

www.sintef.no



**SINTEF Unimed**

Postadresse: 7465 Trondheim
Besøksadresse: Olav Kyrres gt. 3
Telefon: 73 59 25 90
Telefaks: 73 59 63 61

Foretaksregisteret: NO 948 007 029 MVA

SINTEF RAPPORT

TITTEL

BRUKERORIENTERT ELEKTRONISK PASIENTJOURNAL**Forbedring av lesbarhet basert på systematisering av fri tekst**

FORFATTER(E)

Hroar Piene

OPPDRAGSGIVER(E)

Norges Forskningsråd

RAPPORTNR. SFT78 A030101	GRADERING Åpen	OPPDRAGSGIVERS REF. Berit Nygaard, Medisin og Helse	
GRADER. DENNE SIDE Åpen	ISBN 82-14-03233-4	PROSJEKTNR. 780024	ANTALL SIDER OG BILAG 42 inkl vedlegg
ELEKTRONISK ARKIVKODE i:\7801 adm\780024	PROSJEKTLÉDER (NAVN, SIGN.) Hroar Piene, dr.med.	VERIFISERT AV (NAVN, SIGN.) <i>[Signature]</i>	
ARKIVKODE	DATO 2003-06-24	GODKJENT AV (NAVN, STILLING, SIGN.) Tonje Hamar, forskningsdirektør	<i>[Signature]</i>

SAMMENDRAG

Opplysninger i pasientjournaler vil ofte foreligge i fri tekst. Innholdet er gjerne relatert til såkalte "kontaktepisoder", ved hver episode føyes det til en rapport eller notat i pasientjournalen. Selv om opplysninger kan foreligge som måleverdier vil tolkninger foreligge som tekst. Slike tekster er ikke-standardiserte og er i utgangspunktet ikke egnet til noen form for logisk eller statistisk behandling e.a.

For å karakterisere innholdet i medisinske pasientjournaler er det utviklet et system for gjenfinning og karakterisering av enkeltbegreper i fri tekst. Systemet bygger på gjenfinning av nøkkelord knyttet til diagnoser, behandling/tiltak, lokasjoner, symptomer, sykdomsagens, legemidler og laboratorieundersøkelser. En rekke underliggende kataloger over enkeltord er utviklet for å utgjøre basis for tekstsøk. Dersom et nøkkelord gjenfinnes i teksten registreres hvilket nøkkelord samt dets posisjon. Dette tillater karakterisering av setninger i forhold til innholdet av nøkkelord og oppbygging av meningsfulle tekststrenger.

Som illustrasjon av muligheter er det laget ulike grensesnitt for å demonstrere konsekvenser av prinsippene. Fra et grensesnitt som viser automatisk uttrukne basale pasientdata kan en "klikke" seg fram til genererte korttekster eller hel tekst.

Evaluering viser at å penetrere en tekst via tekstutvalg fra forekommende nøkkelbegrep til generert korttekster kan gi tidsinnsparing på 30%. Oppfatningen av at en tekst er komplett i forhold til originalen krever at symptomer er inkludert, i tillegg til diagnosetermer, tiltak og lokasjoner samt medikamenter og lab-undersøkelser, i den presenterte nøkkelordlisten og i korttekstene som er basert på denne.

Denne studien har vist at det er mulig å bruke nøkkelordkataloger til å karakterisere innholdet av medisinske tekster. På basis av lister av identifiserte nøkkeltermer kan man bygge opp ulike måter å representere innholdet i teksten. I dette prosjektet er det i anvendelsessammenheng bare utprøvd noen muligheter.

STIKKORD	NORSK	ENGELSK
GRUPPE 1	Helse	Health
GRUPPE 2	Medisin	Medicine
EGENVALGTE	Informasjonsteknologi	Information technology

INNHOLD

1	INDELING – BAKGRUNN FOR ARBEIDET	3
2	METODE	
	2.1 Materiale	5
	2.2 Organisering av tekstfiler.....	5
	2.2.1 Tekstredigering og avsnittsidentifikasjon	5
	2.2.2 Identifikasjon av enkelt-ord og setninger	6
3	KARAKTERISERING AV TEKST	7
	3.1 Generelle medisinske begreper	7
	3.1.1 Etablering av ordsamling	7
	3.1.2 Karakterisering av begreper – manuell gjennomgang	8
	3.1.3 Søk i ordstammer	9
	3.1.4 Medikamenter	9
	3.1.5 Røntgen-(og andre bilde-)undersøkelser.....	9
	3.1.6 Klinisk kjemiske undersøkelser	10
	3.2 Søkestrategier.....	10
	3.2.1 Lemmatisering	10
	3.2.2 Søking i ordliste over medikamenter og laboratorieundersøkelse.....	11
	3.2.3 Sammenstilling av nøkkelord	12
4	FRAMSTILLINGER AV TEKSTEN.....	13
	4.1 Setningsklassifisering	13
	4.2 Fremstilling av nøkkelord	13
5	FREMSTILLINGER AV FRI TEKST	17
	5.1 Visning av nøkkelbegreper	17
	5.2 Kort tekst.....	17
	5.3 Hel tekst 17	
6	EVALUERING.....	18
	6.1 Tidsgevinst.....	18
	6.2 Evaluering av datarepresentativitet.....	19
7	BRUKERGRENSENITT.....	21
8	OPPSUMMERING OG KONKLUSJON	23

1 INNDELING – BAKGRUNN FOR ARBEIDET

En pasientjournal i papir- eller elektronisk versjon inneholder medisinske opplysninger som stort sett foreligger i fri tekst. Innholdet er gjerne relatert til såkalte ”kontaktepisoder”, dvs innleggelses, polikliniske besøk, eller dagbehandlinger/dagkirurgi, og ved hver kontaktepisode føyes det til en innleggelsesrapport eller et notat i den del av pasientjournalen som avsatt for legens opptegnelser. I tillegg blir dokumentasjon i pasientjournalen systematisert/plassert under ulike hovedbolker som ”klinisk kjemi”, ”patologisvar”, ”bildeundersøkelser”, ”kurveark” osv. Selv om en spesifikk undersøkelse er selvstendig dokumentert i pasientjournalen, er det vanlig at undersøkelsene refereres til i den løpende (lege-)journalen hvis de er betydningsfulle for å bedømme pasientens tilstand. På denne måten blir svært meget av det som dokumenteres omkring pasienten notert i fri tekst.

Helsetilsynet har utarbeidet forslag til hvordan pasientjournalen hensiktsmessig bør inndeles og organiseres (Statens helsetilsyn 1994). De elektroniske pasientjournalssystemene (EPJ: elektronisk pasientjournal) som brukes norske sykehus i dag følger samtlige den emneorienterte inndelingen som er anbefalt av Helsetilsynet. Selv om denne inndelingen primært er beregnet på papirbaserte pasientjournaler, er den oppfattet å være hensiktsmessig som emneorientert ordningsprinsipp også i EPJ-sammenheng (jfr KITHs forslag til norsk EPJ - standard, KITH 2000). Dette medfører at den tekstlige presentasjon av innleggelsesjournaler, løpende notater, poliklinikknotater osv., mer eller mindre foreligger som en løpende tekststreng. Å søke seg fram i de medisinske opplysningene i EPJ-er fra tidligere opphold ved å rulle teksten over skjermen (”scrolling”) kan derfor være svært problematisk pga det begrensede lese-vinduet som en vanlig arbeidsstasjon tilbyr.

Til dette kommer at innskannet tekst fra pasientkontakter før EPJ ble innført som eneste tillatte lagringsmedium, ikke kan leses på annen måte enn som tekstbilder, side for side. Dette gjør overgang til komplett papirløs pasientjournal, en reell EPJ, komplisert mht tilgjengeligheten til eldre data (informasjon om erfaringer ved Arendal HF).

For mange formål vil det være m.a.o. være gunstig om man kan trekke ut essenser fra EPJ, spesielt den tekstlige delen, og som kan lagres som en alternativ innfallsport til EPJ. Via en slik innfallsport bør en kunne få enkel tilgang til hovedproblemer og hovedtiltak knyttet til den enkelte kontaktepisode, få tilgang til essensen av det tekstlige journalinnholdet, gjerne koplet til grafiske fremstillinger av laboratedata, og videre tilgang til hele journalen der dette er relevant. En slik tilgang tilsvarer å kunne lese en ingress (eller et abstrakt) før en eventuelt leser en hel (vitenskapelig) artikkel.

