Emnetilknytting av ustrukturert sykepleiedokumentasjon i elektronisk pasientjournalsystem gjennom tradisjonelle tekstgjenfinningsteknikker, klassifisering og rammeverk

Masteroppgave ved

Informasjonsforvaltningsgruppen,
Institutt for Datateknikk og Informasjonsvitenskap,
Fakultet for Informasjonsteknologi, Matematikk og Elektronikk,
Norges teknisk-naturvitenskapelige universitet

Asbjørn Eidevik Østby,
Trondheim, 15.6.2005

Forord

Som sykepleier med stor interesse for databehandling, falt det meg naturlig å vie arbeidet i informatikk mastergradstudiet mitt til et problemområde der jeg kunne bruke min kunnskap fra begge fagfelt.

Det er mitt håp at denne Masteroppgaven kan bidra i skapelsen av verktøy og teknikker som kan forbedre hverdagen til sykepleiere med tanke på søk og navigering i sykepleiedokumentasjonen.

Det er mange som har vært villige til å hjelpe meg under arbeidet med denne Masteroppgaven. Jeg vil spesielt rette takk til følgende institusjoner og personer:

NTNU: Heri Ramampiaro (veileder), Øystein Nytrø, personal og medstudenter ved IDI, spesielt IF-gruppa.

Sabaklass representant: Karl Øyri.

HEMIT: Unni Ulltveit, Rut Naversen, Rolf Holte.

KITH: Jostein Ven, Torbjørn Nystadnes.

DIPS ASA: Kåre Flø, Liv Haugen.

Siemens (DocuLive EPR): Tore Gravdal.

Foruten disse så har mange personer med kunnskap om sykepleiedokumentasjon og elektroniske pasientjournaler bidratt ved å svare raskt på henvendelser og spørsmål. Dette har vært av stor betydning for å oppklare misforståelser. Ingen nevnt, ingen glemt.

Til slutt vil jeg rette en stor takk til min samboer Ragnhild Elisabet Ulfsnes som har hjulpet meg med den stilmessige redigeringen av oppgaven, og for å ha vært en god støttespiller gjennom hele prosessen.

Asbjørn Eidevik Østby,

Trondheim, våren 2005

Sammendrag

Oppgaven ser på et område som det i dag er implementert lite støtte for i eksisterende EPJ-systemer (Elektronisk Pasientjournal), knytting av emner til ustrukturert sykepleiedokumentasjon ved hjelp av tradisjonelle søketeknikker og rammeverk.

Søking og navigering er dårlig utviklet i mange EPJ-systemer, spesielt for ustrukturert dokumentasjon. Knytning av emner til denne typen dokumentasjon kan derfor lette sykepleiernes hverdag, og metoden som presenteres i denne oppgaven kan være et verktøy i denne sammenhengen.

Automatisk klassifisering er knyttet til mange usikkerhetsfaktorer. Disse usikkerhetsfaktorene forsøkes belyst ved vise til sykepleiedokumentasjonens natur og sprikende struktur, noe som vanskeliggjør en regelbasert tilnærming. Oppgaven argumenterer for at man kan tolerere feilkilder så lenge man er klar over at disse eksisterer, og at nytteverdien av å søke på emner fremfor enkelttermer er større enn problemene disse usikkerhetsfaktorene gir.

Bruk av eksisterende rammeverk fremfor egendefinerte klassifiseringssystemer er en naturlig innfallsvinkel til en slik problemstilling. Oppgaven forsøker å vise ett prinsipp, mer enn en favorisering av ett spesielt rammeverk. Likevel vil ett rammeverk, Sabaklass versjon 2.0N, bli mer omtalt enn andre på grunn av at det har vært benyttet i implementasjonen av en prototyp i forbindelse med oppgaven.

Oppgaven viser hvordan slik emnetilknytning kan foregå ved hjelp av en prototyp som er blitt utviklet for formålet. Prototypen benytter vektorrommodellen (VSM) sammen med en ordliste knyttet til rammeverket for å klassifisere ustrukturert sykepleiedokumentasjon. Valget av vektormodellen fremfor andre, tradisjonelle tekstgjenfinningsteknikker, er gjort ut ifra modellens egenskaper til å rangere delvise treff.

De konkrete resultatene av prototypingen kan være diffuse. Dette er i stor grad knyttet til konstruksjonen av ordlisten, og vanskeligheter knyttet til å skaffe en god testsamling. Likevel peker de foreløpige resultatene i favør for en slik søkemetodikk, fordi den kan benyttes sammen med eksisterende søkemetodikker, og den representerer en fordel ved ukjent dokumentasjon.

Summary

English title:

”Association of topics to unstructured nursing records in Electronic Patient Record-systems through traditional Information Retrieval techniques and nursing classification frameworks”

This Master Thesis approaches an area of research which has little implemented support in existing EPR (Electronic Patient Record) systems – topic association in unstructured nursing records. Search and navigation is little supported for such documentation in general in EPR-systems. Topic association through classic information retrieval techniques may therefore represent a tool which makes everyday use easier for nurses.

Automatic classification is connected to many elements of uncertainty. These elements are maybe even more present for nursing records due to the nature of nursing and the different structuring schemes (or lack there of) that exists. The thesis arguments that use of automatic classification may have its good use if those elements of uncertainty is taken into account, and that topic based searching is better than single term searching in some situations.

Using existing nursing frameworks and taxonomies is a natural choice over creating an own organisation scheme for such topics. The Thesis tries to show a principle more than favour one framework over another, but one framework is mentioned in particular in the thesis. The reason for this is that it is used in a prototype made for showing how topic association can be done. The framework is known as CCC (Clinical Care Classification) or previously as HHCC (Home Health Care Classification System).

The prototype uses VSM (vector space model) and a thesaurus related to CCC to make the association happen. The VSM is chosen over other classical information retrieval techniques because it can ranks hits, and give partial hits.

Absolute results may be difficult to evaluate due to the lack of good nursing records for testing in the prototype. The vector space model (VSM) represents a much tested and used method in digital libraries in general. The main difference this thesis presents, is the use of multiple searches based on a nursing classification framework, and a thesauri related to the framework. This thesaurus is nowhere near complete, and much work is required to make it so. Due to these factors, integration of the method in real EPR-systems is not advised yet.

The usability analysis points towards great potential for the method; it may be used with existing EPR-systems and frameworks, and represent an advantage when faced with unknown nursing records.

Definisjoner

I oppgaven blir det brukt mange uttrykk som kan være ukjente for leseren. For å hjelpe på dette, har jeg hentet begrepsavklaringer fra SfID's veileder [24].

Terminologi er en liste over begrep eller faguttrykk brukt innenfor et bestemt fagområde

Et **begrep** er ”*En tankemessig enhet dannet ved abstraksjon på grunnlag av egenskaper felles for en eller flere referenter.” (ISO CD 1087-1)*

Dvs. et begrep refererer til ting, fenomener, vesener som eksisterer i virkelighetens verden eller kun i vår tanke- og forestillingsverden.

En **term** er å forstå som et språklig uttrykk for et begrep og svarer til det norske ordet faguttrykk.

Nomenklatur er en fortegnelse over navn og spesialuttrykk innenfor et fag, en vitenskap og kan brukes synonymt med fagterminologi.

I en **klassifikasjon** er ulike begrep eller ting ordnet i forhold til hverandre i en struktur, klasser eller grupper som gjør det mulig å forstå sammenhengen mellom begrepene. Det finnes ulike prinsipper for å klassifisere og det er grovt sett to ulike måter å strukturere en klassifikasjon på; hierarkisk eller aksialt.

Taksonomi betyr en ordning av begreper eller klassifikasjon.

En kode er et forutavtalt system av ord, tall eller andre tegn.

Kodeverk er en samling av begreper eller en klassifikasjon hvor hver tilhørende begrepsdefinisjon er tilknyttet en unik kode innenfor samlingen eller klassifikasjonen.

Siden det i teksten kan oppstå misforståelser på tross av disse definisjonene, må jeg nevne disse:

En **term** kan i databehandlingssammenheng være resultatet av en prosess som finner ordstammen til et ord (stemming), og trenger ikke nødvendigvis være ett faguttrykk.

Ordet **klassifisere** har to betydninger. Den ene er å lage et klassifiseringssystem (kodeverk), den andre er å benytte dette systemet for å beskrive et objekt i forhold til et eksisterende klassifiseringssystem, og gi det koder eller beskrivende termer [25].

Nomenklatur: I et nomenklatur kan kodene kobles til et medisinsk begrep, og disse kan kombineres for å forme enda mer komplekse begreper. Dette kan gi en mengde muligheter for kodekombinasjoner [25].

Forskjell mellom taksonomi og nomenklatur: Merriam-Webster (2000) differensierer taksonomi *"the study of the general principles of scientific classification"* fra nomenklatur som er *"a system or set of terms or symbols, especially in a particular science, discipline or art"* [58].

I tillegg finnes det noen uttrykk som er brukt som ikke dekkes av veilederen:

Annotasjon: I denne oppgaven er lenking bruksområdet for annotasjon som er mest fremtredende. Annotasjon beskrives slik: *"[annotation] has been construed in many ways: as link making, as path building, as commentary, as marking in or around existing text, as a decentring of authority, as a record of reading and interpretation, or as community memory"* [57].

Invertert fil: representasjon av tekstfil der overflødige ord er fjernet, inneholder termer. Termene representeres bare en gang, men antallet forekomster og lokalisasjon registreres.

Morfem: Den minimale betydningsbærende enhet i språket, abstrakt enhet [53].

Ontologi: Læren om det eksisterende (ordenes mening). I databehandlingssammenheng er ontologier er en måte å beskrive meningene og relasjonene til en term.

Ord: samling av bokstaver som utgjør et meningsbærende eller kommuniserende enkeltelement.

Scope: Definisjonsområde

Stemming: prosess som forsøker å finne ordstammen til et ord (altså et token, og resultatet, en term, behøver nødvendigvis ikke være et faguttrykk)

Thesaurus: Kontrollert vokabular med semantiske relasjoner [55].

Token: her: ord

Vokabular: Ordliste over terminologien i et bestemt fagfelt [55].

Forkortelser

ANA - Amercian Nurses Assosiation

ANSI - American National Standards Institute

CAP - College of American Pathologists

CINAHL - Cumulative Index for Nursing and Allied Health Literature

CPRI - Computer-based Patient Record Institute's (CPRI's) Codes and Structures Work Group

DLS - Distributed Link Service

DTD - Document Type Definision

EPJ - Elektronisk Pasient Journal

HHA's - home health agencies

HHCC - Home Health Care Classification system, også kjent som Sabaklass og CCC

HISB - Healthcare Informatics Standards Board (HISB)

HL7 - Health Level Seven (standard for elektronisk utveksling av klinisk, administrativ og økonomisk informasjon mellom helseorienterte datasystem)

ICD - International Statistical Classification of Diseases and Related Health Problems (ICD-10)

ICN - International Council of Nurses

ICNP - International Classification of Nursing Practice

IRC - Internet Relay Chat

ISO - International Standards Organization Technical Committee (TC)

JCAHO - Joint Commission on Accreditation for Health Care Organization's

LOINC -Logical Observation Identifiers Names and Codes

NANDA - North American Nursing Diagnoses Association

NEL - Norsk Elektronisk Legehåndbok

NIC - Nursing Intervention Classification

NIDSEC - ANA's Nursing Information and Data Set Evaluation Center

NOC - Nursing Outcome Classification

NTNU - Norges teknisk-naturvitenskapelige universitet

OASIS - Outcome and Assessment Information Set

OCR - Optical Character Recognition

PPS - OASIS Prospective Payment System

RAP - Resident Assessment Protocols

RDF - Resource Description Framework

RSS - RDF Site Summary RSS (andre: Rich Site Summary, Really Simple Syndication)

RTF - Rich Text Format

SAX - Simple API for XML (bibliotek i programmeringsspråket Java (Sun Microsystems))

SNL - Standardized Nursing Languages

SNOMED RT - Systemized Nomenclature of Mecicine - Reference Terminology

SNOP - Systemized Nomenclature of Pathology

SOAP - Måten legejournaler tenkes oppbygd på: Subjectiv, Objective, Assessment, Plan

SOAP - Simple Object Access Protocol

tf-idf - term frequency-inverse document frequenzy

TREC - Text REtrieval Conference, holdes av National Institute of Standards

UMLS - Metathesaurus of Unified Medical Language System fra NLM (National Library of Medicine

UIO – Universitetet i Oslo

VSM - Vector Space Model

WHO - World Healt Organization

XML - eXtended Markup Language

Innholdsfortegnelse

1 Innledning

Oppgaven henvender seg i første rekke til lesere med bakgrunn innen informatikk og informasjonsforvaltning. I andre rekke er det et ønske at sykepleiere med god forståelse for informatikk og/eller sykepleiedokumentasjon i EPJ-systemer skal ha utbytte av å lese den. Spesielt kapitlene om bruksområder for emnetilknytning er tiltenkt en slik lesergruppe.

Denne adresseringen fører til tidvis mer inngående omtale der mer forklaring har vært nødvendig for å imøtekomme begge grupperingene.

1.1 Målsetning og problemspesifikasjon

Målsetningen for oppgaven er å knytte emner til ustrukturert tekst i sykepleiedokumentasjonen. Dette er interessant fordi det kan lette gjenfinning av slik informasjon for sykepleierne ved å kunne søke etter begreper fremfor enkeltord.

Med ustrukturert tekst menes tekst som ikke har blitt klassifisert av rammeverk eller annen form for klassifiserende strukturering i et elektronisk pasientjournalsystem. Med emner menes emner hentet fra et rammeverk for sykepleie. Med emnetilknytning menes søk basert på en predefinert ordliste. I å ”lette gjenfinning” ligger muligheten for benytte emnetilknytningen som utgangspunkt for søk og navigering.

1.2 Avgrensning

Fremtredende syn på sykepleiedokumentasjonen i EPJ (Elektronisk Pasient Journal) systemer i denne oppgaven heller mot EPJ som passivt oppbevaringssted (repository) mer enn et aktivt arbeidsverktøy for planlegging, vurdering, evaluering osv. Dette er fordi oppgaven tar for seg problemstillingen med å knytte emner fra allerede eksisterende, ustrukturert tekst til et rammeverk. Dette er et syn som er benyttet for å gjøre det enklere å fremstille budskapet til denne oppgaven. Oppgaven beveger seg tidvis inn på bruk av EPJ mer aktivt, spesielt i omtale av hva emnetilknyttingen kan benyttes til.

Synet på sykepleiedokumentasjonen (definisjonsområde, "scope"): Et dokument brukes som samlebegrep for en rapport, ett sammendrag, et brev eller lignende. Det vil si at en "rapport" ansees som å være et dokument, og den samlede sykepleiedokumentasjonen til en pasient ansees som dokumentsamling. Dette er gjort for å forenkle behandlingen av inndata, og siden ingen form for metadata benyttes som skiller de ulike dokumenttypene fra hverandre. Om definisjonsområdet snevrer inn eller utvider seg i omtale av de ulike teknikkene som nevnes i oppgaven, vil dette bli nevnt.

Det er i denne oppgaven utarbeidet en prototyp for emnetilknytning av slike rapporter. Selve prototypen er helt løsrevet fra alle EPJ-systemer og tar ikke hensyn til noen vanskeligheter de beskrevne operasjonene ville ha hatt i en virkelig setting. Slike problemer blir diskutert i denne oppgaven, men tekniske aspekter ved integrering i spesielle, eksisterende EPJ-system er utenfor rekkevidde for oppgaven.

1.3 Oversikt over oppgavens oppbygning

Først kommer en generell omtale av problemer knyttet til synet på at enkeltord kan bygge opp til kunnskap (dersom en ser rammeverkene som representasjon av kunnskap, og ikke informasjon), samt litt om problemer som er knyttet til sykepleiedokumentasjon til forskjell fra andre medisinske profesjoners dokumentasjon.

"State of the Art" viser til viktige teknologier og aspekter relatert til problemstillingen oppgaven tar opp: å knytte emner til sykepleiedokumentasjon fra et rammeverk automatisk.

- Systemer og teknikker for språkprosessering og klassifisering
- Systemer som inneholder sykepleiedokumentasjon, og hvordan de forholder seg til denne i dag.
- Rammeverk for sykepleiedokumentasjon og høyereliggende nomenklatur, samt litt om standardisering.

Videre følger en oversikt over klassiske gjenfinningsteknikker, og nærmere beskrivelse av dem som er brukt i oppgaven. Et innblikk i hva hyperlenking kan tilføre av funksjonalitet når man får knyttet emner til sykepleiedokumentasjon vil bli presentert.

Det resterende av oppgaven er en beskrivelse av hvordan teknikkene er implementert i prototypen, og problemstillinger knyttet direkte til denne. Videre kommer en evaluering av prototypingen, og kritikk av løsningen. Viktige elementer i evalueringen er spesielt kost/nytte forholdet løsningen representerer.

1.4 Bakgrunn

Det finnes EPJ-systemer (elektronisk pasientjournal) som benytter sykepleierammeverk til å standardisere innlegging slik at man kan enklere finne frem og bruke sykepleiedokumentasjonen mer aktivt. Men mye av dokumentasjonen som legges inn er fritekst uten struktur, og mye av dokumentasjonen er fra tiden før rammeverk ble innført for å strukturere dokumentasjonen.

Dagens EPJ-systemer har valgt forskjellige rammeverk og forskjellige løsninger med tanke på utforming av standarder for innlegging. Gjenfinning og fremstilling av funn er også forskjellig. Bruken av systemene er i høyeste grad forskjellig - ikke all dokumentering følger systemets standard ved innlegging. Sykepleierne har en utstrakt bruk av individuelt eller avdelingsspesifikt utviklet dokumenteringsstil - ofte råder et kaos med egne forkortelser, internt språk osv (bl.a. [16]). Dette er en situasjon som gjør det vanskelig å konstruere et datasystem som kan klassifisere tekst automatisk.

Dokumentasjonen brukes både som informasjonsformidling og arbeidsredskap av sykepleierne. Et eksempel på at sykepleiernes arbeidsform endrer seg, er at den tradisjonelle "rapport" kommer til å forsvinne og bli erstattet med såkalt "stille rapport" (som betyr at pleieren skal lese seg til informasjonen han/hun trenger). Dette stiller store krav til funksjonalitet til et elektronisk system.

For dokumentasjon som ikke har noen spesiell struktur knyttet til seg ved innleggelse, oppleves lesning som tidsorientert for brukeren (sykepleieren). Et tidsorientert (også kalt kronologisk eller journalorientert) perspektiv er som en pergamentrull eller "dorull" når en skal finne igjen aktuelle problemstillinger og angivelser i teksten. En må enten vite når noe skjedde, eller lese seg igjennom en større del av tekst der aktuell nedtegnelse finnes. For å applisere andre perspektiver (aktuelle er problemorientert og kildeorientert/prosessorientert), trenger en mer opplysninger, gjerne fra dokumentasjonens innhold. Det problemorienterte synet er interessant for denne oppgaven - altså at en vet hva man er interessert i, men ikke hvor eller når dette evt. er dokumentert i sykepleiedokumentasjonen. For å applisere et problemorientert syn på hele sykepleiedokumentasjonen, må en kunne klassifisere ustrukturert tekst (som er utgangspunktet jeg tar her). Dette kan benyttes til å knytte begrepene i teksten til en høyereliggende struktur, helst et kodeverk.

Gjenfinning av relevant informasjon i en spesifikk kontekst bør ikke være basert på at en må lese igjennom mengder av (for situasjonen) uvesentlig informasjon. Tradisjonelle gjenfinningsteknikker kan avhjelpe situasjonen sykepleierne står overfor - mye informasjon, svake søkeverktøy og delvis dårlig struktur på informasjonen. Å finne en søkemetode som kan begrepsfeste fri tekst tilfredsstillende med tanke på alle forvanskende elementene er ikke lett. En innfallsvinkel er å si at vi trenger noe som ikke er for rigid, men som kan gi brukbare resultater. Så kan innleggingen forbedres innen de rammene det spesifikke EPJ-systemet gir, og fremtidige systemer kan glede seg over å ha en bedre struktur å jobbe med.

Arkitekturen i elektroniske pasientjournaler kan variere fra system til system, men det er vanlig å ha en slags databasestruktur for å lagre informasjon, og benytte XML og stilskjema

for fremstilling av informasjonen. Jeg har valgt å abstrahere meg vekk fra denne arkitekturen, og benytte selve teksten som utgangspunkt for å lage et system som kan finne termer. Disse kan benyttes som grunnlag for å plassere teksten inn i ett rammeverk. Selv om det er utviklet mange klassifikasjonssystemer innen sykepleie, har ingen foreløpig oppnådd allmenn bruk slik som f. eks ICD har gjort innen medisinen og ICPC innen allmennlegetjenesten [24]. Klassifikasjonssystemer tilgjengelig på norsk er ikke så mange i antall. Likevel har vi allerede har en situasjon der de største systemene benytter delvis forskjellige rammeverk. Derfor vil en innfallsvinkel som kan tilpasses eksisterende EPJ-systemer og ulike rammeverk være nyttig.

Et eksempel på hvordan dette kan gjøres vises i en prototyp som benytter en kjent informasjonsgjenfinningsteknikk (vektorrommodellen), et rammeverk (Sabaklass 2.0N), og en autogenerert ordliste basert på rammeverket.

Oppgaven vil ikke beskrive i detalj hvordan en navigerer etter et problemorientert syn. Dette er utenfor scopet til oppgaven, men at emnene (eller behandlingskomponentene om en benytter Sabaklass' terminologi) kan benyttes til slik navigering vil bli berørt, og eksempler på bruksområder for funksjonalitet på dette området, beskrives overflatisk.

1.4.1 Personlige erfaringer, og ønske om bedret søkefunksjonalitet

Mitt arbeid i denne oppgaven er inspirert av personlige erfaringer med sykepleiedokumentasjon i elektronisk form i ett av de største EPJ-systemene i Norge, ved et av de største sykehusene. I systemet som benyttes der, kan gjenfinning av dokumentasjon karakteriseres som manuell ved at en må lese igjennom mye tekst før en finner det en leter etter. Opplæringen i dokumentering i systemet har også vært dårlig. Min opplevelse er at personalet oppfatter gevinsten av å ha EPJ, men utnytter programvaren dårlig. I bunn og grunn gjøres det som det alltid har vært gjort, nå bare "på data". Uten å ha gjort en grundig analyse av dette, tror jeg det dels skyldes at grensesnittet er gjort så likt papirformatet som mulig, og dels at personalet blitt lært opp til å bruke systemet på et tidligere tidspunkt. Opplæringen har ikke blitt fulgt opp når systemet har utviklet seg og fått flere funksjoner. Enkelte av pleierne har ikke gått kurset i det hele tatt, og skal få hjelp av såkalte "superbrukere" på avdelingen som har fått en grundigere innføring. Problemet er at når superbrukerne skal vise de andre hva som er nytt, er det et gap i kunnskap, slik at det hele fortoner seg som et grunnkurs. Dette skjer gjerne i tiden da man skal skrive rapport, og superbrukeren selv skal skrive sin egen sykepleierapport. Dette gjør at de mer avanserte funksjonalitetene i dokumenteringssystemet aldri blir brukt.

I min avdeling har det aldri vært kultur for å skrivepleieplaner. Dette gjelder også for papirversjonen av sykepleiedokumentasjonen. Dette skyldes mange grunner, men den som nevnes oftest er tidspress. Pleieplaner har blitt nedprioritert, noe som har konsekvens for bruken av pleieplaner i EPJ-systemer også. Studier i forbindelse med DIPS' innføring av VIPS og NANDA i sitt EPJ-system, viste at pleieplaner ble skrevet i bemerkelsesverdig høyere grad etter innføringen av dette i forsøkssystemet. Det kan være flere årsaker til dette:

- pleierne så større behov for denne funksjonaliteten,
- det ble fokusert på at dette var en funksjonalitet som var nyttig
- det ble satt av mer tid til rapportskriving
- det var et resultat av at man merket et bedre utbytte ved lesing av rapporter og pleieplaner

Resultatene av studiene er publisert, bla [38].

Poenget med å nevne dette, er at strukturell innlegging har mange fordeler, og er sannsynligvis den beste metoden for å sikre at man kategoriserer riktig i ett gitt rammeverk. Men dersom verktøyet ikke brukes, har man ikke strukturert tekst å jobbe med heller.

Strukturert innlegging har også ulemper, blant annet at dokumentasjonen nå blir systemspesifikk (her skynder jeg meg å nevne at jeg ikke har glemt at det finnes utvekslingsformat mellom systemene), og at fremstillingen av informasjonen kan oppleves som forvansket og oppstykket. Noen av erfaringene fra eksisterende EPJ systemers struktur i spesielle maler, er at det påvirker tilegningen av informasjonen negativt ved lesing av teksten. Eksempelvis kan ikke den strukturerte teksten i enkelte journalsystem sammenstilles og presenteres i andre kategorier enn kategoriene de er opprettet under. Likeledes kan det oppleves som søking etter informasjon forvanskes. Dette kan skyldes mange forhold, ikke minst sykepleiernes (manglende) erfaring med datamaskin som arbeidsverktøy. Men, det blir pekt på av bl.a. andre Grimsmo og Broseet [10] at mer intelligens i programvaren og bedre grensesnitt er viktig for bruk av EPJ slik at det blir et verktøy mer enn en belastning for pleiepersonalet. De viser til disse innfallsvinklene for opptak av data:

- Styrt strukturering av inndata ved hjelp av dynamiske/interaktive skjemaer med transformasjon av innholdet til lettleselig tekst. Det fungerer når helsepersonell i bytte får raskere og enklere registrering.
- Utvikle verktøy som kan gjenkjenne innholdet i fritekst og automatisk organisere og kode dette, dvs. fri brukeren for krav om strukturering.

- Innebygge assistanse som vil tillate bruk av synonymer, forskjellige skrivemåter (inklusive forkortelser og feil), vilkårlig rekkefølge og gjenbruk av tidligere brukt eller standard tekst.
- kvalitetssikringssystemer hvis mål blir å tillate større marginer ved at de blir mer effektive og selektive
- dynamiske behandlingsprotokoller basert på tilfanget av informasjon i EPJ-systemer
- Kombinasjoner av disse er også mulig. En kan flytte struktureringen av informasjonen fra brukergrensesnittet til lavere lag i programvaren. Å få støtte av systemet til å dokumentere riktig i systemet er viktig når det er så mange behandlingsprotokoller å forholde seg til at den enkelte helsearbeider mister oversikten.

Med tanke på de store mengder med sykepleiedokumentasjon som finnes i ustrukturert form i EPJ-systemer er tankegangen om å kunne kategorisere fritekst forlokkende. Det har vært gjort mye på dette området innen andre profesjoner allerede, spesielt for legejournalen. Problemstillingene ved å gjøre dette for sykepleiedokumentasjon, er mange. Likevel kan en tenke seg at et unøyaktig hjelpesystem er bedre enn ingenting, og hvis en kan kombinere et slikt hjelpesystem med en eksisterende løsning, kan en kombinere styrken fra begge - det ene utelukker ikke det andre. Det trenger ikke være for dyrt eller vanskelig dersom en benytter kjente og prøvde metoder; der vil en på forhånd også vite noe om feilrate. Det er uansett en pleier som avgjør hvordan en skal benytte rammeverket når en skal *legge inn* informasjonen. Å tilføre tradisjonelle søketeknikker fra det digitale biblioteks verden i større grad enn det er gjort i dag, er nyttig.

En praktisk løsning som f.eks. å få opp kategoriene automatisk ved å holde pekeren over datofeltet (som ofte angir en rapport i EPJ systemer) er et slik bruksområde. Mange vil kjenne igjen dette fra såkalte RSS (RDF Site Summary) systemer som benyttes i nettbaserte aviser, og funksjonaliteten blir vist i prototypen utviklet for denne masteroppgaven (bare for behandlingskomponenter). Eksempelet er også en funksjonalitet som ville la seg kombinere med begreper hentet fra manuelt strukturert informasjon i EPJ-systemet. Greier en å få til dette, er ikke veien lang til såkalte TopicMaps, og en kan skape en helt annen innfallsvinkel til all tekst, også tekst som i utgangspunktet fortoner seg som en gammeldags pergamentrull.

2 Sentrale betraktninger forut for automatisk klassifisering

Dersom leseren av denne oppgaven ikke har kjennskap til historikk knyttet til sykepleiedokumentasjon, benyttes anledningen her til å si at sykepleiere har lange tradisjoner i å dokumentere alt de gjør, og har etter nyere (2001) lovgivning plikt til å dokumentere (helsepersonelloven) [5].

Viktige hendelser og personer for sykepleiernes tradisjoner vedrørende dokumentering, er ”den hippokratiske legeed”, systematisering i Florence Nightingale's arbeid under Krim krigen, og klassifiseringstankegang fra nyere tids teoretikere som Virginia Henderson, Dorothea Orem m.fl.

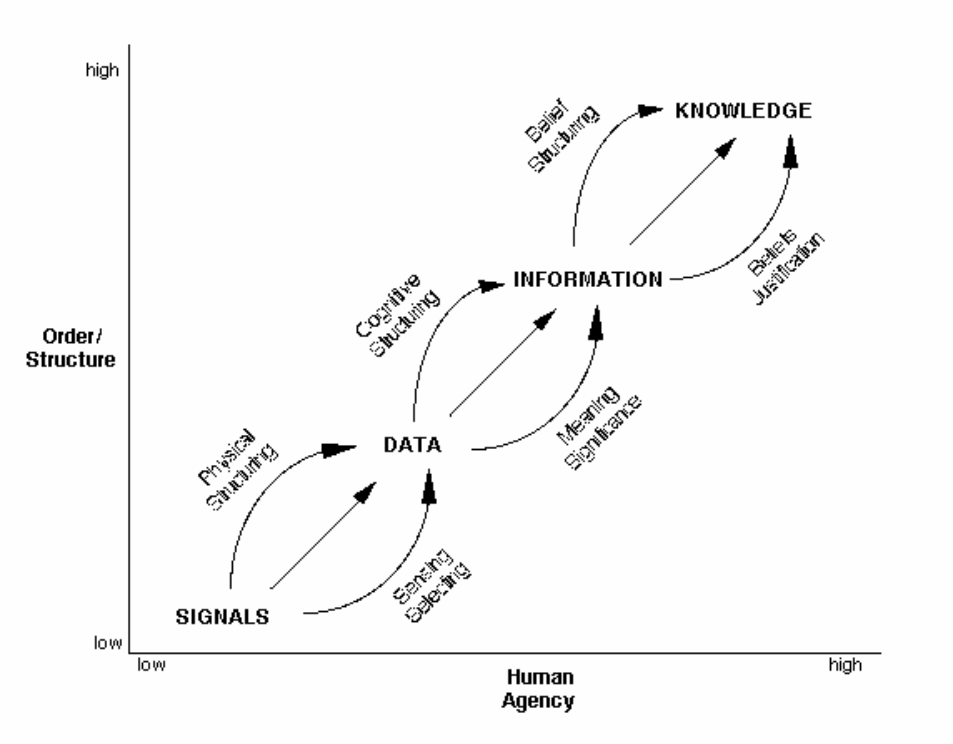
Naturlig språkprosessering for databehandling er et stort fagfelt. Forsøk på å fange opp begreper i tekst og klassifisere dem har så mange lingvistiske aspekter at det er vanskelig å peke på alt som er av betydning. For sykepleiedokumentasjon er det en del aspekter som kommer inn i tillegg til de "vanlige" problemene. Betraktninger som gjøres i de følgende kapittel er allmenngyldige uavhengig av klassifiseringssystem og teoretikere, men prosessering av sykepleiedokumentasjon møter disse problemene, fordi det er tradisjoner for å dokumentere forskjellig. Noen av disse "uvanene" blir nevnt for å belyse dette.

2.1 Om synet på data, informasjon og kunnskap

Det finnes mange måter å betrakte data, informasjon og kunnskap på. En ikke uvanlig måte å se det på, er at den ene er et grunnlag for den andre. En slik betraktning kan eksemplifiseres slik:

Språket vårt består av å organisere tegn eller signaler slik at det blir meningsbærende. Blant tegnene finner vi tall og bokstaver, og disse kan utgjøre en symbolsk enhet vi kan forholde oss til. Tallet 37 er et eksempel på ett symbol vi forstår. Det består av to tegn, og kan utgjøre det som kalles data. Om det er data eller ikke, kommer an på settingen. Sammenstiller vi det med tallene 36 og 38 har vi plutselig en tallrekke. Sier vi at dette er en skala, og at 37 er bestemt på grunn av en utregning eller måling, er det enda større grunn til å kalle det data. Vi kan sette verdien 37 i en større kontekst; la oss si at tallrekken 36, 37, 38 inngår i en skala som benyttes for å verdisette kroppstemperatur, basert på et system utviklet av Anders Celsius (1701-44) [52]. Hvis 37 er en verdi som kommer oss i hende som følge av en rektal måling av ett menneske via et instrument som heter termometer, er sannsynligheten for at vi kan kalle dette informasjon ganske stor. Denne informasjonen har i seg selv ingen stor nytteverdi, om det ikke benyttes i en enda større kontekst. En mulig vinkling på dette, er at den som leser

informasjonen har kunnskap om termometer, celsiusskalaen, og hva som er vanlig kroppstemperatur for et menneske. Dersom en slik person finnes, kan en nå avlede mange forhold rundt denne informasjonen, og sette det inn i en kunnskapssammenheng.

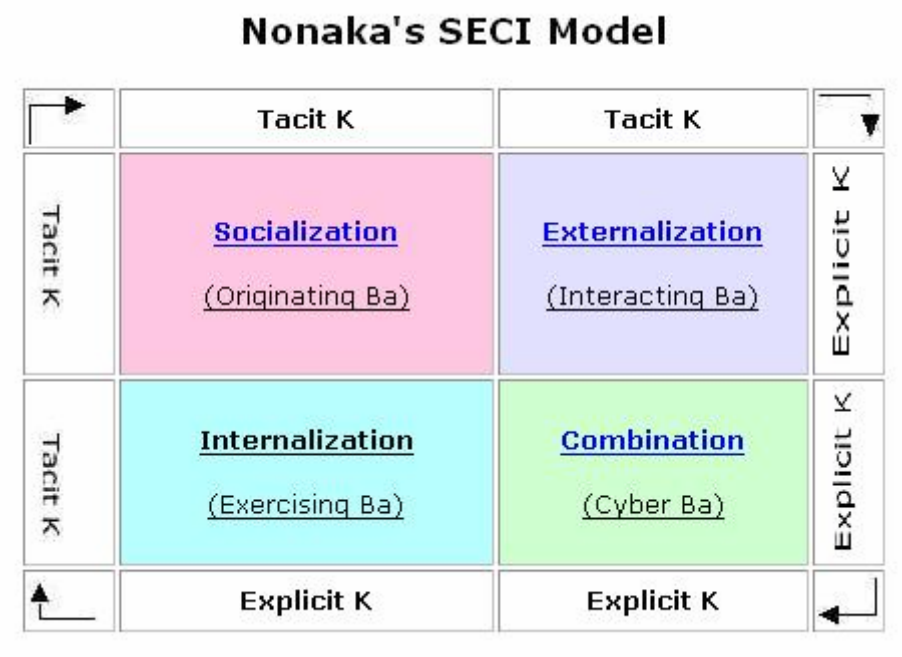


Figur 2-1 Fra signal til kunnskap,

Figur 2-1 er hentet fra [4], og viser den tenkte prosessen å bevege seg fra signal til kunnskap. Figuren viser at krav til orden og struktur, og menneskelig medvirkning øker fra lav til høy når en nærmer seg kunnskap.

I denne oppgaven kan en kritisere forsøket på å automatisk knytte begreper til klassifisering som et hopp fra data til kunnskap uten å gå veien om kognitiv resonering. Med dette menes at man forsøker å gi enkeltstående data, et ord, mening og innhold gjennom klassifisering uten at en bruker kunnskap aktivt for å gjøre dette. Et motargument til dette kan være at man bruker informasjon til å komme frem til klassene, fordi det er flere ord som inngår i beregningene og at rammeverkene ivaretar semantikken. Ergo har en ikke hoppet bukk over informasjonssteget, for det er ikke hvert enkelt ord som danner grunnlaget for klassifiseringen, det er alle ordene i sammenheng som gjør det. Motargumentet til dette igjen, er at en kan ikke kalle dette kunnskap, bare relasjon i gitt kontekst.

Arbeidet til Nokaka m.fl. [20] mener kunnskap er en følge av Sosialisering, Eksternalisering, Kombinering, og Internalisering. I Figur 2-2 vises SECI modellen, som illustrerer tankegangen:



Figur 2-2 SECI modellen for kunnskap

Siden sykepleiedokumentasjonen er eksplisitt skrevet, er den å betrakte som eksternalisert. Men, som vi vil se i kapittel 2.3, er ikke eksternalisering så enkelt, og informasjonen må gå flere runddanser i SECI modellen avhengig av hvem som leser den og hva som gjøres med den. For deler av sykepleiedokumentasjonen er det i lys av slike anskuelser vanskelig å avgjøre om det er informasjon eller kunnskap vi snakker om.

Emnetilknytningsmetoden denne oppgaven beskriver, kortslutter delvis SECI-modellen også siden den ikke benytter semantikk og logiske slutninger for å knytte emner til sykepleiedokumentasjonen. Benyttes metoden som hjelpemiddel og ikke ansees for å sitte med kunnskapen, kan man si at den inngår i kunnskapsbygging etter SECI-modellen.

Disse aspektene skal ikke debatteres mer i dybden her, og de er trukket frem mest for å gjøre leser oppmerksom på at slike betraktninger får store konsekvenser for systemer som angriper en så kompleks problemstilling som denne oppgaven gjør, og videre lesning må foregå med dette i bakhodet.

2.2 Dokumentasjon av sykepleie i pasientjournalen (SfID)

I august 2002 kom det ut en veileder fra NSF, Sykepleiernes forum for IKT og Dokumentasjon (SfID), som het "Dokumentasjon av sykepleie i pasientjournalen". Den er senere blitt revidert og det er lagt inn et nytt kapittel om klassifikasjoner, koder og terminologier. Veilederen adresserer i hovedsak sykepleiere, og poengterer at selv om EPJ systemer for sykepleie har eksistert i over 10 år, er fokuseringen på struktur og felles

terminologi ny for mange. Veilederen er lagt opp generelt, og berører tema som har med lovgivning, EPJ standarder, faglig utvikling, klassifisering og strukturering innen elektronisk dokumentasjonshåndtering.

Strukturering er viktig for å utnytte potensialet datamaskinen som verktøy har når det gjelder informasjonshåndtering. Eksempelvis er beslutningsstøtte nevnt. Derfor er kravene til dokumentasjonen annerledes enn den er for papirdokumentasjon. Veilederen trekker frem ISO-18104 standarden for sykepleie referansemodell. Det er utviklet en modell for sykepleiediagnoser, og en for sykepleietiltak; disse beskrives som:

- Sykepleiediagnose
  - Fokus - hvem som skal motta hjelp (individ, familie, andre)
  - Bedømmelse - flere begreper; eks: aktuell, potensiell, risiko for
- Sykepleiehandling
  - Prosess der en tilsiktet tjeneste gis til en mottaker som har behov for helsehjelp. Handlinger må være rettet mot et mål.

Veilederen poengterer at klassifikasjonssystemer bare utgjør en del av det totale dokumentasjonssystemet. Den beskriver også kort de ulike systemene som er viktige for sykepleiedokumentasjonen i Norge (noen av dem er beskrevet i denne oppgaven i hovedkapittel om klassifisering), og andre viktige klassifikasjonssystemer i helsevesenet forøvrig.

Veilederen og alle prosjektene NSF og SfID er involvert i (se bl.a. kapitlene om KITH, S@mspill og NSEP senere) viser at forbundet er med i utviklingen av klassifisering og EPJ. Dette er av enorm betydning, slik at sykepleierne får være med i utviklingen av verktøy de må forholde seg til i årevis fremover. Likevel tror jeg forbundet har en stor oppgave foran seg når det gjelder å få den enkelte sykepleiers bevissthet rundt betydningen av klassifiseringssystemene opp.

2.3 Generelt om språkbruk i Sykepleiedokumentasjonen

Undersøkelser viser at den skriftlige dokumentasjon av sykepleie ofte er ufullstendig og lite systematisk. Svakheter med sykepleiedokumentasjonen er bl.a. disse [16]:

"Sykepleiernes formidling er preget av muntlig og upresis språkbruk. Dette gjenspeiles i sykepleiedokumentasjonen. Lokale diagnoser, skjulte pleieplaner og journalen som

beskjedbok gjør at dokumentasjonen ikke blir mulig å tolke for andre enn personalet på avdelingen".

Sykepleiernes måte å dokumentere og planlegge pleien på er veldig forskjellig fra avdeling til avdeling, fra kommune nivå til statlig nivå. KITH har laget en veileder [5] for hvordan dokumentasjon bør foregå for sykepleie i EPJ-system både med tanke på bruk av klassifikasjonssystemer og fri tekst for å få bedre systematikk i dokumenteringen.

I likhet med at nye teknologier har påvirket ungdommens språk (f. eks mobiltelefonen og IRC (Internet Relay Chat)), kan en håpe på at taksonomier og EPJ-systemer vil føre til endringer i språkbruken for sykepleiere i retning av bedre konkretisering. Standardisering av vokabular generelt og for EPJ spesielt har mange fordeler, bl.a. struktur for nedtegning og presisering av begreper for innhold [11]. Men standardiseringen og dokumenteringen påvirker også arbeidsmetodene til sykepleierne, med både positive og negative erfaringer knyttet til innlegging av dokumentasjon, og gjenfinning av den [63].

Et eksempel på ufordringen å lage et automatisk klassifiseringssystem kan illustreres fra følgende rapport hentet fra kommunehelsetjenesten:

EKSEMPEL:

"12.04.2005 - Bruker hadde akkurat sovnet da jeg kom".

"13.04.2005 - Bruker lå og sov, mannen ville ikke at jeg skulle vekke henne".

