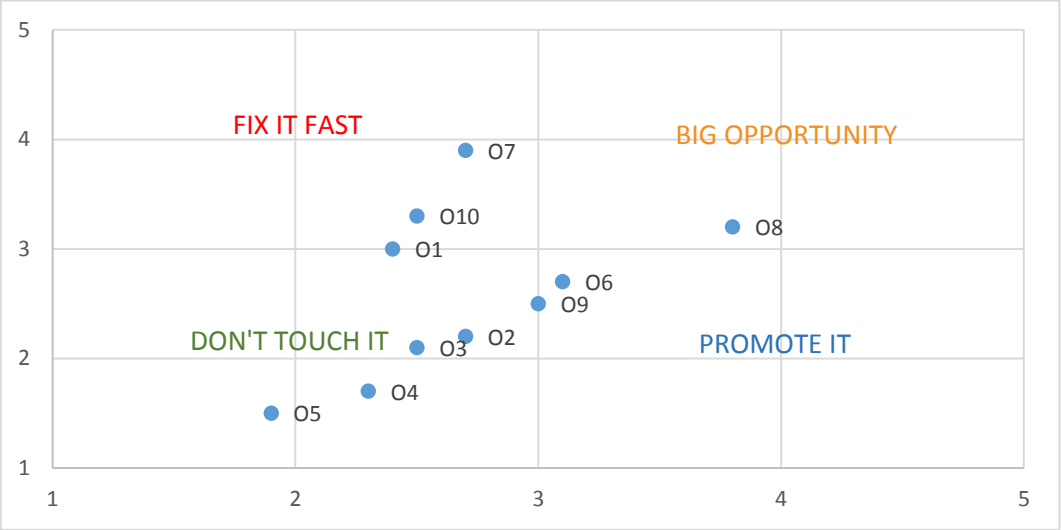
Figur 28: Deltakernes selvrapporterte grad av opplevd vanskelighet og interesse for aktivitetene å annotere entiteter (venstre), opprette ekvivalensrelasjoner (andre til venstre), annotere relasjoner (andre til høyre) og å bruke verktøyet generelt (høyre)

Også i denne runden regnet jeg ut de gjennomsnittlige verdiene for deltakernes *forventinger* om en oppgaves vanskelighetsgrad og deres *opplevde* vanskelighetsgrad, og la verdiene i et spredningsplott etter systemet til Albert og Dixon for å se hvilke aspekter som bør prioriteres. Som sagt havner oppgaver brukerne både forventet at skulle være enkle og som de opplevde som enkle i feltet nederst til høyre, «don't touch it»-sektoren. Oppgaver som viste seg å være vanskeligere enn forventet befinner seg øverst til høyre, i «fix it fast»-sektoren. Nederst til venstre havner oppgaver som var lettere enn brukerne forventet, i «promote it»-sektoren, og øverst til venstre ligger oppgaver som både var forventet og opplevd som vanskelige, i «big opportunity»-sektoren. (Sitert i Tullis & Albert, 2013, s. 132–133)

Tabell 8 på neste side viser de gjennomsnittlige verdiene for forventet og opplevd vanskelighetsgrad fordelt på oppgave, og figur 29 viser spredningsplottet basert på tabellen.

	Gjennomsnittlig forventet vanskelighetsgrad	Gjennomsnittlig opplevd vanskelighetsgrad
O1	2,4	3
O2	2,7	2,2
O3	2,5	2,1
O4	2,3	1,7
O5	1,9	1,5
O6	3,1	2,7
O7	2,7	3,9
O8	3,8	3,2
O9	3	2,5
O10	2,5	3,3

Tabell 8: Gjennomsnittlig forventet og opplevd vanskelighetsgrad fordelt på oppgave



Figur 29: Spredningsplott basert på gjennomsnittlig forventet og opplevd vanskelighetsgrad fordelt på oppgave

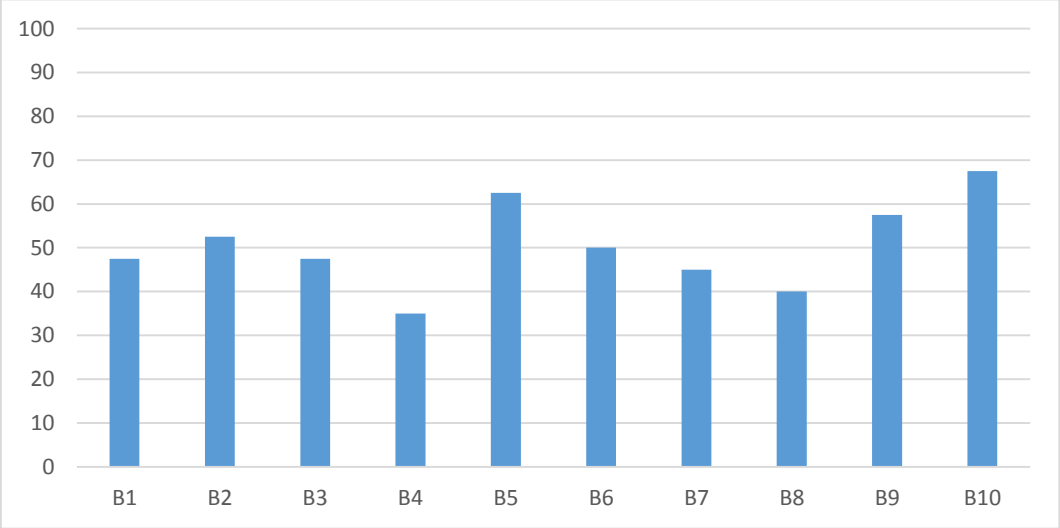
En interessant ting vi kan lese av spredningsplottet er at mens den første relasjonsoppgaven, oppgave 7, ligger i «fix it fast»-sektoren, ligger den identiske oppgave 8 i «big opportunity». Dette indikerer at aktiviteten føltes mye vanskeligere første gangen deltakerne gjennomførte den, enn andre gangen. Dette stemmer med tilbakemeldingene jeg fikk verbalt både underveis i testen og i intervjuet etter at testen var gjennomført: det største problemet lå i å finne ut hvor man annoterer relasjoner, ikke i annoteringen selv.