Automatisk genererte sammendrag kan for eksempel også brukes for å sende ut pre-epikriser, d.e. sammendrag med de viktigste medisinske opplysningene til bruk for fastleger e.a. som skal følge opp en pasient etter et sykehusopphold.

Det bør m.a.o. ikke herske tvil om nytteverdien av å kunne ekstrahere det medisinske innholdet i form av korttekster o.l., spørsmålet er hvordan det kan gjennomføres.

I visse typer fritekst, som vitenskapelige artikler, må forfatteren selv oppgi en rekke relevante nøkkelord som senere kan brukes til katalogisering og søking. Dette tilsvarer i pasientjournalen de innleggelses- eller utskrivingsdiagnoser som noteres og som skal uttrykke selve essensen ved pasientens tilstand. Men disse diagnosene alene kan være misvisende m.h.t. hovedproblemet ved

en aktuell kontakt. Som eksempel kan diabetes (sukkersyke) gi en rekke følgetilstander/-komplikasjoner fra karsystemet (årebetennelser/koldbrann, synsproblemer/blindhet, infarkter osv.) som ikke nødvendigvis kommer frem dersom man kun bruker "diabetes" som nøkkelord for den medisinske problematikken. Å legge om dagens praksis i retning av at den som har ansvar for journalen også skal oppgi en rekke nøkkelord i tillegg til diagnosen som uttrykk for journalinnholdet, oppfattes som urealistisk. Et alternativ er at dette gjøres automatisk av et dataprogram som løper samtidig med det aktuelle tekstbehandlingsprogrammet og som undersøker journalteksten mht forekomst av signifikante ord og begreper som, i tillegg til hoveddiagnosen, beskriver pasientens medisinske problem.

Behandling av naturlig språk er en gren innen datalingvistikken som i økende grad blir betydningsfull (Bradley & Rockwell, 1992). Datalingvistikk beskjeftiger seg bl.a. med uttrekk av essenser fra ulike tekster. Tradisjonelt gjøres dette ved å telle ordforekomst i en tekstmengde; det er de moderne raske datamaskinene som har gjort slike teknikker mulig. Antakelsen er at ord som forekommer hyppigere i en viss tekst enn den statistiske forekomsten i andre, normale, tekster, uttrykker et spesifikt innhold meningsinnhold for den aktuelle teksten. Typisk for medisinske notater i EPJ er imidlertid at de er korte og derfor lite egnet til en slik statistisk vurdering av tekstinnholdet. Selv trivielle ord kan få "betydning". En annen sak er at denne metoden bare kan fortelle at et ord forekommer hyppigere enn normalt, men sier ingen ting om ordet som sådant.

En alternativ metode er å sjekke om den aktuelle tekstens ord eller begreper samsvarer med ord eller begreper i kataloger over ord og begreper med tilhørende definisjoner. Problemet med denne metoden er først å fremst å etablere slike kataloger, som må være mest mulig komplette, og dernest å definere eller klassifisere begrepene i katalogen. Store prosjekter i engelsktalende land finnes på dette området (UMLS, GALEN, SNOMED).

I det NFR-prosjekt som beskrives her er et dataprogram samt et sett nøkkelordkataloger basert på å gjenfinning av ord, dvs programmet baserer seg på den alternative metoden som kort er skissert ovenfor. Grunnelementene i programmet kan tilpasses ulike typer brukergrensesnitt.

2 METODE

2.1 Materiale

Materialet består av i alt ca 60 tekster fra originale pasientjournaler der alle opplysninger som kan tjene til å identifisere enkeltpasienter er fjernet. Tekstene er stilt til rådighet fra to sykehus: St. Olavs Hospital og Kristiansand SH. Fra St. Olavs Hospital omfatter materialet ulike avdelinger (hud, revmaotologi, infeksjonsmedisin, nevrologi, øye) mens fra Kristiansand SH er det indremedisinsk og kirurgisk avdeling som er representert. Blanding av avdelinger har en viss betydning i den forstand at de formelle journaloppsettene varierer, noe som øker utfordringen mht å finne algoritmer som behandler tekstene uten hensyn til oppsettene. Alle tekstene er behandlet i sin opprinnelige form uten retting av de feil som evt måtte finnes i den originale teksten. Dette medfører at feiltolkninger og feilstavinger i tekstene kunne få konsekvenser for den senere tekstbehandlingen.

2.2 Organisering av tekstfiler

2.2.1 Tekstredigering og avsnittsidentifikasjon

Filen med den originale journalteksten ble lest inn i programmet. Tomme linjer, som bare fyller skjermen ble fjernet, og delte ord ved linjeslutt sammenføyd. Faste avsnittsmarkeringer (se nedenfor) som ikke var avsluttet med punktum eller kolon, ble tilføyd et kolon. Setninger som ikke var avsluttet med punktum ved slutten av linjen ble tilføyd denne sluttmarkeringen.

Faste tekstavsnittsmarkeringer i teksten som:

- *Innleggende lege,*
- *Fast lege,*
- *Journal ved,*
- *Innleggelsesdiagnose, Problemstilling, Årsak til innleggelsen*
- *Familie/Sosialt,*
- *Tidligere sykdommer,*
- *Aktuelt,*
- *Medikamenter,*
- *Allergier,*
- *Stimulantia,*
- *Naturlige funksjoner,*
- *Status presens,*
- *Vurdering/Resyme*

og varianter av slike vanlig brukte tekstavsnitt ble identifisert, registrert og markert. Disse tekstavsnittene var i liten grad standardisert, selv om en så tendenser til at ulike sykehus eller avdelinger brukte visse lokale "normaloppsett". Ved en realisering av systemet kan det være aktuelt å bruke standard avsnittsmarkeringene som er lagt inn som faste avsnitt i journal, epikrise, m.m., hvis dette gjøres.

2.2.2 Identifikasjon av enkelt-ord og setninger

Ord i teksten ble identifisert med ordskilletegnet ("blank") eller "/" og startposisjonen for hvert ord ble registrert.

Setninger, som ble definert som minste selvstendig meningsbærende enhet, ble identifisert.

Hensikten var å bryte opp teksten i mindre enheter som hver i og for seg kan leses i sammenheng.

Hovedalternativene ble definert ved:

1. Ny overskrift
2. Punktum/kolon+ordskilletegn+stor forbokstav
3. Komma+ordskilletegn+liten forbokstav
4. Ordene "men", "slik at" og "selv om" i teksten, selv om komma var utelatt..

Avvik fra hovedregel:

1. Ved enkeltord som oppfyllte betingelsene både før og etter ordet brytes det ikke etter ordet. Eksempel: "Allergier: Husstøv. Timotei." registreres som én setning.
2. Ikke brudd etter pga., U.a., tabl. mb. Eksempel: mb. Menière i samme setning (mb. er kort for morbus=sykdom)
3. Ikke brudd ved oppramsinger atskilt med komma. Ved oppramsinger beholdes hele ramsen som én setning.

Setningenes startposisjon identifiseres og registreres. Det avsnittet setningen står i identifiseres dersom teksten var delt inn i avsnitt. (I korte journalnotater vil den ofte(st) ikke være det.)

3 KARAKTERISERING AV TEKST

Etter tilretteleggingen av teksten, med identifikasjon av avsnitt, setninger og enkeltord, gjennomføres den for å identifisere nøkkelbegreper knyttet til medisin generelt og til legemidler, kliniske-kjemiske laboratorieundersøkelser samt bildeundersøkelser. Objektene som det søkes etter kan endres uten at grunnstrukturen i programmet forandres.

Nedenfor gjennomgås de ordlistene som det søkes i og som er spesielt utviklet for dette prosjektet, søkeprinsippene som anvendes samt noe om den primære behandling av søkeresultatet.