Språklig logikk noe av det vanskeligste å forstå for en datamaskin. Til forskjell fra vanlig prosa tekst er sykepleiesjargong snevret ned til domenet ned til helserelaterte problemstillinger. Likevel har friteksten i rapportene likheter til prosa på grunn av det holistiske (hele mennesket) synet sykepleierne har på pleien. Eksempelet over kan analyseres med språklig logikk til å ha mange betydninger, bla at pasienten lå og sov, og mannen ville ikke at pleieren skulle vekke henne - men det var greit at han gjorde det. Siden ikke mer er sagt i rapporten antar vi at hun ikke ble vekt av noen, og hun fikk sove videre ved dette tilsynet. Det virker lite trolig at man ved løsninger som skisseres i denne oppgaven skulle få noe nyttig ut av en slik formulert rapport. Er det i det hele tatt mulig for en datamaskin å "forstå" dette? Med mindre du som leser dette har en del erfaring med pleie og omsorg, menneskelige reaksjonsmønstre og mer kjennskap til resten av journalen, vil du heller ikke få mye ut av slik rapportering. Resten av sykepleiedokumentasjonen, viser at pasienten lider av mange sykdommer, er i konflikt med ektefelle pga. forestående innleggelse på sykehjem, og har nedsatt kognitiv kapasitet. Med erfaring som sykepleier, og ved å ha lest om problemene og oppførselen til pasienten tidligere, kan man anta dette: Foruten at hun er trøtt og sovner før

hjemmesykepleien kommer, kan det å legge seg før det kommer noen til huset ha med hennes reduserte allmenntilstand å gjøre, hun har en følelse av mangel av kontroll på livssituasjonen, føler seg fremmedgjort i forhold til avgjørelser for sin fremtid osv. En sykepleier vil kunne kategorisere dette ganske enkelt, men ikke en datamaskin. Forslag hentet fra Sabaklass' behandlingskomponenter for diagnoser, er AKTIVITET, ERNÆRING, EGENOMSORG, KOGNITIV, SIKKERHET, SAMSPILL til dette, men man kunne argumentert for et annet utvalg av "sykepleiediagnoser". Dette er tolkninger og vurderinger som det ikke er mulig å mappe til et rammeverk automatisk. Det har kanskje heller ingen betydning - vi må fortsatt lese rapportene, vi kan ikke få alt gjennom høynivå rammeverk.

Egenutviklet terminologi er et annet fenomen. Artikkelen "Fin i kontakten" illustrerer problemstillingene utmerket [16]. Det er et problem at det er utbredt terminologi i posten med formuleringer som f. eks at en pasient er "fin i kontakten" eller går fra å være til fin til hyggelig. Formuleringer i rapporten kunne vært "har bedret sin mellommenneskelige relasjon" og dette kan mappes enklere (eksempelvis til Sabaklass Diagnose: M 32 - Forandring i sosialisering). Men en må huske at sykepleiedokumentasjon også er et arbeidsinstrument, og om det gir like stor mening for den aktuelle posten å skrive på den mindre spesifiserende måten, må en la datasystemet tillate dette. En kan lokalt (f. eks for hver post) ha mulighet til å tilpasse ordlister til systemet, får så å la pleierne benytte disse ordlistene som hjelp til å mappe til et mer "høyverdig" språk. Denne løsningen er ikke optimal, men gir en mulighet til å tilpasse et automatisk begrepstilknytningssystem den lokale terminologien. Eksempelet viser at det til og med for mennesker er mye tvetydighet og lesning mellom linjene i rapportene. Derfor kan det ikke skade om en får støtte i å lese tvetydig dokumentasjon, selv om datamaskinen kan feile i forsøket. Vi må forholde oss til hvordan virkeligheten er, og søke forbedringer i fremtiden.

Som om ikke dette var ufordring nok, så kommer feilstaving av enkeltord og feil bruk av begreper i tillegg, samt henvisninger til udokumenterte hendelser, og ufullstendige gjengivelser av data. For ikke å snakke om egendefinerte forkortelser [16]. Dette må en forholde seg til, siden lovverket gjør det vanskelig å endre på allerede innlagt dokumentasjon. Forkortelsene vil det la seg gjøre å fange opp vha ordlister, men disse måtte i så fall spesiallages på grunn av at sykepleiernes fantasi på dette området overgår vanlige norske forkortelser. Det finnes teknikker for å analysere om et ord feilstavet automatisk, og noen er mer avanserte enn andre. En av de enklere metodene for dette er N-Gram stemming (se senere). Når det gjelder feil begrepsbruk, er situasjonen verre, men løsbar. Blanding av data og løsrevne henvisninger er et prosesseringsproblem man muligens ikke får til å løse.

Til nå er sykepleiedokumentasjonen beskrevet som om den ikke har noen struktur i det hele tatt. Dette er en betraktning som ikke stemmer. Eksempelvis er mange sykepleiere flinke til å skrive emneord før tekst, og på det viset er strukturering allerede til stede i teksten. Dette foreligger gjerne på formen:

- Ernæring:
- Sirkulatorisk:
- Sosialt:
- Annet:

Ofte er ikke denne stilen knyttet til noe spesielt rammeverk, men er en skikk sykepleieren har hentet fra utdanningen sin, eller det er en måte å gjøre det på som avdelingen har kommet frem til. Organiseringen kan være hentet fra en sykepleieteoretikers anbefalinger (Dorothea Orem, Virginia Henderson), eller prosjekter rettet direkte mot et mer standardisert oppsett. VIPS modellens funksjonsområder er f. eks mye brukt [11] (se Vedlegg C). I enkelte EPJ-systemer (f. eks DocuLive) finnes det maler som sykepleierne kan tilpasse etter sitt behov til en slik organisering. Dette er nok det nærmeste en kommer en organisering som kan sammenlignes med legenes SOAP (Subjective, Objective, Assessment, Plan) som gjerne utnyttes av systemer som bedriver automatisk klassifisering innen deres domene (denne sammenligningen er noe søkt, fordi den ikke tar hensyn til pleieplanene). Likevel er ikke slik organiseringsform i teksten uniform nok til man enkelt kan mappe den til rammeverkene definitivt. Problemet er at organiseringen egentlig egner seg best til oversikt i et papirbasert system, og at man drar med seg skikken til EPJ-systemer fordi det delvis er lagt opp til det i disse systemene.

En tilnærming til å utnytte slik tekstlig struktur, er f.eks. å gi slike ord ekstra vekt ved likhetsutregning, og kanskje også ordene i påfølgende setning(er). Allerede strukturert informasjon som følge av bruk av implementerte rammeverk og pleieplaner, og den strukturering EPJ-systemet ellers kan gi, vil være nyttig informasjon å forholde seg til ved f.eks. bygging av thesaurus. Dette er ikke gjort i prototypen, men bruk av metadata ekstrahert fra teksten, maler eller systemet vil gjøre automatisk klassifisering mer nøyaktig. Kombinering av metoder er en fordel, men krever spesiell tilpassing. Dette er ikke beskrevet i denne oppgaven, da utgangspunktet er at dokumentasjonen antaes å ikke være strukturert utover det som er nevnt over.

3 State of The Art

3.1 Eksisterende systemer for automatisk klassifikasjon av ustrukturert tekst

Det har i skrivende stund ikke lykkes meg å finne systemer som tar for seg automatisk klassifisering av ustrukturert sykepleiedokumentasjon på norsk. Flere autoritetspersoner innen klassifisering av sykepleiedokumentasjon i Norge har blitt kontaktet i jakten på lignende prosjekter: Norsk Sykepleieforbund, representanter for de omtalte rammeverkene i oppgaven, høyskoler og universiteter, Siemens, Dips, HEMIT, m.fler. At det ikke er funnet noen, betyr ikke at de ikke finnes, men det kan bety at det ikke er publisert mye av det. En kan forundres over dette, siden rammeverk for sykepleiedokumentasjon har eksistert i tiår i USA, og det er gjort substansielle arbeider innen andre profesjoners domener, spesielt legejournalen.

Dette kapittelet omtaler kort prosjekter som er beskrevet i artikkels form. Det fører for langt å gå inn på detaljene for hvert enkelt, da de representerer arbeid gjort over flere år av flere personer. Hensikten med omtalen er å vise et utsnitt av hva som kan gjøres med automatisk klassifisering, og hva andre prosjekter har fått til. Som en på forhånd kunne anta, ser en at kompleksiteten øker suksessivt med nøyaktigheten en ønsker å oppnå.

Bruk av VSM på sykepleiedokumentasjon

Applikasjonsområdene for vektorrommodellen er mange. Dersom en oppnår gode nok resultater kan en tenke seg at man kan bruke dem til beslutningsverktøy og få mer ut av dokumentasjonen. Som et eksempel på at dette kan være mulig, er denne artikkelen interessant [27]: Man har i artikkelen (fra 1996) trukket ut pasientdetaljer fra en database og laget dokumenter i tekstlig form som det så utføres likhetsberegninger på vha vektorrommodellen (VSM). Denne teknikken sammenlignes med resultatene de oppnådde i ett "case-based reasoning" system. Artikkelen studerer muligheten for å forutsi hva som er beste behandlingsmetode for en pasients nyoppståtte problem, samt å forutsi hvilken behandling som kan være aktuell når et kjent problem krever endring i behandling. Systemet er altså et "decision support" (beslutningsstøtte) verktøy. Pasientgruppen som studeres er innlagt ved en kirurgisk avdeling. Pasientdokumentasjonens nåværende struktur i databasen blir utnyttet ved at de f.eks. gir attributter som kjønn, alder, diagnosegruppe osv et forutbestemt prefiks når de danner dokumentene. De gjør dette for å kunne tillate sammenligning av tilfeller og historie, men fastholder at de ikke benytter dette for å lage sekvenser i representasjonen.

Teknikkene som benyttes er VSM og et mer avansert regelbasert system. Sett i kontekst av denne oppgaven, er den interessant som argumentasjon for at enkle, kjente teknikker har sin styrke sammenlignet med, eller i samarbeid med, mer avanserte teknikker. Artikkelen peker på at tekst som er strukturert i en database, kan inneholde informasjon som ikke kan utnyttes til fulle fordi den er knyttet til en bestemt databasestruktur. Det påpekes i konklusjonen i arbeidet at representasjonen av pasienttilfeller og dokumenter er veldig nært knyttet til gjenfinningsytelsen, og at det ikke ble utført individuelle målinger for betydningen av disse for å bevise betydningen hver representasjon har. I korthet viste resultatene at enkel tekstgjenfinning i det omtalte systemet var bedre enn den regelbaserte metoden for den ene problemstillingen de undersøkte ("Change Treatment"). For den andre ("New Problem"), måtte vekting læres ved regresjon, men ble da forbedret i forhold til den regelbaserte metoden med opptil 18 %.

Vekting (se bl.a. kap 4.6.1) av spesielle termer er et av forbedringspotensialene som ikke er utnyttet i tilstrekkelig grad i prototypen denne oppgaven henviser til, bl.a. fordi den ikke forholder seg til metadata fra EPJ-systemet.

Metoder som kan påvirke til bedre resultat

Synet på termer i metoden som denne oppgaven beskriver er i likhet med mange andre former for NLP (Natural Language Prosessing) at en term er atomisk. Teknikker knyttet til NLP for å forstå tekst og tale investeres det hundrevis av arbeidstimer i, og EPJ systemer og det medisinske domenet er et naturlig nedslagsfelt for disse. Problemet er at dette er grov overforenkling av virkeligheten; termer er relatert til hverandre. Artiklene [3] og [29] representerer et viktig tillegg til NLP. Arbeidet (fra 1998) peker på at å betrakte de store mengdene med medisinsk tekst som "en sekvens av tegn" er nyttig, men ikke egnet for å organisere informasjon. Mengdene alene gjør problemstillingen med å organisere enda mer aktuell, og likevel fortsetter EPJ systemer å lagre tekst ustrukturert.

"Morpho-semantems" ble utviklet som teknikk på 70-tallet, men artikkelforfatterne hevder at først nå kan den komme til bedre utnyttelse på grunn av konteksten EPJ gir. Leksikalsk analyse bryter som oftest opp sammensatte ord, og man betrakter betydningen av dem som enkelheter (semantiske atomer). Generelt kan en si at et "morpho-semantem" peker til et elementært konsept, og ikke kombinasjonen av to eller flere konsepter i domenet. Som eksempel på et slikt konsept, brukes på engelsk "Aortic valve insufficiency" og den tyske ekvivalenten "Aortenklappeninsuffizienz". Artikkelforfatterne oppfatter dette som et betydelig skifte i synet på språkrepresentasjon. Parseren som er utviklet for formålet er

avansert, og benytter flere ordbøker (men det totale behovet for ordlister reduseres). Den morfo-semantiske ordboken, som er delvis vanskelig å lese for mennesker, er særdeles viktig for å kunne ekstrahere kunnskap ut av tekst automatisk.

Hovedpoenget er altså at morfemene er bedre til å ekstrahere et dokuments mening enn de enkeltstående ordene. Femgangsåten er å

- konvertere ord til deres helt grunnleggende former
- hente de grunnleggende formene fra ett leksikon
- dekomponere "morpho-semantems" av ordene
- analysere termene med tanke på sammensatte uttrykk
- analysere setninger med tanke på spesialtegn, negering, oppdeling osv.

Blant fordelene med metoden nevnes at man kommer mer finkornet inn på kunnskapen i teksten, man kan redusere ordlistenes størrelse, man kan følge det medisinske domenets hang til å lage nye uttrykk bedre, raskere prosessering, større grad av nytteverdi selv ved mislykket parsing og blant vestlige land er flerspråklige leksikon lettere å utvikle.

Begrensningene er at slik parsing er mindre kjent blant lingvister i databransjen, og tilgjengelige verktøy derfor begrenset. Man må benytte tagging eller lignende teknikker, flerspråklige leksikon av denne typen finnes ikke, den regelbaserte fremgangsmåten trenger tidkrevende finjustering, den kan ikke basere seg kun på leksikalsk analyse alene (underliggende domenemodell trengs).

Effekten "morpho-semantic parsing" har på ordlistene gjør den attraktiv for denne oppgaven. Brukt i prototypen for denne oppgaven, ville en slik teknikk betydd at man kunne operere med færre ord i thesaurusen, kanskje bare høyereliggende begreper som lignet dem i rammeverkene. I tillegg ville man ha større kontroll over fraser, og kunne utnyttet dem bedre, f. eks til vekting. En annen mulighet ville være å fri thesaurusen fra rammeverket i større grad, slik at den kunne benyttes på flere områder. Implementasjonen ville blitt betydelig mer kompleks, og beveget seg bort fra en enkel og billig løsning. Hvorvidt metoden hadde egnet seg godt for sykepleiedokumentasjon er også et sentralt spørsmål.

Automatisk klassifisering applisert på radiologi

At noen faktisk får til å klassifisere automatisk etter et rammeverk er artikkelen [47] et eksempel på. Artikkelen (fra 2001) beskriver et system (infoRAD) gjort for å strukturere radiologenes beskrivelse av en røntgenundersøkelse i fri tekst form. Disse rapportene

inneholder medisinsk informasjon, og studiens NLP (Natural Language Prosessor) finner eksistensen, egenskapene, lokalisering, og diagnostiske tolkninger av funn i rapportene. Motivasjonen for systemet er ganske lik motivasjonen for å strukturere fritekst i sykepleierapporter: Skaffe seg oversikt, kunne finne igjen spesielle hendelser uten for mye annen støy, generelt utnytte medisinsk informatikks muligheter til å finne ut årsak, hvor, hvor lenge, utvikling og finne grunnlag for vurderinger. Vanskelighetene med å strukturere fri tekst for domenet radiologi, har store likhetstrekk med de som eksisterer for sykepleiedokumentasjon. Artikkelen beskriver hovedpunktene i fremgangsmåten for å komme frem til et strukturert skjema for informasjonen. Stegene er

- En strukturell analyse for å finne seksjoner i rapporten som kan kategoriseres som "Prosedyrebeskrivelse", "Historie", "Funn" og "Inntrykk", samt setninger innen disse seksjonene.
- Leksikalsk analyse som finner semantiske og syntaktiske trekk for ord vha et medisinsk leksikon
- En parser som fastslår relasjonene mellom ord i setningene
- En semantisk tolker for lenkene parseren fant og laget i det forrige steget, denne produserer så avhengighetsdiagram og en mengde (sett) logiske relasjoner
- En konstruktør for en ramme som samholder de individuelle logiske relasjoner i strukturerte rammer (frames).

Systemet produserer logiske relasjoner for funn (f. eks "fortetning") i teksten på formen "has_existence", "has_location", "has_size", "has_size_trend", som videre gies attributter som verdi, grad av sikkerhet, hvor, når, dimensjon osv.

Metoden innebærer avansert analyse av struktur av rapporten i sin helhet, på setningsnivå og ordnivå. På ordnivå har de utviklet sitt eget leksikon basert på publiserte radiografiske kilder og faktiske rapporter, og benytter ikke mer overordnede ressurser som f. eks UMLS (universell metathesaurus for medisinsk språk) fordi de mener at slike ikke har stor nok detaljgrad for å støtte NLP systemer som dette. Systemet benytter AI (Artificial Intelligence) for å lære seg semantisk tolkning basert på treningssett og et på forhånd tagget treningseksempel. I tillegg har utviklerne mulighet til å indikere logiske forbindelser manuelt for systemet. Systemet ble påbegynt i 1995. I 2001 var klinisk fokus serie-CT (Computer Tomografi) rapporter innen pediatri for pasienter med lungemetastaser.

3.2 Eksisterende EPJ systemer

Kompetansesenter for IT i helsevesenet AS (KITH) har formulert en definisjon på hva en Elektronisk Pasientjournal (EPJ) er:

”*En elektronisk pasientjournal (EPJ) er en pasientjournal hvor informasjonen er elektronisk lagret på en slik måte at den kan gjenfinnes ved hjelp av EDB-verktøy".*

Det eksisterer i dag en rekke systemer som har med pasientbehandling å gjøre, og KITH sier videre

"Den samling av opplysninger som utgjør en pasients EPJ f.eks. på et sykehus trenger ikke nødvendigvis å være håndterer av et og samme EPJ-system. Ofte vil det være slik at det i tillegg til opplysningene i det generelle EPJ-systemet finnes opplysninger i flere spesialiserte system som inngår i pasientens EPJ."[49]

En EPJ kan altså være en samling av systemer, og trenger ikke nødvendigvis integrere alle systemer. Fra ”Forskrift om pasientjournal" (des.2000) finner man denne definisjonen om hva en pasientjournal er:

”Samling eller sammenstilling av nedtegnede /registrerte opplysninger om en pasient i forbindelse med helsehjelp jfr. helsepersonelloven paragraf 40"

Lov om Helsepersonell § 40:

"Journalen skal føres i samsvar med god yrkesskikk og skal inneholde relevante og nødvendige opplysninger om pasienten og helsehjelpen, samt de opplysninger som er nødvendige for å oppfylle meldeplikt eller opplysningsplikt fastsatt i lov eller i medhold av lov. Journalen skal være lett å forstå for annet kvalifisert helsepersonell.

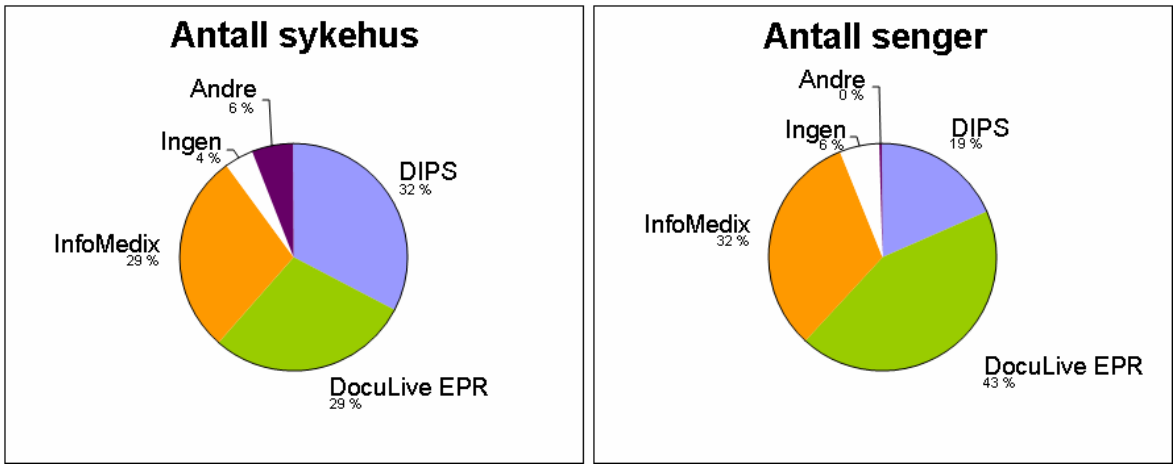
Det skal fremgå hvem som har ført opplysningene i journalen.

Departementet kan i forskrift gi nærmere regler om pasientjournalens innhold og ansvar for journalen etter denne bestemmelse, herunder om oppbevaring, overdragelse, opphør og tilintetgjøring av journal."[54]

Man kunne sikkert også med fordel referert til paragraf 46 i samme lov, som sier at journalen kan føres elektronisk. Slike definisjoner forsøker å poengtere og oppsummere hva man mener med en EPJ. Den første definisjonen poengterer at informasjonen skal være

søkbar vha EDB verktøy. Hva som ligger i dette sier definisjonen ikke mer om. Som et eksempel på hvor problematisk det er å relatere slike definisjoner til et spesifikt system og deres løsninger, kan en hente frem begrepet "Papirløs EPJ". Uttrykket er hentet fra Dips' foredrag på HelsIT2004 [26] i forbindelse med skanning (optisk digitalisering av papir) av dokumentasjon. I forhold til "Forskrift om pasientjournal" skulle skannet dokumentasjon være like bra som noe, men om det dekker KITH's definisjon blir vel i stor grad en implementasjonsavhengig diskusjon.

EPJ som begrep har altså mange fasetter, og det finnes mange systemer som kan kalles EPJ-systemer. I denne sammenheng nøyer vi oss med å se på løsningene to viktige systemer, Siemens "DocuLive EPR" og DIPS ASA's "DIPS", har i forbindelse med elektronisk lagret sykepleiedokumentasjon (skannede dokumenter er utelatt her). Figur 3-1 viser andelsprosenten de to omtalte systemene representerte i 2002 [56].



Figur 3-1 Fordeling av somatiske sykehus og sykehussenger tilknyttet EPJ, 2002

3.2.1 DIPS

DIPS har allerede implementert bruk av NANDA i sitt system, og ser nå på NIC. Med tanke på at NANDA ble publisert på norsk i 2003 (versjon fra 2001-2002) og at NIC forventes lansert på norsk snarlig (våren 2005), må de sies å være langt fremme i bruken av klassifiseringstaksonomier innen sykepleiedokumentasjon. De har også et system som fokuserer på sykepleiedokumentasjon i større grad enn konkurrentene, og har en del interessante vinklinger på prosessmodellen til sykepleierne.

Størrelse og utbredelse (i Norge)

DIPS ASA er et norsk firma som ble etablert privat i 1997. Det er en viktig aktør innen levering av papirløse journalsystem til sykehus, og dekker 35 % markedsandel på somatiske sykehus, og hele 65 % markedsandel på psykiatriske institusjoner i Norge. (Tall fra 2002, 3 % variasjon fra Figur 3-1).

Sykepleiedokumentasjon

DIPS er som det første firmaet i Norge som leverer journalsystem for sykehus som har integrert støtte for sykepleiedokumentasjon [37], [26],

Firmaet innledet ett samarbeid med Diakonhjemmet Sykehus i Oslo og Sentralsjukehuset i Sogn og Fjordane (Førde Sjukehus) for å jobbe med et prosjekt ("Elektronisk sykepleiedokumentasjon") som hadde som mål å utvide funksjonaliteten i DIPS' journalsystem for sykepleiere. Prosjektet startet i 1999 og ble avsluttet i 2003. De har tatt utgangspunkt i sykepleiernes behov, og tilpasset dem systemets eksisterende funksjonalitet og integrert dem i DIPS slik at flere kan bruke dokumentasjonen (imøtekomme tverrfaglige behov, basert på lesetilgang). Noen av resultatene fra prosjektet er nå implementert i systemet. På deres informasjonssider på nettet [36] finner man de viktigste:

- Eiendelsregistrering
- Mulighet for å registrere ansvarlig sykepleier (primærkontakt)
- Mulighet til å få presentert sentrale opplysninger om pasienten og oppholdet i en "stay- on-top" funksjon (G1 funksjon, se Vedlegg D)
- Mulighet til å knytte innlagte pasienter til team og få presentert sengepostlisten teamvis
- Mulighet til å få skrevet ut pasientlister med sentral informasjon om pasienten basert på teamtilhørighet (rapportark)
- Fleksible måter å presentere sengepostlisten på som brukeren selv kan definere og lagre i egne profiler

Det er fokusert på å kunne legge inn sykepleiedokumentasjon strukturert i DIPS. Systemet håndterer sykepleiedokumentasjonen som en gruppe som inngår i pasientens totale, kontinuerlige journal. Innen denne gruppen kan sykepleierne definere ulike dokumenttyper, som f. eks innkomstnotat, utskrivingsnotat mm (se Vedlegg A). Man kan legge inn alle typer dokumenter sykepleierne bruker, og det finnes maler til disse. Disse malene er utviklet med

bakgrunn i klassifikasjonssystemene NANDA og NIC. Dips vektlegger at strukturert innlegging er en fordel for uthenting, man kan bl.a. lettere nyttegjøre seg dokumentasjonen i forskning og statistikk. I tillegg kan man vise journal flettet fra flere faggrupper, slik at journalen fremstår helhetlig.

Sykepleiejournalens elementer kan inngå i ett nytt dokumentformat som blir kalt "Behandlingsplan", der alle disse elementene kan vises som sammenhengende tekst basert på kronologi, eller som enkeltelementer. Behandlingsplanen i DIPS består av fritekst og "Pleieplanelementer". Disse elementene består av sykepleiediagnoser, problemer pasienten har, sykepleietiltak og - forordninger. Grunnen til å lage et eget dokumentformat for dette, er at man nå kan presentere elementene i en helhetlig pleieplan. Pleieplanen viser en oversikt over alle pasientens sykepleiediagnoser, tiltak og forordninger mm. Systemet har lagt til rette for at en sykepleier kan finne veiledende pleieplaner til en problemstilling, selv om dette ikke er levert med fra DIPS. Årsaken til at DIPS ikke leverer dette, har med faglig ansvarsforhold å gjøre. Dette er for tiden noe det enkelte sykehus vil måtte ta ansvar for, for det finnes ikke noen overordnet godkjent standard enda (NIC er ikke oversatt eller godkjent i skrivende stund). Sykehus som selv har laget slike pleieplaner kan hente dem ut i Behandlingsplanen.

DIPS angir at de forsøker å tone ned "Sykepleieprosessen" som modell for dokumentasjon, selv om det vil være mulig å følge denne modellen også i bruk av journalsystemet (forskjellen ligger altså i hvordan man dokumenterer og hvordan man bruker journalen som planlegningsverktøy). Sykepleiedokumentasjonen er forsøkt strukturert på ulike kategorier, og "VIPS" modellen er benyttet som inspirasjonskilde, selv om de ikke bygger på denne modellen i DIPS. De hevder at struktureringen i "VIPS" modellen fungerer som bindingsmiddel opp mot sykepleieprosessen i for stor grad, og frykter en påvirkning i negativ retning for arbeidsflyten i dokumenteringen. Videre mener de at sykepleieprosessen er for rigid for en elektronisk journal i dokumentasjonssammenheng (et tiltak vil kunne være aktuelt å bruke for flere av pasientens problemer). Lineære sammenhenger som sykepleieprosessen er et uttrykk for, vil gjøre systemet lite fleksibelt [36], [40].

Søking og navigering

Strukturert innlegging der forskjellige dokumentformater kan inngå gir mulighet for mange løsninger når det gjelder fremstilling av dokumentasjonen som finnes i systemet. Men samtidig kan en slik kategorisering muligens skape problemer for fri navigering i teksten. Teksten i journalsystemet kan ikke sammenstilles og presenteres i andre kategorier enn kategoriene de er opprettet under. DIPS har muligheten til å bruke vanlig tekstsøk i den

kontinuerlige journalen, og slik finne frem til det som er skrevet om det enkelte funksjonsområdet. Eksempelvis vil en kunne se hva som er skrevet om en pasients Eliminasjon ved å finne frem kontinuerlig sykepleiejournal fra siste opphold (dette vil være alle sykepleiedokumenter og "rapporter" sammenstilt kronologisk som sammenhengende tekst), velge "Søk" og skrive eliminasjon. Deretter kan en bla seg nedover i journalen og se alle registreringer rundt dette funksjonsområdet. Et slikt søk fordrer at man har lagt inn emneordet "Eliminasjon" ved innlegging av dokumentasjonen. (Basert på opplysninger fra Kåre Flø, og Liv Haugen ved DIPS' Trondheimskontor, som lot meg se det i praksis.)

3.2.2 Doculive

Siemens norske pasientjournalsystem – ”DocuLive EPR” har fra desember 2004 støtte for pleieplaner i sin løsning for sykepleiedokumentasjon.

Størrelse og utbredelse (i Norge)

DocuLive brukes i dag av:

- Universitetssykehuset i Nord-Norge
- Rikshospitalet
- Ullevål universitetssykehus
- Helseregion Midt Norge bruker en sentral løsning for alle 8 sykehusene i regionen (flersykehus modell)
- Helse Bergen og Helse Fonna bruker Siemens pasientjournal på sine sykehus, totalt 6 sykehus er nå knyttet opp mot Haukeland Sykehus med flersykehusmodellen
- Akershus Fylkeskommune har fire sykehus i drift, og har i dag over 6 års produksjon
- Passert 1 mill. elektroniske journaler fordelt på 30 000 brukere og 21 sykehus

Systemet er utviklet i samarbeid med alle 5 regionsykehus i landet, og har siden 1999 vært et internasjonalt samkjørt prosjekt. I Norge har Siemens ca. 40 utviklere som jobber med DocuLive EPR, av totalt 400. I andre verdensdeler markedsføres det relaterte produktet ”Soarian”. Soarian er en webbasert pasientjournal, der man bruker systemet som en portal til å høste data fra flere undersystemer å få dem presentert i ett grensesnitt.

Sykepleiedokumentasjon

Innenfor sykepleiedokumentasjon har Siemens alternativ per tiden ikke tilknytning til diagnostiske rammeverk som f. eks NANDA og NIC/NOC. DocuLive baserer seg i større grad på de ulike former for sykepleienotater og dokumentasjonsformer papirjournalen har i dag. Et bredt utvalg av maler skal hjelpe brukerne (sykepleierne) til å strukturere dokumentasjonen sin bedre. Malene er bygd på VIPS-modellen (se Vedlegg C), og ved hjelp av standardtekster og kontrollert vokabular i form av ordlister som er tilpasset dokumentets type, kan man påvirke en mer uniform struktur. Dette betyr at en innkomstjournal har en mal som er tilpasset dette datainnsamlingsformålet, mens et problemnotat (eller en annen mal) vil ha et sett standardtekst tilpasset dette formålet og denne dokumenttypen. Systemet gir mulighet til å knytte emneord (også disse er basert på VIPS) til dokumentasjonen, og disse emneordene kan knytte notattyper sammen.

Sykepleiedokumentasjonen består i dag av

- Løpende journalnotater (G2)
- Problemer med mål, tiltak og evalueringer
- Sykepleiesammenfatninger og diverse brev.
- Oversikt over problemer, tiltak, notater og evalueringer i eget vindu.
- I tillegg har løsningen team-ark og status-ark under utvikling.

I systemet fremstilles sykepleiedokumentasjonen kronologisk, men mulighet til å lage pleieplan gir en annen synsvinkel i tillegg. Problemløsende tilnærming til dokumentasjon i sykepleiedokumentasjonen har vært et fokusområde i systemet. Pleieplanen kan f. eks bestå av problemstillinger, tiltak og evalueringer; slik "Sykepleieprosessen" legger opp til. I tillegg kan en bruker (dvs. sykepleier) legge inn notater i pleieplanen, og disse kan være av ulike dokumenttyper som f. eks innotat, utnotat, overflytting osv. Evaluering av problem, mål og tiltak skal være mulig å gjøre underveis, samtidig som en kan angi grad av måloppnåelse og skrive sammenfatning. Disse tilpassningene gir muligheter for assosiasjoner mellom dokumenttyper, og i utgangspunktet slik brukeren selv vil; f. eks:

- Problem - tiltak - notat
- Tiltak - problem - notat
- Notat - problem - tiltak
- Evaluering - problem - tiltak.

Man unngår et hierarki med mange regler, og en kan lage en mange til mange konstellasjon av f. eks problemstillinger og tiltak som er knyttet til hverandre.

Søking og navigering

Siemens har lagt opp til at notatyper kan knyttes sammen ved hjelp av emneord, som f. eks funksjonsområder og tiltaksområder. De har flere presentasjonsmuligheter der en kan se i fulltekst i ulike sammenstillinger uavhengig av hvor de er dokumentert. Alle notater knyttet til tiltak og problem kan vises, eller alle dokumenter av samme type, eller dokumentasjon knyttet til et funksjonsområde. Organisering av notatene kan bestemmes av bruker, slik at man kan ha flere tiltak til ett problem, og at flere problem dekkes av samme tiltak. Når en ser sykepleiejournalen som kontinuerlig tekst, er pleieplanelementene indisert med små ruter under teksten der det er rubrikker for de ulike problemene, tiltakene og evalueringene. For å se dem i sammenheng, dukker ett nytt vindu opp [70].

Når ordet "emneord" benyttes her, menes det emneordet som blir lagt inn av bruker når sykepleieren oppretter et notat. Strukturen ligger i innleggingen, så en må passe på å legge inn riktig emneord. Disse emneordene er hentet fra VIPS-modellen (se Vedlegg C). Emneordene for problem er hentet fra "Pasientstatus" i Sykepleieprosessen Fase 1, mens for tiltak er de hentet fra Fase 3.

Siemens DocuLive planlegger ikke automatisk klassifisering av fri tekst på grunn av den kliniske kompleksiteten. [44].

3.2.3 Sammenligning

De to systemene legger opp til støtte for at sykepleiedokumentasjon ikke bare skal være oppbevaring av dokumentasjon, men at den også skal fungere som et verktøy for planlegging av pleie. Man har muligheten for å legge inn sykepleiedokumentasjon fortløpende som før, men pleieplanstøtte og muligheten for uniform, strukturert innlegging gir anledning for uthenting etter et større spekter av valgmuligheter. Tabell 3-1 viser en oversikt over dokumenttyper og notattyper fra de to systemene (se også Vedlegg A og Vedlegg B).

Tabell 3-1 Dokument- og notattyper

DIPS	DocuLive
	Spl. sammenf. SplBrev til pas SplBrev om pas Meld. utskrivn.klar pas.
innkomstnotat	Spl.Besøk
notat	Spl.Inn-Notat
oppsummeringsnotat	Spl.Oper.rapp
overføringsnotat	Spl.Overfl.notat
sluttnotat	Spl.Pol.notat
tverrfaglig planlegging	Spl.Sammend.
	Spl.notat
	spl.Tlf.notat
	Spl.Ut-notat

Søkefunksjonaliteten i begge systemene er knyttet til emneord som pleierne har lagt inn. (DIPS har i tillegg muligheten for søk på enkelttermer (fritekstsøk) i dokumentasjonen). Systemene forsøker å ikke legge begrensninger på føring, men teksten som er i diagnoser og dokumenttyper er knyttet til systemets bakenforliggende struktur, med de begrensninger det kan påføre fremstilling, tilegning og bruk [56]. Gevinsten av struktureringen overskygger ulempene, men det er et poeng at man i fremtiden kan ha et annet syn eller behov som ikke var påtenkt i utviklingen av det nåværende system. Begge systemene vektlegger at de har en åpen arkitektur som kan tilpasses slike behov, og de har begge mulighet for utveksling av informasjon igjennom meldingsutveksling, selv om de har ulike løsninger på det. Dette er veldig positivt med tanke på fremtidig bruk av informasjonen.

Ulikhetene mellom systemene er mange, og utviklingshistorikken for dem er også ulik. DIPS ble konstruert for medisinsk dokumentasjon, mens DocuLive er et eksisterende tekstadministrasjonssystem som er tilpasset formålet. På sykepleiedokumentasjonssiden ligger DIPS muligens ett hestehode foran på grunn av at de har valgt NANDA og NIC som kan kodifiseres, mens DocuLives valg av VIPS-modellen ikke uten videre kan mappes opp i høyreliggende nomenklatur som SNOMED. Det er i denne sammenheng verdt å nevne at verken NANDA eller NIC foreløpig offisielt anbefalt av NSF som standard rammeverk for

sykepleiedokumentasjon, men det er tegn som tyder på at det vil bli det. Det finnes uenighet blant fagpersoner innen sykepleiefaget rundt hvilket rammeverk som best balanserer behovene for struktur og praktisk nytte.

Kommunikasjon med EPJ-systemene

Kommunikasjon med DocuLive: Hvordan DocuLive faktisk kan kommunisere med andre systemer, har det ikke lykkes meg å finne ut. I forbindelse med Soarian sier de at dette kan foregå "sømløst". Siden Soarian er en webbasert portal, vil bruk av XML og XSLT være aktuelle teknologier. De hevder at det er gjort stor innsats for å være kompatibel med meldingsutveksling, noe som antagelig innebærer bruk av HL7.

Kommunikasjon med DIPS: Systemet er bygd opp modulært, og kan benyttes innen de fleste områder av journalføring. DIPS har en politikk om at andre systemer skal kunne kommunisere med deres system, derfor har de laget et grensesnitt som gjør det mulig å utvikle moduler som kan integreres i systemet, det såkalte ”DIPS Link”. Integrasjonen gjøres enten via DIPS-Link, et programgrensesnitt mot DIPS, via DIPS DCOM-Server som er et Com/XML-grensesnitt mot DIPS eller via direkte integrasjon fra DIPS. I tillegg benyttes også EDIFACT som er godkjent fra ISO (International Organisation for Standards) og er en standard for syntaks i databaser. Standarden er blant annet nyttig for meldingsutveksling (ISO 9735- 9:2002)[48].

3.3 Klassifisering

I dette kapittelet omtales klassifiseringsystem for sykepleieobjekter; diagnoser, tiltak og evalueringer.

Det må generelt foreligge en del betingelser for å bedrive klassifisering[25] (engelsk):

1) Domain completeness
2) Nonoverlapping classes (mutual exclusiveness)
3) Suitable for its purpose
4) Homogeneous ordering (one principle per level)
5) Clear criteria for class boundaries
6) Unambiguous and complete guidelines for application
7) Appropriate level of detail

For å lage maskinassisterte kodesystemer må i tillegg disse forholdene foreligge (engelsk):

1) Allow for the use of synonyms
2) Allow for the use of lexical variations
3) Insensitive to spelling errors
4) Reliability
    a. consistent operation (insensitive to ordering of terms)
    b. correct

Årsaken til at en ønsker å kunne klassifisere, kategorisere eller sette diagnose på sykepleie, har mange begrunnelser. Planlegging og evaluering av pleie er ett nivå, faglig utvikling, økonomiske aspekter og forskning et annet. Sammenknytningen av sykepleie og informasjonsteknologi forbedres gjennom et godt klassifikasjonsverk.

Det har blitt jobbet med å kunne klassifisere sykepleie i forhold til dette i flere tiår. Det startet gjerne som lister eller en til to aksiale systemer, men generelt kan en si at maskinassisterte kodesystemer for sykepleie har sin oppstartstid fra tidlig på 90-tallet. NSF, gjennom SfID, har laget en tidligere omtalt veileder som kom ut i august 2002, og ble revidert i 2003. I revisjonen er det lagt inn et nytt kapittel om klassifikasjoner, koder og terminologier [24].

NSF sier dette om klassifikasjon på sine hjemmesider [62]:

"Klassifikasjonssystemer i sykepleien er systematiske framstillinger av begreper som er spesifikke for sykepleiefaget. Slike systemer er nødvendige for å synliggjøre sykepleieres bidrag til pasientenes helsetilstand. Dette er en av intensjonene som ligger til grunn for utvikling av sykepleieklassifikasjoner."

Det finnes forskjellige måter å strukturere og klassifisere sykepleie etter. NSF angir disse som de viktigste klassifiseringssystemene for norsk sykepleie [62]:

- NANDA - North American Nursing Diagnosis Organisation
- NIC - Nursing Intervention Classification
- NOC - Nursing Outcome Classification
- ICNP - International Classification of Nursing Practice
- HHCC - Home Health Care Classification system (også kalt CCC eller Sabaklass)

I dette kapittelet vil noen av dem bli presentert. I tillegg vil et høyereliggende nomenklatur, SNOMED, bli presentert, og behovet for standardisering nevnes ved å vise til et utvalg institusjoner som arbeider med dette.

3.3.1 VIPS

VIPS står for Velvære, Integritet, Profylakse og Sikkerhet. Det er ikke ett kodeverk, men en systematisering av ulike pleiedata (se Vedlegg C). Nøkkelbegrepene finnes i ulike former i en stor del av sykepleielitteraturen og representerer innholdet i sykepleie - en standardisering og strukturering av sykepleiernes dokumentasjon. I Sverige har VIPS-modellen fått stor utbredelse, og her i Norge har Rikshospitalet innlemmet modellen i klinisk bruk. VIPS har også vært organiseringsform i EPJ-systemer til en viss grad.

VIPS-modellen bygger ikke på en spesiell sykepleieteori eller definisjon. Den kan dermed benyttes sammen med ulike teoretiske perspektiver, dokumentasjonssystemer og innenfor ulike spesialområder. VIPS-modellen er konstruert hierarkisk i tre nivåer og kan derfor implementeres i IT-systemer. Det overordnete nivå er en operasjonalisering av sykepleieprosessen som består av sykepleieanamnese, sykepleiestatus, sykepleiediagnose, planlagte og utførte sykepleietiltak, sykepleieevaluering og sykepleiesammenfatning. Disse kalles hovedsøkeord. På neste nivå i modellen er hovedsøkeordene sykepleieanamnese, sykepleiestatus og sykepleietiltak operasjonalisert i søkeord. Eksempler på søkeord er; sosial bakgrunn, livsstil, aktivitet respirasjon/sirkulasjon, stell, legemiddelhåndtering og miljø. Siste nivå er operasjonalisert i undersøkeord som gir forslag til innholdet i søkeordene. Eksempel på innholdet i søkeordet miljø er; tilpassing og strukturering av miljøet/omgivelsene fysisk, psykisk og sosialt. Selve dokumentasjonsinnholdet kan være i fritekst under søkeordene [24].