På samme måte er det at oppgave 10, å slette en relasjon, befinner seg i «fix it fast»-sektoren misvisende. Oppgaven fikk ufortjent høy opplevd vanskelighetsgrad fordi listen med relasjoner ikke synes i skjermbildet før man scroller ned på grunn av postens lengde, og alle

deltakerne kommenterte at selve slettingen var veldig enkel, men å finne ut hvor man gjorde det var veldig vanskelig første gangen.

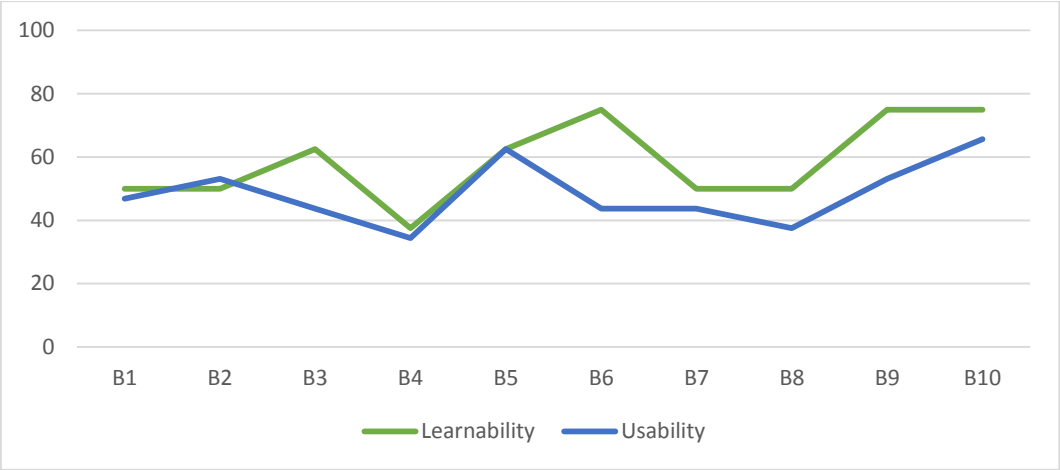
4.2.3.1 SUS-score

Figur 30 viser SUS-scoren fra de forskjellige deltakerne. Som vi ser i figuren var den høyeste scoren 67,5, og den laveste 35, gjennomsnittet er 50,5. Som nevnt tidligere vil en score på over 68 tilsi at systemet er vurdert som over gjennomsnittlig bra. Med andre ord vurderte deltakerne annoteringsverktøyet som noe under gjennomsnittet godt å bruke. Det fikk en lavere score enn i den forrige testen, men det er ikke særlig sammenlignbart da delen av verktøyet som ikke var inkludert i første runde helt klart fikk overvekten av negative tilbakemeldinger i andre runde.



Figur 30: SUS-score fordelt på deltaker

Igjen brøt jeg ned poengsummen som beskrevet i 3.3.1 Datainnsamling for å se hvordan den generelle brukskvaliteten stilte seg i forhold til hvor lett brukerne følte verktøyet var å lære seg. Figur 31 på neste side viser en moderat korrelasjon mellom de to verdiene. Den gjennomsnittlige verdien for *learnability* var 58,7 og den gjennomsnittlige verdien for brukskvaliteten 48,4. Svaret på hvorfor vi i denne runden endte opp med en lavere score for *usability* enn *learnability* kan ligge i vanskelighetene deltakerne hadde med relasjonsannoteringen. Flere deltakere nevnte at de slet med å finne ut *hvor* man annoterte relasjoner i forbindelse med blant annet påstand 5 «jeg synes at de forskjellige delene av systemet hang godt sammen» og 6 «jeg synes det var for mye inkonsistens i systemet (det virket "ulogisk")» i SUS.



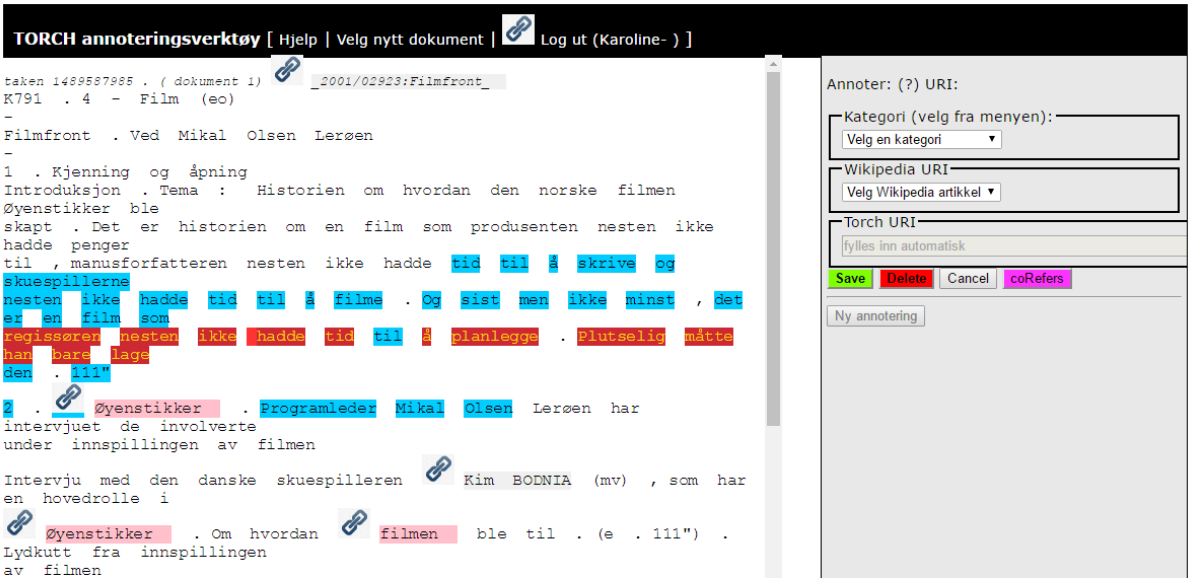
Figur 31: Resultatet av å bryte ned SUS-scoren til learnability- og usability-aspektene

4.2.4 Observasjoner og intervju

Igjen kategoriserte jeg problemene jeg observerte etter mønstrene til Burghardt (2014). Også i denne testen lå hovedparten av problemene i kategorien generelt grensesnitt. Som sagt skriver Burghardt at dette er problemer som påvirker den generelle opplevelsen av verktøyet, uten tilknytning til et spesifikt stadium i annoteringsprosessen. I denne runden identifiserte jeg følgende mønstre: trygg utforskning (safe exploration), hjelp for domenespesifikke funksjoner (help for domain-specific functions) eller utilstrekkelig dokumentasjon (insufficient domuentation), mulighet til å angre (changes in midstream), og tilvenning (habituation).