3.1 Generelle medisinske begreper

3.1.1 Etablering av ordsamling

En katalog over generelle medisinske begreper ble dannet ved å søke alle norske ikke-allmenne ord i tilgjengelige medisinske databaser:

- ICD-10 katalogen
- ICD-9 katalogen
- ICPC katalogen for allmennt medisin
- NCSP katalogen for kirurgiske prosedyrer
- NIS kodeverket for operative inngrep (som gjaldt før NCSP ble standard)
- DRG katalogen

En enkel tekstbehandler ble utviklet. Denne søkte gjennom tekstene i de gitte tekstfiler (i enkelt tekstformat) og identifiserte og talte opp alle enkeltord. Enkeltord ble definert som tekststrenger atskilt med skilletegnene punktum, komma, kolon, semikolon, blank, kontinuasjonstegn eller skråstrek. Alle slike skilletegn ble fjernet fra ordene. Hvert enkeltord ble sjekket mot allerede eksisterende ord i den nøkkelordfilen som var underetablering og forekomsten talt opp. Hvis ordet ikke eksisterte fra tidligere ble det lagt inn på alfabetisk korrekt plass. Hvis ordet allerede eksisterte ble tallet for forekomst av ordet inkrementert.

Etter tillegg av alle de nevnte katalogene/kodeverkene ble hele ordlisten gjennomgått manuelt for å:

- 1) Ta ut ord med høy forekomst og som ikke spesifikt kunne knyttes til det medisinske fagområdet. Ord med høy forekomst var hovedsakelig allmenne norske ord. En del ord som kan knyttes til medisinske begreper, men ikke selv er spesifikt medisinske, av typen adverb, adjektiver og preposisjoner, som bidrar til å kvalifisere de medisinske begrepene, ble beholdt.
- 2) Fjerne artefakter og forkortelser av typen "o.a.", "etc", "m.fl." og lignende
- 3) Fjerne ord med endelser som anga bestemt form (-en, -et), flertall (-er) eller genitiv (-s). Disse ordene ble fjernet dersom ordets grunnstamme fantes i listen; alternativt ble bare endelsene fjernet.
- 4) *Ad hoc* tillegg ble gjennomført på basis av tekst i aktuelle journaler

Etter denne prosessen stod man igjen med ca 16 000 ord.

3.1.2 Karakterisering av begreper – manuell gjennomgang

Ordlisten på ca 16 000 ord ble gjennomgått manuelt for å karakterisere hvert enkelt av dem. Hvert ord fikk et suffiks som anga en medisinsk begrepsklasse. I klassifiseringen ble følgende inndeling brukt:

- D: begreper som angir en sykdomstilstand ("disease") eller diagnose.
- T: begreper som angir et diagnostisk eller terapeutisk tiltak.
- S: begreper som angir et symptom
- L: begreper som angir en lokalisasjon
- R: begreper som har mening i relasjon til begrepene ovenfor, for eksempel adverb, adjektiver og visse preposisjoner
- A: begreper som angir et sykdomsagens
- 0: begreper som i hovedsak har administrativ status

Klassifiseringen bød på en rekke problemer og det måtte ofte foretas avveininger med tilhørende valg av løsninger. Noen klassifikasjoner var enkle, f.eks ble alle sammensatte ord som endte på en definert sykdom som -kreft, -cancer, -anemi, -blødning, ord med -itis eller -itt endelser (uttrykker betennelsestilstand) o.a. gitt D-klassifikasjon.

Mer uspesifikke begreper som "kvalme" eller "kløe" ble generelt gitt S-klassifikasjon. Skillet mellom symptom (S-klassifikasjon) og sykdom (D-klassifikasjon) var imidlertid ikke alltid enkelt. Det ble valgt å bruke D-klassifikasjon dersom det aktuelle ordet beskriver en medisinsk tilstand som peker mot en viss sykdom, mens mer generelle symptomer som kan være felles for en rekke sykdommer ble klassifisert som S. I sammensatte begreper vil mange symptomer gå over til å bli D-begrep, et eksempel er symptomet blødning (S) som kan opptre sykdommen i mageblødning (D).

R-klassifikasjon ble satt på ord (adjektiver, adverb, preposisjoner) som kan ha relasjon til D-, T-, L- eller S-klassifiserte begreper. Dette ble imidlertid ikke gjennomført konsekvent for adjektivenes del. En rekke adjektiver kan i medisinsk språkbruk være uttrykk for en sykdom når de står alene og de ble derfor klassifisert som D-begreper. Et eksempel er adjektivet *allergisk* som uttrykker det samme som en som *har allergi* i medisinsk språkbruk.

Skillet mellom klassifikasjon av adjektiver som D- eller R-begreper gikk i forhold til svaret på spørsmålet: "Kan pasienten være <adjektivet>? Hvis svaret var Ja ble klassifikasjon D, ellers ble klassifikasjon R. Eksempel:

1. Kan pasienten være *allergisk*? (Ja. *allergisk* D-klassifiseres).
2. Kan pasienten kan være *psoriatisk*? (Nei. *psoriatisk* R-klassifiseres)

Pkt 2 fungerer dersom andre ord i teksten peker mot sykdommen (*psoriasis*), hvis ikke vil slike opplysninger falle ut ved tekstbehandlingen. Dette ble mindre problematiske med den sammenkoplingen av nøkkelord som ble innført. I normale fall vil ordet "psoriatisk" i teksten bli fulgt av et kvalifiserende ledd som f.eks *hudlesjon*. Iflg sammenkoplingsreglene (se nedenfor) vil en ende opp med *psoriatisk hudlesjon* som et nytt felles D-klassifisert nøkkelbegrep til erstatning for det to (R-ord + D-ord => D-begrep (to ord)). Det hadde ikke hatt noen betydning om psoriatisk hadde blitt klassifisert som D-begrep, sluttresultatet ville blitt det samme (D-ord + D-ord => D-begrep).

Begrepene klassifisert som "0" fikk ingen praktisk senere betydning i den foreliggende programversjonen.

I tillegg til klassifikasjonen beskrevet over ble hvert ord gitt en kode som fortalte hvorvidt det var et komplett ord eller et avkortet ord. Avkortede ord ble brukt der det av erfaring viste seg at det kunne foreligge feilstavinger i teksten. I eksemplet nedenfor er ordet ”tykktarmsdivertikkel” forkortet til ”tykktarmsdivertik” fordi divertikkel kan feilstaves som ”divertikel” og derfor unnslipe et søk i en tekst etter begreper som overensstemmer med listen.

Et eksempel fra listen, med alle ord som har ordstammen ”tykk” kan illustrere klassifikasjonen:

Tabell: Eksempler på klassifisering av nøkkelbegreper

Begrep:	Kompletthet:	Klasse:
tykk	Fullstendig	Symptom (S)
tykktarm	Fullstendig	Lokalitet (L)
tykktarmcancer	Fullstendig	“Disease”/diagnose (D)
tykktarmkreft	Fullstendig	“Disease”/diagnose (D)
tykktarmkrøs	Fullstendig	Lokalitet (L)
tykktarmscancer	Fullstendig	“Disease”/diagnose (D)
tykktarmsdivertik	Avkortet	“Disease”/diagnose (D)
tykktarmskreft	Fullstendig	“Disease”/diagnose (D)
tykktarmskrøs	Fullstendig	Lokalitet (L)

Legge merke til at begge ordstammene *tykktarm-* og *tykktarms-* forekommer i listen. Slike alternative skrivemåter, som her stammer fra de originale underliggende kodeverkene, er beholdt for å gjøre relevansen høyest mulig overfor vanlige skrivemåter i vanlig tekst og derved muliggjøre mest mulig komplette søk.

3.1.3 Søk i ordstammer

En forenklet søkeprosedyre ble implementert for å unngå å gå glipp av begreper som av noen grunn ikke fantes i listen over. Prosedyren besto i å undersøke om ordets endelse var:

- -kreft
- -betennelse
- -fissur
- -fistel
- o.a., i alt 75 ulike alternativer.

I så fall ble ordet registrert i hht den klassifikasjonen som endelsen representerte.

3.1.4 Medikamenter

Alle medikamentnavn og generiske preparatnavn, til sammen 1850, ble samlet i en alfabetiske liste som grunnlag for søk etter medikamenter i teksten.

3.1.5 Røntgen-(og andre bilde-)undersøkelser

Med utgangspunkt i prosedyrekodeverket for radiologi (NORAKO 2000) ble det etablert en liste over undersøkelsesprosedyrer innen radiologi på til sammen 38 modaliteter.