VIPS kan integreres etter de ønsker og behov brukerne av et system måtte ha. Som eksempel på dette kan DocuLive nevnes. Bruk av VIPS-modellen på dokumentasjonen representerer et godt utgangspunkt for organisering og bruk av dokumentasjonen, samt at det vil gjøre mapping til kodifiseringsverk enklere enn om teksten hadde ingen struktur. VIPS foreligger ikke som en ferdig definert datastruktur og siden VIPS ikke er et kodeverk, kan det ikke automatisk mappes inn i andre kodeverk eller høyereliggende nomenklaturer. Friheten modellen gir, skaper problemer i så måte.

3.3.2 Sabaklass 2.0

Bakgrunn for systemet

Sabaklass 2.0N er den norske oversettelsen av Clinical Care Classification (CCC), tidligere kjent som Home Health Care Classification (HHCC) System. Systemet ble anerkjent formelt av ANA (Amercian Nurses Assosiation) i 1991.

Sabaklass består av to supplerende taksonomier: Sabaklass Taksonomi for Sykepleiediagnoser og Sabaklass Taksonomi for Sykepleietiltak. Hver av de to taksonomiene benytter "behandlingskomponenter" som danner et standardisert rammeverk for dokumentasjon av pasientbehandling i kliniske settinger.

Klassifiseringssystemet oppstod i forbindelse med "The Home Care Project" ved Georgetown University School of Nursing utført av Virginia Saba mfl. Prosjektet ønsket å kunne vurdere pasienters behov for ressurser og klassifisere pleien for å kunne måle resultatet av pleien de fikk. Det ble samlet inn pleiedata som kunne beskrive objektivt slike behov. Forskningen baserte seg på den største samling data om pasienter som fikk hjemmesykepleie noen sinne [69].

De to taksonomiene diagnose og tiltak ble konstruert med basis i de innsamlede opplysningene som ble lagt inn i en database. Man utviklet en metode for å beskrive vanlig brukte begrep, og man appliserte så sorteringsteknikker for å samle begrepene i grupper. Man matchet de sorterte opplysningene med pasienter, og baserte taksonomiene således på en blanding av statistisk analyse, empiri og klinisk vurdering. Rammeverket kan ivareta dokumentering, vurdering og evaluering av (hjemme)sykepleie over tid, på tvers av settinger, pasientgrupper og geografisk plassering. Det kan beskrive trinnene i sykepleieprosessen: vurdere, diagnostisere, måle, gjennomføre og evaluere [67],[69].

Oppbygging og innhold

Sabaklass' to taksonomier, diagnose og tiltak, inneholder begge 21 "Behandlingskomponenter" som representerer funksjonelle, helseatferds-, fysiologiske og psykologiske handlingsmønstre for pasientbehandling. Disse behandlingskomponentene knytter de to taksonomiene sammen, og kan brukes med andre helserelaterte klassifikasjonssystemer.

Diagnoser består av 176 termer - 54 overordnede diagnostiske kategorier, 122 underordnede kategorier. Man sammenlignet innsamlet materiale (40 361 diagnoser og pasientproblemområder) og NANDA i utviklingen. NANDA termer ble valg og tilpasset, men i tillegg ble 50 diagnoser spesifikke for hjemmesykepleie inkludert.

Tiltak består av 197 termer - 72 hovedkategorier og 125 underkategorier. Det innsamlede bakgrunnsmaterialet (73 529 tiltak og oppgaver), og HCFA's behandlingskoder dannet grunnlag for disse.

Taksonomienes behandlingskomponenter kodifiseres ved hjelp av fem alfanumeriske tegn. Kodestrukturen er basert på WHO's ICD-10. Det første tegnet (A til U) representerer behandlingskomponenten, de to neste tegnene er tall som representerer hovedkategori for diagnoser og tiltak (01-*nn*). Om hovedkategorien har et underpunkt (1-9), følger punktum eller mellomrom, og det femte tegnet er modifikator for dette underpunktet. Modifikatorene beskrives som resultatkriterie for sykepleiediagnose (1, 2 eller 3), og handlingstype for sykepleietiltak (1, 2, 3 eller 4).

Denne oppbygningen brukes til å lage en ”forventet resultat-akse” og en ”aktuelt resultat-akse”. Disse (de nevnte modifikatorene 1,2,3) kan splittes opp i tre resultatkriterier:

- forbedret
- stabilisert
- forverret

En kan slik måle eller evaluere omsorgsprosessen, eller måle endringer underveis i behandlingen. Rammeverket fordrer 4 handlingstyper (de nevnte modifikatorene 1, 2, 3, 4):

- Vurdere/Observere - samle og analysere data om helsestatus
- Behandle/Utføre - gjennomføre en terapeutisk handling
- Veilede/Undervise: tilføre kunnskap og ferdighet
- Administrere/Henvise - koordinere og videreføre

Samhandling med andre systemer og standarder

Sabaklass er registrert som HL7 språk og kan integreres med:

- LOINC (Logical Observation Identifiers Names and Codes)
- SNOMED RT (Systemized Nomenclature of Mecicine - Reference Terminology)
- UMLS (Metathesaurus of Unified Medical Language System fra NLM (National Library of Medicine))
- CINAHL (Cumulative Index for Nursing and Allied Health Literature).

Sabaklass er godkjent av en rekke standardiseringsorganisasjoner, som ANSI, HISB, ICNP, ICN, ISO, og såkalte HHAs.

3.3.3 NANDA

Bakgrunn for systemet

NANDA er et sykepleiediagnoserammeverk utviklet av North American Nursing Diagnoses Association. NANDA regnes som en pioner innen diagnostisk klassifisering av sykepleie, da de allerede i 1973 opprettet en gruppe for å jobbe med denne problemstillingen på "First National Conference on Classification of Nursing Diagnoses". Det ble til et samarbeidsprosjekt med kanadierne i 1982. De er en medlemsdrevet organisasjon der praktiserende sykepleiere selv skal ha mulighet til å påvirke utviklingen av klassifikasjonssystemet[9].

NANDA's taksonomi har blitt revidert mange ganger, og er stadig i utvikling. I Norge er det NSF's forlag ”Akribe” som utgir rammeverket. Det publiseres i form av en bok, og siste versjon er fra 2001-2002, men under oversettelsesfasen kom det ut en ny revisjon av NANDA [19]. Utviklingen har gått fra å være en liste på 80-tallet, til å bli et konseptuelt system. Systemet blir kalt Taksonomi II, og ble første gang presentert på NANDA-konferansen april 2000. Nanda er et selvstendig rammeverk, men nevnes ofte sammen med taksonomier for tiltak (NIC) og resultater (NOC). Disse utgjør en omfattende klassifisering av sykepleiens problemområder.

Oppbygging og innhold

NANDA Taksonomi II består av domener, klasser, diagnostiske begreper og diagnoser. Taksonomien sies å være multiaksial, dvs at den har 7 akser som en klassifiserer etter. En akse er her definert som *"en dimensjon av den menneskelige reaksjon som blir vurdert under den diagnostiske prosessen"* [19]. Sykepleiediagnoser kan formuleres ved hjelp av et ”PES statement (problem, etiology og signs & symptoms)”:

Problem	*Eitology*	*Signs & Symptoms*
Diagnostic label	Cause/contributing riskfactors	Defining characters

Den multiaksiale strukturen gjør at deskriptorene er adsklit fra det diagnostiske konseptet. Dette gjør at en kan skape et kontrollert vokabular. Men NANDA advarer også mot at denne strukturen gjør det mulig å konstruere meningsløse diagnoser (som f. eks *svekkede dagliglivets aktiviteter, foster*)

Aksene er representert i de navngitte/kodede sykepleiediagnosene ved sine verdier. I noen tilfeller er de eksplisitt navngitt, i andre tilfeller er de implisitt. I noen tilfeller vil enkelte akser ikke være aktuelle eller relevante. De 7 aksene er:

- Akse1 Det diagnostiske konseptet

  Taksonomi II består av 13 domener, som igjen er delt inn i klasser (se Vedlegg F). I domenene er det diagnostiske begreper og diagnoser. (De er 101 stk. og er ikke vedlagt pga copyright, punkt 2, side 246 i [19]). Disse begrepene er rotbegrepet i det diagnostiske utsagnet.

- Akse2 Tid

  Varighet av en periode, eller et intervall (akutt, kronisk, intermitterende, kontinuerlig).

- Akse3 Tjenestemottaker

  Den delen av befolkningen sykepleiediagnosen retter seg mot (individ, familie, lokalsamfunn, målgruppe).

- Akse4 Alder

  Den varighet av tid eller tidsintervall et individ har eksistert. Det finnes 12 verdier som spenner seg fra "foster" til "gammel eldre voksen".

- Akse5 Helsestatus

  Posisjon eller rangering i et helsekontinuum (i god form, risiko for, aktuell).

- Akse6 Deskriptor

  Bedømmelse som avgrenser eller spesifiserer betydningen av en sykepleiediagnose/ diagnostiske konseptet, 25 angitte verdier.

- Akse7 Topologi

  Alle typer deler / områder av kroppen (vev, organer, anatomiske steder eller strukturer, 17 aktuelle verdier).

Samhandling med andre systemer og standarder

NANDA kan samhandle med bl.a. NIC, NOC, SNOMED og UMLS

3.3.4 NIC

Bakgrunn for systemet

NIC er en forkortelse for Nursing Interventions Classification og er et klassifikasjonssystem for sykepleietiltak. Boken "Nursing Interventions: Treatments for Nursing Diagnoses" kom først ut i 1985. 1987 formet de et forskningsteam for sykepleietiltak ved universitetet i Iowa. På 90-tallet ble NIC anerkjent av ANA, og inkludert i UMLS. NIC anerkjennes av flere tunge aktører innen sykepleie, og oversettes til flere språk. I 2000 dannes "the NNN Alliance" som fokuserer på samhandling mellom NANDA, NIC og NOC, og NIC er i SNOMED. NIC fremstår i dag med 4. revisjon.

Klassifiseringen inneholder tiltak sykepleieren iverksetter hos pasienten og for pasienten, både direkte og indirekte. Nytteverdien av å klassifisere sykepleietiltak ligger i klinisk dokumentasjon, videreføring av informasjon, integrasjon av data over ulike systemer og settinger, effektivisere forskning, effektivitetsmålinger, kompetanseevaluering, avlønning og undervisningsopplegging [33].

Utviklerne av NIC definerer tiltak som *"any treatment, based upon clinical judgment and knowledge, that a nurse performs to enhance patient/client outcomes."* Det er meningen at NIC skal kunne brukes i alle settinger, fra akutt til langtidspleie, i intensivavdelinger og i hjemmesykepleie. Det finnes et norsk utvalg som jobber med å oversette NIC til norsk, og har ett mandat til å forvalte oversettelsen (Norsk Redaksjonsutvalg for klassifikasjonssystemene NANDA, NIC og NOC). Den siste publikasjonen, og som redaksjonsutvalget har basert oversettelsen på, er [7].

Oppbygging og innhold

Et NIC-tiltak består av et hovedbegrep som det er knyttet en unik kode til, en definisjon og en rekke tilknyttede sykepleiehandlinger/forordninger som er de konkrete aktivitetene pleieren gjør for å utføre NIC-tiltaket. NIC-systemet består nå av 486 tiltak og til sammen mer enn 12000 aktiviteter/forordninger [33].

Inndelingen foregår i såkalte nivå, og det er 3 av disse. Disse består av domene, klasse og tiltak. I NICs 4 revisjon er det sju domener, 30 klasser og 514 tiltak.

De 7 domene er (engelsk):

- Physiological: Basic
- Physiological: Complex
- Behavioral

- Safety
- Family
- Health System
- Community

Disse domenene er identifisert av et tall (1til 7) og klassene under dem har en bokstav som angir hvilken klasse det er. Det er forskjell på store og små bokstaver i denne sammenheng. Hvert av de ulike tiltakene i klassene har ett unikt nummer (kode) knyttet til seg. Kodene til tiltakene består av fire tall, og er kontekstfrie. Det betyr at de ikke er organisert i en logisk orden (selv om de i utgangspunktet var det i revisjon 2, og det er fulgt opp delvis siden)[7].

Klassifikasjonssystemet er så stort at de vil være vanskelig å forholde seg til for en sykepleier. Tiltakslistene til NIC fremstår som enormt detaljert og det er vanskelig å få oversikt (bokutgaven av klassifikasjonen er totalt på 1062 sider). Riktig bruk av kodene fordrer god støtte i et EPJ-system. "The NIC/NOC Letter" [32] sier at det for hver NANDA diagnose eksisterer en liste mulige NOC utfall, og for disse vil det være en liste av mulige NIC tiltak. Et dataprogram er under utvikling for dette. De stadige revisjonene av rammeverket er et problem både for brukerne og utviklerne av slike system, og kostnadene tilknyttet dette likeså. Heldigvis er kodene i rammeverket unike, og en kode som er utgått brukes ikke på nytt.

Samhandling med andre systemer og standarder

NIC kan knyttes til NANDA, Omaha System og NOC, samt OASIS og RAP (Resident Assessment Protocols). NIC er registrert som HL7 kompatibelt, og er mappet til SNOMED. NIC er anerkjent av ANA, og imøtekommer NIDSEC's (ANA's Nursing Information and Data Set Evaluation Center) retningslinjer. NIC er inkludert i UMLS og CINAL. NIC er også knyttet til JCAHO (Joint Commission on Accreditation for Health Care Organization's) og ansees som et klassifiseringsystem som kan møte standarden for uniform databehandling [7].

3.3.5 NOC

Bakgrunn for systemet

NOC står for Nursing Outcomes Classification, og er et rammeverk for å klassifisere og evaluere effekten til sykepleietiltak fra NIC (Kodeverk for sykepleieresultater/evaluering). Nytteområder for NOC er standardisering av dokumentasjon, bruk av klinisk informasjon og

utvikling av kunnskapsgrunnlag i sykepleie. Utviklingen av NOC startet i forbindelse med et forskingsteam som hadde fokus på evaluering i 1991, og beskrives i fem faser:

Fase I: pilotarbeid for å teste metodologien (1992-1993)

Fase II: konstruering av resultatene/evalueringene (1993-1996)

Fase III: konstruering av taksonomien og kliniske tester (1996-1997)

Fase IV: evaluering av målingsskalaer (1998-2002)

Fase V: raffinering og klinisk bruk (1997- dd)

Sykepleieresultater er et mål for tilstand, på individuelt, familiært eller samfunnsnivå. Oppførsel eller oppfattelser kan måles kontinuerlig og påvirkes av sykepleietiltak. Rammeverket skal kunne benyttes i alle settinger og for alle pasientpopulasjoner. Siden evalueringene beskriver pasientstatus, kan andre yrkesgrupper benytte dem for evaluering for deres tiltak.

Oppbygging og innhold

NOC kommer nå i 3. revisjon, og består av 330 resultater (311 individuelle, 10 familiære og 9 samfunnsrelaterte). Hver av disse har en definisjon, en liste av indikatorer som kan brukes til å evaluere pasientstatus i forhold til evalueringene, en målrangering, kildehenvisning for bakgrunnsdata, en fempoengs skala for pasientstatus, og en kort referanseliste for utviklingen av resultatet/evalueringen. 76 av evalueringene har en tilleggsskala.

NOC evalueringene grupperes i 7 domener (engelsk):

- Funcitonal Healt
- Physiologic Healt
- Psychosocial Healt
- Healt Knowledge & Behavior
- Percieved Healt
- Family Healt
- Community Health

Domene er delt inn i til sammen 31 klasser, og de 330 evalueringene hører under disse. Hver av evalueringene er identifisert av et unikt kodenummer. Klassifiseringen er under

kontinuerlig oppdatering for å imøtekomme nye evalueringer, eller revidere gamle der det er nødvendig. Dette foregår etter en fire års syklus [34].

Samhandling med andre systemer og standarder

NOC evalueringer kan knyttes til NIC og NANDA. I tillegg kan det benyttes til Omaha Systems' problemer, Gordons funksjonelle mønster, Taxonomy of Nursing Practise, OASIS, og ulike systemer brukt i sykehjem (RAPs - Resident Admission Protocols). ANA har anerkjent NOC, og det er inkludert i UMLS og SNOMED mapping er underveis. Det har blitt godkjent for bruk gjennom HL7. Det er oversatt til flere språk, og norsk oversettelse planlegges.

3.3.6 SNOMED

Bakgrunn for systemet

SNOMED (CT) er en forkortelse for Systemized Nomenclature of Medicine - Clinical Terms. Systemet har sin bakgrunn fra SNOP (Systemized Nomenclature of Pathology), og har også vært benyttet for veterinærmedisin. SNOMED ble første gang utgitt i 1974, eies av College of American Pathologists (CAP), og det nåværende navnet er SNOMED International [59].

SNOMED er ikke et klassifiseringssystem for sykepleiedokumentasjon, men et nomenklatur for medisinsk terminologi. Systemet er stort, og sykepleieklassifikasjon inngår som en liten del av det. Systemet har internasjonal utbredelse, og kan derfor benyttes for å bedre internasjonalt samarbeid innen helserelaterte problemstillinger.

Oppbygging og innhold

SNOMED International vedlikeholder SNOMED CT's tekniske design, arkitekturen for og det faktiske hovedinnholdet (content core). I dette ligger også concept table, descripton table, relationships table, history table, og teknisk referanseguide. I SNOMED International er diagnostiske termer fra ICD (det internasjonale kodeverket for klassifisering av diagnoser) inkludert i disease/diagnostic modulen (D-kodene). SNOMED finnes på flere språk (engelsk, spansk, tysk m.fl.), og norsk oversettelse vurderes av KITH. SNOMED består av (forskjellige tall finnes):

- 11 akser (uavhengige moduler)
- 19 begreper på toppen i systemet
- 344.000 begreper totalt i dag

- mer enn 975 000 beskrivelser av begreper
- 1 362 960 relasjoner mellom begreper

SNOMED er stort, det leveres i dag på 4 CD-rom disker (eller DVD-rom), og estimeres til over 12 GB ukomprimert [43].

SNOMED består av hovedinnhold (Core Content), en rekke begreper organisert etter et system av hierarkier og akser, som også har innbyrdes relasjoner (interrelaterte definisjoner og begreper). Det betyr at begreper med lik mening/innhold plasseres i samme hierarki. Nomenklaturet er bygd opp av akser (11 stk) der hver av disse er et komplett hierarkisk klassifikasjonssystem. Aksene tildeler koder til veldefinerte semantiske kategorier, og kodene vises i en hierarkisk orden. Aksene gjør at en kan skille mellom ulike uttrykk innen f. eks topologi, morfologi osv. De 11 aksene i SNOMED International er vist i Tabell 3-2 [59].

Tabell 3-2 De 11 aksene i SNOMED International

Axis	**Definition**	**Description**
T	Topography	Anatomic terms
M	Morphology	Changes found in cells, tissues and organs
L	Living organisms	Bacteria and viruses
C	Chemical	Drugs
F	Function	Signs and symptoms
J	Occupation	Terms that describe the occupation
D	Diagnosis	Diagnostic terms
P	Procedure	Administrative, diagnostic and therapeutic procedures
A	Physical agents, forces, activities	Devices and activities associated with the disease
S	Social context	Social conditions and important relationships in medicine
G	General	Syntactic linkages and qualifiers

Tabell 3-3 SNOMED's 19 toppbegreper.

attribute	physical force
body structure	physical object
context-dependent categories	procedure
disease	qualifier value
environments and geographical locations	social context
events	special concept
findings	spesimen
observable entity	staging and scales
organism	substance
pharmaceutical / biologic product	

Det finnes 19 toppbegreper i SNOMED, disse er vist i Tabell 3-3.

Generelt kan det sies at det innen konseptene finnes "parent-child" hierarkier, men at det ikke er hierarkier på tvers av konseptene. Det finnes andre typer relasjoner mellom konsepter fra samme eller andre hierarkier (*qualifying* eller *defining relationships*). Det er også en form for beskrivelseslogikk som muliggjør formell representasjon for meningen av konseptene og deres innbyrdes forhold, slik at man algoritmisk kan finne synonymer, redundans og hierarkier.

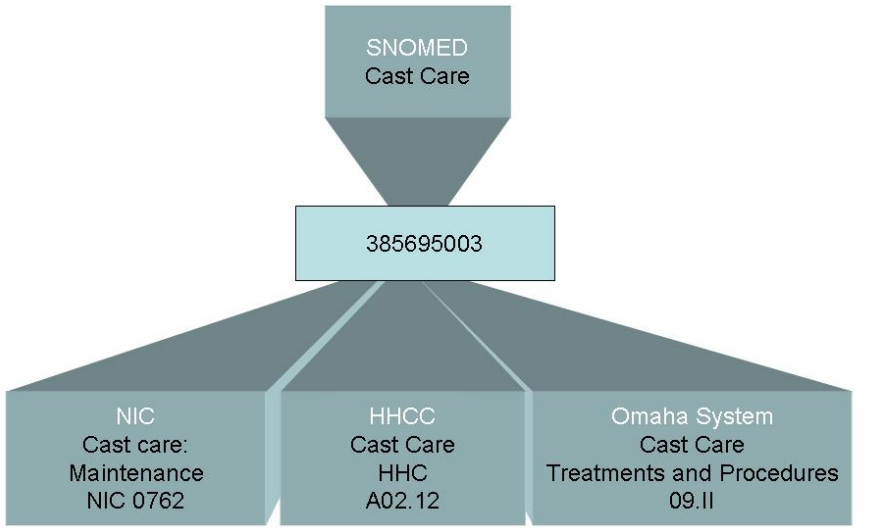
I SNOMED CT kan flere standardiserte SNL (Standardized Nursing Language) bli mappet opp. Eksempelvis kan sykepleiediagnoser fra NANDA finnes innen "Clinical Findings", mens tiltak fra NIC finnes innen "Procedures" og evalueringer fra NOC er modellert inn i entiteter i hierakiet "Observable". Sabaklass er også innlemmet på dette viset fra juli 2004.

Når en har flere klassifiseringer, kan sykepleiekonsepter mappes inn i SNOMED CT på 3 måter:

1) Det finnes et identisk sykepleiekonsept i SNOMED
2) Et sykepleiekonsept er synonymt med et eksisterende konsept, og legges til som et synonym
3) Et sykepleiekonsept er nytt (ikke allerede i SNOMED CT terminologien). I så fall legges de til, gies en kode, modelleres ut ifra nåværende attributter, og plasseres etter en IS_A (parent-child) relasjon, plassert i rette hierarki.

Uansett får konseptet en intern (ikke utgitt) "identifier" som representerer kildeterminologien. Dette er en form for markering som gjør at disse konseptene kan bli entydig identifisert, slik at en kan generere og befolke "mapping tabeller". Disse tabellene gir lenker mellom den spesifikke sykepleieterminologien og SNOMED CT. Som eksempel nevnes det at NIC mapping tabellen identifiserer relasjonslenkene mellom NIC kildeterminologien og SNOMED CT.

Eksempelet i Figur 3-2 er hentet fra [71]:



Figur 3-2 Eksempel på SNOMED/SNL mapping

Mapping foregår fra SNOMED til klassifiseringssystemet, ikke fra klassifiseringssystemet til SNOMED. På norsk, i klassifikasjonene omtalt i dette kapittelet, vil eksempelet i figuren se slik ut:

> NIC: 0762 | Pleie ved gipsbehandling: Vedlikehold (Norsk oversettelse av NIC fra [40])
>
> Sabaklass: A AKTIVITET 02.1 - Gipsbehandling - Handlinger utført for å håndtere gips (Norsk oversettelse av Sabaklass 2.0N)

Samhandling med andre systemer og standarder

I denne sammenhengen er det kanskje mest interessant å vite hvilke "mappings" som finnes for SNOMED CT og sykepleieklassifiseringer. Disse eksisterer i dag:

- NANDA Nursing Map
- NIC Nursing Map
- NOC Nursing Map

- PNDS Nursing Map
- HHCC (kjent som Sabaklass) Nursing Map (juli 2004)
- Omaha System Nursing Map (juli 2004)

3.3.7 Om rammeverkene

Å bruke rammeverk på sykepleie er ikke helt problemfritt. En har etter hvert fått et knippe klassifiseringssystemer for sykepleie, og selv om noen ser ut til å få preferanse, kan det bli uenigheter om valg blant disse. Grimsmo og Broseet [10] peker på at sykepleiedokumentasjon med sine pleie- eller tiltaksplaner skiller seg litt ut i klassifiseringssammenheng. Kravene til strukturering for å få til prospektive tiltaksplaner med målsetninger og resultatevaluering er høye, men hvor langt en skal gå i denne sammenheng er fremdeles uklart. De mener det gjenstår betydelig forskning rundt både struktureringen, utviklingen og evalueringen av sykepleiens klassifikasjonssystemer. I en oversikt fra 1998 [12] var det ingen av taksonomiene NSF nevner som viktigst som møter alle krav foreslått av CPRI (Computer-based Patient Record Institute) til karakteristika som kreves for å innlemmes i EPJ-systemer. Dette er viktige aspekter for fremtidens bruk av informasjonen; forskning, beslutningsstøtte og kvalitetsmåling.

Felles for alle de nevnte klassifikasjonssystemene i dette kapittelet er at de er utviklet i USA, og de blir kritisert for at de er tilrettelagt for andre forhold enn norske, gir et reduksjonistisk menneskesyn, passer best for et stereotypisk bilde på hva en pasient er, og har problemer med validitet og reliabilitet. Begrepet "sykepleiediagnose" er også omdiskutert da det får tankene mer over mot medisinsk språkbruk [24]. Slike ting er vanskelig å unngå når en forsøker å klassifisere, men det er utvilsomt fordeler for databehandling at en kan begrepsfeste eller klassifisere dokumentasjonen.

Sabaklass ble utviklet med hjemmesykepleie og ambulatorisk behandling generelt i tankene, og skal nå fungere som et felles språk for sykepleie og andre helsetjenester. NANDA har et mer universelt utgangspunkt, og er spesielt utviklet for å fungere på tvers av systemer og i alle settinger der sykepleie blir utøvd. NANDA hadde i en utarbeidelsesfase 21 domener, men vurderte det til for mange til å være praktisk. De opererer i dag med 13 domener, og et varierende antall klasser under disse (til sammen 46). Sabaklass opererer med 21 behandlingskomponenter for sine diagnoser. Til gjengjeld har Sabaklass en tilhørende tiltaks-taksonomi, som er likt oppbygd. NANDA opererer ikke med tiltak, men NIC er et naturlig valg i så måte. I en sammenligning av disse klassifikasjonssystemer i NIC-boken [7] hevdes

det at det er uklart hvor mye Sabaklass brukes til tross for sin store spredning. De sier også at Sabaklass ikke har vært oppdatert siden sin første publikasjon i 1993 [7].

3.3.8 Standardiseringer

Innen EDB har standardisering og valg av foretrukket teknologi alltid vært et stridens eple. Fra opprinnelsestiden til Internet har vi striden om foretrukne og ”beste” protokoller. Standardiseringer for optiske lagringsmedia er i dag er et annet høyaktuelt diskusjonstema. For begge er det interessant å se at gamle løsninger får nytt liv når situasjonen endrer seg, og at en er nødt til å dra med seg gamle løsninger i lang tid når en lager nye standarder.

Sykepleie er universell, og et felles "språk" styrker kunnskapsutveksling. Likevel er det fryktelig mange faktorer å vurdere med tanke på kvalitet, kostnad, informasjon og kunnskap, og hva struktureringen gjør med disse faktorene. Det finnes institusjoner i Norge som spesielt (og generelt) ser på standardisering og klassifisering i helsevesenet, for å hjelpe til med valg og utvikling for norske forhold.

KITH

KITH (Kompetansesenter for IT i Helse- og Sosialsektoren) ble etablert som aksjeselskap i 1990 og eies i dag av Helsedepartementet (59,5 %), Kommunenes sentralforbund (30 %) og Sosialdepartementet (10,5 %). Frem til juni 2003 eide Sør-Trøndelag fylkeskommune 30 % av aksjene i KITH. Disse ble som et resultat av helsereformen overtatt av Helsedepartementet før generalforsamling 24. juni 2003. KITH jobber med standardisering av IT i helsevesenet, men ikke direkte med spesifikke journalsystemer. Samordning, med den effektivisering og utvikling det medfører, kan spare helse-Norge for store utgifter. Standardisering vil være en viktig del for å få til en slik samordning.

KITH har følgende kjernevirksomhet:

- Standardisering - Etablering av formelle, tekniske eller helsefaglige krav, retningslinjer eller beskrivelser til gitte problemstillinger innen området helseinformatikk.
- Kravspesifisering - Spesifisering av generelle krav til datagrunnlag og brukerfunksjonalitet i IT- systemer i helsevesenet, samt generelle krav til kommunikasjon og integrasjon mellom ulike IT- systemer.
- Rådgiving - KITH skal være helsevesenets sentrale rådgiver og kompetansesenter innenfor områdene helseinformatikk og IT.

Fra 2005 til 2010 vil KITH jobbe med "Strategi for Standardiserings- og Samordningsprogrammet" forkortet til SSP. Det er spesielt meldingsutveksling som er målet med programmet, men dette drar med seg flere elementer, bla koding og klassifisering (KoK-programmet). Avgrensing av hva som hører innen hvilket program, ivaretaes ved kontakt mellom programmene. SSP har disse hovedsatsningsområdene:

- Informasjonsutveksling
- Informasjonssikkerhet
- Elektronisk pasientjournal (EPJ)

Siden sykepleiedokumentasjonen er en del av EPJ-systemene, gjelder disse tingene også for denne typen dokumentasjon. I 2002 bevilget Sosial- og Helsedirektoratet midler til Sørlandet sykehus Arendal, som i samarbeid med KITH utarbeidet "Kravspesifikasjon for elektronisk dokumentasjon av sykepleie – Nasjonal standard" [1]. Denne ble godkjent i 2004. I tillegg har NSF selv laget en veileder [24]. For å følge opp dette arbeidet, ble det satt i gang et prosjekt for å utvikle et elektronisk dokumentasjons- og beslutningsstøttesystem for sykepleie. Dette støttesystemet bygger blant annet på klassifikasjonssystemene NANDA, NIC, NOC og ICD-10. Disse oversettes i regi av NSF og gies ut av Akribe forlag, men Sosial og Helsedirektoratet forutsetter at KITH er med for å sikre en offisiell norsk oversettelse[6].

S@mspill

S@mspill 2007 er Sosial- og helsedirektoratets strategi for elektronisk samhandling i helsevesenet. Forløperen til planen het "Si @!". ”Si @!” hadde som formål å få til et Nasjonalt helsenett, standardisering av elektroniske meldinger, telemedisin og økt samhandling med publikum via internett. I S@mspill 2007er fokuset fremstilt slik:

- Hovedsatsing 1: god informasjonsflyt
    - Helhetlig og veldefinert informasjonsgrunnlag
    - Nasjonalt helsenett
    - Informasjonssikkerhet
    - Elektronisk pasientjournal
    - Konsolidere utbredelse av elektronisk meldingsutveksling
    - Fagstøtte og kunnskapskilder
- Hovedsatsing 2: elektronisk samarbeid med nye aktører
    - Involvere pasienter, brukere og pårørende
    - Elektroniske resepter til apotekene og forskrivningsstøtte

- Elektronisk samarbeid mellom kommunal helse- og sosialtjeneste og spesialisthelsetjenesten

I forhold til denne strategien vil også strukturering av sykepleiedokumentasjonen få stor betydning, særlig med tanke på kommunikasjon mellom de ulike sykepleietilbudene, både mellom sykehus, hjemmesykepleie og sykehjem. Løsningene må basere seg på gjeldende standarder og kravspesifikasjoner, eksempelvis kravspesifikasjon for elektronisk dokumentasjon av sykepleie og elektronisk dokumentasjonssystem for pleie- og omsorgstjenesten [74]. S@mspill planen nevner også ELIN, et samarbeid SHdir/KITH/Nasjonal ITK, som skal bidra til enighet om felles krav og innhold i standarder og løsninger for samhandling mellom pleie- og omsorgstjenestene, helseforetak og allmennleger.

NSEP

Nasjonalt Senter for Elektronisk Pasientjournal ble opprettet ved NTNU (Norges teknisk-naturvitenskapelige universitet) i 2003, med midler fra Norges Forskingsråd. Det overordnede målet er forskning som kan bidra til utvikling og implementering av elektroniske pasientjournalsystemer som kan forenkle hverdagen for helsepersonell. Senteret tilhører det medisinske fakultet, Institutt for nevromedisin, men skal styrke og utvikle et tverrfaglig forskningsmiljø med kompetanse fra helse, IKT- og samfunnsfag. Senteret har fått tildelt midler for fem års drift. Foruten representanter fra faggrupper på universitetet, er styret satt sammen av representanter fra næringslivet, helseforetaket, KITH og norsk forskningsråd.

NSEP skal drive forskning og kunnskapsutvikling i front av anvendelsen av EPJ-systemer i helsetjenesten. Forskningen vil ta utgangspunkt i problemstillinger som har sitt utspring i helsetjenesten, men som også har et langsiktig perspektiv som kan produsere ny grunnleggende og generisk kunnskap, og slik ha et potensial til industrirealisering.

NSEP skal etablere prosjekter som til sammen har en bredde tilstrekkelig til å styrke og bygge opp kompetanse innenfor pasientjournalens tre hovedfunksjonsområder som [61]:

- dokumentasjons- og arbeidsverktøy (opprinnelig funksjon)
- kommunikasjons- og samhandlingsverktøy
- informasjons- og kunnskapsbase

For utvikling av funksjonalitet knyttet til sykepleiedokumentasjon, betyr NSEP en mulighet til tverrfaglig diskusjon med andre profesjoner, og en mulighet til å se muligheter og begrensninger i løsninger basert på deres kunnskaper. En annen viktig funksjon NSEP har, er

å skape ett kontaktnett mellom faggrupper for å få en mer helhetlig oversikt over mulighetene og problemene. Dette kan få et mer praktisk perspektiv frem, og bidra til felles løsninger.

4 Basisteknologier

4.1 Gjenfinningsteknikker

Informasjonsgjenfinning (ofte forkortet IR for Information Retrieval) har lange tradisjoner innen databehandling, og det er skrevet mye og omfavnsrikt om temaet. IR er representasjon, lagring, organisasjon av og tilgang til informasjonsobjekter [2]. Målet er lett tilgang til informasjon som brukeren er ute etter, men det er allment kjent at dette ikke er enkelt. En kan dele problemstillingen i to: det brukeren leter etter (spørring, query), og det man leter i (dokumentasjon, dokumentsamling). Det finnes mange måter å behandle spørringer, dokumentsamlinger og svardokumentene på.

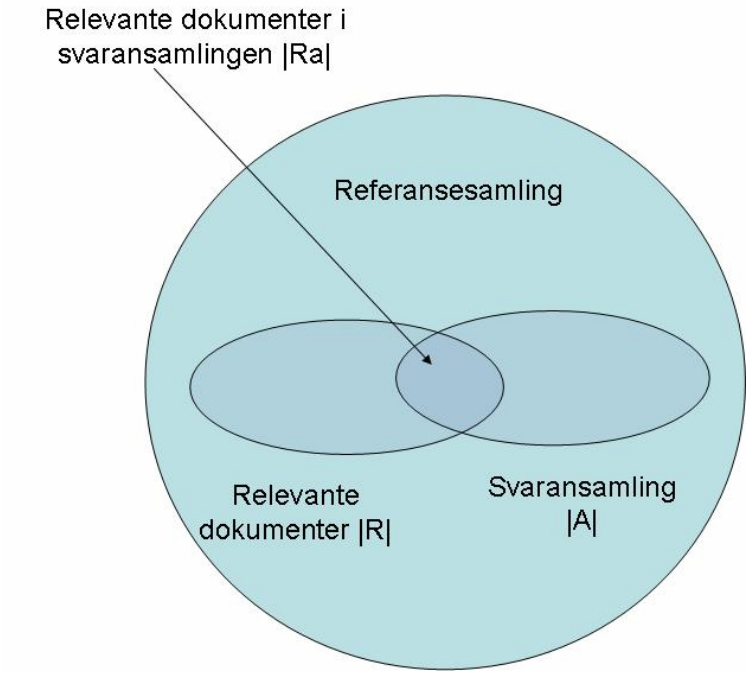
Det vil i det følgende beskrives noen tradisjonelle gjenfinningsteknikker. Dette kapittelet konsentrerer seg mest om dokumentsamlingen sett som ren tekst uten annen struktur enn en rekke ord. Vektorrommodellen (VSM) beskrives i større detalj enn de andre, da denne er valgt ut i forbindelse med implementasjonen av en prototyp.

4.2 Presisjon, tilbakekalling og spørring

En kan ikke snakke om gjenfinning av ustrukturert tekst uten å nevne presisjon (precision) og tilbakekalling (recall). Dette er ord som brukes når en evaluerer søkings kvalitet (det finnes også andre parametere som ikke omtales her; forventet søkelengde, tilfredsstillelse, frustrasjon). Med dette menes at det er ikke sikkert at søketeknikken er presis i sine funn, altså at ikke alle gjenfunnede dokumenter handler om det man var ute etter. Søketeknikken kan også ha for liten tilbakekalling, altså at det i mengden av relevante dokumenter bare ble funnet igjen en andel av relevante dokumenter.

Man har et informasjonsbehov I fra en referansesamling. R er de på forhånd antatte relevante dokumentene til dette behovet. $|R|$ tilsvarer alle dokumentene i referansesamlingen. Man benytter en aktuell søketeknikk (eller strategi - det kan dreie seg om flere teknikker i kombinasjon), og sitter igjen med svaransamlingen A. $|A|$ er antallet dokumenter i denne mengden. $|Ra|$ er mengden dokumenter i snittet av mengdene R og A. Vi kan da si at

- *Precision* - |*Ra*| / |*R*| = Andel relevante dokumenter som hentes frem ved et søk. Tilsvarer spesifisitet.
- *Recall* - |*Ra*| / |*A*| = Hvor mange dokumenter av maks mulige relevante dokumenter i en samling som hentes frem ved et søk. Tilsvarer sensitivitet.



Figur 4-1 Precision - Recall forhold [2]

Oppfatningen av at det er en byttehandel mellom presisjon og tilbakekalling er vanlig - en kan øke tilbakekallingen på bekostning av presisjonen [73]. Vurderingen på kvaliteten av søket er ikke lett å gjøre for en bruker på grunn av at han ikke presenteres med mengder, men heller med en form for rangert liste eller på annet vis organisert fremstilling.

Informasjonsbehovet *I* er ikke det samme som selve spørringen (query). Spørringen må formuleres slik at den avhengig av aktuell søketeknikk dekker informasjonsbehovet best mulig. Det er mange teknikker knyttet til spørringen, og disse kan igjen påvirke precision/recall i forhold til informasjonsbehovet. Selv med strukturert innlagt tekst trenger ikke tilbakekalling av ønsket informasjon gjennom en gitt spørring dekke informasjonsbehovet. Men det er sikrere at du får hva du spør etter om du søker i tekst som har struktur. Generelt kan en si at kostnadene med å forsøke å gi fritekst høy presisjon og god tilbakekalling, øker med nøyaktighetsgraden informasjonsbehovet har, grunnet i arbeidet som må gjøres med preprosesseringen av referansesamlingen. Bland annet av overnevnte grunner er strukturert innlegging og gjenfinning etter på forhånd angitte kriterier til enhver tid

fritekstsøkets overmann. Det betyr ikke at en ikke kan se forholdet fra flere vinklinger, og kategorisering av ustrukturert tekst har sitt anvendelsesområde.

Siden EPJ-systemer allerede er presset i forhold til nettverks og prosesseringskapasitet, er det viktig å forsøke å tilpasse søketeknikk etter informasjonsbehov og hvilken kostnad en kan tillate seg i forhold til systembelastning. Metodene som diskuteres i denne oppgaven er velkjente i sine ytelser, men selve presisjon og tilbakekallingsgraden er implementasjonsavhengig. Jeg må nøye meg med å gjengi andres uttalelser og vurderinger av de valgte teknikkene, og gi en generell oppfattelse av prototypen fordi grunnlaget for precision/recall evaluering ikke er godt nok foreløpig (se senere kapittel).

4.3 Ordbehandling

I vanlig språksammenheng er det vi kaller et ord ikke vanskelig, det er en sammenstilling av bokstaver som gir mening. Ser en litt nærmere på dette, er det ikke så enkelt. For eksempel er forvolder ordet "stopp" i den dagligdagse oppfattingen av det ingen vanskeligheter i utgangspunktet. Men sammenstiller en "stopp" med "stopp?" eller "stopp!" har en på grunn av tegnene "!" og "?" tillagt en annen betydning en man først antok. I datasammenheng er ord (word) ikke nødvendigvis det samme som "ord" beskrevet over, og en opererer med flere begreper for å forsøke å skille dette. Ett eksempel er begrepet term, som i denne sammenheng betyr ordstammen til et ord.

Noen ord betraktes som viktigere enn andre i informasjonssammenheng. Dette belyses av at de 10 mest brukte termene i det engelske spåket utgjør 20-30 prosent av en tekst [46]. Ord man i denne sammenhengen anser som lite interessante, er altså ord som gjør det vanskelig å skille det ene dokumentet fra det andre. Man snakker om såkalte "stoppord", ord og termer som kan fjernes med den fordelen at man kan representere teksten som en samling termer som kan bli opptil 40 % mindre enn utgangspunktet (her antaes at like ord representeres en gang, i en såkalt invertert liste) [2]. Ord som er av en slik karakter, er gjerne artikler, preposisjoner, bindeord, noen verb, adverb og adjektiv.

Det klassiske, i flere betydninger enn en, eksempelet på hvilke problemstillinger en slik komprimering av en tekst gir, er utsagnet "Å være eller ikke være". Ved å applisere stoppordlister har vi fjernet hele søkestrengen. Ingen av ordene betraktes som å være betydningsfulle - selv om filosofer kan hevde at dette er alt som er! Alternativt vil ordet "være" bli stemt til "vær" og vi kan ende opp med å finne værmeldingen eller siste nummer av "Sau og Geit". Løsningen på problemstillingen er mange, og frasesøk er nærliggende. Derfor

kan det i vår sammenheng lønne seg å ta vare på hele teksten i tillegg til den inverterte filen ved indeksering [2].