Tilvenning betyr i denne konteksten at brukere forventer at systemet skal være konsekvent når det kommer til handlinger som dobbeltklikking og plassering av lignende funksjoner. I vårt tilfelle var det som sagt kun én deltaker som fant ut at man måtte dobbeltklikke på relasjonsikonet for å åpne menyen for relasjonsannotering, og det er et kritisk problem. De fleste deltakerne forventet å finne en knapp for relasjonsannotering nært knappen for ekvivalensrelasjoner.

Muligheten til å angre, eller gå tilbake steg i en prosess, er også viktig. I denne testen angret enkelte deltakere på teksten de hadde markert, men ved å klikke «cancel» forsvant ikke den markerte teksten. Dette er rent kosmetisk, verktøyet har ikke lenger denne teksten «markert» for annotering, men for en bruker ser det dramatisk ut. Et deltaker forventet at denne teksten skulle bli umarkert av å klikke på ESC, men dette har ingen funksjon. Et skjermbilde hvor det er markert unødvendig tekst og klikket «cancel» kan ses i figur 32 på neste side.



Figur 32: Skjermbilde av verktøyet når mye tilfeldig tekst markeres

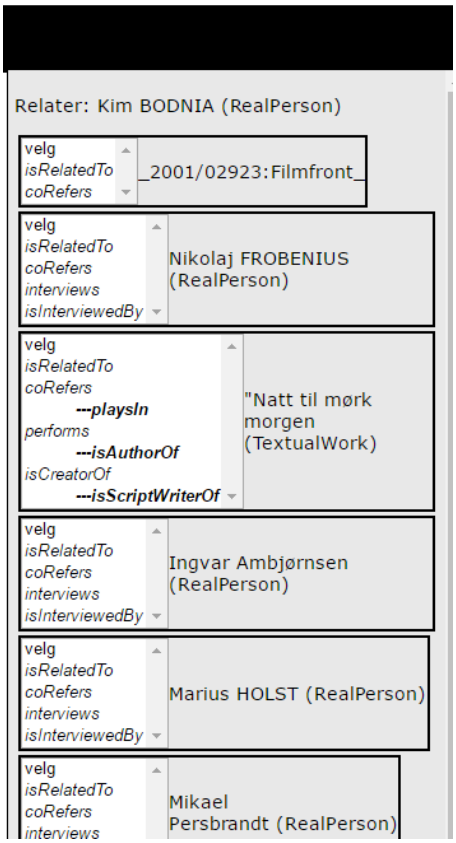
I behandling av entiteter ligger knappene for å lagre/endre/slette i menyen til høyre, noe ingen deltakere hadde problemer med å finne, men for å slette relasjoner må man scrolle ned til under tekstposten, hvor alle relasjonene ligger i en liste (figur 33). Dette forvirret de fleste brukerene, og gjorde som nevnt tidligere at aktiviteten mest sannsynlig ble vurdert som urettferdig vanskelig. Som en deltaker sa: «Det er jo superlett når man vet hva man skal!». En annen kilde til forvirring i forbindelse med relasjonsannoteringen var at knappen for å lagre er grå og har standardteksten «submit» eller «send», i motsetning til i entitetsmenyen hvor den er grønn med teksten «save». En deltaker sa rett ut «jeg synes det er rart at man ikke sletter relasjoner på samme måte som man sletter annotasjoner (sic)».

Subjekt	Relasjon	Objekt	Slett
Øyenstikker (Movie)	hasDirector	Marius HOLST (RealPerson)	x
Øyenstikker	coRefers	Øyenstikker (Movie)	x
Mikael Persbra	playsIn	Øyenstikker (Movie)	x

Figur 33: Skjermbilde av en liste med relasjoner

I relasjonsannoteringen er det slik at når man skal lage relasjoner til en valgt entitet, dukker alle de annoterte entitetene opp i menyen til høyre. Dette er for å kunne annotere relasjoner mer effektivt, men det virket i dette tilfellet mot sin hensikt: flere deltakere var veldig usikre på om man måtte velge noe i alle boksene som dukker opp. Det som i teorien vil spare mye tid for eksperter, vil for ikke-eksperter presentere for mye informasjon på en gang, og bare

være en kilde til forvirring. Under viser figur 34 et eksempel på hvordan en relasjonsmeny kan se ut i en post med mange annoterte entiteter.



Figur 34: Eksempel på skjermbilde ved relasjonsannotering

Også i denne testrunden fikk jeg inntrykk av at deltakerne var usikre på om de hadde fullført oppgavene. I oppgaven for å slette annoteringer sa en bruker «Det er litt skummelt at jeg bare kan slette sånn uten videre, kanskje?». Det ble også etterlyst tilbakemelding fra verktøyet om at det arbeidet under oppslag mot wikipedia.

En deltaker uttalte at «jeg tror verktøyet er greit å jobbe med hvis man har kjennskap til det fra før, men jeg synes enkelte ting er litt "gjemt" eller ulogisk plassert. Jeg tenker det ville vært enklere hvis jeg hadde en bruksanvisning, eller om jeg hadde fått et kurs før bruk. Dessuten er det litt tregt», og dette speiler inntrykket jeg fikk av de andre deltakerne i varierende grad.

4.2.5 Oppsummering av funn og anbefalinger etter andre testrunde

Funnene i den andre testen samsvarer på mange måter med funnene i den første testen: resultatene indikerte at annotører med rimelig datakompetanse kan ha høy suksessrate uten å være domeneeksperter, og vi ser at deltakerne foretrekker og forventer at grensesnittet fungerer som tradisjonelle applikasjoner. Jeg fant heller ikke her noen klar sammenheng

mellom deltakernes opplevde vanskelighetsgrad og deres interesse i å delta i videre aktiviteter av samme type.

Også her sank deltakernes *opplevde* vanskelighetsgrad noe utover i testen, men siden de opplevde verktøyet som lite intuitivt, *forventet* de at oppgavene skulle være vanskelige. Vi kan derfor fremdeles anta at verktøyet har mye å hente når det kommer til hvor lett det er å lære. Selv om det ble ikke var intuitivt for deltakerne, mente de fleste at det var både enkelt å bruke og effektivt etter å ha gjennomført en oppgave eller to av samme type.