3.1.6 Klinisk kjemiske undersøkelser

Med utgangspunkt i kodeverk for klinisk-kjemisk lab (NKKL-10) ble det laget en liste over 120 ulike laboratorieundersøkelser. Listen ble *ad hoc* komplettert med vanlige kjente klinisk-fysiologiske undersøkelser.

3.2 Søkestrategier

Etter identifikasjon av ordene i den medisinske tekst som analyseres (se kap.), ble hvert tekstord input til to alternative søkealgoritmer som sammenholdt tekstordet med nøkkelordlistene:

1. Søk i liste over ordstammer.

Vilkår for aksept:

- Søkeord > Ordstamme, og
- Ordstamme = søkeordets siste del.
- Begrepsklasse søkeord = Begrepsklasse for ordstamme

2. Søk i liste over 16 000 begreper

Vilkår for aksept:

- Søkeord > Listeord (unntak av endelse (se nedenfor om lemmatisering)
- Begrepsklasse søkeord = Begrepsklasse for listeord

For den generelle listen (ca 16 000 ord) var søket lagt opp som "binært søk". Denne algoritmen er kort beskrevet:

- Start: Undersøk om tekstordet ligger foran eller etter det midterste ordet i listen (alfabetisk rangering)
- Hvis etter: Flytt nedre grense for listen til midten av listen
- Hvis foran: Flytt øvre grense til midten av listen
- Iterer: Gjenta fra start med nytt "midtre ord"

Denne prosedyren halverer den aktuelle oppslagslisten for hver iterasjon. Det betyr at 16 000 ord undersøkes med maksimalt 14 oppslag ($2 \exp 14 = 16\,384$). Å søke igjennom en tekst på ca 100 ord for å finne eventuelle nøkkelord tar i størrelsesorden sekund(er). Prosedyren krever at den opprinnelige ordlisten transformeres til random filtype for å tillate søk avhengig av posisjon i listen; i motsetning til sekvensielle søk som alltid vil starte med ord nr1 på listen.

Dersom et ord ikke kan finnes, men det finnes en beste tilpasning som er et avkortet begrep (se over) beholdes det opprinnelige tekstordet som det søkte nøkkelordet. I eksemplet over vil *divertikkel*, *divertikler*, *-divertikkeler* og *-divertikkler* alle komme ut i *sin opprinnelige skriveform* med klassifikasjon D.

Output fra prosedyren er det funne nøkkelordet samt klassifikasjonen.

3.2.1 Lemmatisering

Bøyningsendelser medfører problemer ved søking pga alfabetrekkefølgen. Det vil ofte kunne skje at en bøyningsform av en ordstamme kommer flere plasser nedenfor selve ordstammen og med sammensatte ord på plassene inn i mellom. I eksempelet vist i tabellen ovenfor vil det eventuelle ordet "tykkertarmen" komme etter "tykkertarmcancer", mens det logisk sett burde stå etter "tykkertarm". Dersom det ikke er eksakt overensstemmelse mellom et søkt ord og ordlisten vil ikke søket lykkes og alternative strategier, som for eksempel kan gå ut på å finne samsvar med unntak av de siste 1-2 karakterer, vil også kunne mislykkes. Lemmatisering formaliserer en alternativ tilleggsprosedyre ved å definere hvilke typer endelser en kan fjerne uten å endre meningsinnholdet i søkeordet.

Dersom et søkeord hadde en endelsesom uttrykte flertall, genitiv eller bestemt form ble endelsen fjernet ("lemmatisering") fra ordet før søket ble gjentatt. Den funne ordstammen sammen med klassifikasjonen ble output fra algoritmen. Ord som "eksemet" ville m.a.o. bli avkortet til *eksem* og identifisert og klassifisert som "eksem /D" når det forekom i teksten. (En opsjon kunne også være å la det opprinnelige ordet bli stående som det ettersøkte nøkkelordet med søkeordets aktuelle endelse og den funne klassen.)

Lemmatisering vil bare ha effekt virker dersom det faktisk finnes et ord i ordlisten som tilsvare ordstammen etter at endelsen er fjernet. I medisinske tekster forekommer endelser som f.eks "-ene" ikke bare som flertall bestemt form, men også selvstendig som i ordet "migrene". Man kan derfor, som dette eksemplet viser, ikke konsekvent sløyfe alle "-ene"-endelser i listen, man må også vurdere om ordstammen faktisk har mening..

Ord som ender på *-et* og *-t* kan alle uttrykke bestemt form ("bekken-*et*", "endometrie-*t*") eller fortidsform ("bekymre-*t*"). Pga usikkerheten om det ville være korrekt å avkorte med *-et* eller *-t* ble i slike tilfelle listen ikke endret, slik at det i listen finnes en rekke ord som uttrykker bestemt form ved endelsen *-et*. (Eksempel: Ord som ender på "...område*t*" ble beholdt selv om ordstammen er "...område")

Ved den senere presentasjonen av nøkkelordene (se kap) gjør lemmatiseringen det enklere å unngå å gjenta ord og begreper som allerede er registrert, for eksempel unngå visning av både "eksem", "eksemet" osv. hvis begge ord forekommer i teksten.

3.2.2 Søking i ordliste over medikamenter og laboratorieundersøkelse

For disse ordlistene, som er betydelig kortere, brukes sekvensiell søking. I stedet for å søke på enkeltord i teksten søkes det etter overensstemmelse mellom et ord i listen og setninger i teksten. Dvs, når et ord fra listen overensstemmer med teksten i setningen markeres det hvor ordet forekommer samtidig som setningen som sådan klassifiseres (se nedenfor).

For søk etter medikamentopplysninger i teksten ble det laget en særlig strategi:

1. For hver setning lokaliseres et evt medikament i hht listen over spesialpreparater og generika (sekvensielle søk). Hvis det lokaliseres et medikament utføres:
 2. setningen analyseres for forekomst av tall-karakterer (ASCII 48 - 57) og posisjonen av første tall finnes.
 3. Posisjonen av første tall sammenlignes med posisjonen av (første karakter i) legemiddelnavnet. Hvis
 - a. tallposisjonen kommer først i setningen vises sekvensen fra tallposisjonen til og med legemiddelnavnet; hvis
 - b. legemiddelnavnet kommer først vises sekvensen fra legemiddelnavnet og ut setningen.

På denne måten ville medikamentopplysninger av formen 10 mg Morfin, og Morfin 10 mg x 3, begge bli registrert og vist korrekt.

For laboratorieundersøkelser (inkl. rtg og andre bildeundersøkelser) ble det laget en noe varierende strategi. Her ble setninger med lab-relaterte ord først identifisert, deretter ble setningene undersøkt mht tallforekomst. Hvis tall forekom ble setningen fra om med typen lab-undersøkelse til setningsavslutning vist. Setninger som: "og det målt blodtrykk på 130/90

mmHg." vil bli gjengitt som "Blodtrykk på 130/90 mmHg." Hvis det ikke forekom tall ble de nøkkelordene som setningen inneholdt oppsummert etter typen lab-undersøkelse.¹

3.2.3 Sammenstilling av nøkkelord

Ordene i nøkkelordlisten forekommer ett og ett. I mange medisinske begreper er det imidlertid to eller flere ord som inngår. Alle disse ordene kan være av sykdomsbeskrivende art, men like gjerne er et ord sykdoms- (eller symptom-)beskrivende og andre ord av mer kvalifiserende art.

Kvalifiseringen kan eksempelvis være relatert til lokalisasjon eller alvorlighetsgrad. Det er viktig at slike distinksjoner kommer med i analysen. Dette ble løst på følgende måte:

To påfølgende ord i teksten, som begge var nøkkelord, ble koplet sammen til ett begrep med én felles klasse dersom klassifikasjonsrekkefølgen var

Tabell: Regler for kombinasjon av nøkkelbegreper

Regel:	Ord N-1:	Klasse ord N:	Fellesterm	Resultat felles klasse:
1	Klasse lik ord nr N	Klasse lik ord nr N-1	Ord (n-1)+ord(n)	Lik N = N-1
2	Klasse D, S eller T	L eller R		D, S eller T (Lik N-1)
3	S	L	Ord (n-1)+ord(n)	D
4	L eller R	D, S, T		D, S eller T (Lik N)

I klartekst var prinsippet at to sammenhengende ord, som begge var nøkkelord, ble slått sammen til ett begrep med klasse lik den felles klassen (hvis det forekom), eller til D-, S- eller T- klasse begrep dersom ett av ordene hadde klasse L eller R.