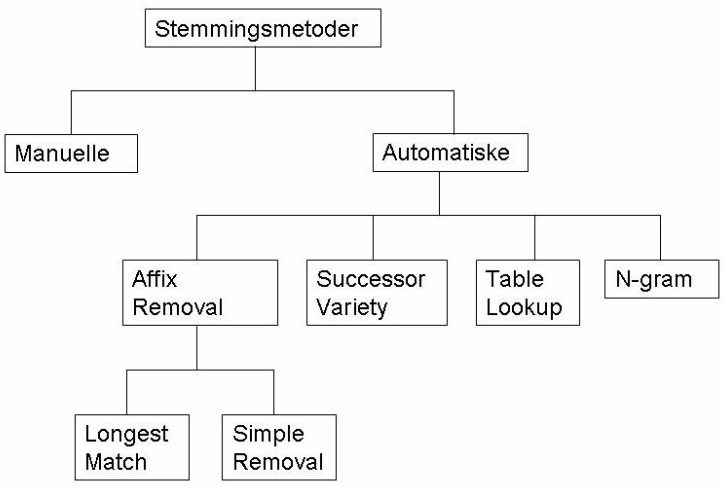
Den stemte termen "vær" har allerede ledet oss til neste problemstilling. Språket består av en rekke ord og fraser som forvansker naturlig språk prosessering, for eksempel:

- Ord som betyr det samme (synonymer)
- Like ord som har ulik betydning (polysemi)
- Språklige bilder
- Verdilading og bibetydning
- Språkutvikling (ordene har annen betydning nå enn tidligere)

Denne listen er ikke komplett. [17] sier at NLP (Natural Language Prosessing) begrenses av størrelsen til ordlister:

"The Webster's Third International Dictionary contains approximately 500,000 entries among them one can evaluate that 200,000 words belong to the medico-technique language ... Therefore, compound expressions in place of single words are quite common in medical language".

Når en snakker om ordbehandling i IR-sammenheng, er det gjerne i forbindelse med å finne ordstammer (grammatikalske røtter). Dette kalles "stemming". Stemming er en delvis omstridt metode å behandle ord på, og det finnes kritikk til å gjøre dette [2]. Kritikken går både på ytelse og resultat. Stemming kan gjøres på flere måter, figuren under viser en oversikt over hovedmetoder. I denne oppgaven omtales metoder fra N-Gram, affix removal og table lookup, alle automatiske stemmingsmetoder.



Figur 4-2 Oversikt "Stemmingsmetoder" [46]

4.3.1 N-Gram

N-Gram stemming er en metode som ikke produserer en ordstamme, men beregner en likhet mellom indeksord ved hjelp av klynger og likhetsutregning [2]. Navnet N-Gram stammer fra begrepet diagram som betyr samling av tegn. *N* refererer til antall tegn (som i bigram, trigram, ..., n-gram).

Metoden går kort ut på å kalkulere unike bigram (eller trigram) for hvert enkelt av ordene i et dokument. Eksempelvis vil ordet "statistical" bestå av

- bigram totalt: st ta at ti is st ti ic ca al
- bigram unike: al at ca ic is st ta ti

Har en flere ord som ligner, f. eks. statistic (at cs ic is st ta ti), finner en de felles unike bigram for disse ordene: at ic is st ta ti.

Alle ordene behandles på denne måten, og en legger resultatene inn i en matrise for likhetsutregning. Det dannes så klynger for ord som er like (termer med like n-gram blir liggende nær hverandre), og slik kommer en frem til en felles ordstamme for de likeste ordene. Problemet er selvfølgelig å finne et fornuftig likhetsmål.

Metoden nevnes her pga den positive effekten metoden har på feilstaving. Feilstaving og skrivefeil forekommer i sykepleiedokumentasjon på lik linje med annen tekstlig dokumentasjon. Når en skal behandle et dokument på basis av enkeltstående ord, kan et feilstavet ord få konsekvenser som påvirker resultatene. Ideelt sett burde feilstavinger og skrivefeil blitt fanget opp ved innleggelse av dokumentasjonen, men det skjer ikke alltid. Feil kan også oppstå som følge av databehandling, f.eks ved OCR (optical character recognition) eller i utvekslingsformat som ikke støtter visse tegnsett.

4.3.2 Affix Removal - Porterstemming

Porterstemming er en vanlig brukt metode for å komme frem til ordstammene til tekst i IR-sammenheng. Metoden hører under "Affix Removal - simple removal" i Figur 2-1. "Affix Removal" er metoder som foretrekkes, fordi de er intuitive, enkle og kan implementeres effektivt [2]. "Affix" kan bety både forstavelser og endinger, og Porterstemmeren er en metode som fjerner endinger. Dette er fornuftig da de fleste varianter av ord skapes av ulike endinger, ikke forstavelser. Dette betyr at ordene "liggesår" og "liggesårene" blir like, fordi "liggesårene" stemmes til "liggesår". Metoden er ikke så raffinert at den får til lemmatisering av ord. Man får ikke likhet med en tekst som omtaler "ligge" og "sår", selv om det hadde vært

ønskelig. Det ville forøvrig være vanskelig å finne et kriterium for når slik oppdeling er ønskelig i indekseringssammenheng.

Algoritmen ble beskrevet av Porter i 1980, og finnes i dag i ulike varianter. Porter driver ett nettsted der utvikling av algoritmen for flere språk og programmeringsspråk fortsatt pågår [72]. Metoden deles inn i flere steg og regler. Siden det er forskjell på språk, er det forskjellige regler for hvor en skal søke etter endinger. Dette kalles en regiondefinisjon, og benevnes R1 og R2 i algoritmen. For norsk er kun R1 gjeldene.

- R1 er regionen etter den første "ikke-vokal"som følger en vokal, eller null ved slutten av ordet om det ikke finnes en slik "ikke-vokal".
- R2 er regionen etter den første "ikke-vokal"som følger en vokal i R1, eller er null ved slutten av ordet om det ikke finnes en slik "ikke-vokal".

Ordet "ikke-vokal" benyttes fordi det kan forekomme andre tegn enn konsonanter. Definisjonen av vokaler varierer fra språk til språk, derfor må algoritmen få dette beskrevet eksplisitt for språket den skal stemme ord fra.

Etter at man har kjørt R1 vil en sitte igjen med variablene C, V, L og ø for henholdsvis konsonant, vokal, vilkårlig bokstav og null. Kombinasjoner av disse utgjør mønster. Man benytter tegnene (*) og (+) for å gjengi om et mønster forekommer i 0-mange eller 1-mange repetisjoner [2].

Alle de følgende stegene må utføres

- Steg 1: Søk etter den lengste av disse endingene i R1, og utfør handling
  - (a) *a e ede ande ende ane ene hetene en heten ar er heter as es edes endes enes hetenes ens hetens ers ets et het ast* --> slett
  - (b) *s*--> slett hvis en gyldig forestående s-ending foreligger (Det kan hende at gyldig s- ending ikke forekommer i R1)
  - (c) *erte ert* --> bytt ut med *er*
- Steg 2: Dersom ordet ender med *dt* eller *vt* i R1, slett *t* (for eksempel meld*t* --> meld, operativ*t* -- > operativ)
- Steg 3: Søk etter den lengste av de følgende endingene i R1, og hvis den finnes, slett den (*leg eleg ig eig lig elig els lov elov slov hetslov*)

Porters algoritme er fritt tilgjengelig, og kan lastes ned fra nettet [72]. Porter har utviklet et eget programmeringsspråk som kalles Snowball, som algoritmen er implementert i. Det

finnes en generator for ulike programeringsspråk fra Snowball, blant annet Java. Koden støtter stemming på 12 språk. Det er denne norske stemmeren som er benyttet i prototypen omtalt i denne oppgaven.

4.3.3 Table Lookup

Table Lookup er en metode som lagrer indekstermer og alle deres stammer i en tabell. En bruker tabellen når en indekserer og utfører spørringer. Problemet med denne metoden er at den er plasskrevende. Fordelen er at oppslag kan gjøres effektivt ved hjelp av hashing-algoritmer. Disse typene algoritmer er blant de raskeste en datamaskin kan utføre. En Table Lookup metode har ikke samme problemer med nøyaktigheten som de andre metodene har, og i tillegg tilrettelegger de godt for lemmatisering av ord (orddeling). Implementasjon er applikasjonsavhengig, og beskrives ikke her.

NorKompLeks

NorKompLeks er et norsk komputasjonelt leksikon (NorKompLeks) utviklet ved NTNU i samarbeid med norsk forskningsråd og Telenor. Arbeidet foregikk ved Institutt for språk- og kommunikasjonsstudier i perioden 1996-2000, og oppryddings- og vedlikeholdsarbeid pågår kontinuerlig. Leksikonet er maskinleselig og finnes får språkformene bokmål og nynorsk. Ordutvalget er primært basert på Bokmålsordboka og Nynorskordboka (begge fra Leksikografiseksjonen ved Institutt for nordistikk og litteraturvitenskap ved Universitetet i Oslo).

NorKompLeks fullform er en tekstfil på 31 Mb. Oppbyggingen er alfabetisk, og ord beskrives med kjønn, form, type og normalform. En finner også de ulike endingene i leksikonet. I tillegg finnes filer for bl.a. grunnform, argumentstruktur. Selv om størrelse er et problem med slike leksikon, kan en ha gevinst i forhold til andre bruksområder i samme implementasjon, f.eks kontrollert vokabular og korrigering av skrivefeil ved innlegging av dokumentasjon. Leksikonet inneholder ikke fagtermer spesielt for sykepleie, men det foregår forsøk på å benytte det til kunnskapsekstraksjon fra elektroniske pasientjournaler ved NSEP, der fokuset bl.a er automatisk anonymisering av journaler og medisinsk språkprosessering.

Oslo-Bergen Taggeren (for bokmål og nynorsk)

En annen spennende ressurs for semantisk disambiguering (forhindre feil meningsfortolkning) er "Oslo-Bergen Taggeren" [76]. Den er et samarbeidsprosjekt mellom Universitetet i Oslo og Universitetet i Bergen, og kan benyttes til å sette grammatiske tagger på ord. Eksempelet i Figur 4-3 er hentet fra "Morfologisk disambiguering" taggeren tilbyr:

"<Pasienten>"	"pasient" subst appell mask be ent
"<spiste>"	"spise" verb pret tr1 i1 tr11 rl9 pa3 pr6 tr4
"<en>"	"en" det mask ent kvant
"<halv>"	"halv" adj ub m/f ent pos
"<bolle>"	"bolle" subst appell mask ub ent
"<grøt>"	"grøt" subst appell mask ub ent
"<til>"	"til" prep
"<middag>"	"middag" subst appell mask ub ent
"<.>"	"$." clb <<< <punkt>

Figur 4-3 Morfologisk disambiguering vha "Oslo-Bergen Taggeren"

Taggeren er heller ikke spesielt tilpasset sykepleiedokumentasjon, men inneholder flere funksjonaliteter, bl.a. "Syntaktisk og navnedisambiguering", noe som kan utnyttes for å anonymisere sykepleiedokumentasjon (grunner til å ville gjøre det omtales senere).

Å sammenligne "Oslo-Bergen Taggeren" med table lookup er muligens feil, men implementasjonen ville i prototypen betydd å koble seg opp mot en server via en SOAP (Simple Object Access Protocol) protokoll og således representert en form for oppslag. Oppslaget via nettet og SOAP tar tid.

4.3.4 Collocations

Collocations, eller sammenstilling, er et utrykk bestående av to eller flere ord som sammenfaller med en vanlig måte å uttrykke seg på [18][45]. Selv om denne oppgavens foreslåtte metoder ikke handler om formalslutninger, finnes det metoder for å forbedre semantikken til funnene noe. Som eksempel kan "hjerteinfarkt" nevnes. Står ordet "hjerteinfarkt" etter ordene "familiær opphopning av" eller "risiko for" har det vesens betydning. Det finnes flere måter å finne collocations på:

- Frekvens
- Gjennomsnitt og varianter av avstand mellom ordet og ordsammenstillingene som er aktuelle
- Hypotesetesting
    - *t*test
    - c^2 test

Her omtales bare frekvens. Frekvenstelling er den enkleste måten å finne collocations i en tekstsamling på. Hvis to ord opptrer sammen ofte, tolkes det som bevis på at de har en spesiell funksjon som ikke bare er et resultat av deres kombinasjon. En må forholde seg til stoppord når en forsøker å finne collocations, fordi slike ord forstyrrer statistikken (ordene *av* og *for* i eksempelet over). Metoden går ut på å telle opp alle bigram, og sortere dem etter hyppighet. Består bigramene bare av stoppord, fjernes de fra listen. Justeson og Katz [15][45] har i tillegg foreslått å lage et filter som kan fastslå om ordene er adjektiv, substantiv eller verb. Metoden er utviklet med tanke på å danne automatiske ordlister, er basert på kvantitet med et minimum av språklig kunnskap tilføyd. Den fungerer best med faste fraser.

Collocations er viktige for å fjerne dobbeltbetydninger, tolke språklige bilder eller sikre konsekvent tolkning av begreper. Collocations er svakere enn semi-morfologi (jamfør [3] og [29]), men vil gi større mulighet til å forholde seg til fraser når en beregner likhet mellom et sykepleiedokument og et rammeverk. En kan fremdeles benytte samme søkemetode, men samtidig ta hensyn til at enkelte ord får fundamentalt annen betydning sammen med et annet. Med dette menes at en kan ha en liste med fraser knyttet til thesaurusen, og påvirke vektingen om en finner collocations. Men om det er vanskelig å bygge en thesaurus med enkeltord, er det enda vanskeligere å bygge en generell thesaurus som støtter de viktigste collocations på grunn av de mange forskjellige skriveformene.

4.4 Thesaurus

Ordet "thesarus" har blitt brukt mange ganger i denne oppgaven allerede. Ordet har gresk og latinsk opprinnelse, og ble brukt som en referanse til "skattekammer av ord" [2]. Ofte har det samme mening som ”synonymordliste”, og har i IR- sammenheng disse karakteristikkene:

- en forberedt liste over viktige ord i et gitt kunnskapsdomene
- for hvert ord i denne listen, finnes det et sett relaterte ord

Hovedhensikten med en thesaurus er vanligvis disse:

- gi standardisert vokabular (eller systemreferanser) for indeksering og søk
- hjelpe brukere med å lokalisere termer for å bedre kvaliteten på spørringer
- gi klassifiserte hierarkier som tillater utvidelse og innskrenkning av spørringene slik at de best mulig er tilpasset brukerens informasjonsbehov

Innholdet i en thesaurus er gjerne substantiv, eller verb i gerundium (verbalsubstantiv). I tillegg er det svært nyttig å kunne identifisere et begrep basert på kombinasjonen av to ord,

gjerne et adjektiv og et substantiv. Ofte blir det representert i thesaurusen på formen "substantiv, adjektiv". Det kan være vanskelig å avgjøre om en skal basere seg på entalls- eller flertalls form i thesaurusen. En løsning kan være å stemme termer på samme vis i thesaurusen som i indeksering av dokumentsamlingen en skal bruke thesaurusen i, noe som dog kan resultere i problemer for gjenbruk.

Fordelen med en thesaurus er bl.a disse: normalisering, struktur, reduksjon av støy, identifisere indekseringstermer med klar semantisk mening, støtte bruk av fraser, gjenfinning basert på begreper istedenfor enkeltord samt organisering i kategorier og underkategorier [2]. Ordnett (semantiske nett, 'word nets', 'concept nets') er avanserte thesaurier. Disse er databaser over ord, der deres ulike betydninger er utskilt. Relasjoner og semantiske relasjoner mellom dem finnes også (som hyponymi og hyperonymi (over- og underbegreper)). Et eksempel er hentet fra ”WordNet 2.0” sin nettbaserte søkemotor, der det engelske ordet "hand" ga 14 hypernymer, bla "terminal part of the forelimb in certain vertebrates". I domener der en tildels kjenner innholdet, som i EPJ, er slike egenskaper svært nyttige. Det finnes i dag en rekke medisinske thesaurier, og UMLS er kanskje en av de mest kjente. Selv om thesaurier er nyttige i sammenheng med automatisk klassifisering og NLP, viser det seg at store overordnede thesaurier blir for lite spesifikke på et slikt bruksområde. Ofte ender applikasjoner som forsøker å gjøre dette opp med å konstruere sine egne thesaurier [47].

Fordelen med en godt utviklet thesaurus er at den også kan benyttes som et verktøy under katalogiseringen og dokumenteringen av en kilde eller et datamateriale. Thesaurusen vil tilby et autorisert utvalg av stikkord og termer som kan benyttes som stikkord eller indekser for ulike elementer (kontrollert vokabular). Om en slik tesaurus benyttes systematisk i dokumentasjonsarbeidet, vil det bli langt enklere å lokalisere relevante kilder når disse gjøres til gjenstand for et søk [51].

4.5 Inverterte filer

Inverterte filer er en representasjon av et informasjonsobjekt som har som formål å gjøre søk mot informasjonsobjektet raskere. Konstruksjon av en slik representasjon kan gjøres på et utall forskjellige måter, men en vanlig oppfattning av struktureringen er at den er basert på en ordliste og forekomst av ordene i objektet. Noen skiller mellom inverterte filer og inverterte lister. For inverterte filer er hvert element i listen en peker til et dokument eller filnavn, mens for en invertert liste er elementene i listen ordposisjoner [2].

Søking opp mot en invertert fil, har 3 hovedtrinn:

- Søk etter ord - ord (eller mønster) i spørringen isoleres, og man leter etter disse i ordlisten. Fraser og nærliggende ord deles opp i enkeltord i spørringen
- gjenfinning av forekomster - listen over forekomster som er funnet gjengies til brukeren
- Manipulering av forekomster - forekomstene prosesseres for å finne fraser, nærheter eller boolske operasjoner. Andre manipuleringer kan være nødvendige.

Foruten å gjøre søkeoperasjoner raskere, kan en invertert fil gjøre samlingen en utfører søk på, mindre. For eksempel kan 1 GB av tekst fra TREC samlingen (en samling med dokumenter som ofte benyttes til å evaluere søkemotorers egenskaper) komprimeres til 5 Mb i en invertert fil. I tillegg kan en benytte stemming og normalisering å få ordsamlingen enda mindre [2]. Siden størrelsen på den inverterte filen består av både ordlisten og forekomstlisten, vil det totale plassbehovet være større. Man mener representasjonen av informasjonsobjektet i en invertert fil som oftest vil komme på 30-40% [2],[46].

Inverterte filer benyttes i prototypen, og en beskrivelse av hvordan det er løst der følger senere. Løsningen der gjør at en invertert fil øker behovet for lagringsplass eller minne. Begrunnelsen for dette, er at en må forholde seg til lovverket i en EPJ - en kan ikke lage en representasjon av originalteksten uten videre. Forøvrig utfører prototypen ikke alle hovedtrinnene i de beskrevne søkeoperasjonene.

4.6 Søketeknikker

Søketeknikker er et omfattende begrep. Denne oppgaven befatter seg med "klassiske søketeknikker", som er et nokså uklart begrep. Teknikkene som er beskrevet i tidligere kapittel inngår i dette begrepet. Ordet "klassisk" er ikke så veldig nyttig for å definere dette heller, da teknikkenes opprinnelsesår varierer kraftig. Et forsøk på avgrensning kan være teknikker som er blitt utviklet for å administrere vitenskapelig litteratur, og som senere har fått utbredelse i forbindelse med søk på internett. Baeza-Yates et.al [2] beskriver teknikkene slik:

"The classic models in information retrieval consider that each document is described by a set of representative keywords called index terms. An index term is simply a (document) word whose semantics helps in remembering the document's main themes".

En hovedgruppering av teknikkene kan se slik ut:

- boolsk modell (”boolean model”)
- vektor modell (”vector space model”)
- propabilistisk modell (sannsynlighetsberegning eller ”probabilistic model”)
- klyngeanalyse (”clustering”)
- neurale nettverk (”neural networks” eller ”Artificial Intelligence”)

Siden teknikkene har ulike styrker og svakheter, er det en ufordring å velge riktig teknikk for ens problemstilling. En løsning er å implementere flere, og benytte dem i applikasjonen som en ser det passer best, eller i kombinasjon. Mange av teknikkene kan benytte mye av den samme representasjonen innad, slik at når man først har fått opparbeidet seg indekstermer, er ikke kostnaden alt for stor i implementasjons- eller ytelsessammenheng til at en kan gjøre dette (et utsagn med modifikasjoner, selvfølgelig - det er mest basert på tidligere utsagn om at indekseringskostnader gjerne står for halvparten av kostnadene, og det ikke sikkert dette utsagnet passer så godt for neurale nettverk som for de fire andre).

På bakgrunn av at det verken har funnes en god dokumentsamling å jobbe med, og heller ikke en godt utviklet thesaurus, har valget av IR metode for prototypen til denne oppgaven blitt farget av det. En metode som gir anledning til rangering og delvise treff ville passe best i en slik situasjon.

De boolske modellene ble vurdert for svake for formålet. De boolske modellene gir ikke delvise treff (selv om ”Fuzzy Boolean” kunne vært benyttet). Det ble implementert en helt primitiv, ikke ferdig versjon av boolsk logikk i prototypen for å kunne se om man i det hele tatt fikk likhet mellom samlingen og spørringene.

Sannsynlighetsberegning og vektorrommodellen er klare kandidater, men det er diskusjon hvilken av disse som yter best [2]. Den probabilistiske metoden har disse karakteristika:

- En må gjette den initiale delingen av dokumenter i relevante og ikke-relevante mengder
- Den behandler ikke frekvens (alle vekter er binære)
- Den antar at hver term er uavhengig

Jeg har valgt å benytte en vektormodell for sammenligning av spørring og dokumentsamling i denne mastergradsoppgaven. Valget er bl.a basert på at den ikke er for

komplisert å implementere, den yter bra sammenlignet med andre [2], den gir anledning til påvirkning av vekting, og det kan benyttes flere likhetsutregninger (mål) om en ønsker det.

Klyngeanalyse er en spennende teknikk i denne sammenheng. Jeg anser klyngeanalyse som nyttig for denne typen problemstilling, og det vil være naturlig å se mer på anvendelsen av klyngeanalyse i forbindelse med sykepleiedokumentasjon. Klyngeanalyse er likevel ikke blitt benyttet i denne oppgaven, da vektormodellen ble vurdert til å passe bedre ved denne anledningen. En av årsakene er blant annet at klyngeanalyse ofte forutsetter at en må predefinere hvor mange klynger en ønsker (f.eks. de 21 behandlingskomponentene til Sabaklass), men det kan være vanskelig å velge de mest fornuftige. Et alternativ kunne være å kombinere klyngeanalyse med andre teknikker. Klyngeanalyse ble ansett til å passe bedre til et større scope enn det som var satt som ramme for denne oppgaven.

Om neurale nettverk inngår som "klassisk" kan diskuteres. Mange av teknikkene som går under AI (Artificial Intelligence) eller "kunstig intelligens" i dag har vært kjent lenge, og har stor utredning i problemdomener som ligner det oppgaven tar for seg. Grunnlaget for dokumentbehandling med disse teknikkene kan være så like de fire andre nevnte teknikkene, at representasjonen en allerede har skapt i indekseringssammenhengen kan benyttes også til neurale nettverk (jamfør Baeza-Yates et.al's definisjon). Neurale nettverk ble en periode vurdert som aktuell metodikk for dette prosjektet, men videre arbeid ble avsluttet. En etterlevning av dette er at prototypen kan skrive ut thesaurus med tilhørende klassifiseringskoder i en tekstfil, noe som muligens kan brukes til AI implementasjoner senere. Årsakene til at arbeidet ble avsluttet var tidsfaktoren, at alternativet kom for sent på banen, hadde mer avansert implementasjon og forventninger om usikre resultater. En enkel, billig og brukbar metode var tross alt målet for dette prosjektet. At neurale nettverk og AI kommer til å spille en viktig rolle for applikasjoner som forsøker å automatisk klassifisere sykepleiedokumentasjon i fremtiden, bl.a i forbindelse med beslutningsstøtte (jamfør metoder som i [27]), er sannsynlig.

4.6.1 Vektorrom modellen (Vector Space Model)

Valg av IR-metode for bruk i prototypen falt på den såkalte vektorrommodellen (VSM). Modellen er utviklet av Gerard Salton m.fl. på slutten av 70-tallet og begynnelsen av 80-tallet (1983)[14]. Selv om utregningen av likhet er krevende på grunn av alle vektorene, yter metoden bra, og står seg overraskende godt sammenlignet mer avanserte systemer [2]. Den er mye prøvet og brukt i IR sammenheng. Modellen har blitt bearbeidet av mange, og utvidet med både xml og sub-vektorer for bedre å ivareta semantikk. I sin originale form er den er

forholdsvis enkel å implementere, den støtter ulike rangeringsalgoritmer og likhetsmål, og kan tilpasses samlingen ved å benytte nye kalkulasjoner av vekter (”relevance feedback”) om en ønsker det. Den kan støtte frasesøk ved tilpasset implementasjon. Den normaliserer dokumentene, slik at lange dokumenter ikke får større betydning enn korte.

Svakheter med metoden er at den i utgangspunktet ser ord som atomiske, og at dokumenter ansees som "bags of words". Hvert eneste ord i samlingen får plass i en vektor, noe som gjør at enkelte av vektorene kan være glisne. Det finnes heller ingen teoretisk basis for antagelsen av "term-rom" i vektorene, dette er gjort for å visualisere.

For å lage en søkemotor basert på vektormodellen, må følgende steg utføres[30]:

1) Samle inn dokumenter det skal utføres søk på
2) Lage term/dokumentrom
3) Forberede en spørring til samme termrom
4) Sammenligne spørringen vektor mot dokumentenes vektor
5) Returnere en liste av dokumenter som ligner mest, rangert etter avstand

Metoden forutsetter at man kan lage vektorer av termene. Dette gjøres på basis av en term-dokument matrise. Normalisering eller stemming av termene reduserer antallet termer, slik at en unngår for mye støy når en teller opp antall termer. Det dannes så en term- frekvens matrise (ofte forkortet *tf*). En må også holde rede på antallet dokumenter termen forekommer i, dokumentfrekvens (forkortet *df*).

Å gi dokumenters indekseringstermer betydning basert på frekvens, kalles vekting. Vektingen brukes for matematisk å kunne uttrykke termens betydning for dokumentet i samlingen. En indeksterm som forekommer tusenvis av ganger i flere hundre dokumenter, er en dårlig indeksterm. En term som forekommer få ganger i få dokumenter i samlingen derimot, kan være en viktig indekseringsterm fordi den kan konkretisere akkurat det brukeren er ute etter.

For å forhindre at lange dokumenter med hyppig gjentatte termer skal få forrang over korte, konsise dokumenter (Zipf's lov[1]), dannes en invers dokument frekvens matrise (ofte forkortet *idf*) . Vi har nå laget vektorer for alle termene og dokumentene i vår *tf- idf* matrise. En vektor kan sammenlignes med en matrise av flyttallsverdier, den har retning og størrelse.

[1] Empirisk regel som sier at en terms egenskap som indeksterm er avhengig av antall ganger termen forekommer i dokumentsamlingen [46]. ”[Zip’s Law] states that the *i*-th most frequent word appears as many times as the most frequent one divided by i^{θ}, for some $\theta \geq 1$” [2].

Det finnes mange måter å gjøre dette på. Jeg har valgt å følge anbefalinger fra [2]:

Ligning 1 viser frekvens av term *i* i dokument *j*:

$$f_{i,j} = \frac{freq_{i,j}}{\max(freq_{l,j})} \qquad \text{Ligning 1}$$

Ligning 2 viser den inverse dokument frekvens for term *i* i dokumentet:

$$idf_i = \log\frac{N}{n_i} \qquad \text{Ligning 2}$$

Ligning 3 viser vektingen av termene *i* dokumentet *j*:

$$w_{i,j} = f_{i,j} \cdot \log\frac{N}{n_i} \qquad \text{Ligning 2}$$

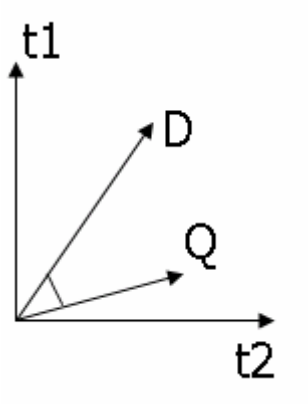
Når man så har konstruert tf-idf matrisen, er neste steg å forberede spørringen. Spørringen må gjøres om til en vektor for å kunne sammenlignes med dokumentets vektor. Salton og Buckley har foreslått formelen i Ligning 4:

$$w_{i,q} = \left(0.5 + \frac{0.5 \cdot freq_{i,q}}{\max(freq_{l,q})}\right) \cdot \log\frac{N}{n_i} \qquad \text{Ligning 3}$$

Dette betyr at spørringen også normaliseres. I prototypen har dette liten betydning siden spørringen baseres på thesaurusen som vanligvis har fjernet duplikater av termer.

4.6.2 Likhetsmål

Når en har konstruert dokumentvektorer og vektor for spørringen, gjenstår det å sammenligne disse vektorene. Figur 4-4 illustrerer et tenkt forhold mellom termer (t1 og t2), det aktuelle dokumentet (D) og spørring (Q). Avstanden mellom dokumentet og spørringen i vektoren utslagsgivende for om en mener dokumentet er relevant for spørringen.



Figur 4-4 termer (t1, t2), dokument (D) og spørring(Q) [60]

Likhetsutmålingene har omtrent samme ytelse uavhengig av vektormodell som benyttes [28]. Beregningene kan gjøres på flere måter, disse nevnes ofte:

- DICE-koeffisient
- Jaccard-koeffisient
- Cosinus-koeffisient
- Overlapp mål
- Asymmetri mål

I denne oppgaven er cosinus valgt. Cosinus likhetsmålet er et normalisert uttrykk basert på å multiplisere de to vektorene sammen. Likheten mellom dokument og spørring blir lik 1 (eller en desimalverdi like i nærheten) når dokumentet matcher spørringen, og 0 når dokumentet ikke inneholder en eneste term fra spørringen. Formelen som er benyttet i denne oppgaven ser slik ut (Ligning 5):

$$sim(d_i, q) = \frac{d_i \cdot q}{|d_i||q|} = \frac{\sum_j (w_{i,j} \cdot w_{q,j})}{\sqrt{\sum_j w^2_{i,j}} \cdot \sqrt{\sum_j w^2_{q,j}}}$$ Ligning 4

Valget av likhetsutmåling er basert på personlig preferanse, og ut fra at den ofte benyttes. Det har ikke blitt foretatt noen analyse av de ulike metodene i dette studiet. I artikkelen [27] bruker de Jacard fordi den viste seg å gi et mer balansert likhetsmål for treningssettene i deres implementasjon.

4.6.3 Rangering

Når en har foretatt en spørring, og beregnet likheten som beskrevet over, vil en motta en mengde (desimaltall) for hvert av dokumentene som angir avstand til spørringen. Det er først nå man kan vurdere kvaliteten til metoden, f.eks basert på presisjon og tilbakekalling som ble omtalt tidligere. Siden det er vanskelig å forholde seg til en mengde, kan det lønne seg å organisere den før en presenterer den som resultat av spørringen. Valg av organiseringsmetode får altså konsekvenser for hva brukeren (eller systemet) får returnert og kan forholde seg til. Det er i hovedsak 3 måter å organisere etter:

- Organisere etter likhet
    - Synkende sortering basert på likhetsutmålingen
    - Benytte tidlige treff til å generere tilbakemelding til systemet (relevance feedback)
- Velge de *k* første dokumentene (der k er en angitt verdi)
- Plukke dokumenter innen mengden som ble returnert basert på et kriterium, og returnere disse

I prototypen ble det valgt å sette en verdi for hva som ble betegnet som likt, og returnere alle dokumenter som kom over denne verdien. Dette valget ble gjort fordi implementasjonen foretar ikke bare en spørring, men ca 300. Løsningen ble å gjøre likhetsmålet justerbart av brukeren. Dette er en veldig diskutabel måte å gjøre det på, fordi det er fryktelig vanskelig å velge en god verdi (kvaliteten på relevans av returnerte dokumenter vil variere med verdien, og størrelsen på dokumentsamlingen og spørringen).

5 Støtteteknologier

Motivasjonen for å knytte emner til dokumenter i denne oppgaven var å kunne navigere i en dokumentsamling etter disse emnene. Oppgaven dreier seg mindre om dette enn tenkt initialt, på grunn av den store arbeidsmengden som var knyttet til den første problemstillingen. Jeg ønsker likevel å vie dette kapittelet til å omtale noen teknikker jeg ser for meg kan brukes i større omfang for sykepleiedokumentasjon i EPJ-systemer enn det gjøres i dag, også fordi de er nærliggende å bruke når man har emnetilknyttet fri tekst.

5.1 Ontologi

Ontologi er et begrep en hører ofte i disse dager, men det har ulikt innhold for forskjellige faggrupper, og det har også forskjellig betydning innen disse faggruppene.

En mye referert definisjon i it-sammenheng er denne:

"*Ontology is an explicit representation of a conceptualization, the conceptualization includes a set of concepts, their definition and inter- relationships*" (Gruninger og Lee, 2002).

Ontologi kan sees på som utplukking av viktige konsepter i et domene, relasjonene som eksisterer mellom disse konseptene. Ontologien er en slags typeangivelse av begrepene. Ontologien skal hjelpe oss til å ha et kontrollert vokabular med lik forståelse av konsepter. Dette får man ved å beskrive entiteter eller konsepter, og hvordan de er relatert til hverandre.

Dette betyr at man lager et oppsett for å forklare hva begreper i samlinger av termer betyr. Deretter knytter man disse betydningene opp mot hverandre, og finner relasjonene mellom dem. Ontologi er blitt sett på som nøkkelen for danne semantisk web. Semantisk web er en utvidelse av eksisterende web, ved at informasjon blir gitt veldefinert mening. Ontologi kan brukes for å standardisere semantiske termer ved hjelp av hierarkiske taksonomier av ord som handler om spesielle emner [2].

Det er mange som jobber med problemstillingene å kunne gi ustrukturert tekst en meningsbærende, overordnet struktur i forbindelse med webteknologi. Siden EPJ møter mange av de samme problemstillinger som web også sliter med, er det nærliggende å ønske å utnytte dette arbeidet. Ontologier er vanskelige å bygge, og de er gjerne domenespesifikke [77]. Ser en på taksonomiene i f.eks NANDA og Sabaklass som en slik domenespesifikk ontologi, kan en lettere se for seg bruksområder som en kjenner fra web. Disse kan knyttes enda mer overordnede "ontologier" som SNOMED, og en kan knytte utenforliggende ressurser til teksten på en ny måte. Dette igjen betyr at en lettere kan tilpasse seg andre systemers sykepleietaksonomier, for ikke å snakke om andre ressurser; prosedyrehåndbøker,

felleskatalog, Legemiddelhåndboken, ICF funksjonsområder, og finne hjelp relatert til et tema raskt. Sammenligningen er ikke søkt; DocuLive "tagger" allerede sykepleiedokumentasjon, de benytter hyperlenking, og de har utviklet webportal for systemet (Soarian). SNOMED har blitt knyttet til såkalte emnekart (TopicMaps).

5.2 XML

XML, eXtensible Markup Language er definert av W3C, og er en videreføring av SGML (Structured General Markup Language). XML har hatt innflytelse på å kunne ivareta semantikk på web og i digitale bibliotek generelt. Innen helseinformatikk har XML fått stor betydning for strukturering og standardisering, både med tanke på utvekslingsformater mellom systemer, men også som intern representasjon. Ulempen med XML er en forholdsvis stor overhead ved parsing av taggene. Det er heller ikke noen intelligens knytet til xml, det er rett og slett et språk som gir mulighet til å markere tekst på en hensiktsmessig måte. XML må benyttes i samhandling med andre teknikker for å kunne brukes utover tekstrepresentasjon. RDF (Resourse Descripiton Framework) er eksempel på en viktig standard som kan uttrykke semantikk sammen med XML. Ontologier forholder seg ofte til RDF på web.

Det finnes mange teknikker som baserer seg på XML, bl.a TopicMaps og XLink. TopicMaps er såkalte emnekart, og hovedhensikten med emnekart er å støtte begrepsbasert navigering i ressurser. XLink er en metode for å gi hyperlenker semantikk, og en mulighet for å støtte toveis navigering mellom ressurser.

5.3 TopicMaps

Emnekart kan sammenlignes med en indeks i en bok, bare tilpasset digitale ressurser. De har en egen standard, kalt XTM (ISO/IEC 13250:2003) (det finnes to standarder, HyTM er den andre). Steve Pepper ved Ontopia, som siden 1996 har han ledet Norges delegasjon til ISO-komitéen SC34, er en autoritetsperson på området. Han mener emnekart kan betraktes fra forskjellige perspektiver [65]:

- Informasjonsforvaltningsperspektivet - et nytt paradigme for organisering, gjenfinning og navigering i informasjonsressurser
- Kunnskapshåndteringsperspektivet - en formalisme for kunnskapsrepresentasjon optimalisert for informasjonshåndtering
- Biblioteksutviklingsperspektiv - en mulighet for å samle all kunnskap om et emne, spesielt relasjoner til andre emner og informasjonsressurser

Det er tre hovedbegreper som inngår i emnekart ("The TAO of TopicMaps"):

- Topic - emnet informasjonen handler om (sammenlignes med emnene i en bok)
- Assosiation - relasjoner mellom emner (sammenlignes med 'se også' relasjonene i en bok)
- Occurenses - relevante instanser av relasjoner mellom emner og informasjonsressurser. (Dette er en sammenføyning av emnet, relasjonen og ressurser, kan sammenlignes med sidetallene i boken)

Opprettelsen av emnekart kan gjøres manuelt, halvautomatisk eller automatisk. Det finnes et gratis verktøy for å debugge og prototype emnekart, kalt "Ontopia Omnigator". Emnekart kan benyttes i flere nivå, å inngå i stadig større sammenhenger, eller infiltreres i andre emnekart. Dette kalles scope. Scope gjør at en kan ta høyde for at kunnskap ikke er absolutt, men har kontekstspesifikke aspekter. Dermed er det mulig å presentere kunnskap fra forskjellige synspunkter. Pepper nevner også at TopicMaps kan være en måte å hjelpe kunstig intelligens til å tolke semantikk i et informasjonsobjekt bedre [22]. "Omingator" har støtte for RDF.

Disse egenskapene med emnekart gjør at de passer spesielt godt for navigering i sykepleiedokumentasjon. Ved opprettelsen av emnekart kan en benytte allerede strukturert informasjon som finnes i EPJ-systemet (manuelt), man kan utvide med foreslåtte eller egne relasjoner (halvautomatisk), eller man kan forsøke knytte ustrukturert tekst automatisk til emnekartet. Dette kan f.eks være basert på metoden denne oppgaven foreslår for å finne emner i sykepleiedokumentasjonen.

Siden emnekart støtter forskjellige kontekster, betyr det at en kan benytte ulike organiseringer (VIPS), rammeverk eller taksonomier (NANDA, SabaKlass, NIC, NOC) sammen, og utnytte styrkene fra de ulike. Emnekart har ulike scope, noe som gjør at man kan benytte dem til navigering i enkeltpasienters dokumentasjon, få overblikk over avdelingens pasienter, eller i en større sammenheng alt etter som systemet og lovverket tillater. Dessuten finnes det en ferdig utviklet søkestrategi for emnekart dersom en satser på Ontopia's løsning, og denne kan bli integrert og tilpasset EPJ-systemet [64].

www.lumrix.net har forsøkt å lage en intelligent søkemotor som baserer seg på XML og emnekart der en kan vise en SNOMED CT representasjon. De konkluderer med gode resultat og at emnekart er et kraftfullt verktøy [39]. Likevel kan emnekart ha begrensninger i en slik sammenheng, da SNOMED kan ha multiple relasjoner som ikke lar seg representere med TopicMaps og RDF.

5.4 XLink

Xlink står for XML Linking Language. Xlink er en standard utviklet av W3C, og er i hovedsak en måte å representere hyperlenker. En hyperlenke er primært tenkt å være en presentasjon for et menneske, men kan benyttes til prosessering av en datamaskin.

- En Xlink defineres til et uttrykkelig relasjonsforhold mellom ressurser eller deler av ressurser.
- En ressurs defineres som gjort utrykkelig ved et Xlink lenkeelement, som er et Xlink konformt XML element som sikkert uttrykker eksistensen av en lenke.

XLink er gratis, og er tilgjengelig under en såkalt GPL lisens (GNU General Public License) [80].

Det finnes to klasser av lenkeelementer for Xlink, enkel eller utvidet. En Xlink (kan) består av 6 elementer, der to av dem er selve lenkeelementene, mens de andre beskriver, eller gir karakteristikk til lenken. En oversikt over elementene og attributtene, og hvilke som må eller kan forekomme, ser ut som i Figur 5-1:

	simple	extended	locator	arc	resource	title
type	R	R	R	R	R	R
href	O		R			
role	O	O	O		O	
arcrole	O			O		
title	O	O	O	O	O	
show	O			O		
actuate	O			O		
label			O		O	
from				O		
to				O		

Figur 5-1 Xlink type attributter (required (R), optional (O))

Styrken til Xlink ligger i at man kan uttrykke mening, flere roller og flere ressurser (eller deler av ressurser) i en lenke. Ofte brukes linkbaser for å forenkle administrasjonen av lenkene. Linkbasene må være uttrykt i XML.

5.5 Lenking i EPJ's sykepleiedokumentasjon

Det finnes flere teknologier i denne sammenheng jeg burde nevnt, og jeg gjør TopicMaps og Xlink urett ved å gi en så kortfattet og ufullstendig introduksjon. Den tekniske omtalen

avsluttes her, for poenget med å nevne dem er å forsøke å gi et utvidet perspektiv på hva emnetilknytning av ustrukturert dokumentasjon kan brukes til. Web'en har fått større mulighet til å bære semantikk (the semantic web), og webgrensesnitt over databaser er mer og mer vanlig i bedrifter (gjerne "Lotus Notes" kombinert med et "off-the-shelf" portal som f.eks ”Verity” [78]).

Dette gjør at vi kan forvente at disse teknologiene mer og mer blir brukt i EPJ-systemer, og et eksisterende eksempel er Siemens Soarian. I "DocuLive EPR" har man også tatt i bruk lenking på dokumentnivå, der enveis lenking til et annet dokument (A til B), brev eller ressurs kan forekomme dersom det er hensiktsmessig. For sykepleiedokumentasjonen er bl.a å referere til noe noen andre har skrevet, for at pleieren skal slippe å skrive dette på nytt, et bruksområde. Lenkene legges inn av bruker. Det finnes situasjoner der lenking som har en mer overordnet mening enn A-B er ønskelig:

En pleier sitter med sykepleiedokumentasjonen foran seg og snakker med hjemmesykepleien pr telefon. Han vil dobbeltsjekke noe fra sammendragsnotatet mot en rapport, men husker ikke eksakt hvilken dato dette ble skrevet. Dersom den som skrev notatet ikke har lenket til den rapporten, betyr det leting ved å lese igjennom rapportene til en finner aktuell nedtegnelse.