Det var relativt få semantisk meningsløse feil i resultatet av annoteringene i denne runden også. Vi har fremdeles noe å hente med tanke på integrerte retningslinjer, da forsøket med hjelpetekst i mouseover viste seg å være særdeles lite effektivt, som nevnt i 4.2.2.

Igjen er det fare for at verktøyet ble vurdert som vanskeligere enn nødvendig både fordi brukerne undervurderte egne evner, og fordi oppgavene var utformet på en slik måte at de avsluttet flere av oppgavene frustrerte. For eksempel tror jeg at oppgave 10, som beskrevet i 4.2.3, ville blitt vurdert mer fordelaktig dersom de hadde blitt bedt om å slette flere relasjoner.

Hovedproblemområdene for verktøyet slik det fremstår nå er derfor inkonsekvens: bruken av dobbelt-/enkeltklikking er ikke intuitivt for brukeren, og deltakerne forventet å bli presentert med lignende menyvalg i de to typene annoteringer, men ble frustrerte når de ikke fant det de lette etter. En deltaker sa «det var bare sykt irriterende at jeg ikke skjønte hva jeg skulle gjøre av meg selv» i forbindelse med sletting av relasjoner.

Også i denne testen fikk jeg inntrykk av at det var en bratt læringskurve, men at verktøyet er simpelt etter innledende bruk.

Tabell 8 viser en oppsummering av de konkrete brukskvalitetsproblemene som ble avdekket i den andre testen. Også her kategorisert etter Burghardts bruksvalitetsmønstre (2014) og rangert etter alvorlighetsgradene beskrevet i 3.3.1.1.

Problem, kategori og alvorlighetsgrad	**Beskrivelse**	**Forslag til løsninger**
Problem: Inkonsekvent bruk av klikk/dobbeltklikk **Kategori**: Generelt grensesnitt: tilvenning **Alvorlighetsgrad**: 4 - Høy	Brukere forstår ikke intuitivt at det kreves dobbeltklikk på relasjonsikonet for å åpne menyen for relasjonsannotering	Endre til enkeltklikk, eventuelt en godt synlig hjelpetekst

Problem: Markert tekst forsvinner ikke ved «cancel» eller ESC **Kategori**: Generelt grensesnitt: mulighet til å angre **Alvorlighetsgrad**: 3 - Høy	Markert tekst forblir farget selv når bruker klikker «cancel» eller ESC. Dette utgjør ikke en systemfeil, men det er meget forvirrende for bruker	Sørg for at fargen forsvinner fra markert tekst når bruker angrer eller går tilbake
Problem: Høy læringskurve **Kategori**: Generelt grensesnitt: trygg utforskning **Alvorlighetsgrad**: 3 - Medium	Verktøyet er ikke intuitivt ved førstegangsbruk, men oppleves som enkelt etterhvert	Innfør en opplærings-/øvelsessekvens (tutorial) i begynnelsen av første økt, hvor brukeren får mulighet til å utforske systemet og opprette en annotering uten konsekvenser Trinnvis bruksanvisning Videoveiledning
Problem: Inkonsekvent plassering av mulighet for sletting **Kategori**: Generelt grensesnitt: tilvenning **Alvorlighetsgrad**: 3 - Medium	Brukere forventer knapp for sletting av relasjoner i menyen til høyre, i likhet med entiteter	Legg til slette-knapp i menyen til høyre
Problem: Ordlyd i knapp for å lagre relasjon **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Knappen for å lage/oppdatere leser «submit» eller «send», brukere forventer «lagre» i likhet med entitetsannotering	Endre ordlyd til «lagre»
Problem: Inntrykk av fryst system under Wikipedia-oppslag **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Bruker tror verktøyet har fryst når det bruker tid på oppslag mot Wikipedia	Tilbakemelding om at systemet arbeider
Problem: Manglende tilbakemelding på om en annotering er lagret korrekt **Kategori**: Generelt grensesnitt: hjelp for domenespesifikke funksjoner **Alvorlighetsgrad**: 3 - Medium	Brukere uttrykte usikkerhet rundt om en annotering var utført korrekt, og om resultatet deres var lagret	Tilbakemelding om gjennomført lagring i systemet
Problem: Forvirring rundt opprettelse av anker **Kategori**: Annoteringsprosessen: opprett anker **Alvorlighetsgrad**: 3 - Medium	Å starte markering av anker midt i ordet/frasen virker fremmed for brukeren ved første gangs bruk	-

Problem: Forvirring rundt innganger til funksjoner relatert til relasjoner **Kategori**: Generelt grensesnitt **Alvorlighetsgrad**: 3 - Medium	Brukerne forventer å kunne slette relasjoner i menyen til høyre, og å kunne endre dem i listen nederst på siden Brukere forventer også å finne en knapp for relasjonsannotering i nærheten av knappen for ekvivalensrelasjoner	Vurder å gjøre det mulig å behandle relasjoner i menyen til høyre Vurder å plassere en knapp for relasjonsannotering i menyen for valgt entitet, i nærheten av «coRef»
Problem: Hjelpetekster **Kategori**: Generelt grensesnitt: hjelp for domenespesifikke funksjoner/ utilstrekkelig dokumentasjon **Alvorlighetsgrad**: 3 - Medium	Brukere ser ikke hjelpetekstene i systemet	Gjør hjelpeteksten synligere, for eksempel ved et ikon for verktøytips
Problem: Overveldende lang post **Kategori**: Visning av primærdata **Alvorlighetsgrad**: 4 - Lav	Brukere føler posten som skal annoteres er lang	Gjør det mulig å endre mengde tekst som vises
Problem: Bruker vil åpne Wikipedia-URI **Kategori**: Ønsket funksjon **Alvorlighetsgrad**: 4 - Lav	Bruker ønsker å åpne Wikipedia-lenker for å 1) forsikre seg om at det er korrekt URI, eller 2) lese mer av ren interesse	Gjør lenken klikkbar og åpne siden i ny fane
Problem: Lite tiltalende grensesnitt **Kategori**: Ønsket funksjon **Alvorlighetsgrad**: 4 - Lav	Bruker har inntrykk av at grensesnittet er «gammeldags»	-

Tabell 9: Oppsummering av brukskvalitetsproblemer etter første test

5. Konklusjon og videre arbeid

Formålet med denne masteroppgaven var å foreta en formativ evaluering av det grafiske brukergrensesnittet til annoteringsverktøyet TORCH. Jeg ønsket å få et innblikk i brukernes opplevelse av verktøyet og identifisere eventuelle brukskvalitetsproblemer. Undersøkelsene var ledd i en prosess hvor det endelige målet er at verktøyets grensesnitt skal ha tilstrekkelig høy brukskvalitet og være intuitivt nok til at hele eller deler av annoteringsprosessen kan crowdsources og fremdeles resultere i annoteringer av god kvalitet

Ved første øyekast kan det virke som verktøyet er i dårligere forfatning etter andre test fordi jeg fant en rekke flere og alvorligere problemer enn ved første test, men ved direkte sammenligning kan vi se at en mange av problemene jeg avdekket i den første testen er eliminert og at hovedvekten av problemene jeg fant i den andre testen ble avdekket i

forbindelse med relasjonsannotering. Ved å se på deltakernes opplevelse og tilbakemeldinger rundt entitetsannoteringen alene kan vi se en indikasjon på at formative brukertester kan ha en positiv innvirkning på grensesnittet.