Legg imidlertid merke til regel 3: Et S-begrep går over til D-klassifisering dersom det er koplet til en lokalisasjon. Eksempel: "pruritus" (S) "vulva" (L) = "pruritus vulva" (D)

Andre eksempler på slike koplinger er (jf teksteksempel side):

1. "Commotio" /D+ "cerebri" /L = "Commotio cerebri" /D (regel 2 i tabellen)
2. "Venstre" /R + "parieto" /L+"occipitalregion" /L = "Venstre parieto occipitalregion" /L (først regel 4=> venstre parieto /L, deretter + occipitalregion /L med regel 1)
3. "fullstendig" /R + "amnesi" /S => "fullstendig amnesi" /S (regel 4)

Nøkkelbegrepene (enkeltord eller sammensatt av flere ord) slik de nå fremkom, ble registrert mht innhold og posisjon i teksten.

¹ Det kan lages logiske varianter på slike strategier og der det tas hensyn til vanlige begreper som brukes for måleenhet eller for dosering, som "mmHg", "tablett" osv for å bygge opp nye uttrykk som alltid følger formatet "navn + antall + måleenhet (+ hyppighet)". Som regel vil det imidlertid være tilstrekkelig å vise prøvenavnet/medikamentnavnet + resten av setningen, men dette forutsetter at det allerede er satt et setningsskilletegn mellom hver labprøve/legemiddel.

4 FRAMSTILLINGER AV TEKSTEN

4.1 Setningsklassifisering

De ulike setningene i en medisinsk tekst kan være mer eller mindre meningsbærende. I prinsippet er de det alle, men i noe ulik grad og meningsinnholdet vil være ulikt i de forskjellige tekstavsnittene.

For en overordnet bedømmelse av pasientproblematikken er det oftest hoveddiagnosen og noen hovedsymptomer sammen med terapivalg som er viktig. En mer detaljert vurdering kan baseres på en mindre omfattende utsiling av tekstelementene, og en fullstendig vurdering av alle elementer kan kreve journalteksten komplett.

Dette krever en form for klassifisering av setningene mht tekstelementene som inngår. Følgende klassifisering ble brukt:

- Setninger som inneholdt ett eller flere ord som pekte på en diagnose eller medisinsk problem (D-klasse nøkkelbegrep) eller et tiltak (T-klasse nøkkelbegrep), ble klassifisert som *REL (relevant)*.
- Setninger som inneholdt informasjon om symptomer, ble klassifisert med *SYM*.
- Setninger som inneholdt informasjon om lokalisasjon, ble klassifisert med *LOC*.
- Setninger som inneholdt informasjon om et eller flere legemiddel i hht medikamentlisten, ble klassifisert som *MED (medication)*.
- Setninger som refererte seg til ulike undersøkelser ble klassifisert som *LAB*.
- Setninger som inneholdt spørsmål eller antakelser, ble klassifisert som *QUE (question)*. Et slikt spørsmål/antakelse ble definert å være knyttet til setninger som inneholdt ordene "mulig", "kanskje", "mistanke", "usikker", "prøve", "sjekke om", "vel dreie seg om", eller "spørsmål".

En enkelt setning kunne være klassifisert med ingen, en, eller flere av de ulike klassifikasjonene.

4.2 Fremstilling av nøkkelord

Nøkkelordene ble sortert i forhold til klasse og klassene samlet setning for setning. Hvilke setninger som skulle inngå ble, dersom det gjaldt en inkomstjournal, definert til å være de i avsnittene for diagnose, aktuelt, observasjoner (og tilsvarende) samt resymé (og tilsvarende). For journalnotater og lignende kortere og mer ustrukturerte tekster ble hele teksten omfattet av analysene.

Reglene for nøkkelordsammenstillingen var:

REL- (og SYM) klassifiserte setninger

Alle D-type nøkkelbegrep, som forekom i *REL*-klassifiserte setninger, ble samlet i en separat tekststreng. (Observer at hvert nøkkelbegrep kunne bestå av flere ord som følge av utført nøkkelordkombinasjon.) Nye tekststrenger ble lagt til den endelige strengen dersom nøkkelordene som inngikk ikke allerede forekom i tekststrengen. I denne vurderingen ble det til en viss grad tatt hensyn til at samme ord kan forekomme i ulike former, f. eks. det ble det ikke skilt mellom *sinusitis* og *sinusitt* som har samme ordstamme. Hvis for eksempel *sinusitis* forekom i D-begrepslisten ble senere bruk av *sinusitt* i teksten ikke inkludert. Derimot ble det ikke gjort forskjell på ulikartede ord som *cancer* og *kreft*. Hvis begge begreper ble brukt ville også begge bli lagt til i tekststrengen.

Hvis den aktuelle setningen som ble behandlet var av *QUE*-type ble tekststrengen avsluttet med et ?-tegn.

Man kunne velge hvorvidt samlingen skulle omfatte enkeltsetninger, avsnitt eller hele teksten samlet.

Alle T- (og S-) type nøkkelord ble samlet på prinsipielt tilsvarende måte.

Eksempel:
Følgende inntømsjournal:

26.10.1997 INNLEGGELSE NEVROKIRURGISK AVD NEVROKIRURGISK S ENGEPOST B3

Innl.diag.: Commotio cerebri.

Journal ved: bbb bbb/ bbb.

Familie/

Sosialt: Pasienten er samboende, for tiden avtjener han militærtjeneste.

Tidligere sykdommer: Opr. testisretensjon som barn, stort sett frisk.

Aktuelt: Pasienten har i kveld vært ute på byen og i beruset tilstand falt ned en trapp, visstnok etter å ha blitt dyttet. Husker intet av det aktuelle og har hatt litt hodepine etterpå. Ikke vært kvalm eller kastet opp. Innlegges til observasjon.

Allergier: Ingen kjente.

Naturlige funksjoner: U.a.

Stimulantia: {fritekst}.

Medikamenter: Ingen faste.

Status: presens: 26.10.1997 kl. 0300. Pasienten er en ## år gammel mann i vanlig hold og god almentilstand. Virker lettere beruset, men forøvrig adekvat og gjør greit rede for seg. Koopererer bra ved undersøkelsen. BT: 110/70. Puls: 80 regelmessig. Caput: Det sees en større subcutan hevelse i venstre parieto occipitalregion. Collum: Ingen ømhet, ingen stenose over halskar. Cor, pulm, abdomen: Normale forhold. Ekstr.: Ingen ytre tegn til skade. Otoskopi: Bleke trommehinner bilat. Ved orienterende nevrologisk undersøkelse finnes runde og egale pupiller med normal lysreaksjon. Skarpe papillegrensener bilat. ved oftalmoskopi. Normale og sidelike kraft- og tempoprestasjoner i alle ekstr. Normale og sidelike reflekser.

Resymè: 26.10.1997 v/ bbb bbb. 20 år gammel mann som innlegges etter at han i kveld har falt ned en trapp i beruset tilstand, visstnok blitt dyttet ned. Fullstendig amnesi for det aktuelle, men ved undersøkelse i mottagelsen er han våken og adekvat. Det er tatt

(hvis dette leses i elektronisk versjon vil en kunne observere hvordan avsnitt og deler av teksten framheves med farger)

Tekstanalysen dette resultatet mhp utvalgte nøkkelord ga som resultat:

”PROBLEMER:”

commotio cerebri; fissur venstre parietalregion?

TILTAK:”

observasjon

MED-klassifiserte setninger

For inntømsjournaler ble det satt som begrensning at MED-setninger skulle finnes i Medikament, Aktuelt, Resymè eller Vurderings-avsnittene, for notater ble det ikke satt noen slike begrensninger. Setningen ble organisert slik at legemiddelordet ble første ord, og deretter fulgte resten av setningen slik den var registrert. Det ble ikke gjort noe forsøk på å strukturere setningene der legemiddelnavn forekom ut over dette, f. eks. å redigerer setningene slik at de konsekvent ble av formen: ”Legemiddel” *n* mg, *m* x daglig, og liknende, selv om dette er fullt mulig. Fremgangsmåten medførte at forskrivninger som var beskrevet i klartekst, for eksempel: ”*Marevandosen ble justert*” eller: ”*Lasix ble seponert*”, fremkom i klartekst.