Dersom en hadde benyttet emnetilknytning og Xlink i situasjonsbeskrivelsen over, kunne en se for seg at en XLink lenke dukker opp. Denne kunne f.eks ha en oversikt over notater som er skrevet av samme pleieren som skrev sammendragsrapporten, kanskje også organisert etter emner. Det meste av metadata (datoer, forfattere etc.) som trengs for å konstruere den tenkte funksjonaliteten finnes allerede i EPJ systemet, og kan genereres automatisk når en bytter aktivt dokument. Emnekart (TopicMaps) kan brukes til å orientere seg i pasientens dokumentasjon, og navigere til ulike emner. Men dersom pasienten kun har vært innlagt for ett hovedproblem, vil dokumentasjonen stort sett revolvere rundt dette emnet. Derfor vil semantisk lenking kombinert med flere muligheter være ønskelig (XLink). Om dokumentasjonen er ustrukturert ellers, vil metoder som denne oppgaven har som hovedtema, kunne hjelpe til med å finne emner.

5.5.1 Hyperlenking til eksterne ressurser

Et eksempel på en ekstern ressurs man kan ha ønske om å lenke til fra sykepleiedokumentasjon, er Norsk Elektronisk Legehåndbok (NEL). NEL beskrives som en

medisinsk kunnskapsbase for helsepersonell og publikum. NEL har sitt utgangspunkt i papirutgaven av "Norsk Legemiddelhåndbok", men rettighetene til NEL tilhører Norsk Helse Informatikk A/S. NEL 1 kom i april 1999, og inneholdt ca 1500 filer. Nå er versjon 13 ute, og består av over 7000 filer (tall fra NEL10). Den leveres som CD- rom eller online abonnement, og revideres kontinuerlig. Det er gjort fortløpende kvalitetsforbedringer både når det gjelder innhold og teknologiske løsninger.

NEL er et typisk emneorientert oppslagsverk, der 29 hovedkategorier er utgangspunktet for søk. I tillegg finnes øvrige ressurser og spesialist moduler. Det er vanskelig å si hvordan NEL er strukturert internt da de ikke har opplysninger om dette lett tilgjengelig. En kikk på funksjonaliteten til NEL samt og koden som vises i webbrowseren avslører XML og ASP (Active Server Page) som kan kombinere NEL's dokumenter med ressurser fra bl.a Felleskatalogen, ICD-10 og ICPC kodeverkene. NEL inneholder utstrakt bruk av hyperlenker, annotasjoner, bakgrunnssøk og "pop-ups" ved å holde musen over enkeltord (bare støtte i Internet Explorer). Ordlister (thesauri) er tydelig i bruk for å tillate dette konsistent.

NEL benytter lenking fra A til B i sin webbaserte versjon, noe som gjør at en må bruke "tilbake" funksjonen i browseren, eller starte på nytt fra emnet. Dette gir en tungvindt følelse når en navigerer, selv med en så utstrakt bruk av lenker. Siden NEL har så god infrastruktur, kunne emnekart gjøre navigasjonen enda bedre. Et eksempel på en ressurs som viser styrken til emnekart er Forbrukerrådets portal [42]. Der blir en presentert for forskjellige emner på venstre side, og selve dokumentene vises midt på siden. Velger en "Helse" i emnemenyen, dukker en side med aktuelle dokumenter opp, samt to menyer med underemner og beslektede temaer opp (henholdsvis på høyre og venstre side av dokumentene). Emnene og beslektede temaer er dynamisk oppdatert etter hvilke dokumenter du ser i skjermbildet for øyeblikket. Navigeringen oppleves behagelig når en forsøker å orientere seg om et tema. Tabell 5-1 viser disse:

Det er gjort mye arbeid og forskning innen webløsninger som en kan se i praksis i Forbrukerportalen. Emnene i Forbrukerportalen er på dokumentnivå. En kan se for seg navigasjon i sykepleiedokumentasjonen etter lignede funksjonalitet, med f.eks Sabaklass' behandlingskomponenter eller NANDAs domener og klasser som emner og temaer.

Tabell 5-1 Emner under "Helse" i Forbrukerportalen, og beslektede emner (NB: disse står ikke nødvendigvis i forhold til hverandre i tabellen)

Emner	**Beslektede temaer**
Helse	
Allergi	Matvaresikkerhet
Bakterier	Dyrehelse
Ernæring	Genteknologi
Helsevesen	Mat
Kunstig befruktning	Helsekost
Lege	Forurensing
Medisin	Mosjon, trening
Pasientrettigheter	Personlig pleie, kosmetikk
Smittevern	Akrylamid
Tannlege	

Jeg avslutter dette store temaet her med å nevne at KITH har jobbet med en standardisering for EPJ ("Elektronisk pasientjournal standard - Arkitektur, arkivering og tilgangsstyring"). I mandatet for standardiseringsarbeidet, er følgende punkt interessant i vår sammenheng:

"Spesifikasjon av en grunnleggende, generell journalarkitektur som gir mulighet for en mer fleksibel bruk av pasientjournalen. Merk at dette kun innebærer å spesifisere generelle mekanismer, attributter mv som er nødvendig for å dekke behovet for strukturering av innholdet i journalen, og ikke alle konkrete dokumentgrupper etc. som det er behov for".[50]

Det er altså ønskelig å utnytte de muligheter det elektroniske medium har for dokumentasjonen i EPJ. Å si at emneorientert navigering og hyperlenking er viktig i så måte, er vel å uttrykke seg beskjedent. EPJ-standarden har mekanismer for semantisk hyperlenking, men de er mest relatert til medisinsk journal og kodeverk. Siden standarden ikke sier noe om hvordan lenking for øvrig må implementeres i et spesifikt EPJ-system står en ganske fritt. Teknikkene og ressursene jeg har nevnt har ingenting med EPJ-standarden å gjøre, men å utnytte standarder som er laget for lenking og semantikk lik dem som finnes for web er å foretrekke, spesielt om en tenker på interoperabilitet med andre systemer.

6 Prototyp: "SKTax" - arkitektur og funksjonalitet

Det er blitt laget en prototyp som forsøker å knytte emner til ustrukturert sykepleiedokumentasjon i forbindelse med denne masteroppgaven. Helt kort kan prototypen oppsummeres med at den benytter en kjent informasjonsgjenfinningsteknikk (vektorrommodellen) og et rammeverk for å søke ut emner i innlest sykepleiedokumentasjon. Prototypen har fått navnet "SKTax" med bakgrunn i at det benytter taksonomiene i Sabaklass 2.0N.

6.1 Funksjonelle krav til prototypen

Krav til funksjonalitet er begrenset til den funksjonalitet som er nødvendig for å vise noen av mulighetene et system som dette kan gi. Det er ikke satt krav til defineringen av thesaurusen eller rammeverket utover det som har vært nødvendig for å få prototypen opp å kjøre.

"Søking" er relatert til søket som utføres automatisk mellom thesaurus og dokumentsamling, og ikke søk som er definert av en bruker. Slik funksjonalitet er foreløpig ikke implementert.

6.1.1 Visuelle krav

- Grensesnittet skal vise resultat av søket, der følgende bør fremkomme:
  - Prototypen skal illustrere rammeverket slik at brukeren ser sammenhengen som har blitt knyttet mellom dokumentet og rammeverket
  - Termene som danner grunnlag for likhet, skal vises i grensesnittet

6.1.2 Krav til justeringer

- Brukeren skal kunne påvirke likhetsmålet ved å kunne sette sine egne verdier for dette (se Figur 6-3).
- Brukeren skal kunne redigere thesaurusen for å kunne påvirke utfallet av søket, og tilpasse thesaurusen (se Figur 6-2).

6.1.3 Innlesing og lagring

- Brukeren skal kunne velge hvilken sykepleiedokumentasjon som skal leses inn. Dette begrenses til ren tekst, og til at filer representerer en rapport.
- Bruker skal kunne lese inn en egendefinert thesaurus.

- Bruker skal kunne lagre sin redigerte thesaurus.

6.2 Oversikt over funksjonalitet

I prosessen med å bygge en prototyp ligger det mange utfordringer, og det må foretaes en rekke valg. Jeg vil i det følgende forsøke å forklare mekanismene som ligger bak resultatene. I prototypen kommer resultatene til uttrykk ved å vise rammeverket og markere ordene som danner bakgrunn for emnetilknytningen i teksten, og den tilrettelegger for justeringer slik at en kan forbedre presisjonen. Det er også forsiktig forsøkt å vise bruksområder for emnetilknytningen i prototypen, men dette har ikke vært prioritert da det primære målet har vært selve emnetilknytningen. Det er viktig å presisere følgende om prototypen:

- Den tar ikke hensyn til relasjonene mellom diagnoser og tiltak - treffene er uavhengige resultater av matematisk beregning av likhet mellom dokumentet, dokumentsamlingen og de ulike termene i thesaurusen knyttet til de to taksonomiene i rammeverket.
- Prototypen viser altså IKKE Diagnoser og Tiltak som følge av en intelligent "Sykepleieprosess".
- Den forsøker ikke å finne begreper i selve teksten, det er dokumentet som helhet som utgjør likhet med et eller flere emner i rammeverket.

I de følgende underkapitlene beskrives prototypens arkitektur og funksjonalitet. En vil få ett innblikk i hvordan implementasjonen er, og det beskrives en del forhold rundt problemer som har oppstått og valg som har blitt gjort. Prototypen bærer preg av at den utviklet under tidspress, og av en person. Den er således å betrakte som et forslag til hvordan man kan tilnærme seg problemstillingen å knytte emner til ustrukturert sykepleiedokumentasjon, og at videre raffinering eller omskrivning er nødvendig.

6.3 Valg av rammeverk i prototyp

Valget av rammeverk var en prosess som måtte ta hensyn til flere faktorer. Enten var aktuelle rammeverk ikke oversatt til norsk, for dyre, utilgjengelige, for omfavnsrike, eller for kompliserte til å enkelt la seg implementere i en prototyp som kunne vise en tenkt tilnærming til problemet med automatisk begrepsknytting til ustrukturert tekst. For minst to av de aktuelle kandidatene, var rammeverket kun tilgjengelig i form av papir (bokform) eller i ett tekstformat knyttet til et tekstbehandlingsprogram. I tillegg var prosessen med å få tillatelse til å benytte rammeverkene tidkrevende, og for ett av rammeverkene savnes tilbakemelding om tilgang til dags dato på tross av gjentatte purringer til innehaver.

Bruk av VIPS's emneord (Fase 2, se Vedlegg C) var en kandidat i forbindelse med prototypimplementasjonen denne oppgaven beskriver. Mangel av lett tilgjengelig, ferdig definert datastruktur gjorde at det falt bort i prosjektet grunnet arbeidet med å lage en slik, og at det fantes alternativ som også inneholder kodifisering.

NANDA og NIC var rammeverkene jeg kunne tenkt meg å bruke initialt. De ser ut til å være satsningsområder for NSF, og de representerer et universelt og detaljert kodeverk. Siden NIC ikke var oversatt til norsk, ville NANDA vært mest aktuelt. Grunnet at jeg ikke fikk svar på mine henvendelser til innehaveren av rettighetene i Norge (Akribe Forlag) om bruk av rammeverket, ramlet det bort. Dette er ikke den eneste grunnen. NANDA's syv akser og voluminøse beskrivelse av diagnosene, ville begrenset prototypen til å inneholdte et utvalg av diagnoser for maskinleselig representasjon. Selv om dette ville gi en bedre thesaurus, ville det ikke blitt laget en fullstendig maskinleselig representasjon for prototypen.

Sabaklass rammeverket viste seg å være tilgjengelig og overkommelig for prosjektet. Sabaklass 2.0N var til forskjell fra andre rammeverk ikke belastet med strenge rettighetsbeskyttelser som utelot bruk for enkeltindivider og forskning. Etter en enkel registrering ble fri tilgang gitt til å bruke rammeverket. I tillegg ble kopi av den originale avhandlingen tilsendt på forespørsel. Dette, samt den overkommelige størrelsen på taksonomiene avgjorde valget i favør for dette klassifiseringssystemet. At rammeverket kodifiserer og lar seg mappe inn i andre organiseringer (SNOMED), gir et spennende tilleggspotensial for å tilnærme seg problemstillingen å knytte emner til fri tekst.

Valg og utenforliggende problemstillinger er en situasjon utviklere av EPJ-systemer opplever i virkeligheten. Derfor er dette et poeng i seg selv: flere rammeverk gir flere muligheter, men innebærer også flere kompliserende faktorer. Prototypen trenger ikke et spesielt rammeverk for å fungere, men det må struktureres på en spesiell måte for å fungere. Denne strukturen er tilpasset Sabaklass, så bytte av rammeverk betyr utvikling av en ny XML-parser dersom strukturen må endres. Dette kan gjøres modulært, og trenger ikke bety mye omskrivning av programmet som helhet.

6.4 Utarbeidelse av maskinleselig versjon av Sabaklass

Taksonomiene i Sabaklass 2.0N, tiltak og diagnoser, forelå som 8 tabeller laget i tekstdokumenter skrevet i formatene Microsoft Word (doc) og Acrobat Portable Document Format (pdf) fra Adobe. Med ett slikt utgangspunkt var det en utfordring å få rammeverket maskinleselig. Flere representasjonsformer er mulig, men XML er et fornuftig valg med tanke

på muligheter for senere bruk. For å kunne gjøre samme operasjoner på begge taksonomiene, ble det besluttet å gjøre XML representasjonen lik for diagnoser og tiltak.

Arbeidet avslørte noen mindre feil som hadde sneket seg inn i den norske versjonen av Sabaklass 2.0N. Disse ble korrigert dersom det var nødvendig for å gjøre lesingen uniform, eller de ikke utgjorde noen semantisk betydning. Uoverensstemmelser mellom de to taksonomiene som også dukket opp, men som ikke utgjorde vanskeligheter med innlesing, fikk være som de var (se Vedlegg E). Det må også nevnes at det kan ha sneket seg inn feil i løpet av arbeidet med å skrive rammeverket inn i XML.

Vedlegg H viser et autogenerert DTD (Document Type Definision) for tiltak, samt et utsnitt av XML koden for tiltak og diagnoser. Utformingen av DTD for diagnose vil være likens; unntaket er elementet i DTD'en: <!ELEMENT klassifisering (tiltak+)> som erstattes av taggen diagnose.

De to taksonomiene har definisjoner på ulikt nivå. For diagnoser er feltet <definisjon></definisjon> fylt med beskrivende, overordnet informasjon om behandlingskomponenten. Tiltak er ikke beskrevet på dette nivået, men har derimot et felt <def></def> som beskriver tiltaket nærmere. XML representasjonen er lik i begge tilfellene, men feltene er fylt med verdien "U/A" der det ikke foreligger noen definisjon.

I feltet <klassifisering></klassifisering> finner man selve diagnosene eller tiltakene (kategorier og subkategorier). Disse har i rammeverket et unikt identifiserende nummer i den aktuelle taksonomien, og noen av disse numrene er desimaltall. Tallene kan gå igjen i den motsvarende taksonomien. Ett forhold som man umiddelbart ser om en sammenligner tekstversjonen med den strukturerte versjonen av rammeverket, er at den strukturerte versjonen mangler spesiell markering av subkategorier. Tekstversjonen beskrives (omskrevet) slik:

"Hver av de to taksonomiene bruker fem alfanumeriske tegn for å kode hvert element. ... Det første er en bokstav for å representere behandlingskomponenten, de 2. og 3. tegnene er tall som representerer hovedkategori for diagnoser eller tiltak, det 4. tegnet er blankt eller et desimaltegn for å representere en subkategori. Det femte elementet er en såkalt modifikator" [66].

De siste tallene, desimaltegnene, er det nyttig å gjøre visuell forskjell for mennesker ved hjelp av indentering i fremstillingen. For en datamaskin utgjør ikke en slik fremstilling noen fordel, tvert imot vil den føre til flere sjekker og kall i koden for å få tak i informasjonen. Siden ikke alle punktene hadde underpunkter, og siden numrene er unikt i hver av taksonomiene, fant jeg ikke noen grunn til å vanskeliggjøre XML representasjonen ytterligere

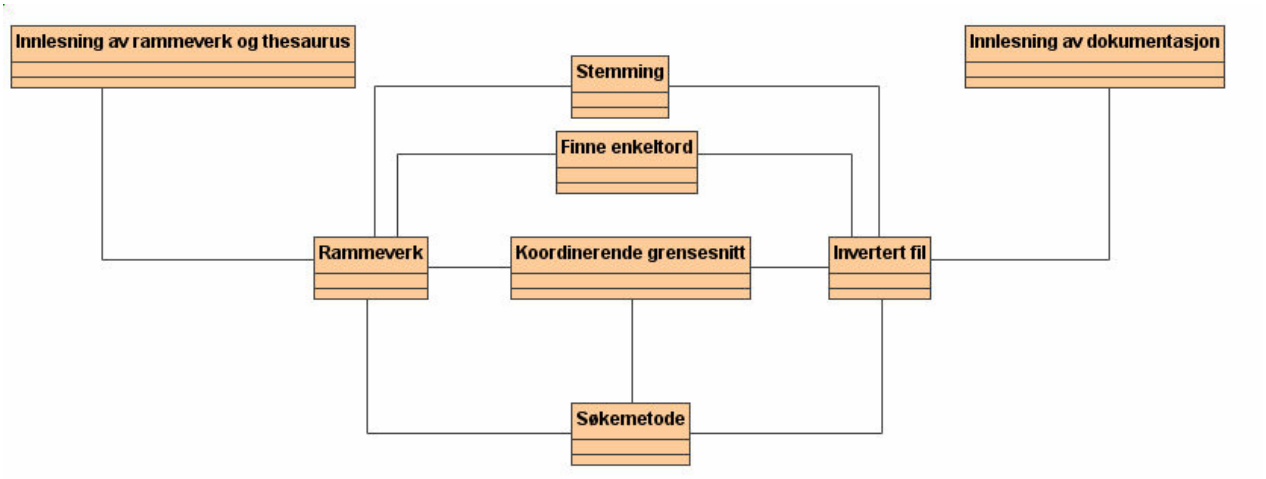
ved å lage tagger for underpunkter for subkategoriene. Derfor ble kategorier og subkategorier likestilt i fremstillingen av XML filen. Modifikatorene *forbedret - stabilisert - forverret* (1, 2, 3), er ikke plassert i XML representasjonen. Dette ble ikke gjort fordi en applikasjon vil kunne tilpasse disse selv, og spesielt siden denne prototypen ikke gjør bruk av dem.

Feltet <thesauri></thesauri> er ikke en del av rammeverket, men er et eget tillegg. Dette feltet blir brukt til å inneholde termer som representerer hvert enkelt av de unike diagnosene eller tiltakene. Dette er gjort for å slippe å ha en egen struktur for disse dataene. Dette feltet inneholder termene som blir brukt i spørringen.

XML versjonen av rammeverket fremstår som følge av de valg og tilpassninger prototypen har hatt behov for, og denne representasjonen kan selvfølgelig være gjenstand for diskusjon. Det understrekes at de norske representantene for Sabaklass er kjent med arbeidet, men har ikke tatt del i det, og om det er forhold rundt denne løsningen som er utilfredsstillende er det ikke Sabaklass eller deres representanter som er ansvarlige for det.

6.5 *Konseptuel Modell*

En Konseptuel modell fremstiller hovedfunksjonaliteter og relasjoner i et problemdomene. En Konseptuel modell over prototypen "SKTax" kan se ut som i Figur 6-1.



Figur 6-1 Konseptuel modell av prototypen "SKTax"

Om en ser på den konseptuelle modellen, ser en at prototypen henger sammen av flere deler. Et slikt modulært prinsipp kalles objektorientering, og er av stor betydning for vedlikehold, forbedring av funksjonalitet eller omskrivning av deler av programmet. Objektorienteringen gir f.eks anledning til å bytte ut søkemetoden med en annen. Dette betyr at en kan finne styrker og svakheter ved andre søkemetoder uten å måtte spesielt tilpasse resten av programmet hver enkelt. Det er imidlertid bare implementert en fungerende søkemetode her.

Prototypen består av i alt 46 klassefiler, inkludert 12 klasser for ulike språk til Porterstemmeren. Siden det fort blir uoversiktlig, er de organisert i pakker, disse er (se Vedlegg G for UML oversikt, og kildekode på vedlagt CD):

- mainpack - pakke som har med administrering av programmet å gjøre; grensesnitt, hjelpemeny samt ordutskilling
- documentops - pakke som forbereder og utfører operasjoner på dokumentsamlingen, inkludert søk
- fileops – pakke som har med innlesning og lagring av filer å gjøre
- porter - pakke som har med porterstemming å gjøre, produsert av SNOWBALL og enkelt tilpasset

Det er 5 klasser som er av spesiell interesse i disse pakkene:

- TokenizeWords - en klasse som benyttes til å isolere ord i teksten, og kan brukes til å skrelle vekk uønskede tegn. Dette er en viktig klasse som brukes flere steder i programmet når det er behov for å finne enkeltord.
- PorterStemmer - en klasse som koordinerer flere klasser. Utfører stegene i Porterstemming-algoritmen. Stemmeren er hentet fra Porters eget nettsted, og eneste tilpassning som er gjort er at inn og ut parametrene er enkelt tilpasset prototypen.
- SabaklassTaxonomy - klasse for intern representasjon av det innleste rammeverket. Det vil instaniseres to versjoner av denne klassen, en for diagnoser, og en for tiltak. Klassen inneholder også de innleste thesaurusene som er laget til hver av kategoriene og subkategoriene til behandlingskomponentene.
- SykepleieDokument - intern representasjon av innlest sykepleiedokumentasjon, og det instansieres like mange versjoner av klassen som det er dokumenter. Inneholder bl.a. dokumentet i sin helhet (alle ord og tegn), informasjon om antall ord, endringsdato osv og en liste over termer som det skal gjøres søk imot. Denne klassen holder også på informasjonen om hvilke emner søket har funnet passer for dette dokumentet.
- MatchVector - klassen som lager vektorer av sykepleiedokumentene, og av thesaurusene, og utfører selve søket.

Ved oppstart av programmet, leses rammeverket med tilhørende thesauri inn, og vises i grensesnittet. Dokumentasjonsfiler leses inn av bruker, og bakgrunnsprosesser som

konstruering av inverterte filer, fjerning av stoppord, utførelse av stemming, og opprettelse av vektorer og søk startes etter de innstillinger som ligger til grunn. Resultatene forberedes for fremstilling i grensesnittet.

6.6 Klasser for behandling av innlest tekst

Prototypens "scope" (definisjonsområde) er sykepleiedokumentasjon for en pasient i et kunstig "EPJ-miljø". En pasients dokumentasjon må være delt opp i rapportvise filer, og en folder i filsystemet kan således sees som samlet sykepleiedokumentasjon for denne pasienten. Prototypen virker ikke som forventet om man leser inn kun en lang fil, da filene danner grunnlaget for vektorer. Filene betraktes som en rekke ord, og kun ren tekst benyttes. Innlesningen "stripper" all form for informasjon som måtte finnes i dokumentet, dette inkluderer linjeskift og tegnsetting. Den eneste form for metadata som taes vare på, er endringsdato for filen.

Innlesning av filer skjer i forskjellige situasjoner i programmet. Viktig behandling av de innleste filene innbefatter utplukking av enkeltord, og stemming av disse. Behovet for innlesning er knyttet til:

- Rammeverk
- Stoppord
- Sykepleiedokumentasjon

TokenizeWords

Klassen som plukker ut enkeltord i situasjoner der er ønskelig, er kalt "TokenizeWords" i prototypen. Den er usynlig for brukeren av programmet, men den er såpass viktig for å forstå problemer at den må omtales. Å lage en god tokeniserer er en vanskelig oppgave, og kan i et virkelig prosjekt være en oppgave for et team. Min versjon er bygd på Javas egen "StringTokenizer", og har et flagg for enkel eller avansert tokenisering. Enkel tokenisering er helt lik standardløsningen til Java, med unntak av at det sjekkes om kontrolltegn er kommet med - disse fjernes i så fall. Flagges det for avansert tokenisering, gjøres ordet om til små bokstaver, og det utføres en rekke sjekker. Sjekkene er i hovedsak utført for å få fjernet tegnsetting fra ordet. Orddeling er ikke mulig, og den tar heller ikke hensyn til fraser. Selve tokeniseringen utføres stort sett ved hjelp av sjekker og iterering, noe som påvirker ytelsen til programmet i stor grad.

6.6.1 Stemming

I programmet er Porterstemming benyttet. Koden er hentet fra Porters websider som tilbyr stemmeren gratis. Stemming etter Portermetoden har mange feilkilder knyttet til seg, men ansees likevel for å gi akseptable resultater i vanlige søkemotorer. Om en kan akseptere denne feilkilden i elektroniske pasientjournalsystemer, bør vurderes. Det er derfor lagt til rette for at man kan bruke andre metoder i prototypen, men dette ville kreve opprettelse av egne bibliotek for formålet. NorKompLeks ble ikke implementert i denne versjonen, dels pga tidsfaktoren og dels pga innskrenkende rettigheter, men tillatelse ble gitt til å benytte det.

6.6.2 Rammeverk

Rammeverket leses inn ved oppstart av programmet, eller en kan hente inn sin egen redigerte versjon av rammeverket. Behandlingen skjer likt i begge tilfellene. Rammeverket har et spesielt format (XML), derfor er det skrevet en parser basert på et Javabibliotek som heter "SAX" (Simple API for XML).

Når rammeverket blir lest inn, blir informasjonen mellom taggene i XML filen blir fordelt i en egen klasse, kalt "SabaklassTaxonomy.java" (se Vedlegg G). Klassen har som formål å holde på taksonomiene og thesaurus. Det blir opprettet en instans av klassen for hver av taksonomiene diagnose og tiltak.

Taksonomiene forberedes for en trestruktur, et såkalt "JTree" i Java, og dette er synlig i grensesnittet. I en virkelig implementasjon vil en ikke ha interesse av å se rammeverket på denne måten i hovedgrensesnittet i det hele tatt. Det er gjort for å visualisere i større grad hvordan dokumentasjon og rammeverk blir knyttet sammen. Jeg valgte ikke å vise undergrupperinger i trestrukturen, da det forvansket fremstillingen. Det vil imidlertid være mulig å vise subkategoriene som inneholder desimaltall som underkataloger i treet ved å forta noen justeringer i koden.

6.6.3 Stoppord

Stoppords betydning er omtalt tidligere, hovedpoenget er at en terms egenskap som indekseringsterm er avhengig av antall ganger termen forekommer i dokumentsamlingen (Zipf's lov). Forskning viser at 20 % av termene i en tekst kan utgjøre 70 % av teksten [46]. Stoppordene er de ordene en ikke ønsker skal være med å representere dokumentet ved utregning av likhet. I denne prototypen slike ord fjernet ved hjelp av stoppordlister som lastes inn. Disse er representert i et eget, enkelt XML-format (se Vedlegg H). Det er ingen spesiell grunn til å representere dem slik, det kunne like gjerne vært en ren tekstfil der linjeskift anga

nytt ord, men XML strukturen sikrer i større grad at feil ikke oppstår. Stoppordlisten må stå til det språket dokumentasjonen de skal appliseres på. Stoppordlistene i prototypen er hentet fra hjemmesidene til prosjektet ”Snowball” [72] og kan redigeres i en utenforstående tekstredigerer. Stoppordene legges inn i en trestruktur for senere bruk i hovedgrensesnittet "GUISabaKlass.java".

6.6.4 Sykepleiedokumentasjon

Innlesningen av sykepleiedokumentasjonen bestemmes av bruker. At brukeren kan velge filer selv, kan illustrere at i et virkelig system kan det være filer en ikke vil ha med i søket, f.eks filer som det allerede er knyttet emner til. Tekst som leses inn trenger nødvendigvis ikke helt mangle struktur, men den blir av programmet behandlet som om den gjorde det. Dette er gjort delvis for å forenkle arbeidet for meg selv, delvis fordi utgangspunktet for programmet er å behandle ustrukturert tekst. Både når det gjelder filer på formatet ren tekst og formatet RTF (som er de eneste filformatene som kan benyttes for sykepleiedokumentasjon i prototypen) overlates tolkingen av tegnene til programmeringsmiljøet. Prototypen opererer med forståelige tegn, kontrolltegn blir rett og slett fjernet fra teksten. Med kontrolltegn menes her linjeskift, "end-of-line", "end-of-file" ol.

Tekst som blir lest inn i prototypen, blir lagt inn i en klasse som kalles "SykepleieDokumentasjon.java". Det instansieres et slikt objekt for hver av filene som leses. Hvert enkelt ord forsøkes å isoleres vha såkalt tokenisering, der ordet plukkes ut av den innleste teksten, og legges inn i en listestruktur. På dette tidspunktet har vi nå laget en representasjon av teksten som består av ord og tegn, men ikke kontrolltegn.

Oppbyggingen av den mye omtalte "inverterte filen" filen er en mer omfattende prosess. En tar nå utgangspunkt i listestrukturen som ble laget i steget foran, og fjerner tegnsetting og stilistiske tegn i teksten. Alle ordene gjøres om til små bokstaver. Så sammenlignes ordene med stoppordlisten for å fjerne uønskede ord fra den inverterte termlisten. Ordene telles opp, og like ord representeres sammen, og har pekere til plassering i teksten. I prototypen stemmes ordene etter Porter-algoritmen og blir til termer her. Andre metoder kan benyttes (mer er ikke implementert), derfor dette ekstra steget: Likhet beregnes på nytt på samme måte som over for å sikre at man ikke har laget flere representasjoner av samme term. Dette er diskutabelt, fordi termer kan forsvinne. Men siden prototypen benytter matematikk for likhet, styrker dette dokumentets mulighet til treff ved søk. Sammenlignes denne løsningen med hvordan inverterte filer beskrives i et tidligere kapittel, forstår en at løsningen ikke er optimal her. Det kan være interessant for leser å vite at indekseringsprosessen er skrevet om to ganger, noe

som betyr at det er rom for å gjøre det annerledes. En av årsakene til at collocations ikke er implementert, henger også sammen med løsningene som er valgt her. Programmering handler ofte om valg, og det en forenkler i en setting betyr mer arbeid i en annen.

Klassen ”SykepleieDokumentasjon.java” inneholder forøvrig noen andre variabler og lister. Den tar imot informasjon fra filsystemet om når filen ble lagret, og denne brukes til å illustrere rapportens dato i grensesnittet, som gjengies etter rekkefølgen de ble innlest. Teksten som ble tatt vare på i sin helhet, blir nå reprodusert i grensesnittet. Dersom en ikke har valgt andre innstillinger før innlesning, vil utregning av likhet mellom vist dokument og rammeverk være neste steg. De utvalgte resultatene for et dokument taes vare på i denne klassen som en liste, slik at disse selv vet hvilke behandlingskomponenter de er mest like.

6.6.5 Søk og likhetsutmåling

Selve sammenligningen av dokumentmengden og rammeverket kan skje på mange måter. Måten prototypen er bygd opp på, gjør at en kan utvikle forskjellige metoder for dette. Klassen som søker ut likhet, "MatchVector.java", er klassen for dannelse av vektorer og spørringer, og ansvarlig for å beregne likhet. Implementasjonen beskrives i kapittel 6.8.

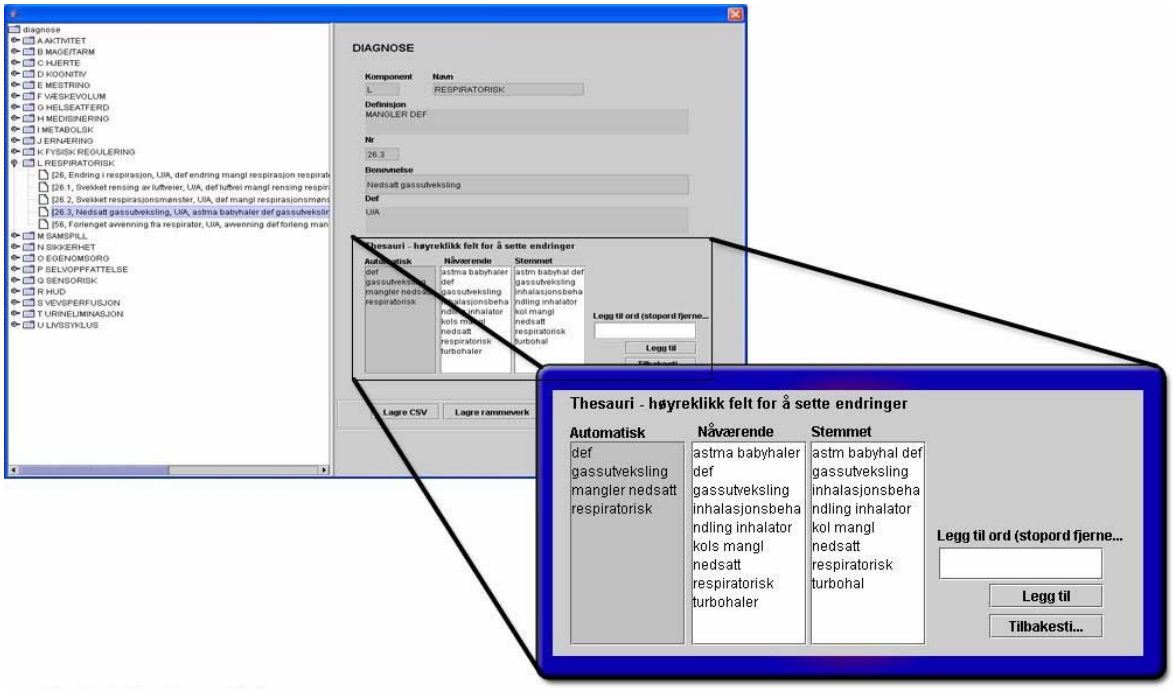
6.7 Thesaurus

Ordlister danner grunnlaget for å komme frem til om et dokument handler om et eller flere emner for mange søketeknikker. Spørringene i prototypen baseres på en såkalt thesaurus, eller ordliste.

Grensesnittet for å redigere thesaurusen inneholder mye funksjonalitet. Det viser hvordan rammeverket ser ut i sin helhet, og en kan kikke på verdiene fra hvert behandlingselement, hver kategori eller subkategori. Thesaurusen opprettes ved å autogenerere ord fra behandlingskomponenter, kategorier og subkategorier i rammeverket. Siden dette neppe er den beste løsningen for å lage thesaurus, har det i tillegg blitt gjort mulig for brukeren å redigere, lagre og hente inn sine egne ordlister. For å forenkle operasjoner innad i prototypen har thesaurusen blitt lagt inn i XML-representasjonen til de respektive kategoriene og subkategoriene i taksonomiene, men det er ikke nødvendig å gjøre det slik. I en virkelig applikasjon ville det være lønnsomt å ha en mer avansert thesauri som har sin egen definerte oppbygning.

Flere strategier var aktuelle for å utvikle thesauri, og for hvilket nivå i rammeverket man skulle forholde seg til. En kunne forholdt seg bare til behandlingskomponentene og laget ordlister som gjorde at tekst ble plassert i forhold til disse 21 overordnede klassene. Siden

rammeverket var såpass overkommelig, kunne man forsøke å komme seg ned på nivået til de nummererte elementene i rammeverket. Videre måtte det besluttes om man ville bruke tid på å utvikle gode thesaurier for et utvalg av disse, eller forsøke å generalisere. Den endelige strategien ble en løsning som omfattet alle elementene i de to taksonomiene, og så legge til en funksjonalitet som gjør at man kan utvide thesaurusen etter hvert. Figur 6-2 viser grensesnittet for formålet i prototypen.



Figur 6-2 Skjermbilde av redigeringsmeny for thesauri. Utsnittet viser thesaurifeltene.

De automatisk genererte ordene til taksonomien kan lettest sees i utsnittet i Figur 6-2, der de angies i ett eget felt ("Automatisk"). De består rett og slett av ord hentet direkte ut fra rammeverket, fra definisjoner fra høyeste nivå (behandlingskomponent), til laveste (subkategori). Ord som finnes i stoppordlisene plukkes ut, men likevel får en inn ord som ikke er meningsbærende eller presiserende nok, inn i thesaurusen. Ett eksempel viser dette best:

- Diagnose
    - ❖ E Mestring - Samling av elementer som involverer evnen til å håndtere ansvar, problemløsning eller vanskeligheter
    - ❖ 11.1 - Ikke optimal mestring i familien
- Autogenerert thesaurus:
    - ❖ ansvar elementer evnen familien håndtere involverer mestring optimal problemløsning samling vanskeligheter

Disse emneordene er lite tilfresstillende, ikke minst fordi det vil være andre elementer på kategori eller subkategori nivå (under Diagnose E Mestring) som inneholder nærmest eksakt samme ordene, og det blir vanskelig å holde det ene diagnoseelementet fra det andre i beregningssammenheng. For tiltak fungerer dette litt bedre, fordi det ikke foreligger noen definisjon på behandlingskomponentnivå, men den er lagt til kategori- og subkategorinivå. Den autogenererte thesaurusen for begge taksonomiene slik de fremstår i prototypen kan kun betraktes som et grunnlag for å få prototypen opp å gå, og for illustrasjonens del.

Det benyttes ikke fraser eller sammensatte begreper ved likhetsberegning i prototypen. Prototypen evner heller ikke å forholde seg til at thesaurusen kunne inneholde begreper som består av to eller flere ord (f.eks Diabetes Mellitus) ved redigering, selv om dette ikke ville være vanskelig å endre på. Thesaurifeltet kan tenkes å kunne inneholde slike begreper også, f.eks markert med fnutter. Eventuelt kan en utvide XML filen med ett nytt felt for fraser og sammensatte begreper. Årsaken til at dette ikke er implementert er at prototypen benytter den mest grunnleggende utgaven av vektorrommodellen som ser på hvert ord som atomisk.

6.7.1 Forberedelse av Thesaurus

Skjermbildet i Figur 6-2 viser et utsnitt av thesaurus-feltene. Disse er "Automatisk", "Nåværende" og "Stemmet". Det første feltet har en grå bakgrunn, noe som indikerer at det ikke kan skrives i feltet. De to andre er hvite, en angivelse av at en kan skrive, klippe og lime i feltene. I tillegg er det felt for innlegging av ord eller fraser, men det gjør det samme som om du opererte direkte i thesaurus feltene. Årsaken til at feltet er der, er at det kan utnyttes for spesiell behandling av ord og fraser, men videre utvikling på det området ble stoppet. ”Tilbakestill” knappen laster default rammeverk, og sletter alle endringer. Høyreklikking på ett av de tre feltene setter det aktuelle feltet til "Nåværende".

Hvorfor 3 thesaurifelter? Ett felt ville være det beste, men siden det går an å stemme på forskjellige måter, ble det vanskelig å programmere støtte for dette i thesaurusen. To innfallsvinkler er nærliggende for en slik løsning:

1) implementere en metode som automatisk finner ut hvilken stemmingsmetode som er brukt, eller

2) lage attributter i XML-filen som sier hvilken stemmingsmetode som er brukt.

Den siste er nok den beste, men det ville antagelig bety en løsrevet thesauri og en parser for dette, noe som ville komplisert og forsinket utviklingen av selve hovedfunksjonaliteten til

programmet. En må manuelt gå igjennom hele rammeverket og høyreklikke samme felt om en ønsker å endre hele thesaurusen.

Dersom en stemmingsmetode er valgt (Porter er originalinnstilling, alternativet er ingen stemming), vil termene bli stemt etter den algoritmen når det høyreklikkes på feltet. Stoppord fjernes også, det hele foregår etter samme prinsippene som for dokumentsamlingen. Dette er av stor betydning, for dersom thesaurusens termer ikke er behandlet på samme måte, får det konsekvenser for likhetsberegning. Feltet "Nåværende" fjerner stoppord om du høyreklikker på det, og "Stemming" gjør det samme. Forskjellen er at "Stemming" alltid stemmer, og en kan således risikere å få dobbeltstemte termer dersom "Nåværende" allerede er stemt. Løsningen må ansees som en minnelig ordning inntil videre.

Når en har gjort de endringer som ønskes, bør en lagre til fil. Prototypen gjør det mulig å lagre to filformater.

- Først og fremst kan den lagre filer på formatet XML som er predefinert etter oppsettet til Sabaklass filene "diagnose.xml" og "tiltak.xml" (se Vedlegg H). Dette er gjort for å ta vare på en redigert taksonomi over tid. En får mulighet til å velge hvor en vil legge filen, men hvis en legger filen til folderen "./files/sabaklass/" og gir den navnet "diagnose.xml" eller "tiltak.xml" (riktig type er selvfølgelig viktig), vil den redigerte filen bli satt til standard. En annen mulighet er å laste inn filer fra et annet sted manuelt, men en risikerer at filene fra plasseringen som er nevnt vil bli lastet dersom feil oppstår på noe vis.

- Filformatet CSV er ren tekst, og står for "comma separated vector", en fil som lagrer nummeret til kategorien/subkategorien og de tilhørende thesaurus ordene (denne filen benyttes ikke til noe i denne prototypen). Årsaken til at thesaurusen til taksonomiene kan skrives til dette formatet, er at en kan tenkes å ha nytte av informasjonen i andre sammenhenger. En kan f.eks importere csv formatet i Microsoft Excel for videre behandling av informasjonen.

Siden arbeidet med å redigere thesauri er arbeidsomt, er det forberedt og lagt ved noen filer som har forskjellige karakteristika slik at en kan prøve ut selv. Det er selvfølgelig ikke meningen at en bruker skal måtte holde på på dette viset i en virkelig applikasjon; redigering av thesauri må bli en jobb som foregår i et større og kanskje tverrfaglig fora (sykepleiere og it-utviklere). Likevel kan det være lønnsomt med mulighet for avdelingsspesifikk tilpassning, jamfør kapittel om hvordan sykepleiere dokumenterer forskjellig.