En rekke av problemene jeg avdekket i denne oppgaven er designfeil vi kan rette på, men funnene mine indikerer også at det ligger et mer abstrakt problem i at læringskurven er veldig bratt. «Det er jo superlett når man vet hva man skal», som en av deltakerne sa, så utfordringen ligger i å informere brukerne om hva det er de må gjøre første gangen de bruker verktøyet på en god måte. For eksempel kan jeg se for meg at en trinnvis brukermanual, en videoveiledning, eller en innledende øvelse hvor brukeren kan utforske systemet og annotere uten frykt for å gjøre noe feil eller «ødelegge noe» vil ha positiv innvirkning på brukernes opplevde vanskelighetsgrad.

Selve resultatet av annoteringene var godt, noe som samsvarer med antagelsen min om at personer uten nevneverdig domenekunnskap kan prestere på høyde med eksperter, og med annen forskning på feltet. Feilene som dukket opp i denne undersøkelsen er i hovedsak feil som i teorien kan unngås dersom retningslinjene er tydelige og lettere tilgjengelige, for eksempel med informasjon-ikoner i stedet for mouseover-tekster i selve verktøyet, eller videoveiledninger som gjør at brukerne ikke behøver å forholde seg til mye tekst.

Det var mer forvirring og feil rundt annotering av relasjoner enn av entiteter, noe som indikerer at antagelsen om at entitetsannotering er mer aktuelt å crowdsource kan stemme. Slik verktøyet fremstår i dag er det mulig at relasjonsannotering fremstår som for komplisert, og lett kan bli overveldende for en ikke-ekspert.

Som forslag til videre forskning ser jeg behovet for å oppdatere grensesnittet og gjennomføre ytterligere brukertester, men også å gjennomføre tester i større skala og se på resultatet av annoteringene. For eksempel ved å gjennomføre en kvantitativ undersøkelse med annotering gjennom crowdsourcing, og måle samsvaret mellom det crowdsourcede resultatet og en gullstandard ved å regne ut F-scoren for forskjellige typer annoteringsaktiviteter, som beskrevet i blant annet Zhai et al. (2013). Eventuelt kunne det vært interessant å undersøke hvilken type feil får dersom man ber testdeltakere annotere hele poster i stedet for å gi dem spesifikke oppgaver slik jeg gjorde i mine undersøkelser. Det ville også vært interessant å gå dypere i brukernes holdninger til annotering, for eksempel ved dybdeintervju.

Litteratur

Alshamari, M., & Mayhew, P. (2009). Technical Review: Current Issues of Usability Testing. *IETE Technical Review*, *26*(6), 402–406. doi:10.4103/0256-4602.57825

Annotasjon. (2009). I *Store norske leksikon*. Hentet 20. juni 2017 fra http://snl.no/annotasjon

Arlov, L. (1999). *GUI-guiden II: brukervennlighet for Windows og Web*. Oslo: IDG Norge Books.

Bontcheva, K., Derczynski, L., & Roberts, I. (under utgivelse). Crowdsourcing Named Entity Recognition and Entity Linking Corpora. I J. Pustejovsky & N. Ide (Red.), *Handbook of Linguistic Annotation*. Heidelberg: Springer. Hentet fra http://www.derczynski.com/sheffield/papers/chapter_crowdsourcing.pdf

Brooke, J. (1996). SUS-A quick and dirty usability scale. I P. Jordan, I. L. McClelland & B. Weerdmeester (Red.), *Usability Evaluation In Industry* (s. 189-194). London: Taylor & Francis.

Burghardt, M. (2014). *Engineering annotation usability - Toward usability patterns for linguistic annotation tools* (Doktorgradsavhandling, Universität Regensburg). Hentet 20. juni 2017 fra http://epub.uni-regensburg.de/30768/

Carletta, J. & Isard, A. (1999), The MATE Annotation Workbench: User Requirements. I *Proceedings of the ACL Workshop: Towards Standards and Tools for Discourse* Tagging, University of Maryland, USA (s. 11-17). Hentet 20. juni 2017 fra http://aclweb.org/anthology/W/W99/W99-0302.pdf

Crowdsourcing. (udatert). I *Oxford Dictionaries*. Hentet 20. juni 2017 fra https://en.oxforddictionaries.com/definition/crowdsourcing

Den Norske Dataforening. (2005). *Usability på norsk*. Hentet 20. juni 2017 fra http://www.dataforeningen.no/usability-paaring-norsk.4599321-134135.html

Dillon, A. (2001). The evaluation of software usability. I W. Karwowski (Red.) *Encyclopedia of human factors and ergonomics* (s. 1110-1112). London: Taylor and Francis.

Dix, A., Finlay, J. E., Abowd, G. D., & Beale, R. (2003). *Human-Computer Interaction* (3. utg). Harlow: Pearson.

Dvergsdal, H. (2015). Crowdsourcing. I *Store norske leksikon*. Hentet 20. juni 2017 fra http://snl.no/crowdsourcing

Faulkner, L. (2003). Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments, & Computers*, *35*(3), 379–383. doi: 10.3758/BF03195514

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., & Dredze, M. (2010). Annotating named entities in Twitter data with crowdsourcing. I *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, California* (s. 80–88). Hentet 20. juni 2017 fra http://dl.acm.org/citation.cfm?id=1866709

Good, B. M., Nanis, M., Wu, C., & Su, A. I. (2015). Microtask crowdsourcing for disease mention annotation in pubmed abstracts. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing, Kohala Coast, Hawaii, USA*, (s. 282–293). doi: 10.1142/9789814644730_0028

Guidelines for annotating NRK SIFT content fields. (2014). [Oslo: Høgskolen i Oslo og Akershus]. Upublisert.