Laboratorierelaterte setninger

De utplukkede opplysningene gjaldt i denne sammenheng klinisk kjemi/fysiologi samt bildeundersøkelser. Hele teksten ble gjennomgått mot ordene i listene over hhv klinisk-fysiologiske/kjemiske undersøkelser og bildeundersøkelser. Resten av setningen gjengitt der slike begreper ble identifisert.

Mht bildeundersøkelser ble først setningene undersøkt mht forekomst av relevante begreper som var relatert til bildeundersøkelser, deretter ble de etterfølgende L- og D-kodete nøkkelordene i den aktuelle setningen gjengitt i den rekkefølge de forkom i teksten, og evt med sluttmarkering for spørrende setning.

Resultatet av denne prosessen ble i teksteksemplet over:

LAB:

OBS: BT: 110/70. Puls: 80 regelmessig. RTG/UL: rtg caput fissur venstre parietalregion?
--

Teksten forteller at "BT:" og "Puls:", som gjenkjennes i nøkkelordlisten for klinisk kjemi/fysiologi, var hhv "110/70" og "80 regelmessig", som var restinnholdet av setningene som starter med hhv "BT:" og "Puls:".

5 FREMSTILLINGER AV FRI TEKST

I programmet fremstilles sluttresultatet av analysene den fri tekst på tre ulike måter:

5.1 Visning av nøkkelbegreper

De utvalgte nøkkelbegrepene vises, etter sortering, i separate ruter som vist i tekstrutene i rapporten. Innholdsmessig er dette selvforklarende

5.2 Kort tekst

Det genereres en korttekst der alle de registrerte tekstavsnittene er med og de setningene innen hver avsnitt som er *REL*-klassifisert blir vist som hele setninger uansett opprinnelse. Innledende ord som ”og”, ”som” og så videre, som ikke har noen betydning i teksten, sløyfes. Et eksempel på en generert korttekst vises i ruten nedenfor (jf journalteksten på side):

Tekstramme: Innleggesjournal

26.10.1997 INNLEGGELSE NEVROKIRURGISK AVD NEVROKIRURGISK SENGEPOST B3.

Diagnose: Comotio cerebri.

Tidligere sykdommer: Opr. testisretensjon som bam,

Aktuelt: Pasienten har i kveld vært ute på byen og i beruset tilstand falt ned en trapp, Innlegges til observasjon.

Status presens: Virker lettere beruset, Det sees en større subcutan hevelse i venstre parieto occipitalregion.

Resymè: 20 år gammel mann som innlegges etter at han i kveld har falt ned en trapp i beruset

tilstand, Fullstendig amnesi for det aktuelle, Men ved undersøkelse i mottagelsen er han våken og adekvat. Det er tatt rtg. caput som kan gi mistanke om en fissur i venstre parietalregion, Pasienten innlegges til observasjon..

være nensiktsmessig.

5.3 Hel tekst

Den hele journalteksten må også kunne vises, enten i sin opprinnelige form (trivielt) eller med uthevede deler av teksten. Her kan en velge flere varianter: Utheve bare visse typer nøkkelbegreper, bare patologiske funn, både patologiske og normale funn, eller setningene som tilsvarer kortteksten.

6 EVALUERING

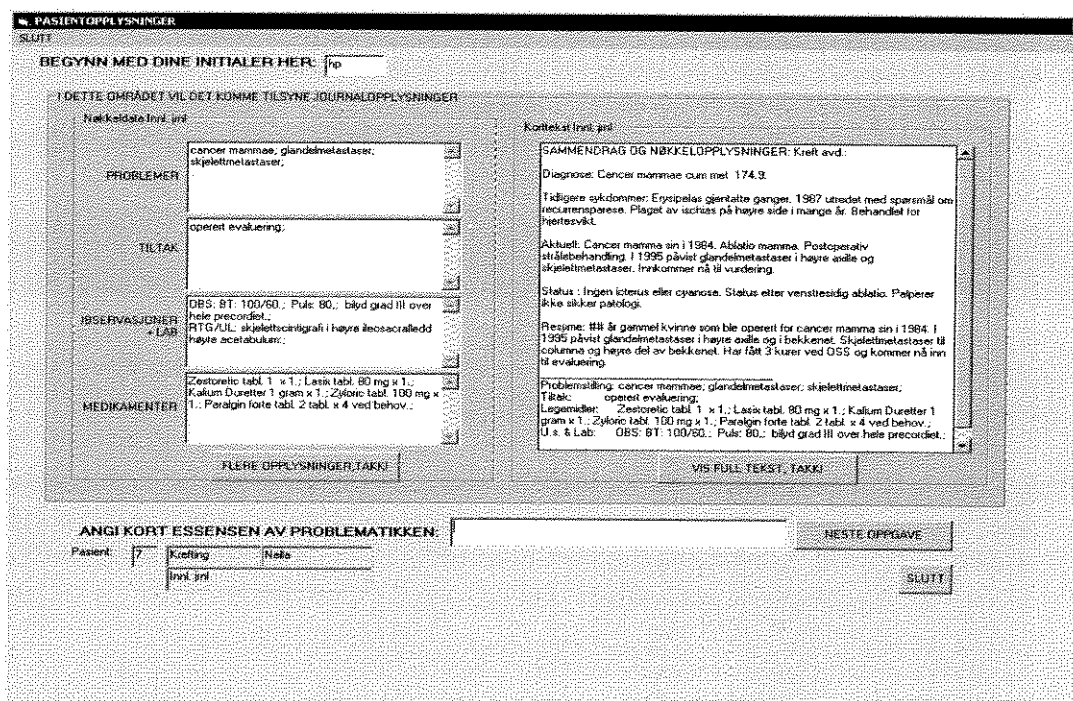
Evaluering en hadde to hensikter:

1. Vurdere hvorvidt lesing av nøkkelopplysninger, evt supplert med sammendrag evt supplert med hel tekst ga noen tidsgevinst i forhold til å lese hele journalen fra først av
2. Vurdere om sammendraget ga et helhetlig eller mangelfullt bilde av journalinnholdet

6.1 Tidsgevinst

For denne evalueringen ble det utviklet et spesielt grensesnitt (se figur). Evaluatør ble satt overfor oppgaven å vurdere hodeproblematikken ved å skrive en kort setning om essensen i journalen. Han kunne gjøre dette enten på basis av nøkkelopplysningene alene (venstre felt i figuren) med sammendraget skjult, eller han kunne åpne sammendraget og få økt tilgang på opplysninger. Dersom dette ikke var tilfredsstillende kunne evaluatør åpne et vindu som viste den opprinnelige journalteksten. Eventuelt (randomisert) fikk evaluatør bare den opprinnelige teksten fra starten av. Hvorvidt evaluatør ble presentert for det ene (eller andre alternativ fra starten av ble randomisert slik at det var lik sjansje for det ene eller det andre alternativ. Tidspunktene for når de ulike vinduene ble åpnet og når evaluatør ga sitt svar ble registrert automatisk. Forbrukt tid kunne så beregnes ut i fra tidspunktene.

*Figur:
Brukergrensesnitt
for evaluering*



PASIENTOPPLYSNINGER
SLUTT

BEGYNN MED DINE INITIALER HER:

I DETTE OMRÅDET VIL DET KOMME TILSYNE JOURNALOPPLYSNINGER

Nøkkedata Innt. pr:

PROBLEMER
cancer mammae, glandulmetastaser, skjeletmetastaser;

TILTAK
operert evakuering;

OBSERVASJONER + LAB
DBS: BT: 100/60. Puls: 80.; blyd grad III over hele precordiet.
RTG/UL: skjelettscitografi i høyre iliacacralledt høyre acetabulum;

MEDIKAMENTER
Zestoretic tabl. 1 x 1.; Lexin tabl. 80 mg x 1.; Kalcium Dureller 1 gram x 1.; Zyloric tabl. 180 mg x 1.; Paragin forte tabl. 2 tabl. x 4 ved behov.;

FLERE OPPLYSNINGER, TAKK!