6.8 Implementasjon av IR - metode

Det omfattende arbeidet med å konstruere en thesauri og en invertert fil av dokumentene, danner grunnlaget for programmets hjerte - likhetsvurderingen, og knytting av emner til et dokument. For å gjøre en slik knytning, benyttes en metode som representerer dokumentsamlingen som vektorer av termer; vektorrommodellen (VSM). Ordene fra thesaurusen brukes til å konstruere en vektor for spørring. Disse to vektorene sammenlignes med et cosinus likhetsmål. Jo nærmere målet er 1, jo likere er de to vektorene, og en kan finne et kriterium som gjør at en mener dokument og spørring er likt nok til at en vil benytte seg av resultatet.

I denne oppgaven er dette løst ved å sette en terskel (desimaltall mellom en og 0) for likhet, og denne er justerbar i brukergrensesnittet (Figur 6-3).

Figur 6-3 Innstillinger for beregning av likhet

I en vanlig søkemotor som vi kjenner fra internett, er gjerne en implementasjon av en vektorbasert søketeknikk tilstede. I et søk i en vanlig søkemotor, stiller brukeren en spørring basert på enkeltord, fraser eller muligens operatorer. Resultatene får en kanskje presentert som en liste, og denne er rangert på noe vis.

Disse prinsippene foreligger i denne prototypen også, men forskjellen er at spørringene er mange flere (med rammeverket Sabaklass er det eksakte antall 182 for diagnoser og 198 for tiltak). Av disse spørringene genereres det 5700 svar dersom en søker i 15 dokumenter. Det må plukkes fra disse, noe terskelen benyttes til. Er likheten større enn terskelen, taes svaret vare på, og dokumentene som hører til svarene annoteres ved å legge dem til en listerepresentasjon i "SykepleieDokument.java".

Som en forstår, er problemet å finne en god terskel. Hva som er en god terskel varierer med dokumentsamlingen og thesaurusen, der antall ord og antall like ord er viktigste parametere. Vektorrommodellen bøter heldigvis noe på problemet med ulik lengde på dokumentene ved å normalisere både dokumentsamlingen og spørringene, men problemene knyttet til terskelen gjelder fortsatt.

6.8.1 Danning av vektorer

Det er laget en egen klasse som inneholder all funksjonalitet knyttet til vektormodellen, kalt "MatchVector.java". Dette er kanskje ikke så lurt om en ønsker å benytte vektorene i annen sammenhenger enn akkurat slik det gjort her, men dette er underordnet og kan tilpasses senere. Klassen får dokumentsamlingen og rammeverket som innparametere. Fra dokumentsamlingen benytter den de preprosesserte inverterte filene, og fra rammeverket thesaurusen.

Første steg er å lage en matrise for hvert ord i den inverterte filen, og resultatet hentes fra et objekt som har mulighet for å ta vare på nødvendig informasjon ("WordCount.java").

Det neste steget er å danne en term_dokument matrise, som danner grunnlaget for den neste matrisen: tf_idf_Matrix. Matrisen blir dannet av en metode som tar inn en term_dokument matrise, og regner ut den inverse frekvensen av en term etter prinsipper tidligere beskrevet (kapittel 4.6.1). Alle disse stegene kunne sikkert vært gjort mer effektivt, men siden det er fort gjort å gjøre feil i implementasjonen av matriseoperasjoner, har oversikt gått foran effektivitet.

Thesaurusen forberedes nå for spørring. De to taksonomienes kategorier og subkategorier slåes sammen, og alle 380 ordlister forberedes en etter en for spørring (se kap. 4.6.2). Implementasjonen er forsøkt holdt så nærme teorien som mulig her, bl.a telles antall forekomster av et ord for bruk i normalisering selv om ordet mest sannsynlig bare forekommer en gang (jamfør forberedelse av thesauri).

6.8.2 Utføring av spørring

Dette foregår fortløpende i programmet, men siden det er en separat handling får dette ett eget kapittel for oversiktens del.

Spørringene sammenlignes med tf_idf matrisen etter cosinusmålet. Cosinus er avstanden mellom de to vektorene (vinkelen), som er et multiplikasjonsprodukt i telleren, og en bruker potens og kvadratrot i nevneren. Dette målet legges inn i en liste for evaluering. Evalueringen er enkel: Dersom målet er høyere enn terskelen, skal det aktuelle sykepleiedokumentet annoteres med verdier fra rammeverket som spørringens vektor representerer. Verdien som taes vare på er om det er en diagnose eller et tiltak, behandlingskomponentens representasjonsbokstav og navn, samt plasseringen til kategorien eller subkategorien i listen over disse. Dette er en løsning som er systemspesifikk, og krever at en kjenner

internrepresentasjonen til systemet. Men det hele lagres som tekst slik at det er mulig å finne igjen for andre metoder som skal lese annotasjonen.

6.8.3 Særegenheter i implementasjonen

Resultatene og hastighetene til metoden som helhet er avhengig av implementasjon. I denne prototypen er det mange forhold som påvirker ytelsen og kjøretidsfølelsen. For alle endringer og oppdateringer som kan få konsekvens for annoteringen av sykepleiedokumentene gjøres alt på nytt, inkludert opprettelsen av vektorer. Eksempler er dersom en endrer likhetsmålet, dannes term/dokument matrisen på nytt. En annen situasjon der dette skjer, er om en redigerer rammeverket, eller svarer ”ja” på at nåværende rammeverk skal settes til gjeldende. Dette er gjort for å slippe å ta stilling til spørsmålet om rekalkulering er nødvendig. En gjør altså en antagelse om at dokumentsamlingen likevel er så liten at kostnadene ved å gjøre dette er ubetydelig. I en virkelig anvendelse ville en ikke kunne gjøre det slik på grunn av overheaden dette gir systemet.

Det er laget en annen metode, direkte likhet, i programmet (se Figur 6-3). Denne metoden sier at om ett ord i rammeverket er likt et ord i teksten, så har vi en match. Dette er en alt for grov tilnærming til sannheten til å være brukbar til noe som helst, men gir muligheten til å se om en finner igjen ord fra thesaurusen i teksten i det hele tatt. Den kan antagelig danne grunnlag for en metode etter en boolsk modell, men må omskrives.

6.9 Bruk av SKTax

Det er laget en hjelpemeny med steg for steg forklaring til hvordan en bruker grensesnittet. Denne er forholdsvis stor med mange skjermbiler, og finnes i programmet (se vedlagt CD). Hovedpunktene gjengies her for å gi en oversikt:

- Krever Java 1.4.2 eller nyere for å kjøres. Grafisk brukergrensesnitt i 1024X768 pikslers oppløsning. Fungerer dårlig i mindre oppløsning.
- Laste inn sykepleiedokumentasjon i ren tekst eller RTF format, organisert i foldere, der en fil representerer en rapport eller notat. Minst to filer må lastes inn. Lange filer fungerer, men gir mye "overhead" i det grafiske miljøet.
- Laste inn stoppordliste i eget format. 2 språk finnes forberedt - norsk og engelsk.
- Benytte porterstemmer - 12 ulike språk (hentet fra "snowball") - ikke laget mulighet i grensesnittet til å benytte annet enn norsk.

- Laste inn SabaKlass taksonomien (kun norsk).
- Redigere en thesauri som er automatisk generert ut fra taksonomien. Denne kan utvides og tilpasses som en bruker måtte ønske (ikke fraser)
- Lagre Sabaklass taksonomien en har redigert til eget XML format, lagre ordlistene i en csv-fil.
- Søk etter termer i den viste sykepleiedokumentasjonen - takler flere ord pr spørring. Ordet markeres, og en kan finne alle treffene i dokumentet ved å klikke "neste" for å navigere til gjennfunnede termer. Dette søkefeltet er foreløpig ikke koblet til emner.
- Hvis en fører pekeren over listen av innleste dokumenter, vil emnene knyttet til dokumentet dukke opp.
- Visning av emner knyttet automatisk til dokumentet, atskilte med pekere direkte til plassering i rammeverket. Eget vindu der dokumentet og emnene vises.
- Hjelpemeny for innføring i bruk av programmet, og overfladisk forklaring av teknikken som er brukt.

6.10 CSV, mulige bruksområder

Det har blitt nevnt muligheten for å generere en kommaseparert fil så mange ganger i denne oppgaven, at det må sies noen ord om dette også. For at det ikke skal oppstå forvirring - denne filen har ingenting med oppgaven eller prototypen å gjøre lengre. Lagringsmuligheten ble opprettet fordi det i en periode ble vurdert å forsøke seg på et nevralt nettverk i et program som heter WEKA. Kompleksiteten gjorde at løsningen ikke ble tilpasset, men lagringsmuligheten er ikke fjernet fra programmet fordi den kan representere en mulighet for at en senere kan gjøre seg nytte av filen.

WEKA er et program utviklet for å teste ulike algoritmer for f.eks similaritetsmål eller treningssett for AI applikasjoner [79], og stammer fra et prosjekt ved University of Waikato på New Zealand.

"It shows how to use Weka's Java algorithms to discern meaningful patterns in your data, how to adapt them for your specialized data mining applications, and how to develop your own machine learning schemes".

WEKA tilbyr et grensesnitt som det ville være mulig for prototypen å kommunisere med, eller en kan hente algoritmer fra WEKA. En annen mulighet er å tilpasse en CSV fil til WEKA's eget filformat (*.arff). WEKA tilbyr en stor mengde algoritmer, og en beskrivelse av

ytelsen til algoritmene. Det ble derimot for vanskelig å operere med treningssett og WEKA i prototypen, siden det ikke var planlagt fra begynnelsen. Bruk av AI ville forøvrig også representert en mer komplisert tilnærming en den oppgaven beskriver.

7 Evaluering

Å evaluere aspekter som oppgaven har nevnt, men ikke sett på i detalj, er for stort til at det lar seg gjøre. Diskusjon, kritikk og evaluering rettes derfor mot de konkrete løsningene oppgaven foreslår.

7.1 Diskusjon

7.1.1 Kritikk til prototypen

Oppgaven prøver å fremstille bruk av VSM som metode for automatisk klassifisering av sykepleiedokumentasjon, uten å bevise at dette er beste metode for formålet. Videre benyttes en prototyp for å vise at fremgangsmåten kan være gyldig, men denne er også for svak til å bygge sikre konklusjoner på.

Valget av VSM for å beregne likhet mellom dokumenter og rammeverk er gjort på ut ifra en vurdering av hvilken av de aktuelle, tradisjonelle metodene, som best passet til problemdomenet.

VSM lider under at man kan få mindre likhet jo større spørringen er, og spørringen blir stor dersom en skal fange opp alle skrivemåtene som kan dukke opp for emnene. Likeledes er både dokumentsamlingens innhold og lengde et problem for metoden. Små dokumenter med viktige termer kan risikere å "drukne" sammen med større dokumenter med de samme termene. Sykepleiedokumentasjon består vanligvis av dokumenter av svært ulik lengde – alt fra en linje til flere sider. VSM modellen kompenserer ved å normalisere både spørring og dokumentsamling, men håndteringen av resultatene i prototypen er det riktig å stille spørsmålstegn ved. Løsningen med å sette en grense for hva som er gyldige treff, og utelukke de andre svarene ser ikke ut til å fungere bra nok til å imøtekomme variablene nevnt over, og den er bare valgt fordi det var enklest å implementere. Skal en få vurdert resultatene VSM kan gi, må løsningen med å sette en terskel vurderes skikkelig mot andre svarhåndteringsmetoder først.

IR-teknikken krever en forberedt thesaurus, og denne er ikke god nok i prototypen. Dette er et forhold det finnes gode muligheter til å forbedre gjennom å bruke bakgrunnsmaterialet til Sabaklass er bygd på til å konstruere en god thesaurus. Hvis metoden benyttes i et system der deler av dokumentasjonen allerede er klassifisert manuelt ved innlegging, kan emneord for thesaurusen beregnes på bakgrunn av statistikk i denne dokumentasjonen, og en kan slik få en bra thesaurus som også er tilpasset lokale forhold.

Valget av metodikk begrunnes i sykepleiedokumentasjonens natur. Sammenlignet med legedokumentasjon, mangler sykepleiedokumentasjonen ofte nødvendig struktur som trengs for å få til en regelbasert eller semantisk tilnærming (til sammenligning følger f.eks legene en SOAP strategi - Subjective, Objective, Asessment, Plan uavhengig av rammeverk de også kan benytte). I fremtiden, når rammeverk har påvirket sykepleierne til mer uniform dokumenteringsstil, blir dette antagelig lettere, men i dagens situasjon tror jeg en løsere søkestrategi kan være nyttig.

Hele prototypen er utviklet for å prøve å vise styrker og svakheter ved IR-teknikker til formålet å klassifisere sykepleiedokumentasjon. Prototypingen er også et ønske om å få vist det i praksis, slik at sykepleiere kan få bedre oversikt og innblikk i problemer knyttet til automatisk klassifisering.

Kritikk til implementasjon av IR-teknikk

Under følger en liste over kjente problemstillinger i programmet:

- Programmet kan ikke kontrollere at thesaurusen inneholder stemte eller ustemte termer, eller en blanding. Problemet er alvorlig, fordi likhetsberegning kan bli utført på feil grunnlag. Dobbeltstemte termer kan også forekomme, og brukeren må påse at thesaurusen er riktig. Dette er et resultat av en inkonsekvent designstrategi basert i ønske om å støtte flere stemmingsmetoder, noe som er underordnet hovedproblemstillingen og burde vært unngått.

- Innlesning av XML-filene (Taksonomifilene "diagnose.xml" og "tiltak.xml") feiler på grunn av størrelsen en array kan ha i Javas SAX-parser. Feilen forårsaker at innholdet mellom taggene i XML-dokumentet delvis kan forsvinne. Feilen øker, eller forskyver seg om man vil (mest sannsynlig), for hver lagring og innlesning av redigerte filer. Det er laget en workaround for dette som ser ut til å fungere i de fleste tilfeller, men å bytte parser kan bli nødvendig.

- Det kan se ut som om xml-filen som blir skrevet når en lagrer rammeverket av og til legger til et mellomrom (eks: <thesauri>ordet </thesauri>) Dette representerer ikke noe problem for prototypen, men gir inkonsistens for utveksling av filformatet.

- Innlesningen av dokumentsamlingen tar ikke vare på noen andre metadata enn filens lagringsdato, og stripper sågar bort all potensiell brukbar informasjon fra teksten (utheving, formatering osv). Innlesning kan være feilaktig dersom tegn ikke er etter tekststandardene (ISO-8859-1, UTF-8, ASCII), og kontrolltegn innenfor standardene

kan også representere ukjente problemer. Kjente problemer er knyttet spesielt til RTF-filer, og RTF filer laget med Microsoft Word 97 eller senere viser seg å være vanskelige å lese med den implementerte løsningen.

- Itereringer og ressurssløsing knyttet til GUI (Graphical User Interface) påvirker brukerfølelsen. I ett kjent tilfelle (markering av ord i GUI med gult) stemmer ikke de fremviste resultatene 100 % med ordene brukt i utregning. I sin ytterste konsekvens kan ønsket om å få til en grafisk fremstilling av resultater ha påvirket de virkelige resultatene, men dette er ikke kjent.
- GUI henger etter når en har foretatt innstillinger. Selv om internrepresentasjonen er oppdatert etter justeringer foretatt av bruker, kan feil resultater komme frem på skjermen.

I tillegg kommer bugs i selve implementasjonen som ikke er kjente. Arbeidet som legges ved denne oppgaven er revisjon 6, der bl.a. hele indekseringsmetodikken for inverterte filer ble omskrevet for å stemme bedre med teorien for indeksering beskrevet i oppgaven. Så langt jeg kan se, fungerer også dannelsen av dokumentvektorer nå etter beskrevet teori for modellen. Årsaken til alle feilene og dårlige valg er selvfølgelig knyttet til at det har vært vanskelig å få oversikt over domenet, og at ressurser som skulle brukes i prototypen ikke har foreligget før implementasjonen delvis var startet. Utviklingen og beslutningene har vært gjort i sin helhet av en person, noe som gjør at grunnlaget likhetsvurderingen beregnes på kan inneholde feilkilder som ikke er kjente. Tidsfaktoren har påvirket utviklingen, testingen og debuggingen kraftig.

7.1.2 Ikke konkrete bevis for god kvalitet på klassifisering vha prototypen

Mangelen på en dokumentsamling som allerede har emner knyttet til seg slik at en kan sammenligne treffene metoden gir med en predefinert fasit, gjør evalueringen av metoden svært vanskelig. At metoden gir gode resultat når man putter inn eksakte søketermer tilpasset dokumentsamlingen i thesaurusen, er ikke bevis godt nok.

Problemene med referansesamling er grunnleggende. Å bedrive forskning og prototyping på dette feltet er vanskelig nok om en ikke i tillegg skal streve med å måtte prøve å skaffe dokumentasjon. Å måtte forholde seg til taushetsplikt når en skal vise frem resultater påvirker arbeidet negativt. NSF bør gå i bresjen og skaffe til veie dokumentasjon i en standardisert samling for slikt arbeid. En slik samling vil også være nyttig for utviklere av EPJ-systemer generelt, og for dem som bedriver undervisning i bruk av slike systemer. Anonymisering av

reell sykepleiedokumentasjon hadde vært det mest ønskelige. Dette er forøvrig et forskningstema som har vært gjenstand for mye debatt, bl.a på grunn av pasientsikkerhet og personvern. Slike samlinger kan ligne TREC, CACM eller ISI [2], men tilpasset et formål som det her er snakk om. Om samlingen er oppdiktet er uvesentlig for typen applikasjoner som denne oppgaven representer, men den må være representativ for reelle forhold.

De statistiske verdiene for hvor bra ett system kan sies å være, måles i treff, bom og støy. Dersom 100 % tilsvarer menneskelig klassifisering, vil man kunne regne 85 % som bra av et datasystem. Det betyr at bom og støy får 15 % til sammen [13]. Det er ikke utført tester på precision/recall i prototypen, siden grunnlaget for å gjøre det har vært så dårlig både med tanke på referansesamling og kvaliteten til thesauri. Etter å ha sett de resultatene prototypen kom opp med, vet man at dette ikke er i nærheten av 85 %. Generelt kan et system som dette, med en godt opparbeidet thesaurus muligens kunne gi opp til 60 % riktig klassifisering, skal en gå etter et anslag basert på uttalelser i [13]. Men siden rammeverkene gjør det mulig å bygge en svært domenespesifikk thesaurus, kan treffene forbedres ytterligere. Muligheten metoden har for raffinering, er også gode. Implementasjonen i prototypen følger algoritmen så tett som mulig, men utnytter ikke tilgjengelige metadata. I et reelt EPJ miljø vil opplysninger om dokumentet finnes på ulikt vis både i og utenfor selve dokumentet, og disse kan påvirke vektingen av enkelttermer. Et eksempel på dette er artikkelen fra [27] som viser til gode resultater i sin versjon av VSM-modellen som overgår de mest pessimistiske tallene fra [13].

Det er sannsynlig at andre teknikker må benyttes i tillegg for å kunne klassifisere riktig på f.eks. subkategorinivå i Sabaklass. Prototypen baserer seg på en søketeknikk, ikke språklig logikk. Å tro at en kan få ut meningen av en tekst ved å plukke ut viktige ord og gjengi meningen basert på ett sett av ord er en stor overforenkling [2]. I tillegg er matchingen mellom dokumentet og rammeverket basert på enda færre unike termer på disse nivåene, og en kan således risikere å ikke finne passende plassering i rammeverket, eller bomme på plasseringen. Derfor er det ingen overraskelse at klassifisering på en slik måte kan gi upålitelige resultat. Måten dette er løst på i prototypen er et resultat av dens evolusjon. Opprinnelig mål var å knytte emner til overordnede emneord fra "VIPS", men Sabaklass gjorde det mulig å spesifisere mer siden det har en konkret oppbygging. Metoden ser ut til å fungere bra når en slår sammen undernivåer til bare funksjonsområdene / behandlingskomponenter, men jeg tror ikke en slik direkte sammenslåing av hierarkiet er fornuftig. En bør i tillegg utnytte mulighetene VSM gir til å redefinere søk basert på delvise resultater. Slik forholdene er per i dag, er det for lite grunnlag til å konkludere, og kanskje må

en innse at denne metoden ikke egner seg for mer enn emnetilknytting på det høyreliggende nivået, og ikke klassifisering på kategori og subkategorinivå.

7.1.3 Diffus fremstilling av resultater i prototyp

Scopet på hva som ønskes sammenlignet med spørringen er bra definert i oppgaven; vektorer laget av enkeltrapporter organisert som filer, og spørringer basert på en predefinert ordliste. Det er derimot litt diffust om jeg mener jeg kommer frem til emner basert på "Behandlingskomponenter" eller deres kategorier og subkategorier i prototypen. Begrunnelsen er gitt i avsnittet over; siden ordlisten består av ord som forekommer fra høyere nivå, kan det være bare ett term som skiller en subkategori fra en annen i den automatisk genererte thesaurusen. Dersom dette ordet ikke forekommer i det hele tatt i dokumentsamlingen, kan en risikere at likhet beregnes som helt lik for to eller flere subkategorier. Denne effekten er tydelig om en gransker similaritetsmålene, der en ser at flere søk har fått samme likhetsverdi. Prototypen er av samme grunn litt vag når det gjelder fremstilling av resultatene.

Det er andre problemstillinger rundt fremstillingen av resultatene som bør belyses. Ordene som blir markert med gult når en klikker på en diagnose eller tiltak, er ikke nødvendigvis ordet som ble benyttet for likhetsberegningen. Årsaken er implementasjonsavhengig, og har med programmeringsmiljøet å gjøre. Klassen som styrer markeringen, er en såkalt "HighLighter". Denne søker etter mønster i teksten, og markerer ord som matcher mønsteret. Mønstrene er også "case- sensitive". Problemet er at ordene er behandlet med tokenisering og stemming, noe som ikke gjør at de trenger å ligne mønsteret "HighLighter" benytter. Ergo får en deler av ord markert, eller de markeres ikke i det hele tatt. Stegene som er tatt for å bøte på dette er å kjøre highlighting to ganger, første gang skal ord som starter med samme bokstavrekke markeres, og andre gang gjøres første bokstav kapitalisert. Markeringen av ord i grensesnittet har ikke med likhetsutmålingen å gjøre, men det hadde vært ønskelig at dette skulle fungert bedre. Markeringen er forøvrig veldig ressurskrevende, og at den kjøres flere ganger ville vært uholdbart i en virkelig applikasjon.

Det finnes et søkefelt i hovedskjermbildet som bare søker etter termer i det aktuelle dokumentet vha ”HighLighter”. Dette feltet bør kunne takle emner som søkeord og gjøre søk i alle dokumentene i tillegg, for å gjøre søkefunksjonaliteten mer fullstendig.

7.1.4 Aspekter som ikke er fulgt opp

Generelt virker det som tiltak fungerer litt bedre enn diagnoser. Dette forklares av at tiltakene har beskrivelse på lavere nivå enn diagnosene har og dermed en mer spesifisert thesauri. Å bruke samme likhetsmål på begge taksonomiene kan være en feilkilde. Jeg har i arbeidet med prototypen ikke operert med bruk av rammeverket på dokumentsamlingen i betydning forsøke å relatere Sabaklass' diagnoser og tiltak til hverandre. Dette hadde blitt for ambisiøst, og heller ikke særlig nyttig med tanke på de foreløpige resultatene prototypen har gitt. Likevel har jeg gjort forsøk på å slå sammen søkene basert på ordlistene fra de to taksonomiene, og slik komme frem til felles emner på behandlingskomponentnivå. Om disse observasjonene kan utnyttes, har jeg ikke tatt stilling til utover dette.

Prototypen abstraherer seg fra virkeligheten. Den gjør antagelser om at man kan holde hele indeksen i minnet, og at systemet vil takle og indeksere for alle brukere. Prototypen gjør også antagelser om at resultater kan lagres, i hvert fall midlertidig og ut ifra brukerens preferanser. Implementert i virkeligheten måtte man kanskje indeksere i stor skala. Det er ikke gjort en totalvurdering av konsekvensene av dette i et reelt system, verken juridisk eller teknisk.

Integrering sammen med eksisterende løsningers bruk av klassifisering ikke vurdert nøye, bare omtalt som at de kan benyttes sammen. Dette kan bli mer komplisert enn oppgaven beskriver. At konflikter mellom allerede klassifisert tekst, og uklassifisert tekst som automatisk klassifiseres kan oppstå, er mer enn sannsynlig. Derfor må tilpassinger gjøres som følge av innføring av den omtalte teknikken i et EPJ-system. En mulig løsning er å utelukke allerede klassifiserte dokumenter fra indekseringsfasen, og heller inkludere dem i navigering (eller annen bruk). Automatisk klassifiserte dokumenter bør markeres som dette siden resultatene er upålitelige. Å benytte de automatiske resultatene som permanent del av dokumentasjonen ser jeg som urealistisk. Videre har jeg ikke forhørt meg om muligheten for å legge slik funksjonalitet inn i et virkelig system f.eks som plug-in eller modul. Men både DIPS og DocuLive påberoper seg å være modulære, så det er ikke umulig at det kan gå. I tillegg har jeg tolket signalene fra representanter fra disse EPJ-systemene at en løsning som her foreslåes kan være interessant.

Ekstern testing av prototyp

Prototypen ble gjort tilgjengelig for testing på en hjemmeside, og sykepleiere og sykepleiestudenter ble kontaktet. Tilbakemelding skulle foregå på et spørreskjema, og intensjonen var å få inntrykk av hva de mente nytteverdien av emnetilknytning på dette viset

er. Responsen på spørreundersøkelsen var så lav at det ikke ga noe verdifullt grunnlag å si noe mer om dette.

7.2 Kost vs. Nytteverdi

De følgende kapitlene debatterer om nytteverdien av omtalte teknikker står i forhold til kostnader.

7.2.1 Enkelt vs. Avansert

Har gamle, tradisjonelle metoder noe å tilby når NLP (Natural Language Prosessing) og AI (Artificial Intelligence) har kommet så langt som "State of the Art"-artikklene [3], [27], [29] og [47] beskriver? Sammenligner man sykepleiedokumentasjon med lege (eller radiolog) dokumentasjon, er sykepleiedokumentasjonen annerledes, men ikke *så* annerledes. Det er fellestrekk for de nevnte teknologiene i artikklene som kan relateres:

- Alle disse bruker NLP som bakgrunn, som delvis kan sies å begynne slik denne oppgaven gjør. Når man har grunnleggende data, kan en lage et ekspertsystem - regelbasert metode for å komme fra observasjon til aksjon (Finner X, gjør Y)
- Bayesian metoder, gjerne kalt AI - Treningssett med kjent innhold, og justere slik at en får et kjent utfall. Kan så appliseres til ukjent tekst. NLP som nevnt over, kan danne grunnlag for slik prosessering. Å benytte AI for å påvirke vektene til VSM modellen ville la seg gjøre slik som i [27].
- Mer eller mindre bruk av universelle thesauruser, foreløpig lages thesauruser spesielt for formålet. I likhet med prototypen, benytter de nevnte prosjektene sine egne thesauruser. Overordnede thesauruser og utveksling av thesauruser som har flere bruksområder er vanskelig å lage.

Denne oppgaven tar ikke for seg å forsøke å semantisk tolke innholdet i dokumentasjonen slik som artiklene tar for seg. Årsaken for dette, er at sykepleiedokumentasjonen ikke møter to vesentlige krav for dette:

"Morphological decomposition and identification precedes semantic analysis. It is only when these two prerequisites are fulfilled that an attempt to grasp the meaning of a whole expression is made possible" [17].

I fremtiden vil EPJ gjøre at sykepleiere endrer vaner, og får bedre opplæring til å benytte eksisterende hjelpemidler. Først når dokumentasjonen er av en slik karakter at god analyse blir mulig automatisk, vil dette skje.

Mye av opplysningene i sykepleierapportene er opplysninger som hører hjemme andre steder (kurve, labrapporter, legejournaler mm) Dette kompliserer situasjonen ytterligere - skal en lage system for å fange opp dette også? I så fall blir et stort puslespill. Ideelt kan en si at vi bare skal lage systemer som fanger opp sykepleie diagnoser og tiltak (evt. evalueringer når disse blir oversatt), men virkeligheten er sammenflettet, og opplysninger fra andre kilder har kanskje rettmessig plass i sykepleiedokumentasjonen i noen situasjoner. Et automatisk system måtte i så fall gjøre valg, og kan hende får vi en situasjon med overlappende klassifiseringer, noe som gjør at vi får inkonsistens i forhold til hvordan andre systemer klassifiserer. Denne problemstillingen er for så vidt gyldig for all form for automatisk klassifisering.

Krav til nøyaktighet når en jobber med liv og helse er naturlig nok høye. Automatisk klassifisering av sykepleie er vanskelig på grunn av sykepleiens natur, den spenner over like mye, og kanskje mer, enn en leges eller radiologs naturlige nedslagsfelt. Bruk av automatisk klassifisering innen sykepleiens domene bør derfor sees i lys av dette. Søking og navigering i dokumentasjonen tåler feil klassifisering, men beslutningsstøtte, forskning og økonomisk evaluering gjør det ikke.

"Noe trengs snart!" er svaret på spørsmålet kapittelet stiller initialt. Vi kan ikke godta at det er tungvindt for sykepleiere som har det hektisk allerede når det finnes forbedringspotensial. Alle de ulike typene notater som finnes i dag i enkelte systemer gjør det vanskelig å få oversikt - et enkelt verktøy for å se hva dokumentene handler om uten å åpne dem er gull verdt. Søk etter enkeltord er ikke holdbart. Denne oppgavens hovedargumenter for å bruke en så enkel tilnærming for å nå dette målet, er disse:

- Kostnad målt i penger og prosessering er lav for metodikken i forhold til andre alternativer.
- Enkel implementasjon kontra alternativene som er svært avanserte, og dyre å utvikle og vedlikeholde.
- Allmenngyldig grunnlag som kan benyttes for videre utvikling om man vil gjøre implementasjonen mer avansert når tiden er moden for det.
- Stadig endring i teknologier, oppdateringer og andre rammeverk gjør at man trenger en tilpassningsdyktig metodikk.
- Kombinere systemer og rammeverk, en tilnærming som ligner "plug-ins" gjør at utviklingen kan foregå parallelt med andre forutsetninger.

7.2.2 Må ha vs. Kjekt å ha

Mulighetene emnetilknytning tilbyr ustrukturert dokumentasjon, gjør at det oppfattes som mer enn en "kjekt å ha" funksjon. Men er dette noe man må ha? Det finnes ikke noe krav til slik gjenfinning i EPJ-standarder.

Nytten av å kunne klassifisere ukjent tekst for søking er stor, noe som gjør teknikken en til en sterk kandidat for å bli ønsket av sykepleiere, og andre yrkesgrupper som har lignende problemer med å finne igjen og navigere i stor og uoversiktlig dokumentasjonsmasse. Men det er viktig å få frem at funksjonalitet som systemutviklere og sykepleiere med gode datakunnskaper ser som nyttige, kan være til plage for pleiere uten slik erfaring. Ved en demonstrasjon av DocuLive's hyperlenkingsfunksjonaliet fra et dokument til en annen ressurs, var spørsmålet fra salen "Hva er vitsen?". Pleieren som spurte var valgt som representant fra en avdelig som "superbruker". Dette er ikke ment som hån, men det belyser at man ikke kan forvente at systemer blir brukt etter hensikten uten at brukerne er lært opp til å bruke dataverktøy generelt, og systemet spesielt. Pleiere som kan et system godt, f.eks et eldre linjebasert kommandosystem med spesialiserte funksjonsknapper, kan ha problemer med å løfte blikket til mer overordnede ideer uten at de har nødvendige datarelaterte bakgrunnskunnskaper.

Det er ingen grunn til grunn til at denne metoden skal forsøke å konkurrere med implementerte løsninger i eksisterende systemer som finnes i f.eks "DIPS". Det fornuftige er å heller se det som et supplement. Selv i systemer som allerede bruker klassifisering for sykepleiedokumentasjonen, er det mange som ikke benytter seg av muligheten til å legge til emneord for teksten. DIPS tilbyr en mer "levende" sykepleiedokumentasjon enn mange av konkurrentene fordi de har et sykepleieplanopplegg som er ment å jobbes med. Strukturering av informasjonen gjør at sykepleierne lettere finner frem til viktig informasjon [63], men at sykepleieplanene gjør det vankeligere å få oversikt ved f.eks utskrivning av en pasient. EPJ-systemer støtter gjerne søk på emner dersom de er lagt inn av den som dokumenterte, men for å få oversikt over pasientens problemer innebærer som oftest å åpne alle de dokumentene en tror kan være aktuelle. Emnebasert gjenfinning er altså nyttig når en skal danne seg et bilde av en ukjent pasient, og dette krever ikke helt eksakt klassifisering.

Å stille en sykepleiediagnose krever analyse, syntese og nøyaktighet i tolkning og forståelse av komplekse kliniske data [19]. Dette kan ikke oppnås med denne metoden. Å basere seg på automatisk klassifisering er uansett forbundet med risiko, uansett kvaliteten på verktøyet. Selv menneskelig klassifiserte dokumenter kan være feil, alt etter hvilket

paradigme man vurderer etter. Som støtteredskap representerer metoden en mulighet for tilgang på informasjon på et høyere nivå enn søk på spesifikke termer. En kan tenke seg at automatisk emnetilknytning kan påvirke til å utnytte klassifiseringen enda bedre enn dagens systemer gjør, ved at det blir lettere for pleieren å velge riktig ved å foreslå ett knippe emner fra systemets rammeverk. På dette viset kan bruk av dokumentasjonen utvide dens nytte, og den blir ikke liggende som "død" kunnskap i en stor database. Argumentasjonen er at det ikke skrives noe av metoden som skal inn i journalen - det er det sykepleieren som gjør.

I en utvidet sammenheng kan metodikken i forbindelse med andre rammeverk og meningsbærende lenkingsmekanismer gi rask tilgang på informasjon slik at utvidet eller ny kunnskap kan dannes. Løsninger basert på vektorrommodellen gir mulighet for å søke på mønstre og avhengigheter som tekst knyttet til en databasestruktur vanskeliggjør, en kan finne kunnskap som ikke var påtenkt å lete etter når databasen var strukturert [27]. At metoden kan eksistere sammen med andre organiseringsmetoder eller klassifiseringsidiomer er en stor styrke, og kan skape mangfold uten å skape kaos. Man er ikke bundet til et spesielt utvalgt rammeverk, men kan tilpasse etter behov. Jo mer avansert en gjør implementasjonen, jo vanskeligere blir selvfølgelig dette.

Bruk av hyperlenking

Nødvendigheten av ekstern hyperlenking er viktig aspekt. Det er lett å bli imponert over egenskapene hyperlenking har, og mulighetene dette gir. Men en må huske at i tillegg til at sykepleiedokumentasjonen er et arbeidsverktøy, er det er knyttet et strengt regelverk til hvordan dokumentasjonen håndteres og brukes. Det er ikke sikkert det er hensiktsmessig å fylle brukergrensesnittet med mange lenker til notatet. En bør unngå at grensesnittet rotes til med informasjon og lenker når en ikke trenger dem. De lovmessige forholdene forhindrer ikke at en kan hente relatert informasjon fra en annen ressurs enn selve EPJ systemet, men en kan ikke tillate ubegrenset tilgang til f.eks internett fra systemet på grunn av sikkerhetsmessige årsaker. Ressurser som NEL kan inngå i det lokale nettverket eller finnes på CD- rom, noe som demper sikkerhetsrisikoen.

I et studieprosjekt fra NTNU i 2001 om primærlegers bruk av elektroniske oppslagsverk, vises det til at papirversjonen av oppslagsverk ofte blir brukt i stedet:

"Mange hadde norsk legemiddelhåndbok og felleskatalogen i elektronisk utgave, men det inntrykket som ble gitt var at de like gjerne slo opp i papirutgaven. En grunn til at legene heller slår opp i papirutgaven kan være at dette tar mindre tid" [8].

Som praktiserende sykepleier samsvarer dette med min erfaring. Oppslagsverk som Legemiddelhåndboken er nyttige verktøy når en raskt trenger en oppfriskning av kunnskapen på ett gitt område, men å lete seg frem til informasjonen en trenger kan foregå like raskt i en bok. En slik situasjon der teknologien og behovene er i utakt, er ikke fremmed for dem som jobber i helsevesenet. Dersom det hadde vært lenker til nærliggende emner knyttet til sykepleiedokumentasjonen direkte, ville slike oppslag kunne foregå raskt, og dermed bli mer brukt.

Et annet tilfelle der jeg har opplevd et lignende behov er i forbindelse med prosedyrer. De nettbaserte prosedyrebøkene som er tilgjengelig ved sykehuset jeg praktiserer ved har hatt en svært svak søkefunksjonalitet, kanskje så ille som boolske søk på eksakte ord og skriveformer. Dette har resultert i bruk av papirversjoner i stedet. Siden papirversjonene gradvis har blitt utdatert, og det har vært fokus på bruk av online prosedyrehåndbøker, har papirversjonene blitt fjernet uten at søkefunksjonaliteten har blitt oppgradert på online ressurser.

Lenking til f.eks prosedyrehåndbøker basert på automatiske søk utvider bruksområdet for dokumentasjonen og letter hverdagen for pleierne. Siden de eksterne ressursene som omtales her gjerne finnes i helseforetaket allerede (en generalisering basert på utbredelsen av Legemiddelhåndboken, og at prosedyrebøker er vanlige) trenger ikke utvidelsen bli for kostbar. Jeg er av den oppfattning at hyperlenking brukt riktig, kan lette hverdagen for pleierne, og sørge for at de benytter oppdatert informasjon. Linkemekanismene trenger ikke å være markert i dokumentet, det kan f.eks. dukke opp som resultat av tastekombinasjoner, eller instansieres i et lite vindu ved siden av om pleieren ønsker det. Måten DocuLive har implementert lenking, med lenker i margen på dokumentnivå, virker som en fin løsning.

For å redusere kostnadene ytterligere og bedre gjenbruk, ville det være ideelt å benytte lenkingsmetoder som ikke var egne, systemspesifikke løsninger, men som bygde på prinsipper om modularitet og standarder. Teknikkene som jeg forslår for slik navigering er kraftfulle og standardiserte. Likevel er det mange problemer knyttet til dem. Emnekart og XLink krever en underliggende struktur for å ivareta semantikk skikkelig (eksempler kan være RDF (resurs- og relasjonsbeskrivelse), FRBR (brukt for å beskrive bibliografiske attributter og relasjoner), PSI (beskrive identifikator til en instans)). Alt dette gjør det hele veldig komplekst, og sammenslåing av f.eks emnekart viser seg å fungere dårlig i praksis. Mange emner som delvis er like og dekker det samme, tap av struktur pga for enkle semantiske modeller, feil assosiasjoner er blant problemene. Bildet som beskrives når en snakker om teknikkene er ofte for rosenrødt og teoretisk, mye arbeid gjenstår for å

standardisere bra nok for at dette skal kunne fungere bra innen EPJ-systemer og mellom EPJ-systemer i praksis.

7.2.3 Kostnader

Nytte/kostnad vurderingen er ikke til å komme forbi når en ser på automatisk klassifisering av fri tekst. Forutsetningene for å benytte metoden "Vector Space Model" i et EPJ system er gode, siden man har et predefinert antall dokumenter å søke i, og et kjent domene for spørringer å sende inn i samlingen. Disse betraktningene vil være allmenngyldige for tiltenkt scope i virkeligheten, altså at en ser for seg metoden applisert innen de rammene og den dokumentasjonen sykepleieren har tilgang til gjennom sin sikkerhetsklarering i systemet. Metoden kan tilpasses slik at den ikke trenger lagre noe eller påvirke originaldokumentene.

En må ta med det følgende i betraktningen: Vil en ha stor nøyaktighet, koster det tid, penger, arbeidskraft og regnekraft. Opparbeidingen av en god invertert fil gjennom av leksikalsk analyse i et system utgjør 50 % av indekseringskostnadene. Dette skulle også gjelde et enkelt system som denne oppgaven omtaler [46].

Hvis en ser på tallene fra TREC samlingen, så kan 1GB dokumentasjon komprimeres til 5 MB indeksert. Med indekseringsord mener jeg ord som brukes i tf-idf matrisen, ikke emneord eller begreper som rammeverket tilbyr. Ved st Olav Hospital var det i 2003 [75]: 47 984 innleggelser og 289 647 liggedøgn. Ganger man liggedøgnene med 3 rapporter, og legger til innleggelsesskrivet får en 916 925 potensielle skriv med sykepleiedokumentasjon direkte. I tillegg kommer ulike administrative skriv, skriv til andre samarbeidspartnere og muligens noen sykepleieskriv fra de 275 074 polikliniske konsultasjonene som ble holdt. Hva dette betyr i antall Gigabyte/Terrabyte har jeg ikke oversikt over, men bare å utvide databasen til å holde på indekseringsordene i sykepleiedokumentasjonen for alle de 650 000 pasientene som sokner til regionsykehuset i Helse-Midtnorge vil utgjøre en betydelig økonomisk kostnad i tillegg til administrasjons- og tjener belastning. Derfor er det urealistisk å gjøre dette for all sykepleiedokumentasjon. Men det er realistisk å gjøre et for aktiv dokumentasjon, evt. indeksere ved behov (tiltenkt scope). Dersom lagring er aktuelt, er det en ideell situasjon for VSM. Siden lovgivningen gjør at dokumentasjonen er statisk, vil ikke indekseringsordene endre seg for det aktuelle dokumentet (med mindre en endrer indekseringsstrategi), og en kan legge til indekseringsord for ny dokumentasjon etter hvert som en så kan innlemme i søket.

Foruten kostnadene med indeksering, trenger VSM så mange ord per vektor at enkel aritmetikk trenger mye tid når den er implementert som i prototypen. For en blanding av store

og små dokumenter, vil de små dokumentenes vektorer representere sløsing av ressurser. Vektorene må rekalkuleres dersom nye dokumenter skal legges til søket for å beregne vektene riktig. Dette gir en stor overhead, noe som i en realistisk setting vil ha betydning, også for kost / nytteforholdet.