Hinze, A., Heese, R., Luczak-Rösch, M., & Paschke, A. (2012). Semantic Enrichment by Non-experts: Usability of Manual Annotation Tools. I P. Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., … Blomqvist, E. (Red.), *The Semantic Web – ISWC 2012* (s. 165–181). doi: 10.1007/978-3-642-35176-1_11

Hoff, K, & Preminger, M. (2015). Usability testing of an annotation tool in a cultural heritage context. I E. Garoufallou, R. Hartley & P. Gaitanou (Red.), *Metadata and Semantics Research, MTSR 2015, Communications in Computer and Information Science, vol 544* (s. 237-248). doi:10.1007/978-3-319-24129-6_21

International Organization for Standardization. (1998). *Ergonomic requirements for office work with visual display terminals (VDTs) — Part 11: Guidance on usability* (ISO 9241-11:1998). Hentet 20. juni 2017 fra https://www.iso.org/obp/ui/#iso:std:iso:9241:-11

International Organization for Standardization. (2010). *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems* (ISO 9241-210:2010). Hentet 20. juni 2017 fra https://www.iso.org/obp/ui/#iso:std:iso:9241:-210

International Organization for Standardization. (udatert). *Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts* (ISO/DIS 9241-11.2). Hentet 20. juni 2017 fra https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:dis:ed-2:v2:en

John, B. E., & Marks, S. J. (1997). Tracking the effectiveness of usability evaluation methods. *Behaviour & Information Technology*, *16*(4), 188–202. doi:10.1080/014492997119789

Jónsdóttir, A. B. (2003). *ARNER, what kind of name is that?: an automatic rule-based named entity recognizer for Norwegian* (Masteroppgave). Universitetet i Oslo, Oslo.

Lewis, J. R., & Sauro, J. (2009). The Factor Structure of the System Usability Scale. I M. Kurosu (Red.) *Human Centered Design, HCD 2009, Lecture Notes in Computer Science, vol 5619.* (s. 94–103). Berlin: Springer. doi: 10.1007/978-3-642-02806-9_12

Marrero, M., Urbano, J., Sánchez-Cuadrado, S., Morato, J., & Gómez-Berbís, J. M. (2013). Named Entity Recognition: Fallacies, challenges and opportunities. *Computer Standards & Interfaces*, *35*(5), 482–489. doi:10.1016/j.csi.2012.09.004

McEnery, T., & Hardie, A. (2011). *Corpus Linguistics: Method, Theory and Practice*. Cambridge: Cambridge University Press.

Mehlenbacher, B. (1993). Software Usability: Choosing Appropriate Methods for Evaluating Online Systems and Documentation. I *Proceedings of the 11th Annual International Conference on Systems Documentation, Ontario, Canada* (s. 209–222). doi: 10.1145/166025.166083

Nielsen, J. (1993). *Usability engineering*. Boston: Academic Press.

Nielsen, J. (1995). *Severity Ratings for Usability Problems*. Hentet 20. juni 2017 fra https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/

Nielsen, J. (2000). *Why You Only Need to Test with 5 Users*. Hentet 20. juni 2017 fra https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/

Petrillo, M., & Baycroft, J. (2010). *Introduction to manual annotation. Fairview Research.* Hentet 20. juni 2017 fra https://gate.ac.uk/teamware/man-ann-intro.pdf

Pickard, A. J. (2007). *Research methods in information*. London: Facet.

Pratchett, T. (1990). *Moving Pictures*. London: Corgi books.

Preminger, M. (2015). TORCH-prosjektet. *Bok og bibliotek*, (2), 58-60.

Pustejovsky, J., & Stubbs, A. (2013). *Natural language annotation for machine learning*. Sebastopol.: O'Reilly Media.

Reidsma, D., Hofs, D. H., & Jovanović, N. (2005). Designing Focused and Efficient Annotation Tools. I L. P. J. J. Noldus, F. Grieco, L. W. S. Loijens, & P. H. Zimmerman (Red.) *Proceedings of Measuring Behavior 2005, the 5th International Conference on Methods and Techniques in Behavioral Research, Wageningen, Nederland* (s. 149–152). Hentet 20. juni 2017 fra http://doc.utwente.nl/65561/

Rubin, J., & Chisnell, D. (2008). *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests* (2. utg.). Indianapolis: Wiley.

Scholtz, J. (2004). Usability Evaluation. I Bainbridge, W. S. (Red.), *Berkshire Encyclopedia of Human-computer Interaction* (s. 750–757). USA: Berkshire Publishing Group LLC.

Snow, R., O'Connor, B., Jurafsky, D., & Ng, A. Y. (2008). Cheap and Fast—but is It Good?: Evaluating Non-expert Annotations for Natural Language Tasks. I *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (s. 254–263). Hentet 20 juni 2017 fra http://dl.acm.org/citation.cfm?id=1613715.1613751

Tallerås K., Massey D., Husevåg A.-S.R., Preminger M. & Pharo N. (2014) Evaluating (Linked) Metadata Transformations Across Cultural Heritage Domains. I S. Closs, R. Studer, E. Garoufallou, & M.-A. Sicilia (Red.), *Metadata and Semantics Research, MTSR 2014, Communications in Computer and Information Science, vol 478* (s. 250-261). doi:10.1007/978-3-319-13674-5_24

Toftøy-Andersen, E., & Wold, J. G. (2011). *Praktisk brukertesting*. Oslo: Cappelen Damm akademisk.

Tullis, T., & Albert, B. (2013). *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Amsterdam: Morgan Kaufmann.

Vikøren, B. M. (2012). Outsourcing. I *Store norske leksikon*. Hentet 20. juni 2017 fra http://snl.no/outsourcing

Zhai, H., Lingren, T., Deleger, L., Li, Q., Kaiser, M., Stoutenborough, L., & Solti, I. (2013). Web 2.0-based crowdsourcing for high-quality gold standard development in clinical natural language processing. *Journal of medical Internet research*, *15*(4), e73. doi: 10.2196/jmir.2426

Zhang, Z. (2013). *Named entity recognition: challenges in document annotation, gazetteer construction and disambiguation* (Doktorgradsavhandling, University of Sheffield). Hentet 20. juni 2017 fra http://etheses.whiterose.ac.uk/3866/

Samtykke

Testen du er i ferd med delta i er en del av TORCH-prosjektet (Transforming Organization and Retrieval in Cultural Heritage) som bygger på et samarbeid med NRK om analysering og bearbeidelser av metadata ved Høgskolen i Oslo og Akershus. Testen er utformet for å evaluere et annoteringsverktøy utviklet i forbindelse med dette prosjektet. Annoteringsverktøyet skal brukes til å utarbeide fasiter for mappinger, transformasjoner, automatiske uttrekk og lenking.