Et fenomen som kan gi problemer, er størrelsesforholdet mellom de store rammeverkene og hvor små sykepleierapporter pleier å være. Vanligvis er situasjonen omvent for implementasjoner som bygger på modellen, og lange spørringer gir som oftest bedre resultat [14]. Dette noe som må vurderes når en bedre og relevant dokumentsamling foreligger, og problemene med å skaffe en slik er tidligere omtalt. Testing av den foreslåtte metodens presisjon og tilbakekalling kan gjøres ved å se thesaurusen som query, og på forhånd definere hvilke emneord som burde dukke opp for hvert enkelt dokument. Samlingen må da ikke endres for å sikre konsistens. Dersom testingen påviser at denne metoden er "bra nok", er den i nytte/kostnad vurderingen et sterkt kort sammenlignet med mer avanserte metoders kostnader.

Med disse betraktningene i mente, kan en sammenligne hva som skal til for mer avanserte regelbaserte systemer: Indeksering kan en anta blir nødvendig uansett. Man må i tillegg trene opp eller manuelt (eller en kombinasjon av disse) fortelle hva som er riktig plassering i ønskede kategorier. Generelt trenger 10-20% av termene i en thesaurus kompliserte regler (muligens flere innen det medisinske domenet). Hvis taksonomien har 1000 nøkkelord, kan 8-900 av dem gjøres automatisk, mens 1-200 av dem bygges manuelt på 10 til 50 timer. Det vanlige antallet dokumenterer som trengs for å lage ett treningssett er ca. 50. Å lage treningssamlinger kan ta opp til en time per term. Når man har treningssettet, starter prosessen med å finne logiske forbindelser mellom termer i dokumentet, og mellom dokumenter i samlingen. Dette returnerer et sett av sannsynlige forbindelser mellom en term og et nøkkelord. Dette fortsetter til man bygger opp nok forbindelser til at man kan kjøre ukjent tekst inn i denne motoren og få kategorisert den automatisk [13].

7.2.4 Innsparingsområder

Det går an å spare i indekseringssammenheng. Eksempelvis kan en indeksere uten pekere, og opprette dem når man trenger dem. Dette vil bety samme prosesseringskostnader senere, men lagringsplass og båndbredde oppfatter jeg som et større problem enn prosesseringskraft og minne i denne sammenhengen, og kan en benytte lokale ressurser (en tykk tjener) i systemet, blir ikke belastningen på det totale systemet for stort. Indekseringsordene kan også være nyttige i flere sammenhenger enn det som foreslåes her (for andre søketeknikker,

forsking og statistikk m.m.), slik at man kan forsvare lagringskostnaden med at en har flere bruksområder dersom en beslutter å lagre ordvektorene.

Kostnader er store i helsevesenet, også innen helseinformatikk. Klassifikasjonssystemene er store og kostbare, både med tanke på utvikling, implementasjon, og rene penger knyttet til lisensiering og provisjon. Kostnadene for å bruke NANDA i et dataprogram er ikke kjent for meg, men det vil dreie seg om beløp i 100 000-kroners klassen. En kan tenke seg at NIC og NOC vil koste noe tilsvarende. VIPS modellen er gratis, men siden den ikke har noe klar definisjon, har den kostnader med å definere det for bruk i dataprogrammet knyttet til seg. Sabaklass er gratis mot at man registrerer bruk av rammeverket. Overliggende nomenklaturer som SNOMED er også dyre. SNOMEDs "mappings" av rammeverkene NANDA, NIC, NOC, Sabaklass og andre sykepleieklassifikasjoner koster $1000 i provisjon hver årlig. Dersom man ønsker emnebaserte søk på ustrukturert tekst, lønner det seg å benytte et rammeverk man allerede har rettigheter til. Siden vi allerede ser at de store leverandørene har valgt forskjellige rammeverk som grunnlag for sykepleiedokumentering i sine EPJ-løsninger, bør søkeverktøy basert på rammeverk kunne støtte flere rammeverk hvis mulig. At søkemetodikken er fleksibel, kan vise seg kostnadsbesvarende også med tanke på de stadige revisjonene av rammeverkene. En tilpassningsdyktig arkitektur er nødvendig i forbindelse med oppgraderinger.

Lovverket pålegger en helseinstitusjon å ta vare på all dokumentasjon, selv etter at pasienten er død. Kostnader knyttet til å lagre på store mengder dokumentasjon som ikke brukes til noe kan sees på som bortkastet. Slik dokumentasjon kan komme til nytte igjen om søk i dem kan gjøres fornuftig og lettvint, f.eks i forskingsøyemed.

Et annet mulig bruksområde for metoden i EPJ-systemet, er for ustrukturert informasjon som lagres som maskinlesbar tekst fra andre kilder. Dette kan dreie seg om dokumentasjon tolket fra skannede eller stemmeprosesserte dokumenter, siden slike dokumenter heller ikke støttes av noen annen form for metadata i seg selv.

Og til slutt bør man vel nevne den viktigste innsparingen: den som er knyttet til sykepleiernes bruk av funksjonaliteten metoden tilbyr. Det blir stadig klarere også i helsevesenet at tid er penger. Dersom metoden kan bidra til at pleierne finner informasjon de leter etter raskere, er hovedmålet nådd.

Kombinasjon av rammeverk

Det ikke lagt opp til dette spesielt i prototypen, og visualiseringen i grensesnittet samt noe spesialiserte overføringer av parametere internt vanskeliggjør utprøving av flere rammeverk.

Et forsøk på å representere "VIPS" i samme xml-format som Sabaklass er blitt utført. VIPS funksjonsområder ble puttet inn i elementene for behandlingskomponenter i XML filen, og alle andre verdier ble satt til "U/A" eller unike tall henholdsvis for tekstlig eller kodifiserte verdier. Selve søket fungerte på samme måte. Dette var for så vidt ingen overraskelse, da det er elementene for emneord/kategorier og thesaurusen som blir benyttet i prototypen. Tallverdier blir kun brukt for å slå opp i trestrukturen til rammeverket i prototypen.

Dette forsøket styrker troen min på at det vil la seg gjøre å kombinere rammeverk om en opererer på emnenivå. Med et annet rammeverk (NANDA, NIC etc.) representert i XML, og ordlister tilpasset det rammeverket, ville det være en smal sak å skrive en parser som om ikke annet kunne søke etter og ta vare på de unike kodene taksonomien til rammeverket har, og benytte dem til oppslag. Dette må leses som en påstand basert på disse gitte forutsetningene.

Forsøket viste også at det må foreligge en annen form for struktur i tillegg som sier noe om hvordan disse emnene fra forskjellige rammeverk skal relateres til hverandre. Å benytte VSM på to forskjellige rammeverk og så relatere disse til hverandre direkte, vil være forbundet med for mange usikkerhetsmomenter. SNOMED finnes for sykepleietaksonomier, og emnekart kan konstrueres for "VIPS" og lignende organiseringer. Dette er likevel ikke så enkelt som det høres ut, blant annet fordi SNOMED ikke støtter toveis lenking, og relasjonene mellom rammeverkene må defineres skikkelig. Relasjoner mellom sykepleierammeverk er nok et forskningsområde vi vil høre mer om i Norge i årene som kommer, og arbeidet som gjøres i forbindelse med å sy sammen NANDA, NIC og NOC i USA [7] tror jeg vil bli toneangivende for løsninger som dukker opp her.

7.3 Lovlighet for slik løsning

Ved forespørsel til de store leverandørene om de planlegger implementering av automatisk klassifisering vha rammeverk i sine EPJ-systemer svarer de at kompleksiteten gjør at de ikke har vurdert det. En logisk slutning å trekke fra det svaret, er at mange av momentene jeg har nevnt vedrørende nytte/kostnad spiller inn, men en kan også tenke seg legale problemer kan ligge til grunn for et slikt svar.

En slik tilnærmingsmetode blir nødvendigvis unøyaktig, ikke minst med tanke på sykepleiedokumentasjonens natur. Slik jeg fremstiller det, er metoden tenkt for hjelp ved søking, for å danne seg et overblikk, og evt. for å greie å klassifisere det en skriver riktig. Dersom pleieren bestemmer seg for å benytte forslaget og legge det inn i journalen, er det hans/hennes ansvar. Men jeg har også argumentert for å benytte metoden for å lage et slags "view" en kan navigere etter, og det kan skape mer tvil. Dette problem-, eller emneorienterte,

"view'et" er altså ikke notert i journalen, men er skapt etter hva pleieren ønsker å finne i ustrukturerte rapporter. Jeg tenker metoden anvendt innen de begrensninger som EPJ systemet setter for pålogget bruker, i prinsippet anvendt på den pasienten sykepleieren ser på akkurat nå, og de rapportene han har tilgang til å lese. Systemet mitt lagrer altså ingen ting; om forslaget som kommer opp anvendes, må pleieren stå ansvarlig for det.

For å bekrefte at dette kan være lovlig, tok jeg kontakt med KITH. Det skulle ikke være noe prinsipielt i veien for å tilføre slik funksjonalitet var svaret derfra.

7.4 Oppsummering

Hovedtema for oppgaven har vært å forsøke å knytte emner til ustrukturert sykepleiedokumentasjon (tekst), ved hjelp av et rammeverk utviklet for denne type dokumentasjon. Slike rammeverk er brukt for strukturert innlegging i EPJ systemer i dag, men er i liten grad utnyttet for å finne igjen informasjon som ikke har emner knyttet til seg. Det finnes store mengder ustrukturert informasjon i sykepleiedokumentasjonen av disse hovedårsakene:

- Gammel dokumentasjon skannes inn, og lagres som bilder.
- EPJ-systemet støtter ikke rammeverk i det hele tatt, eller dokumentasjonen er skrevet før rammeverk har blitt tilpasset EPJ - systemet.
- Tilpasset rammeverk blir ikke benyttet av pleierne.

Metoden som beskrives i oppgaven kan gjøre lite for punkt 1, med mindre den benyttes i sammen med OCR-programvare, men har et potensial for å lette gjenfinning og navigasjon for punkt 2 og 3.

Oppgaven gaper over veldig mange tema, og har derfor blitt mer en oversikt over problemdomene, selv om konkrete forslag til tilnærming blir gitt. Dette er et resultat av at det er så mange aspekter å vurdere i denne sammenhengen, og at det har vært lite beskrivelse av problematikken for sykepleiedokumentasjon å finne. Arbeidet i denne oppgaven har forsøkt å holde seg

- Systemnøytral
- Rammeverknøytral
- Til universelle løsninger som kan tilpasses spesielt, og i kombinasjon

Arbeidet med emnetilknytning av ustrukturert sykepleiedokumentasjon har avslørt at til tross for tiår med utvikling av rammeverk, er det flere forhold som vanskeliggjør forskning

for utenforstående. At det ikke er gjort mer med problemstillingen i Norge, har med at rammeverkene først nå har blitt oversatt, og at de ikke er gjort tilgjengelig for denne typen forskning enda. Jeg håper de som sitter med rettighetene til de ulike rammeverkene forstår at utviklingen avhenger av at forskere og studenter får tilgang til dem, ikke bare de store firmaene som kan betale for dem. Problemer arbeidet har møtt er oppsummert disse:

- Utilgjengelighet av rammeverkene: Rettigheter og mangel på standardiserte, dataleselige formater
- Ingen tilgjengelige programmer, plattformer eller biblioteker som kan danne grunnlag for utgangspunkt. Prototyping må bygge alt fra scratch
- Ingen dokumentsamling for testing, vanskelig å få tak i anonymisert dokumentasjon som kan benyttes til å vise resultater en kommer frem til.
- Lite eller ingen litteratur ute fra andre som har sett på det samme innen sykepleiedokumentasjon. Det meste dreier seg om struktur ved innlegging.

Likheter med andre sammenlignbare profesjoners dokumentasjon er klart tilstede, samtidig som det finnes særegenheter. Arbeidet med prototypen forsøker å vise emnetilknytning etter en generell metodikk som er spesielt tilpasset vha en ordliste og rammeverk. Prototypen forsøker å vise potensialet rammeverk har til å få mer ut av ustrukturert tekst, både til søking og navigering lokalt. Oppgaven beskriver også at slik emnetilknytning kan ha nytte i en større sammenheng. Knytninger til andre rammeverk og eksterne ressurser er eksempler som er nevnt. Teknikkene som er valg ut har bl.a disse fordelene:

- Standardisert, kan benyttes sammen
- Gratis i anskaffelse (GPL lisenser)
- Intelligensen ligger i programvaren, ikke taggingen - kan derfor tilpasses eksisterende løsninger.
- Kan lagres som ren tekst - kan utveksles
- Relasjoner kan opprettes "on the fly"
- Kan danne nye, mer overordnede representasjoner av selve teksten. Søking kan foregå på konseptnivå, ikke på enkelt termer

8 Konklusjon

Hovedmålet med oppgaven var å vise sykepleierammeverkenes potensial til å forbedre sykepleiernes søk etter informasjon i et EPJ-system ved å knytte emner til ustrukturert sykepleiedokumentasjon.

Dette forsøkes gjort gjennom å belyse disse temaene:

- Beskrive sykepleiedokumentasjonens natur og sykepleiedokumentasjon i EPJ-systemer, og delvis hva som gjør det vanskelig å klassifisere automatisk.
- Beskrive rammeverk som finnes på norsk for slik dokumentasjon. Representere ett utvalgt rammeverk i datamaskinleselig format.
- Beskrive en utvalgt gjenfinningsteknikk som kan benyttes til å klassifisere sykepleiedokumentasjon, og vise resultater av metoden i en prototyp. God og etterprøvbar evaluering av resultatene er ikke utført.
- Henvise til bruksområder for resultatene; søk og navigering i dokumentasjonen, og lenking til eksterne ressurser

Sykepleiedokumentasjon

Sykepleiedokumentasjon føres på veldig forskjellig vis, og er av veldig forskjellig karakter. Vanlige tekstprosesseringsproblematikk (NLP) gjelder, men i tillegg kommer spesialiserte utrykk, forkortelser og innblanding av ulike typer data som hører hjemme i kurve, labrapport osv. Innføringen av maler og rammeverk i EPJ-systemer forbedrer strukturen til innlagt dokumentasjon, men det er et problem at man har en blanding av strukturert og ustrukturert tekst. Oppgaven argumenterer for at sykepleiedokumentasjon i dagens situasjon kommer i en særstilling som gjør at man bør velge teknikker som er universelle, noe som kan gå på bekostning av nøyaktighet. Dette trenger ikke være sant i fremtiden når rammeverkene har påvirket innlagt dokumentasjonen til en mer uniform stil.

Rammeverk

Organiseringen VIPS finnes i EPJ-systemer, men er ikke et rammeverk i forstand kodeverk med akser og unikt identifiserbare kategorier. NANDA, NIC og Sabaklass er oversatt til norsk pr. våren 2005. NOC står for tur. Ingen av disse finnes tilgjengelig i standardisert, maskinleselig format på norsk. Sabaklass er blitt representert i XML i forbindelse med denne oppgaven.

IR-teknikk

Vektorrommodellen (VSM) har blitt benyttet sammen med en autogenerert thesaurus basert på termer hentet fra rammeverket i en prototyp. Prototypen utfører så spørringer med thesaurustermene opp mot en samling sykepleiedokumenter, og likhet beregnes med et cosinus mål (avstand mellom spørringens og dokumentenes vektor). En terskel, et desimaltall mellom 0 og 1, brukes for å skille godtatte returnerte svar fra forkastete. Det er problemer knyttet til alle de nevnte forholdene; dokumentsamling, thesaurus og bruk av terskel. En skikkelig evaluering er ikke utført med tanke på precision/recall på grunn av disse problemene.

Bruksområder

Hovedbruksområder oppgaven ser for seg er gjenfinning av ustrukturert sykepleiedokumentasjon etter emner. Men navigering etter emner kunne vært gjort i større utstrekning enn den er i dag i EPJ-systemer om større deler av den var knyttet til emner. Oppgaven foreslår derfor at søketeknikken kan fungere som støtteredskap til å klassifisere riktig, og som grunnlag for å danne lokale emnekart, eller i større perspektiv knytte lenker til andre rammeverk eller eksterne ressurser. To lenketeknikker, TopicMaps og Xlink nevnes, men konsekvensene av å implementere dem i faktiske EPJ-systemer blir ikke debattert i oppgaven.

Hovedkonklusjon

De konkrete resultatene av prototypingen kan være diffuse. VSM i seg selv representerer en utprøvd og mye benyttet metode for søk i andre former for digitale bibliotek. Forskjellen er at man i denne oppgaven foreslår multiple søk basert på et rammeverk og ordlister knyttet til dette. Denne ordlisten er langt fra komplett, og en skikkelig vurdering av metodikken kan ikke foreligge før mer arbeid er gjort med denne. Metodens effektivitet på reell sykepleiedokumentasjon har vært vanskelig å vurdere, siden slik ikke har vært å oppdrive. Disse tingene må sees mer på før integrasjon i virkelige EPJ-miljøer tilrådes.

Nytteverdien kontra ulempene peker i favør for en slik søkemetodikk, den kan benyttes sammen med eksisterende systemer, og den representerer en fordel ved ukjent dokumentasjon.

9 Videre arbeid

I ettersyn er det spesielt tre områder direkte relatert til denne oppgaven som kan være interessante å jobbe videre med. Disse er prototyping, rammeverk og dokumentsamling. I tilknytning til disse er det mange aspekter å se mer på.

Prototyp

Å bygge alt fra grunnen når man ønsker å se på metoder for automatisk klassifisering, krever mye. Inverterte filer, tokeniserere, parsere og stemmingsmetoder er underordnet målet i en slik sammenheng, så utviklingen av en slag plattform eller bibliotek for formålet er ønskelig. Prototypen som er skrevet her trenger kraftig revidering for å kunne bli en slik plattform, så det er i seg selv en oppgave som kan være videre arbeid.

- Revidere og verifisere koden og forbedre effektivitet.
  - Forbedre internrepresentasjon og parameteroverføring
  - Fjerne unødige itereringer, optimalisere
  - Få skikk på implementasjonen av som bruker SAX parseren, evt. bytt parser.
- Forbedre søkefunksjonalitet
  - Forbedre håndteringen av svar, finne en smartere løsning enn bruk av terskel
  - Benytte thesaurus til utvidet/innskrenket søk, eller "relevance feedback".
  - Utnytte muligheten til å vekte enkelttermer bedre
  - Bruke søkefeltet til å lokalisere emner fra dokumentsamlingen. Sjekke om søketermene er emner, og vise dokumenter som har disse emnene knyttet til seg.
  - Foreta en analyse av precision/recall
- Andre søketeknikker
  - Lage flere klasser for søketeknikker for sammenligning.
  - Forsøk med AI - f.eks tilpasse WEKA's grensesnitt for forsøk med nevrale nettverk, eller lage filer WEKA kan benytte
- Skille rammeverk fra thesauri, slik at man kan tilpasse flere rammeverk og thesauruser til programmet.
- Forsøke å generere emnekart, f.eks vha TM4J (TopicMaps for Java)

Rammeverk

Rammeverkene er et eldorado for videre arbeid. Her omtales ikke arbeidet med å utvikle innholdet i rammeverkene.

Ingen av rammeverkene som er omtalt kan skaffes på norsk i dataleselig format [50]. Representasjoner som er gjort, tilhører bedriftene som har utviklet dem for sitt proprietære system. Å lage en standard for hvert enkelt av disse for norske forhold, og som er av en slik utførelse at de kan benyttes av leverandører av EPJ systemer er et stort arbeid. Dette ville være en utfordrende oppgave, og et slikt arbeid bør foregå i samarbeid med KITH og NSF.

Videreføring av representasjonen som er blitt laget i denne oppgaven av Sabaklass 2.0N er en overkommelig masteroppgave for videre arbeid. Med dette mener jeg f.eks å legge til muligheter for attributter, slik at en kan liste opp aktuelle modifikatorer (resultatkriterier eller handlingstyper)) for kategoriene/subkategoriene. I tillegg bør tager som ikke inneholder informasjon fjernes, fordi tomme tagger bare skaper overhead. Forbedring av XML representasjonen er et utfordrende og spennende arbeid som kanskje kan få internasjonale konsekvenser. Det finnes ingen felles XML (eller annen dataleselig representasjon) tilgjengelig. Sabaklass finnes i dag på følgende språk [68]:

Portugisisk	Norsk
Slovensk	Kinesisk (under utarbeidelse)
Tysk	Finsk (under utarbeidelse)
Koreansk	Nederlandsk (under utarbeidelse)
Spansk	

Sabaklass er bygget opp av empiri og statistikk. Bakgrunnsmaterialet i Sabaklass var av et slikt omfang at det var umulig for meg å sette meg detaljert inn i det. Men det er av stor interesse i denne sammenheng, fordi en kan tenke seg at dette bakgrunnsmaterialet ville være av stor verdi for å konstruere en bedre thesaurus enn jeg har gjort. Videre føring av prototypen til en mer avansert søketeknikk kunne bruke et slik materiale som grunnlag for andre metoder for tilnærming av problemstillingen. Metoden som ble brukt til å konstruere taksonomiene (delvis vha klyngeanalyse) kan kanskje reverseres med den forskjell av det nå finnes et høyereliggende rammeverkt å knytte begrepene i friteksten til.

Referansesamling

Et av de store problemene med denne oppgaven var å finne en relevant referansesamling å jobbe med. Dette er en problemstilling alle som skal forske på dette vil stå i om det ikke gjøres noe med det. En slik samling vil også være nyttig for de store leverandørene og deres

brukere - både i sammenheng med utvikling av funksjonalitet og i undervisningsøyemed. En referansesamling som er stabil og etterprøvbar ville gjøre det lettere å se hvilke resultater en oppnår.

- En slik samling vil trenge emner og kanskje klassifiserte elementer som kategorier knyttet til seg. Denne tilknytningen må skje manuelt, for å være sikker på at det blir riktig. I så fall vil det være en oppgave for fagpersoner (sykepleiere), men å utarbeide representasjon vil være informatikerarbeid.
- Anonymisering av sykepleiedokumentasjon. Det gjøres mye arbeid for å kunne automatisk anonymisere journalinnhold. Om dette arbeidet må tilpasses spesielt for sykepleiedokumentasjon, må vurderes.
- Tagging av enkeltelementer i teksten: nært relatert både til automatisk klassifisering og automatisk anonymisering. Ressurser som kan være aktuelle er "Oslo-Bergen Taggeren" eller "NorKompLeks"

9.1 Alternative bruksområder

Avslutningsvis kunne det vært morsomt å vise til andre bruksområder for metodikken. Søkemetoden burde fungere bra som gjenfinningsteknikk i andre situasjoner også:

- Benytte det automatiserte systemet som hjelp til å se om det er et problemområde en har oversett
- Tekst kommer deg i hende fra et annet system der du ikke har tilgang til klassifisering som allerede er gjort i det systemet
- Avdelinger som prioriter annerledes, og bruker en annen innfallsvinkel på dokumentasjonen
- På sykepleieskoler: Hjelp til å lete opp konsepter og lage problem/tiltak lister, og så knytte tiltakene til prosedyrebøker. Benytte søketeknikken på oppgavetekster for problemstillinger i problembasert læring

Bruk av metodikken innen disse feltene har ikke vært fokus for oppgaven.

10 Referanseliste

Bibliografi

[1] Bach, Grete: "Kravspesifikasjon for elektronisk dokumentasjon av sykepleie", KITH-Rapport 12-03, 30.november 2003, Revidert utgave, Oppdragsgiver: Helse Sør RHF, v/Sørlandet sykehus HF, Arendal, ISBN 82-7846-174-0 (http://www.kith.no/upload/1101/R12-03DokumentasjonSykepleie-rev1_1-NasjonalStandard.pdf)

[2] Baeza-Yates, R., Ribeiro-Neto, B.: "Modern Information Retrieval", ACM Press, Addison Wesley, 1999, ISBN: 0-201-39829-X

[3] Baud, R.H., Lovis, C., Rassinoux, A.M., Scherrer, J.R.:"Morpho-Semantic Parsing of Medical Expressions", Medical Informatics Division, University Hospital of Geneva, Switzerland, (Publisert i: American Medical Informatics Assosiation, http://www.amia.org/pubs/symposia/D004743.PDF)

[4] Choo, C.W., Detlor, B., Turnbull, D.: "The Structure and Dynamics of Organizational Knowledge", Chapter 2 of "Web Work: Information Seeking and Knowledge Work on the World Wide Web", Kluwer Academic Publishers, 2000, ISBN: 0792364600

[5] Dale, J.G., Angermo, L.M., Dale, C., Mjøsund, N.H., Storteig, M., Bach, G.: "KITH-rapport: Veileder for elektronisk dokumentasjon av sykepleie", Versjon 1.0 Mars 2003, KITH Rapport R 14/03, Oppdragsgiver: Helse Sør RHF, v/Sørlandet sykehus HF, Arendal, ISBN 82-7846-176-7 (http://www.kith.no/upload/1105/R14-03VeilederElektroniskDokSykepleie-v1.pdf)

[6] Dale, J.G., Dale, B. :"Fra fri tekst til faste former" - Tidsskriftet Sykepleien, nr. 21 2004

[7] Dochterman, J.M., Bulechek, G.M.: "Nursing Interventions Classification (NIC)", Mosby Inc., St. Louis, Missouri, 2000, ISBN: 0- 323-00894-1

[8] Eksperter i Team: "Bedre elektroniske pasientjournaler i primærhelsetjenesten - En undersøkelse av bruk, behov og barrierer", FAG SIF0101/03/B – EKSPERTER I TEAM, 25. APRIL 2001, NTNU, (HTTP://PASIENTJOURNALER.IDI.NTNU.NO/), (http://www.idi.ntnu.no/grupper/su/eit2002/eit2001-rapport.pdf)

[9] Gordon, M.: "Nursing Nomenclature and Classification System Development", Online Journal of Issues in Nursing. (Sept. 30, 1998), (http://www.nursingworld.org/ojin/tpc7/tpc7_1.htm)

[10] Grimsmo, A., Broseet, J.: "Kompetansemiljø for utvikling av elektronisk pasientjournal", Norges Forskningråd 2002, Oslo, juni 2002, ISBN 82-12-01725-7 (EPJforprosjektrapport2002.doc)

[11] Hellesø, R., Hjortnæs, A., Holm, R., Husby, E.H., Børmark, S.: "Godt samarbeid bærer frukter - utvikling av sykepleiedokumentasjon i elektronisk pasientjournal" – Tidsskriftet Sykepleien, 2003, nr. 4, (http://www.sykepleien.no/News/NewsShow.asp?NewsID=5094)

[12] Henry, S.B. Warren, J.J., Lange, L., Button, P.:"A Review of Major Nursing Vocabularies and the Extent to Which They Have the Caracteristics Required for Implementation in Computer-based Systems" – The Journal of the American Medical Informatics Assosiation, 1998, Juli-August; 5(4):321-328

[13] Hlava, Marjorie M.K.: "Automatic Indexing: A Matter of Degree", Access Innovation inc. October 2002, (http://www.accessinn.com/papers/?content=automatic_indexing&author=hlava&return_anchor=ai_articles)

[14] Ikehara, S., Murakami, J., Kimoto, Y., Araki, T.: "VECTOR SPACE MODEL BASED ON SEMANTIC ATTRIBUTES OF WORDS", Pacific Association for Computational Linguistics, (http://afnlp.org/pacling2001/pdf/ikehara2.pdf)
[15] Justeson, John S., Katz, S.M.: "Technical terminology: some linguistic properties and an algorithm for identification in text", (1995), Natural Language Engineering, 1(I):9-27.
[16] Karlsen, Rune: "Fin i kontakten - Vanlige svakheter ved sykepleiedokumentasjonen i psykiatriske sykehusavdelinger" – Tidsskiftet Sykepleien, 16 sept 2004 92. årgang (http://www.sykepleien.no/News/NewsShow.asp?NewsID=6362)
[17] Lovis, C., Baud, R., Rassinoux, A.M., Michel, P.A., Scherrer, J.R.: "Medical Dictionaries for Patient Encoding Systems: a Methodology", Artificial Intelligence in Medicine, 14:201-214, 1998
[18] Manning C. D., Shutze H. - "Foundations of Statistical Natural Language Processing", The MIT Press, 1999, ISBN: 0262133601
[19] NANDA: "NANDA - Sykepleiediagnoser: Definisjoner & Klassifikasjon 2001-2002", Akribe Forlag, 3003, ISBN: 2-7950-084-7
[20] Nonaka, Ikujiro, Hirotaka, Takeuchi: "Theory of Organizational Knowledge Creation", Chapter 3 of "The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation", Oxford University Press, 1995, ISBN: 0195092694
[21] Noy, N.F, McGuinnes, D.L.: "Ontology Development 101: A Guide to Creating First Ontology" -, Stanford University, Stanford, CA, 94305
[22] Pepper, Steve: "The TAO of Topicmaps", (http://www.ontopia.net/topicmaps/materials/tao.html)
[23] Røstad, Lillian: "Sikkerhet i elektroniske pasientjournalsystemer", Høstprosjekt SIF8094P1 – Systemutvikling, Institutt for datateknikk og informasjonsvitenskap, Norges Teknisk Naturvitenskapelige Universitet, 23. november 2001 (http://www.idi.ntnu.no/~nytroe/infosikk/h01_sikkerhet_epjsyst_Lillian.pdf)
[24] SfID: "Dokumentasjon av sykepleie i pasientjournalen - En veileder fra Sykepleiernes forum for IKT og Dokumentasjon", SfID, August 2002, Ver. 1.1
[25] van Bremmel, J. H., Musen, M. A.: "Handbook of Medical Informatics", Springer, Houten/Diegem, 1997, ISBN: 3-540-63351-0
[26] Viksjø, T.A: "DIPS: Hvordan støtter EPJ et helhetlig pasientforløp? ", Presentasjon Helsit2004, Trondheim, 22-23. september 2004
[27] Yearwood J, Wilkinson R.: "Retrieving cases for treatment advice in nursing using text representation and structured text retrieval", Artificial Intelligence in Medicine, Volume 9, Issue 1, January 1997, Pages 79-99, (http://www.ncbi.nlm.nih.gov/entrez/utils/lofref.fcgi?PrId=3048&uid=9021060&db=PubMed&url=http://linkinghub.elsevier.com/retrieve/pii/S0933365796003624)

Webressurser

[28] Annexstein, Fred S.: "Indexing and Representation: The Vector Space Model", (http://www.ece.uc.edu/~annexste/Courses/cs690/Indexing%20and%20Representation.ppt)
[29] Baud, Robert H et.al,:"The Power and Limits of a Rule- based Morpho-Semantic Parser", University Hospital of Geneva, Switzerland, PMID: 10566313, (http://www.amia.org/pubs/symposia/D005387.PDF)
[30] Ceglowski, Maciej: "Building a Vector Space Search Engine in Perl", February 19, 2003 (http://www.perl.com/lpt/a/2003/02/19/engine.html)

[31] CNCCE: Center for Nursing Classification & Clinical Effectiveness, (http://www.nursing.uiowa.edu/centers/cncce/)
[32] CNCCE-Newsletter: "The NIC/NOC Newsletter", Center for Nursing Classification & Clinical Effectiveness, (http://www.nursing.uiowa.edu/centers/cncce/nicnocnews.htm)
[33] CNCCE-NIC: "Nursing Interventions Classification (NIC)", Center for Nursing Classification & Clinical Effectiveness, (http://www.nursing.uiowa.edu/centers/cncce/nic/nicoverview.htm)
[34] CNCCE-NOC: "Nursing Outcomes Classification (NOC)", Center for Nursing Classification & Clinical Effectiveness, (http://www.nursing.uiowa.edu/centers/cncce/noc/)
[35] Digi.no: " Helse-Norge inn i fremtiden", (http://dataforeningen.no/ostlandet/arr/20021107.php)
[36] DIPS: "Sykepleiedokumentasjon i DIPS", (http://www.dips.no/dipsnew.nsf/00a580b13b71173dc12569f40028bdba/67d429f9485f601dc1256df6005d80c6/$FILE/SykepleieDok_Skjerm.pdf)
[37] DIPS: DIPS ASA, (http://www.dips.no),
[38] DIPS: Flø, K., Sørbye, L.W. : "Langt igjen til tverrfaglig pasientjournal…" (Kartleggingsrapport 2001, Diakonhjemmets sykehus), (http://www.dips.no/dipsnew.nsf/d8008ff72aa2b44dc1256b3e004c5b12/cc7e5ac53c014ef3c1256b5f00418d5c/$FILE/Kartleggingsrapport%20internett.doc)
[39] Dudeck, Joachim W., Schweiger, Ralf: "SNOMED CT Representation in an Intelligent Search Engine based on XML and Topic Maps (LuMriX)", Institute of Medical Informatics University of Giessen, Germany, (http://www.hl7.de/iamcda2004/finalmat/day2/LuMriX%20SNOMED%20CT%20Acapulco.pdf)
[40] Flø, Kåre: Personlig kommunikasjon, høsten 2004
[41] FMA: Digital Foundational Atonomist Model (aka Foundational Model of Anatomy or FMA), University of Washington, School of Medicine, (http://sig.biostr.washington.edu/projects/fm/index.html)
[42] Forbrukerrådet: "Forbrukerportalen", (http://www.forbrukerportalen.no)
[43] Fung, Kin Wah: "Access to SNOMED Through the National Library of Medicine's Unified Medical Language System", The UMLS Team, National Library of Medicine (Session 4 E- Kin Wah Fung.pdf)
[44] Gravdal, Tore: Personlig kommunikasjon, 2005
[45] Gulla, J.A.: "Forelesningsfoiler" – undervisning ved NTNU i faget "TDT4215-Dokumentforvaltning og tekstanalyse", 2003
[46] Holme, Arvid: "Forelesningsfoiler" – undervisning ved NTNU i faget "MNFIT281-Informasjonsgjenfinning", 2002
[47] infoRAD: Taira, Soderland, Jakobovits: "Automaitc Structuring of Radiology Free-Text Reports" (http://radiographics.rsnajnls.org/cgi/reprint/21/1/237)
[48] ISO: "International Organization for Standardization ", (http://www.iso.org)
[49] KITH: "EPJ - Elektronisk pasientjournal", (http://www.kith.no/epj/)
[50] KITH: "Kompetansesenter for IT i helse- og sosialsektoren AS", (http://www.kith.no/)
[51] LIMBER: "Language Independent Metadata Browsing of European Resources", Norsk Samfunnsvitenskapeilg Datatjeneste Brukermelding Nr. 1 2000, (http://www.nsd.uib.no/nsd/Bruker/00-1/brukermelding001.pdf)
[52] Linacre, E.: "Origin of the temperature scales", 11/'98, (http://www-das.uwyo.edu/~geerts/cwx/notes/chap03/celcius.html)

[53] Lingvistisk institutt, NTNU: ” MORFOLOGI - Studiet av ordets form og struktur”, (http://www.ling.hf.ntnu.no/fag/exfac/hfexfac010/intensiv2003/sprvit/morfologi_overhead.pdf)
[54] LovData: ”LOV 1999-07-02 nr 64: Lov om helsepersonell m.v. (helsepersonelloven)”, (http://www.lovdata.no/all/hl-19990702-064.html#map0)
[55] Lærum , Hallvar: ” Stanford “Short Course” in Medical Informatics - et intensivt resyme av et intensivt kurs”, (http://kvalis.ntnu.no/PublicDocs/20000620_Resyme%20av%20Stanford%20Medical%20Informatics%20Short%20Course.doc)
[56] Lærum, Hallvar: “Den aktuelle situasjon i Norge”, (http://kvalis.ntnu.no/PublicDocs/SUMIT2002/H_Laerum_Aktuelle-Situasjon-Norge.ppt)
[57] Marshall, C.C.: ”Toward an ecology of hypertext annotation”, Xerox Palo Alto Research Center (http://www.csdl.tamu.edu/~marshall/ht98-final.pdf)
[58] MNA: Michigan Nurses Assosiation” Nursing Practice: Overview of the Standardized Nursing Languages: NANDA, NIC & NOC”, (http://www.minurses.org/prac/CENandaNicNoc.shtml#nomen)
[59] NCCH: “Coding Matters, Newsletter of the National Centre for Classification in Health”, Volume 8, no.2, Sept 2001, (http://www3.fhs.usyd.edu.au/ncchwww/site/downloads/coding_matters/vol8no2.pdf)
[60] Nie, Jian-Yun: “Introduction to Information Retrieval”, University of Montreal, Canada, (http://www.eii.edu.au/seminar/NieIntroductionIR.ppt)
[61] NSEP: ”Norsk senter for elektronisk pasientjournal”, (http://www.ehr.ntnu.no/index.htm)
[62] NSF: ”Norsk Sykepleierforbund”, (http://www.sykepleierforbundet.no)
[63] Nørsett, Anne: ”Erfaringer fra Diakonhjemmet Sykehus”, (http://www.dips.no/dipsnew.nsf/0/1b89fadad3bf26b4c1256d480037f9e2/$FILE/Anne%20N%C3%B8rsett%20-%20Sykepleiedokumentasjon%20Diakonhjemmet%20sykehus.ppt)
[64] Ontopia-Engine: “Ontopia Topic Map Engine”, (http://www.ontopia.net/solutions/engine.html)
[65] Pepper, Steve: ”The TAO of Topicmaps”, (Presentasjon), (http://www.xmluk.org/slides/duxford_2003/2003-11-05-Steve-Pepper.pdf)
[66] Sabacare-Framework: “Framework/Structure “, (http://www.sabacare.com/framework.html)
[67] Sabacare-System: “Clinical Care Classification (CCC) System”, (http://www.sabacare.com)
[68] Sabacare-Versions: “Versions/Links”, (http://www.sabacare.com/versionsandlinks.html)
[69] Sabaklass: ”Clinical Care Classification (CCC)©” (http://www.deltadigital.no/Sabaklass/), (http://www.deltadigital.no/Sabaklass/Bakgrunn%20for%20Sabaklass%20System%202.0N.pdf)
[70] Siemens: ”Sykepleiedokumentasjon i DocuLive EPR”, juni 2004, (http://www.medical.siemens.com/siemens/no_NO//gg_hs_FBAs/files/brochures/Nyhetsbrev-sykepleiedokumentasjon.pdf)
[71] SNOMED: “Nursing Language Integration in SNOMED CT®”, (http://www.snomed.org/clinical/documents/NursingExamples.pdf)

[72] Snowball: ”Porterstemmer”, (http://snowball.tartarus.org/), (http://snowball.tartarus.org/algorithms/norwegian/stemmer.html)
[73] Soboff, Ian: “IR Evaluation, Lecture 9”, Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, (http://www.csee.umbc.edu/~ian/irF02/lectures/09Evaluation.pdf)
[74] Sosial- og Helsedepartementet: “Statlig strategi 2004-2007 - S@mspill 2007 - Elektronisk samarbeid i helse- og sosialsektoren”, (http://www.shdir.no/assets/9922/s@mspill%202007.pdf)
[75] St. Olavs Hospital: ” Nøkkeltall”, (http://www.stolav.no/stolav/Om+St+Olavs+Hospital+HF/sykehuset+i+tall/nokkeltall+.htm)
[76] UIO: Universitetet I Oslo: “Oslo-Bergen-taggeren (for bokmål og nynorsk)”, (http://www.hf.uio.no/tekstlab/)
[77] Vasudevan, Venu: “Notes on ontologies, (and their relevance to service trading in an internet service market)”, (http://www.objs.com/agility/tech-reports/9902-ontology.html)
[78] Verity: ”Verity.K2 Enterprise”, (http://www.verity.com/products/k2_enterprise/index.html)
[79] Witten, Ian H., Eibe, Frank: “Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations”, Morgan Kaufmann, October 1999, ISBN 1-55860-552-5, (http://www.cs.waikato.ac.nz/~ml/weka/book.html)
[80] Xlink: “XML Linking Language (XLink) Version 1.0, W3C Recommendation”, 27 June 2001, (http://www.w3.org/TR/xlink/)

11 Vedlegg

Vedlegg A DIPS notatoversikt

Vedlegg B Doculive (4.22) dokumenttyper

Vedlegg C VIPS-modellen (tabell)

Vedlegg D Norgesjournalen

Vedlegg E Feil i SabaKlass 2.0N

Vedlegg F NANDA

- NANDA Taksonomi II: Domener og klasser
- Eksempel på et NANDA - domenes oppbygning:

Vedlegg G Prototyp SKTax: UML

- package mainpack
- package documentops
- package fileops
- package porter
- class TokenizeWords
- class PorterStemmer
- class SabaklassTaxonomy
- class SykepleieDokument
- class MatchVector

Vedlegg H DTD og XML beskrivelse

- tiltak.dtd
- (utdrag) tiltak.xml
- (utdrag) diagnose.xml
- (utdrag) stoppord.xml

Vedlegg I Innhold på vedlagt CD-ROM

Vedlegg A DIPS notatoversikt

SYKEPLEIEJOURNAL	hentet fra [63]
SPL Sluttnotat	SPL Notat
SPL Plan for utskrivelse	SPL Innkomstnotats
SPL Sammenfatning (Epikrise- mappa)	SPL Innleggelsesnotat

Vedlegg B DocuLive (4.22) dokumenttyper

De ulike dokumenttypene som finnes i DocuLive (4.22) er for Sykepleier somatikk:

Spl. sammenf.

SplBrev til pas

SplBrev om pas

Meld. utskrivn.klar pas.

G2: Notater sykepleier

- Spl.Besøk
- Spl.Inn-Notat
- Spl.Oper.rapp
- Spl.Overfl.notat
- Spl.Pol.notat
- Spl.Sammend.
- Spl.notat
- spl.Tlf.notat
- Spl.Ut-notat

I tillegg finnes G3/G4: Pleieplan. Disse mappene er i DocuLive bare oppbevaring for pleieplanene, og er ikke ment som å brukes direkte.

Vedlegg C VIPS-modellen (tabell)

VIPS

Nøkkelbegrep Hovedsøkeord Søkeord

Velvære - Integritet - Profylakse – Sikkerhet

Sykepleieprosessen: Fase 1		**Fase 2**	**Fase 3**	**Fase 4**
Nødvendig bakgrunns-informasjon	Data relatert til søkeorda i pasientstatus gir grunnlag for å vurdere pasientens behov for sykepleie		Tiltak i forholdtil definerte problem. Struktureres etter de 10 søkeorda	Evalueringen skal synliggjøre effekten av utført tiltak
Pasient-opplysninger	**Pasientstatus**	**Pasientproblem Forventet resultat (mål)**	**Tiltak (intervensjon)**	**Evaluering (resultat)**
Generelle: Informasjonskilde	Kommunikasjon	Problemdefinering er et resultat av den vurdering som er gjordt av pasienten sin situasjon med utgangspunkt i pasientstatus. Forventa resultat (mål) kan være i forhold tilett eller flere undersøkeord Eller samla for hele pasientstatus.	Medvirkning	Vurdering av tiltak og pasientproblem Beskrivelse av oppnådde tiltak
Pårørende	Kunnskap / utvikling		Informasjon / undervisning	
Samtykke	Respirasjon / sirkulasjon		Støtte	
Pasientansvarlig sykepleier	Ernæring		Miljø	
Sykepleieanamnese: (nødvendig bakgrunnsinfo.)	Eliminasjon		Grunnleggende sykepleie	
Kontaktårsak	Hud / Vev / Sår		Trening	
Helse- og sykepleiehistorikk	Aktivitet		Observasjon / overvåkning	
Nåværende sykepleie	Søvn		Spesiell sykepleie Sårbehandling	
Allergi / overfølsomhet	Smerte sanseintrykk		Legemiddel-håndtering	
Sosial bakgrunn	Seksualitet / reproduksjon		Koordinering	
Livsstil	Psykososialt. Emosjonelt. Relasjoner.			
	Åndelig / kulturellt			
	Velvære			
	Sammensatt status			
Innkomstnotat Poliklinisk notat	Innkomstnotat Overflyttningsnotat Utnotat Sammendrag Poliklinisk notat Sammefattning - A5			Sammenfattning Poliklinisk notat Overflytnningsnotat

Tabellen er avskrift fra VIPS-modellen versjon 1.0 februar 2003. Den ble laget av Arbeidgruppa for innføring av spl.dok i EPJ v. Ålesund sjukehus, v. Marit Kjersem. Kilden som ble benyttet var Ehnfors, Hjortnæs, (Riksh.) Naversen, Ruth, (St. Olav), Bømark, (Ullevål) Gjengedal/Jakobsen

Vedlegg D Norgesjournalen

Dokument Grupper	Betegnelse	Undergrupper	Ordningsrekkefølge	Fargekoder
A	Sammenfatninger	A1: Personalia A2: Kontaktoversikt A3: Egne epikriser A4: Andres epikriser A5: Sykepleiesammenfatning A6: Pasientorientering	Omvendt kronologisk innen hver undergruppe	Grønn
B	Legejournal	B1: Løpende journal B2: Resultat av/svar på interne henvisninger	Kronologisk i hver undergruppe	Lys blå
C	Prøvesvar – vev og væsker	C1:Klinisk kjemi C2: Patologiske/anatomiske us. C3: Immunologi C4: Klinisk farmakologi C5: Mikrobiologi C6: Hematologi C7: Fertilitet og arv C8: Diverse	Omvendt kronologisk	Rød
D	Organ-funksjon	D1: Hjerte og kretsløp D2: lunge D3: Sansing og motorikk D4: Fordøyelsesapparat D5: Urinveier D6: Reproduksjon	Omvendt kronologisk	Lys brun
E	Bildediagnostikk	E1: Røntgenopptak o.l E2: Ultralyd E3: Scintigrafi E4: Fotografier	Omvendt kronologisk	Magenta
F	Observasjon og behandling	F1: Kurveark F2: Særskilte obs.skjemaer F3: Undersøkelsesplan	Omvendt kronologisk	Lys grønn
G	Sykepleier dokumentasjon	G1: Pasientopplysninger G2: Innkomstrapport m/sykepleienotater G3: Sykepleieplan	Omvendt kronologisk	Blå
H	Rapporter annet fagpersonell	H1: Fysioterapeut rapport H2: Ergoterapeut rapport H3: Sosionom rapport H4: Psykolog rapport H5: Ernæringsfysiolog rapport H6: Fødejournal	Omvendt kronologisk	Lys rød
I	Ekstern korrespondanse	I1: Innleggelsessøknader I2: Eksterne henvisninger I3: Annenhåndsvurderinger I4: Div. Brevkopier	Omvendt kronologisk	Sort
J	Attester/ meldinger/ erklæringer	J1: Offentlige blanketter i forbindelse med fødsler J2: Tilpliktede meldinger J3: Melding om uhell/ skalder/bivirkninger J4: Melding til frivillige registre J5: Pasientsamtykker/ erklæringer/krav J6: Trygdesaker	Omvendt kronologisk	Brun

Hentet fra [23].

Vedlegg E Feil i SabaKlass 2.0N

Arbeidet med å gjøre rammeverket maskinleselig avslørte noen feil i rammeverket.

Tabell 2

L Respiratorisk --> mangler definisjon

Tabell 3

T49.6 Urinretensjon --> T skal vekk (skal bare stå 49.6 Urinretensjon)

Tabell 6

Feil i formateringen av kodene. Eksempelvis skal [A02.2] skal være [A][02.2]. Her finnes flere elementer som må formateres:

A02.1 Gipsbehandling

A02.2 Immobilitetsbehandling

A05.1 Aktive og/eller passive bevegelser

A05.2 Rehabiliteringsøvelser

A61.1 Leiringsendring

B07.1 Stomiskylling

B62.1 Kvalmebehandling

K32.4 ser ut til å mangle.

K32.5 Spyttprøve - er bare nevnt i denne tabellen, ingen andre tabeller i det hele tatt (burde vært i tabell 6, 7 og 8)

Tabell 7

043.1 Aktiviteter i dagliglivet --> koden skal være en bokstav, altså: O43.1

Tabell 8

32.4 mangler

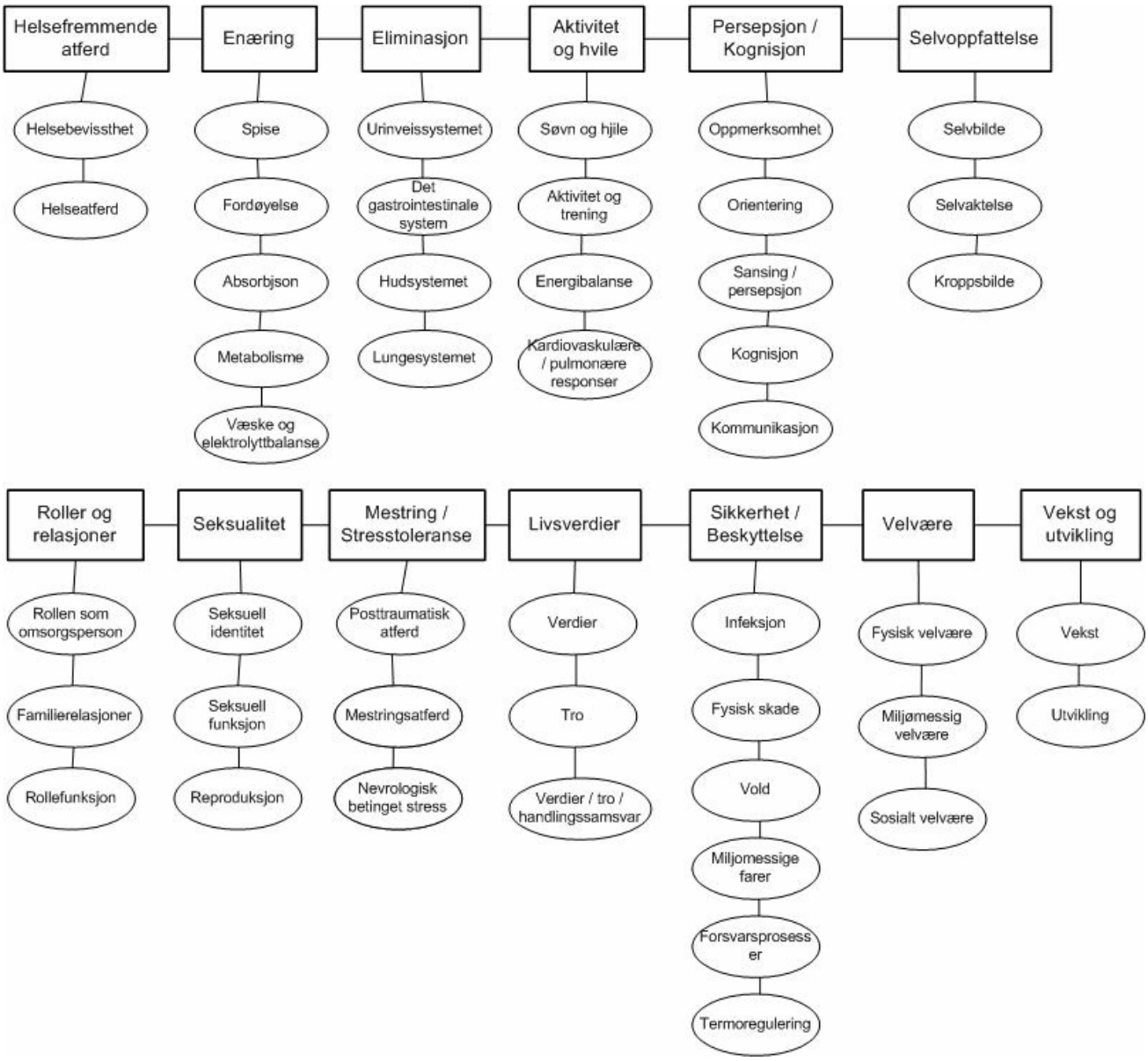
32.5 ingen definisjon.

Andre forhold:

Ulik benevning for behandlingskomponent K: i diagnoser benevnes K som "Fysisk Regulering", i tiltak benevnes K som "Fysiologisk Regulering"

Vedlegg F NANDA

NANDA Taksonomi II: Domener og klasser



Eksempel på et NANDA - domenes oppbygning:

Domene 6 Selvoppfattelse

Bevisst på seg selv.

Klasse 1 Selvbilde

Persepsonen / oppfattningen av hele selvet

Diagnostiske begrep	**Godkjente diagnoser**
Identitet	*00121* ***Identitesforstyrrelse***
	00125 ***Maktesløshet***
	*00152 Risiko for****maktesløshet***
	00124 ***Håpløshet***
Ensomhet	*00054 Risiko for* ***ensomhet***

Klasse 2 Selvaktesle

Vurdering av ens egen verdi, kapasitet, betydning og suksess.

Diagnostisk begrep	**Godkjente diagnoser**
Selvbilde	*00119 Kronisk lavt* ***selvbilde***
	*00120 Situasjonsbetinget lavt****selvbilde***
	*00153 Risiko for situasjonsbetinget lavt****selvbilde***

Klasse 3 Kroppsbilde

En mental forestilling om ens egen kropp.

Diagnostisk begrep	**Godkjente diagnoser**
Kroppsbilde	*00118 Forstyrret* ***kroppsbilde***

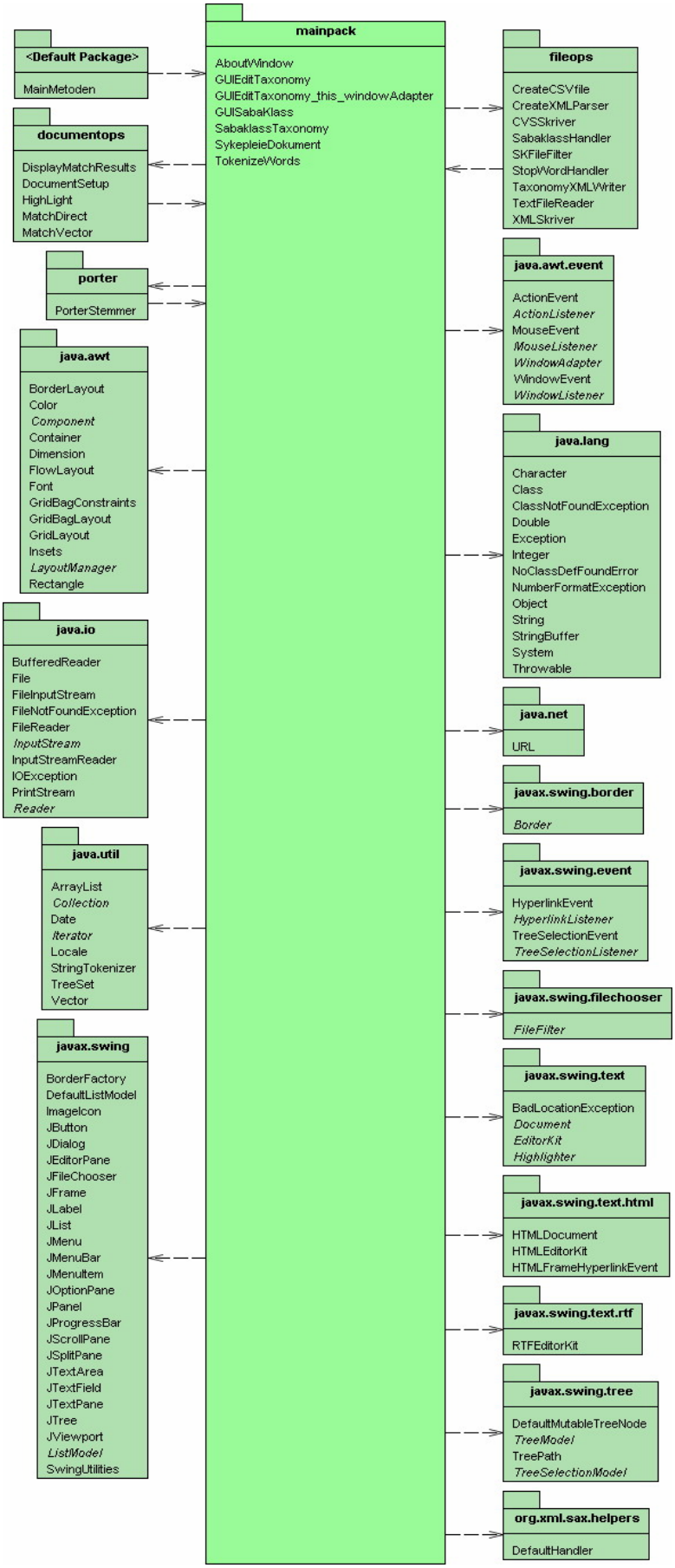
Vedlegg G Prototyp SKTax: UML

Oversikt over pakkene i prototypen "SKTax" og utvalgte viktige klasser

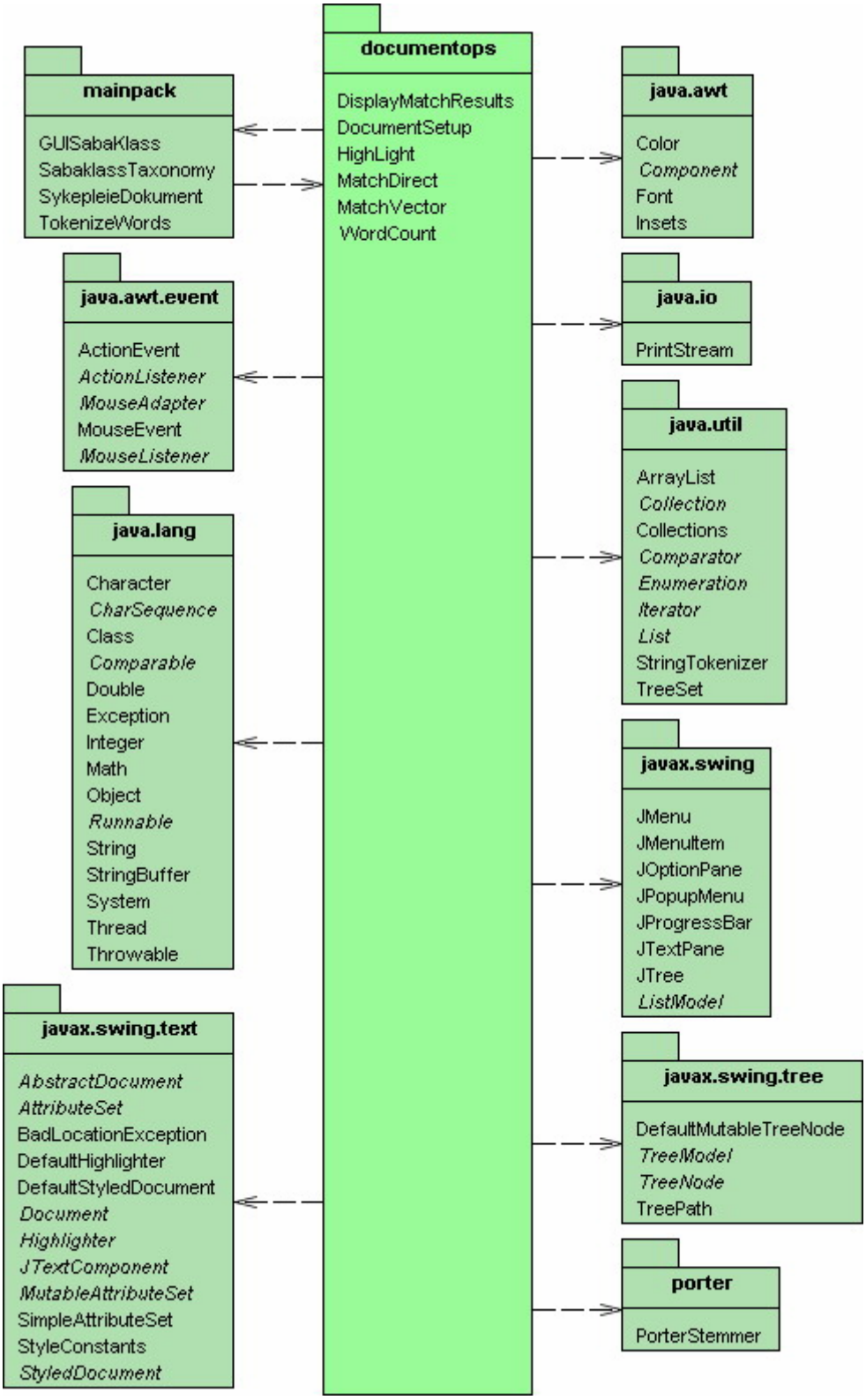
- mainpack
- fileops
- documentops
- porter

- SykepleieDokument
- SabaklassTaxonomy
- TokenizeWords
- MatchVector
- PorterStemmer

package mainpack

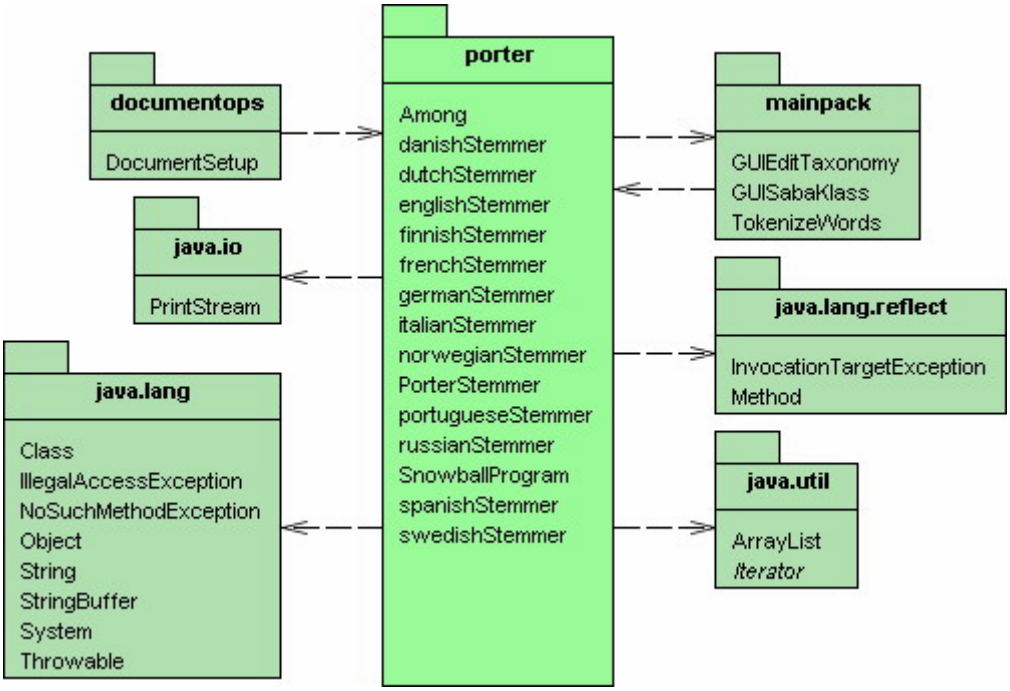


package documentops

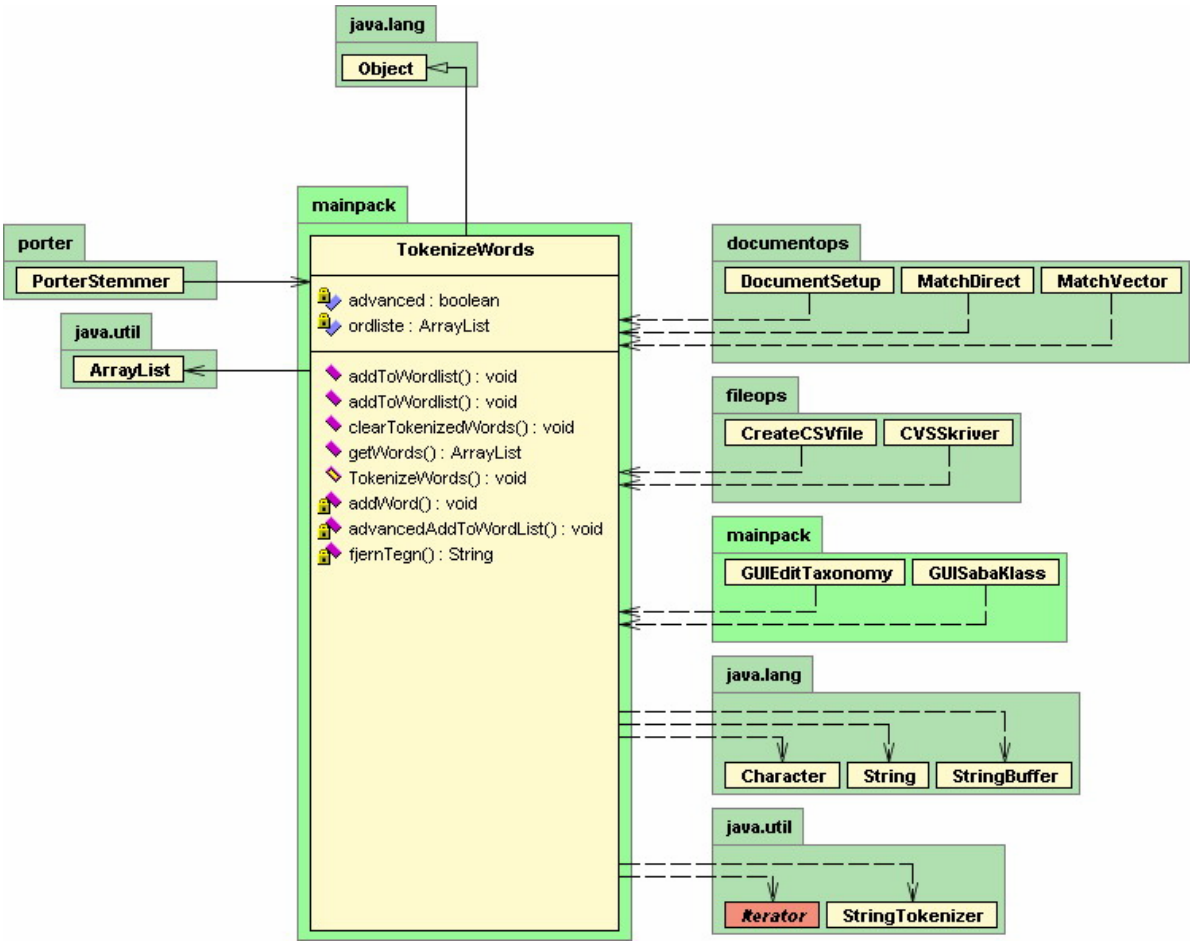


package fileops

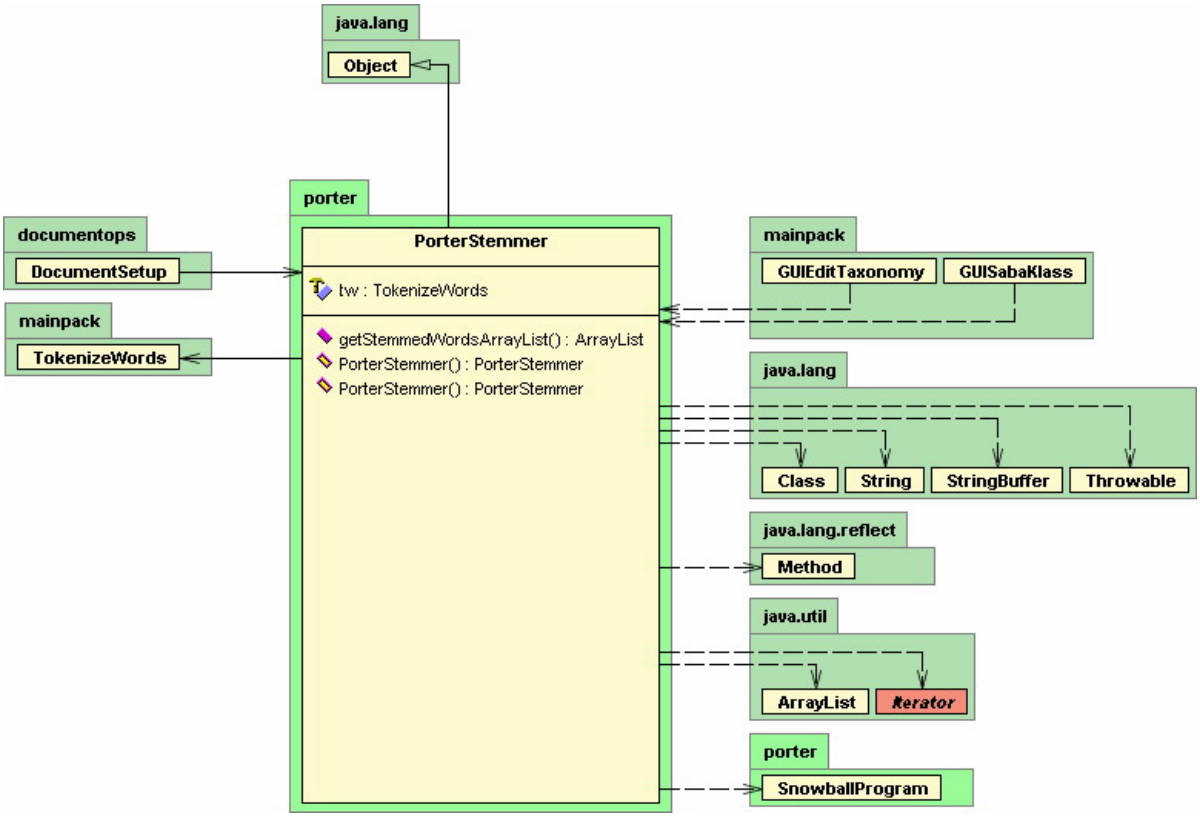
package porter



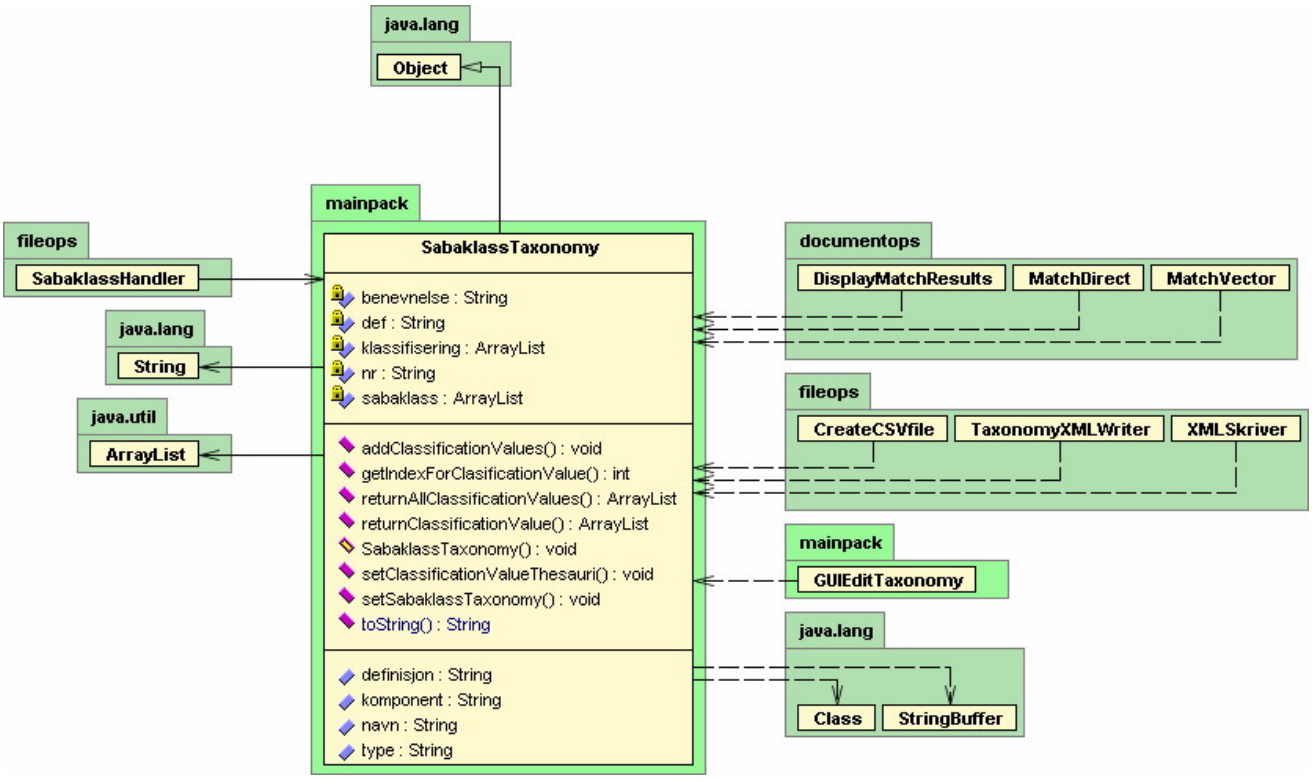
class TokenizeWords



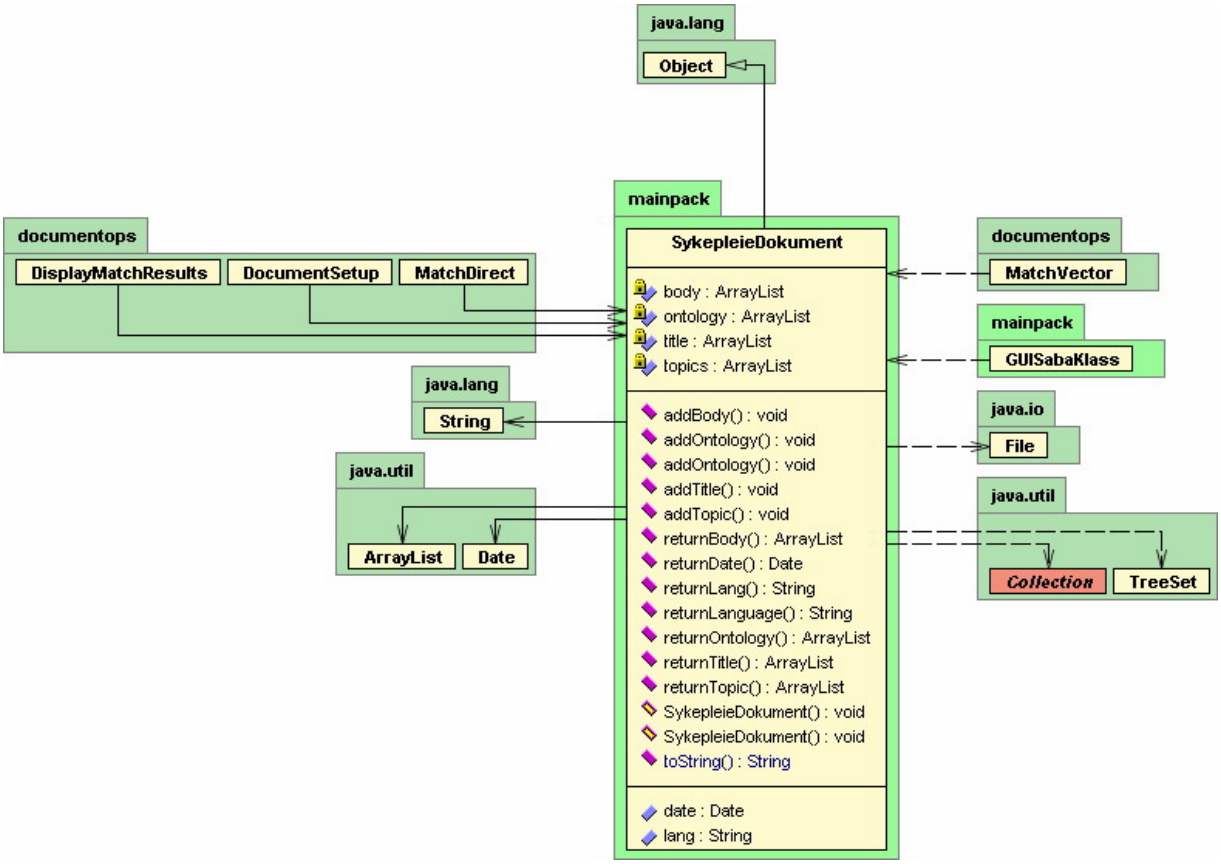
class PorterStemmer



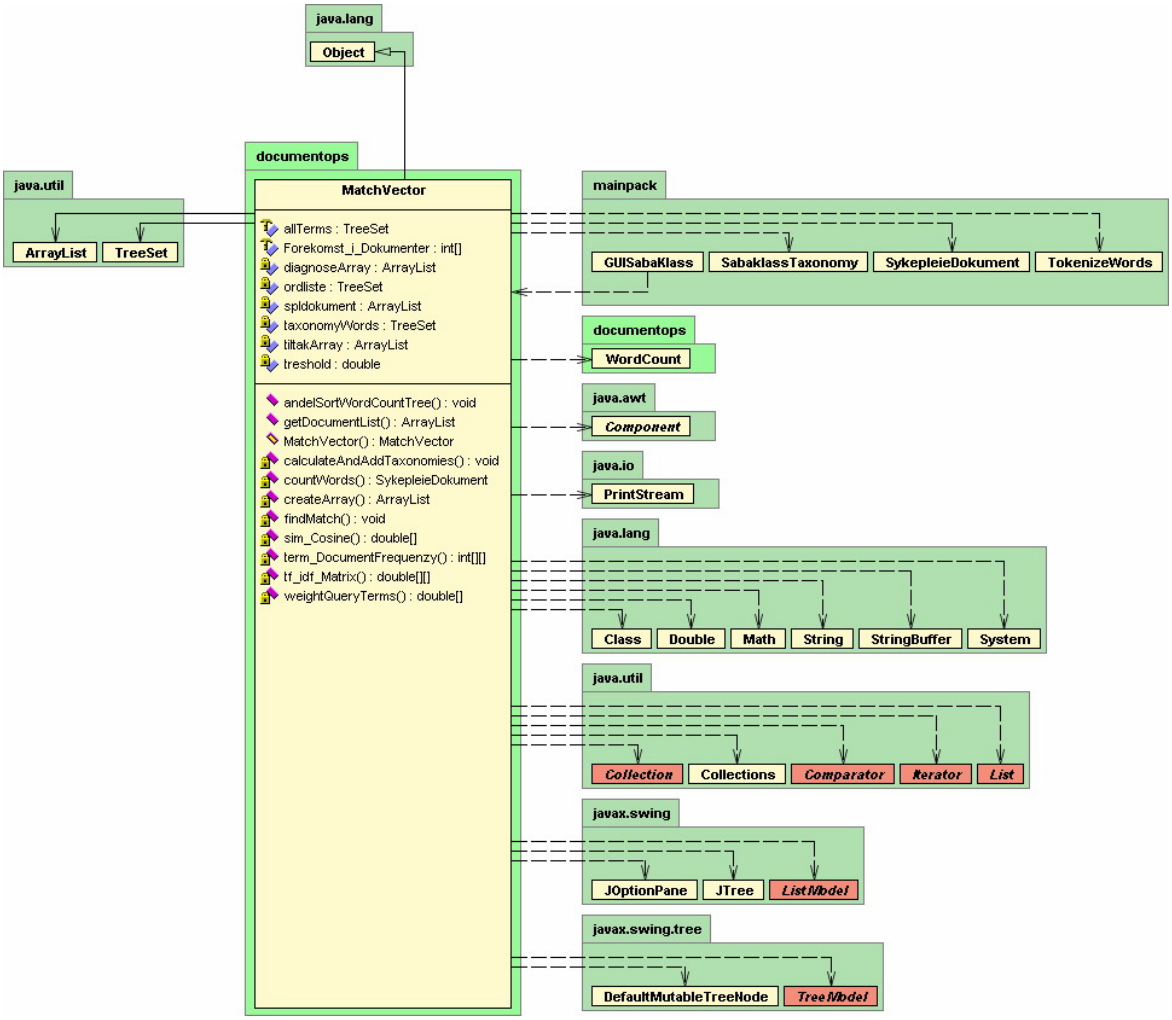
class SabaklassTaxonomy



class SykepleieDokument



class MatchVector



Vedlegg H DTD og XML beskrivelse

tiltak.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT benevnelse (#PCDATA)>
<!ELEMENT def (#PCDATA)>
<!ELEMENT definisjon (#PCDATA)>
<!ELEMENT klassifisering (tiltak+)>
<!ELEMENT komponent (#PCDATA)>
<!ELEMENT navn (#PCDATA)>
<!ELEMENT nr (#PCDATA)>
<!ELEMENT sabaklass (taksonomi+)>
<!ELEMENT taksonomi (komponent, navn, definisjon, klassifisering)>
<!ELEMENT thesauri (#PCDATA)>
<!ELEMENT tiltak (nr, benevnelse, def, thesauri)>
```

(utdrag) tiltak.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<sabaklass>
     <taksonomi>
          <komponent>C</komponent>
          <navn>HJERTE</navn>
          <definisjon>U/A</definisjon>
          <klassifisering>
               <tiltak>
                    <nr>08</nr>
                    <benevnelse>Hjertebehandling</benevnelse>
                    <def>Handlinger utført for å mestre endringer i hjertet
eller kretsløpet</def>
                    <thesauri>endring handling hjert hjertebehandling
kretsløp mestr utført </thesauri>
               </tiltak>
               <tiltak>
                    <nr>08.1</nr>
                    <benevnelse>Hjerterehabilitering</benevnelse>
                    <def>Handlinger utført for å bedre
hjertefunksjonen</def>
                    <thesauri>bedr handling hjert hjertefunksjon
hjerterehabilitering utført </thesauri>
               </tiltak>
               <tiltak>
                    <nr>09</nr>
                    <benevnelse>Pacemakerbehandling</benevnelse>
                    <def>Handlinger utført for å håndtere et elektronisk
apparat som sikrer normal hjterterytme</def>
                    <thesauri>apparat elektronisk handling hjert hjterterytm
håndter normal pacemakerbehandling sikr utført </thesauri>
               </tiltak>
          </klassifisering>
     </taksonomi>
</sabaklass>
```

(utdrag) diagnose.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<sabaklass>
     <taksonomi>
          <komponent>C</komponent>
          <navn>HJERTE</navn>
          <definisjon>Samling av elementer som involverer hjertet, blodkar
og sirkulasjonssystemet</definisjon>
          <klassifisering>
               <diagnose>
                    <nr>05</nr>
                    <benevnelse>Endring i hjertets minuttvolum</benevnelse>
                    <def>U/A</def>
                    <thesauri>blodk element endring hjert involver
minuttvolum samling sirkulasjonssystem </thesauri>
               </diagnose>
               <diagnose>
                    <nr>06</nr>
                    <benevnelse>Kardiovaskulær endring</benevnelse>
                    <def>U/A</def>
                    <thesauri>blodk element endring hjert involver
kardiovaskulær samling sirkulasjonssystem </thesauri>
               </diagnose>
               <diagnose>
                    <nr>06.1</nr>
                    <benevnelse>Blodtrykksendring</benevnelse>
                    <def>U/A</def>
                    <thesauri>blodk blodtrykksendring element hjert
involver samling sirkulasjonssystem </thesauri>
               </diagnose>
          </klassifisering>
     </taksonomi>
</sabaklass>
```

(utdrag) stoppord.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<!--Norske stoppord-->

<!--All the software given out on this Snowball site is covered by the BSD
License (see http://www.opensource.org/licenses/bsd-license.html ), with
Copyright (c) 2001, Dr Martin Porter, and (for the Java developments)
Copyright (c) 2002, Richard Boulton.-->

<!--http://snowball.tartarus.org/norwegian/stop.txt -->

<stopwords>

     <word>og</word>

     <word>i</word>

     <word>jeg</word>

     <word>det</word>

     <word>at</word>

     <word>et</word>

     <word>en</word>

</stopwords>
```

Vedlegg I Innhold på vedlagt CD-ROM

"ROM" i "CD-ROM" står for "Read Only Memory".

Dette betyr for deg som skal kjøre programmet:

- Du får ikke lagret filer til CD-ROM en når du redigerer rammeverket.
- Dersom du kopierer fra CD-ROM'en til harddisken i Windows, vil filene du lagrer ha attributtet "Read only".

Dersom dette forvolder problemer, kjør en installasjon "SKTax.exe" eller pakk ut "SKTax.zip" fra folderen "SKTax_install"

Frem til slutten av sommeren 2005 kan prototypen kan også lastes ned fra http://www.idi.ntnu.no/~asoes/master/

NB! Prototypen "SKTax" krever Java for å kunne kjøre. Miljøet Java insalleres ikke, din maskin MÅ ha dette installert separat. Java kan lastes ned fra: http://java.sun.com/j2se/1.4.2/download.html

Folderne på CD-ROM'en inneholder følgende:

Sabaklass_2.0N	Rammeverket slik det fremstår orginalt, finnes i PDF eller DOC format.
Sabaklass_konvertert	Flere stadier av utviklingen av XML representasjon av Sabaklass2.0N. "ver3" inneholder siste versjon.
SKTax_about	Hjelpemeny for SKTax prototypen. Finnes også inne i programmet.
SKTax	Kjørbar versjon av programmet. Husk at dette er en CD-ROM, og at den ikke kan lagres til.
SKTax_install	Installasjon til harddisk, hvis kjørbar versjon ikke kan brukes.
SKTax_kildekode	Koden til prototypen. Finnes som .java filer, PDF og DOC. Inneholder også JavaDoc.
SKTax_UML	Prototypen vis i "Unified Modelling Language". Bildefiler på formatetene .jpg og .png.