Jeg samtykker i å delta i datainnsamling til masteroppgaven (MBIB9000), utført av Karoline Hoff. Jeg forstår at deltakelse i denne brukskvalitetstesten er frivillig og at testen kan avbrytes når som helst, for eksempel hvis bekymringer eller ubehag oppstår.

Jeg tillater at min utførelse brukskvalitetstesten vil bli dokumentert og brukt som data i masteroppgaven til Karoline Hoff.

De data som brukes vil ikke stå i mitt navn og vil være anonymisert i selve avhandlingen.

Vennligst skriv under nedenfor for å indikere at du har lest og forstått informasjonen på dette skjemaet og at eventuelle spørsmål du matte ha om økten har blitt besvart.

Dato: _____

Underskrift: _____

Tusen takk!

Jeg setter pris på din deltakelse,

Karoline Hoff

Kjønn

Kvinne Mann

Alder

Under 20 21-25 26-30 31-35 Over 36

Utdanning

Hva er den høyeste akademiske graden du har fullført?

VGS
Høgskolekandidat
Bachelorgrad
Mastergrad
Doktorgrad

Eventuelt fagområde:

Erfaring

Hvor godt leser og forstår du norsk?

Godt Veldig godt Utmerket

Hvor godt skriver du norsk?

Godt Veldig godt Utmerket

Har du tidligere erfaring med å evaluere grensesnitt?

Ja Nei

Hvis ja, hvor mange ganger har du vært deltakende i en slik prosess?

1 2 3 4 5 Mer enn fem ganger

Har du tidligere erfaring med emner relatert til brukervennlighet/usability?

Ja Nei

Har du erfaring med annotering eller annoteringsverktøy?

Ja Nei

Hvis ja, kan du kort forklare hvilken erfaring du har?

Kunnskap

På en skala fra 1 til 5, hvor 1 er «ingen kunnskap» og 5 er «veldig kunnskapsrik», hvor godt vil du si du behersker/kjenner til følgende fenomener/aktiviteter:

Tekstbehandling (Word, OpenOffice)

1 2 3 4 5

Emneord, tagging

1 2 3 4 5

Brukervennlighet

1 2 3 4 5

Semantisk web

1 2 3 4 5

Oppgaver

For hver av oppgavene, marker sifferet som best beskriver dine tanker om oppgaven.

Oppgave 1: Annoter to valgfrie personnavn

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 2: Annoter to valgfrie verk

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 3: Annoter et valgfritt sted

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 4: Lag en ekvivalensrelasjon (coRef) mellom to entiteter

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 5: Endre opplysningene i en annotert entitet

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 6: Slett en valgfri entitet

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Opplevd vanskelighetsgrad og interesse for lignende aktiviteter

For hver del, kan du vurdere følgende på en skala fra 1 til 5:

- Vanskelighetsgrad, hvor vanskelig følte du aktivitetene var? (1 = veldig lett, 5 = veldig vanskelig)
- Grad av interesse, hvor interessert er du i å delta i prosjekter med lignende aktiviteter senere? (1 = ikke interessert, 5 = veldig interessert)

Aktivitet	Vanskelighetsgrad	Interesse
Annotere entiteter		
Endre annoteringer		
Opprette ekvivalensrelasjoner		
Bruke verktøyet generelt		

Samtykke

Testen du er i ferd med delta i er en del av TORCH-prosjektet (Transforming Organization and Retrieval in Cultural Heritage) som bygger på et samarbeid med NRK om analysering og bearbeidelser av metadata ved Høgskolen i Oslo og Akershus. Testen er utformet for å evaluere et annoteringsverktøy utviklet i forbindelse med dette prosjektet. Annoteringsverktøyet skal brukes til å utarbeide fasiter for mappinger, transformasjoner, automatiske uttrekk og lenking.

Jeg samtykker i å delta i datainnsamling til masteroppgaven (MBIB9000), utført av Karoline Hoff. Jeg forstår at deltakelse i denne brukskvalitetstesten er frivillig og at testen kan avbrytes når som helst, for eksempel hvis bekymringer eller ubehag oppstår.

Jeg tillater at min utførelse brukskvalitetstesten vil bli dokumentert og brukt som data i masteroppgaven til Karoline Hoff.

De data som brukes vil ikke stå i mitt navn og vil være anonymisert i selve avhandlingen.

Vennligst skriv under nedenfor for å indikere at du har lest og forstått informasjonen på dette skjemaet og at eventuelle spørsmål du matte ha om økten har blitt besvart.

Dato: _____

Underskrift: _____

Tusen takk!

Jeg setter pris på din deltakelse,

Karoline Hoff

Kjønn

Kvinne Mann

Alder

Under 20 21-25 26-30 31-35 Over 36

Utdanning

Hva er den høyeste akademiske graden du har fullført?

VGS

Høgskolekandidat

Bachelorgrad

Mastergrad

Doktorgrad

Eventuelt fagområde:

Erfaring

Hvor godt leser og forstår du norsk?

Godt Veldig godt Utmerket

Hvor godt skriver du norsk?

Godt Veldig godt Utmerket

Har du tidligere erfaring med å evaluere grensesnitt?

Ja Nei

Hvis ja, hvor mange ganger har du vært deltakende i en slik prosess?

1 2 3 4 5 Mer enn fem ganger

Har du tidligere erfaring med emner relatert til brukervennlighet/usability?

Ja Nei

Har du erfaring med annotering eller annoteringsverktøy?

Ja Nei

Hvis ja, kan du kort forklare hvilken erfaring du har?

Kunnskap

På en skala fra 1 til 5, hvor 1 er «ingen kunnskap» og 5 er «veldig kunnskapsrik», hvor godt vil du si du behersker/kjenner til følgende fenomener/aktiviteter:

Tekstbehandling (Word, OpenOffice)

1 2 3 4 5

Emneord, tagging

1 2 3 4 5

Brukervennlighet

1 2 3 4 5

Semantisk web

1 2 3 4 5

Oppgaver

For hver av oppgavene, marker sifferet som best beskriver dine tanker om oppgaven.

Oppgave 1: Annoter et valgfritt personnavn

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 2: Annoter et valgfritt verk

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel				Veldig vanskelig
1	2	3	4	5

Hva var vanskelig?

Oppgave 3: Annoter et valgfritt sted

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 4: Endre opplysningene I en annotert entitet

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 5: Slett en valgfri entitet

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 6: Lag en ekvivalensrelasjon mellom to entiteter

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 7: Annoter to valgfrie, men relaterte entiteter, og registrer relasjonen dem i mellom

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 8: Gjenta oppgave 7, men med to nye entiteter

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 9: Endre en relasjon

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Oppgave 10: Slett en relasjon

Før oppgaven: Hvor vanskelig forventer du at oppgaven vil være?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Etter oppgaven: Hvor vanskelig opplevde du oppgaven?

Veldig enkel Veldig vanskelig

1 2 3 4 5

Hva var vanskelig?

Opplevd vanskelighetsgrad og interesse for lignende aktiviteter

For hver del, kan du vurdere følgende på en skala fra 1 til 5:

- Vanskelighetsgrad, hvor vanskelig følte du aktivitetene var? (1 = veldig lett, 5 = veldig vanskelig)
- Grad av interesse, hvor interessert er du i å delta i prosjekter med lignende aktiviteter senere? (1 = ikke interessert, 5 = veldig interessert)

Aktivitet	Vanskelighetsgrad	Interesse
Annotere entiteter		
Opprette ekvivalensrelasjoner		
Annotere relasjoner		
Bruke verktøyet generelt		

Vedlegg 3: Intervjuguide

1. Hvilket inntrykk har du av verktøyet?

    a. Hva likte du med verktøyet?

    b. Hva likte du ikke med verktøyet?

2. Hva synes du om grensesnittet?

3. Hva ville du endret? Har du forslag til forbedringer?

Vedlegg 4: System Usability Scale

	Sterkt uenig				Sterkt enig
1. Jeg kunne tenkt meg å bruke dette systemet ofte.	1	2	3	4	5
2. Jeg synes systemet var unødvendig komplisert.	1	2	3	4	5
3. Jeg synes systemet var lett å bruke.	1	2	3	4	5
4. Jeg tror jeg vil måtte trenge hjelp fra en person med teknisk kunnskap for å kunne bruke dette systemet.	1	2	3	4	5
5. Jeg syntes at de forskjellige delene av systemet hang godt sammen.	1	2	3	4	5
6. Jeg syntes det var for mye inkonsistens i systemet. (Det virket «ulogisk»)	1	2	3	4	5
7. Jeg vil anta at folk flest kan lære seg dette systemet veldig raskt.	1	2	3	4	5
8. Jeg synes systemet var veldig vanskelig å bruke.	1	2	3	4	5
9. Jeg følte meg sikker da jeg brukte systemet.	1	2	3	4	5
10. Jeg trenger å lære meg mye før jeg kan komme i gang med å bruke dette systemet på egen hånd	1	2	3	4	5

SUS, Norsk versjon ved Dag Svanæs, NTNU 2006

Vedlegg 5: SIFT-post for Filmfront

TORCH annoteringsverktøy [Hjelp | Velg nytt dokument | Log ut (Karoline-)]

taken 1497982805 . (dokument 8) *_2001/02923:Filmfront_*

K791 . 4 - Film (eo)
-
Filmfront . Ved Mikal Olsen Lerøen
-
1 . Kjenning og åpning
Introduksjon . Tema : Historien om hvordan den norske filmen Øyenstikker ble skapt . Det er historien om en film som produsenten nesten ikke hadde penger til , manusforfatteren nesten ikke hadde tid til å skrive og skuespillerne nesten ikke hadde tid til å filme . Og sist men ikke minst , det er en film som regissøren nesten ikke hadde tid til å planlegge . Plutselig måtte han bare lage den . 111"
2 . Øyenstikker . Programleder Mikal Olsen Lerøen har intervjuet de involverte under innspillingen av filmen
Intervju med den danske skuespilleren Kim BODNIA (mv) , som har en hovedrolle i Øyenstikker . Om hvordan filmen ble til . (e . 111") . Lydkutt fra innspillingen av filmen
Intervju med Nikolaj FROBENIUS (mv) , som har skrevet manus til filmen etter romanen "Natt til mørk morgen av Ingvar Ambjørnsen . (e . 443")
Regissør Marius HOLST (mv) intervjues om filmen . Ikke så opptatt av at dette nødvendigvis er en dogmefilm . Beskriver arbeidsmetoden . Om brannen under innspillingen av filmen . (e . 555") . Den svenske skuespilleren Mikael Persbrandt spiller en annen hovedrolle i filmen . Filmsettet måtte ligge like utenfor Stockholm fordi Persbrandt spiller Don Juan om kvelden på Dramaten inne i Stockholm . Persbrandt er forlovet med en annen hovedrolleinnehaver i Øyenstikker , Maria Bonnievie
Intervju med Mikael PERSBRANDT (mv) og Maria BONNEVIE (mv) (på svensk) (e 728") om den hektiske innspillingen . Kort samtale med filmes produsent Sigve ENDRESEN (mv) . Om filmsettet . (e . 850") . Samtale med Maria BONNEVIE (mv) (på norsk) , om det å jobbe med skuespillere som Bodnia og Persbrandt . (e . 1054") 1049" (e . 111")
3 . Filmfront inviterer lytteren inn i det aller helligste hos regissør Marius Holst , et lite rom i kjelleren i den fornemme villaen til New Deal reklamebyrå i Wergelandsvei 21 i Oslo . Samtale med Marius HOLST (mv) om klippingen og ferdigstillingen av filmen . 1110" (e . 1200")
4 . Musikken i filmen er laget av Magne Furuholmen og Kjetil Bjerkestrand
Intervju med Magne FURUHOLMEN (mv) om bruken av musikk i filmen . Korte kutt med musikk fra Øyenstikker , kjenningen "Dragonfly" med Magne Furuholmen . 455" (e 2315")
5 . Kutt med Maria BONNEVIE (mv) om hvordan hun har det før premieren . 050" (e . 2830")
Sluttkjenning
-
*F sist-endret
20100616