Kort tekst Innt. pr:

SAMMENDRAG OG NØKKELOPPLYSNINGER: Kreft evd.
Diagnose: Cancer mammae cum met. 174.3
Tidligere sykdommer: Erysipeløs gjentatte ganger. 1987 utredet med spørsmål om revascularisere. Plaget av isctax på høyre side i mange år. Behandlet for hjertesvkt.
Aktuelt: Cancer mamma sin i 1984. Ablatio mamma. Postoperativ strålebehandling. I 1995 påvist glandulmetastaser i høyre axille og skjeletmetastaser. Inntkommer nå til vurdering.
Status: Ingen icterus eller cyanose. Status etter venstresidig ablatio. Palperer ikke sikker patologi.
Resime: 68 år gammel kvinne som ble operert for cancer mamma sin i 1984. I 1995 påvist glandulmetastaser i høyre axille og i bekkenet. Skjeletmetastaser til colurna og høyre del av bekkenet. Har fått 3 kurer ved OSS og kommer nå inn til evaluering.
Problemløsing: cancer mammae, glandulmetastaser, skjeletmetastaser;
Tiltak: operert evakuering;
Legemidler: Zestoretic tabl. 1 x 1.; Lexin tabl. 80 mg x 1.; Kalcium Dureller 1 gram x 1.; Zyloric tabl. 180 mg x 1.; Paragin forte tabl. 2 tabl. x 4 ved behov. ;
U.s. & Lab: DBS: BT: 100/60. Puls: 80.; blyd grad III over hele precordiet.

VIS FULL TEKST, TAKK!

ANGI KORT ESSENSEN AV PROBLEMATIKKEN:

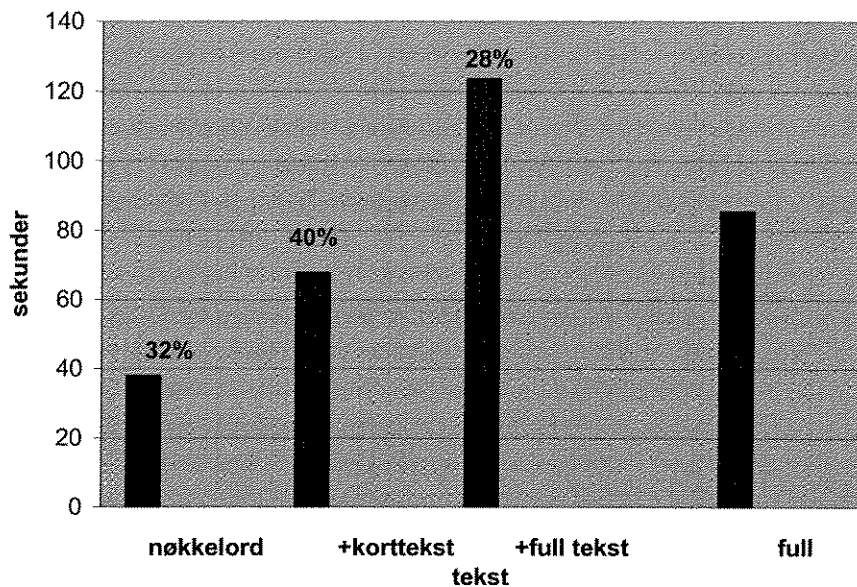
Passert Kjøling Nøkk
Innt. pr:

NESTE OPPGAVE

SLUTT

Seks leger / siste års medisinerstudenter ble brukt som evaluatører. Alle fikk 10 journaltekster, innleggelsesjournaler eller journalnotater fra ulike fagområder, som var valgt tilfeldig.

Resultat av evalueringen fremgår av figur ...



Figur Resultat av evaluering mht tidsbruk

Figuren viser at i 32% av tilfellene kunne evaluator angi essensen i journalen på basis av nøkkelopplysningene alene, og ca 40 sekunder ble forbrukt til å lese opplysningene og avgi svar. I 68% av tilfellene ville evaluator ha tillegg ha a) Korttekst (sammendrag) i alt i 40% av tilfellene, og deretter b) Full tekst i alt i 28% av tilfellene. Å avgi svar på basis av nøkkelopplysninger + korttekst tok ca 70 sekunder i gjennomsnitt, mens ved tillegg av hel tekst tok det ca 120 sekunder før evaluator kunne avgi et svar. Hvis evaluator bare fikk den fulle teksten fra begynnelsen tok det ca 85 sekunder å avgi svaret, i gjennomsnitt.

Mønsteret var helt konsistent for alle evaluatorene: Det tok kortere tid å avgi svaret hvis dette lot seg gjøre på basis av nøkkeldata + sammendrag, enn å lese i gjennom hele teksten og avgi svar. Derimot, hvis nøkkeldata + korttekst var utilstrekkelig tok det lengre tid å gå gjennom nøkkeldata + kort tekst + hel tekst enn det tok å avgi svaret på bakgrunn av bare hel tekst.

6.2 Evaluering av datarepresentativitet

Det er essensielt at brukere av et system som gir automatiske sammendrag kan føle tillit til relevans og fullstendighet av nøkkelord og sammendrag.

Ca 20 anonyme journaltekster ble valgt ut og delt i 2 tilfeldige utsnitt á 10 tekster, alle med opprinnelig tekst samt sammendrag med nøkkelopplysninger. Respondentene ble bedt om å rangere representativiteten av sammendrag + nøkkelord én av i fem kategorier fra "Svært mangelfull" til "Helt dekkende".

Tekstene ble sendt til 8 ulike leger slik at hver fikk en bunke med 10 tekster; hver tekst skulle hvis alle besvarte m.a.o. bli gradert av fire respondenter. Det innkom i alt seks svar

Svarene dekket spekteret fra "Svært mangelfull" til "Godt dekkende", men med tyngdepunkt på den negative siden (mangelfull). I tillegg til graderingen ble respondentene bedt om kommentere

direkte i teksten eller på andre måte. Disse kommentarene var meget nyttige fordi de avdekket følgende:

1. Sammendraget manglet symptomer. I den versjonen som var utarbeidet for evalueringen, var symptomer utelatt; teksten bygget på forekomst av D- og T- begreper.
2. Listen en over medikamenter var ufullstendig noe som medførte at visse viktige medikamentelle behandlingsformer var utelatt.

For øvrig ble det klart at korrekt staving i den opprinnelige journalteksten var avgjørende for vurderingen av mangler. F. eks. ble det anført i ett tilfelle at legemiddelet Ekvacillin manglet i listen over legemidler i nøkkelopplysningene; årsaken tildette var at det var skrevet Ecvacillin (med c!) i teksten og programmet gjenkjente derfor ikke ordet. På samme måte ble tekster i diagnosefeltet av typen: "Ktr. E. Oper." (= kontroll etter operasjon) oversett.

På basis av denne evalueringen kunne det enkelt gjennomføres noen prinsipielle korreksjoner i programmet:

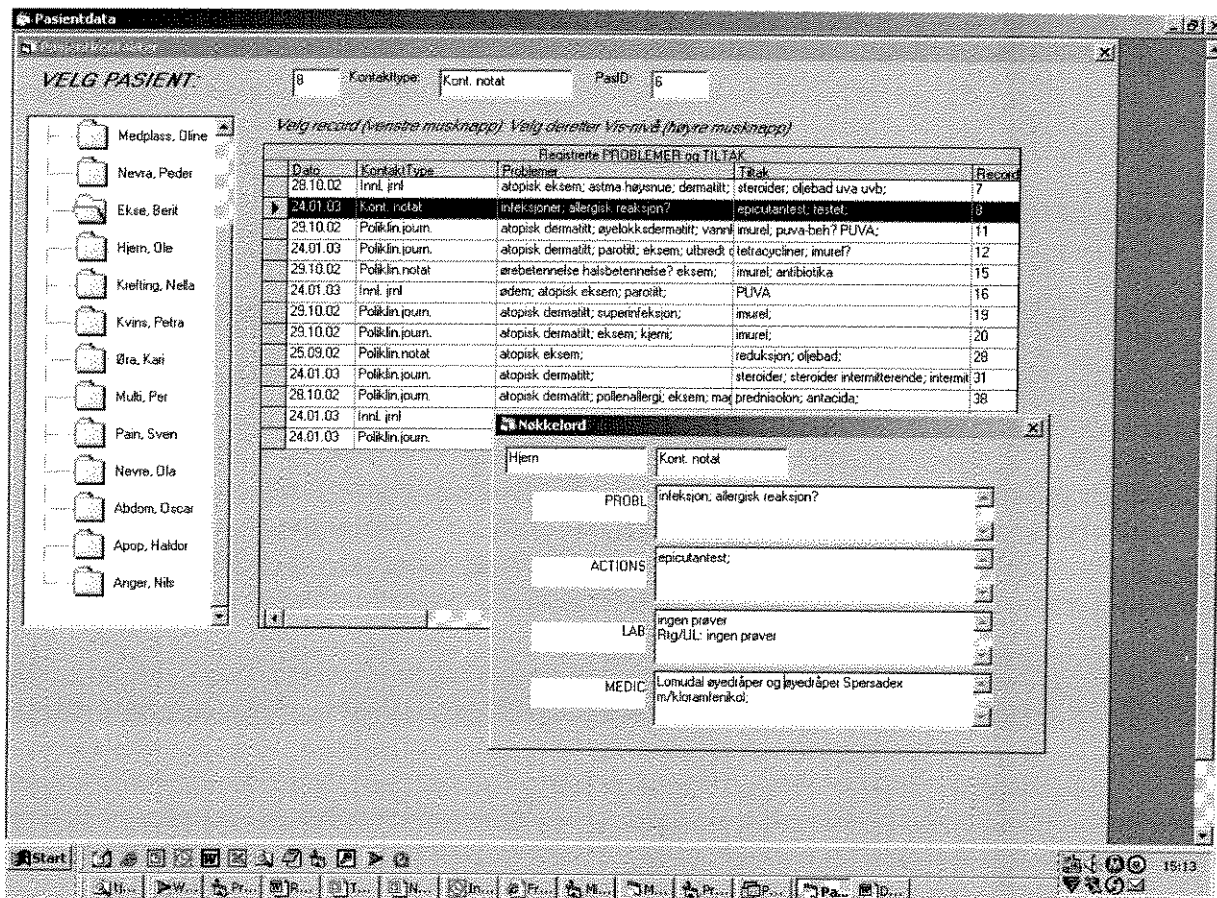
1. Forekomst av S-klassifiserte ord i teksten ble inkludert som basis for generering av sammendrag.
2. Legemiddellisten ble oppdatert ut i fra siste versjon av Felleskatalogen og generiske navn ble inkludert.

Dette førte direkte til at de fleste påviste manglene forsvant. En ny begrenset evaluering ga som resultat en vurdert "kompletthet" på ca 90%. Ulempen var at sammendraget økte i volum. Journaltekster som allerede er knappe og som har et minimum av informative opplysninger, vil knapt bli nedkortet overhodet, utenom at rutinemessige oppramsinger av "Uten anmerkning/U.a." -avsnitt forsvinner.

- En tredje endring som kan overveies er tekstgjenkjenning på basis av similaritet (Ecvacillin *similar med* Ekvacillin.) Dette er imidlertid en oppgave som ligger utenfor denne delen av prosjektet. Det ville formodentlig være enklere å sikre at journaltekstene er korrekte i hht ordlistene før tekstbearbeiding. Dette krever at det finnes en automatisk rettefunksjon i teksteditoren og at ordlistene inkluderes i teksteditorens liste over normaltermer.

7 BRUKERGRENSENITT

Brukergrensesnitt er ikke en vesentlig del av dette prosjektet, men det ble allikevel forsøkt med noen preliminære utforminger. En variant som er vist nedenfor gir oversikt over pasientene, og for hver mappe som åpnes vil man kunne få entabellarisk oversikt over de aktuelle kontaktene for denne pasienten, og i samme tabell, en opplisting av de (automatisk) registrerte problemer og tiltak.



The screenshot shows a software interface for patient data. At the top, there's a window titled "Pasientdata" with a sub-window "VELG PASIENT". Below this, there's a list of patient folders on the left and a table of contacts in the center. A "Nøkkelord" (Key words) window is open over the table, showing details for a specific contact.

Table: Registrerte PROBLEMER og TILTAK

Date	Kontakt type	Problemer	Tiltak	Record
28.10.02	Innl. jnl	atopisk eksem, astma høyeneue, dermatitt	steroider; oljebad uva uvb;	7
24.01.03	Kont. notat	infeksjoner, allergisk reaksjon?	epicutantest, testet	8
29.10.02	Poliklin.journ.	atopisk dermatitt; øyelokkdermatitt; vann	insurel; puva-beh? PUVA;	11
24.01.03	Poliklin.journ.	atopisk dermatitt; parotitt; eksem; ulbredt	tetracycliner; imuref?	12
29.10.02	Poliklin.notat	arebetennelse halsbetennelse? eksem;	insurel; antibiotika	15
24.01.03	Innl. jnl	adem; atopisk eksem; parotitt;	PUVA	16
29.10.02	Poliklin.journ.	atopisk dermatitt; superinfeksjon;	insurel;	19
29.10.02	Poliklin.journ.	atopisk dermatitt; eksem; kjem;	insurel;	20
25.09.02	Poliklin.notat	atopisk eksem;	reduksjon; oljebad;	28
24.01.03	Poliklin.journ.	atopisk dermatitt;	steroider; steroider intermitterende; interni	31
28.10.02	Poliklin.journ.	atopisk dermatitt; pollenallergi; eksem; mo	prednisolon; antacida;	38
24.01.03	Innl. jnl			
24.01.03	Poliklin.journ.			

Nøkkelord window details:

- Hjem: Kont. notat
- PROBL: infeksjon, allergisk reaksjon?
- ACTIONS: epicutanest;
- LAB: ingen prøver; Rtg/UL: ingen prøver
- MEDIC: Lomudal øyediåper og jøydriåpet Spersadex m/kloranfenkol.

Figur Brukergrensenitt, utkast

Ved å klikke på en enkelt kontakt kan en få vist alternativt:

- *Nøkkelord vindu (som vist i figur)*
- *Korttekst vind*
- *Hel tekst vindu*

Den kanskje viktigste verdien av en slik presentasjon kan ligge i den raske oversikten en får over pasientens sykehistorie. I tabell er det gjengitt (den automatiske) oversikten over Problemer/Tiltak for en pasient med poliklinikkbesøk, senere innleggelse og med tilkommen sykehusinfeksjon, organsvikt, men med heldig utgang.

Dato	Kontakttype	Problemer	Tiltak
10.12.01	Poliklin.notat	revmatoid artritt; osteoporose; hjertesvikt; residiv; hydrops; ra-sykdom;	prednisolon; kortison;
18.02.02	Inn.journal	revmatoid artritt; residiverende bronkitter;	operert artrodese venstre;
19.02.02	Kont. notat	cholecystitt; ryggsmerte; pneumoni; septikemi; osteoporose; infeksjon;	antibiotikabehandling;
20.02.02	Kont. notat	septikemi pneumoni; osteoporose; ulcus; melena; colitt; akutt astma	antibiotikabehandling ventrikkelsonde; blodtransfusjon mobilisering;
22.02.02	Kont. notat	febril	ventrikkelsonde;
10.03.02	Kont. notat	venetrombose; lungeemboli; DVT;	profylaktisk

8 OPPSUMMERING OG KONKLUSJON

For å karakterisere innholdet i medisinske pasientjournaler er det utviklet et system for gjenfinning og karakterisering av enkeltbegreper i frie tekster. Systemet bygger på:

1. Gjenfinning av begreper som karakteriserer ulike tekstavsnitt, som "Diagnoser", "Tidligere sykdommer", "Aktuelt" og liknende. Slike begreper gir indikasjon om innholdet i det etterfølgende tekstavsnittet.
2. Inndeling i setninger
3. Gjenfinning av nøkkelord knyttet til diagnoser, behandling/tiltak, lokasjoner, symptomer, sykdomsagens, legemidler og laboratorieundersøkelser. En rekke underliggende kataloger over enkeltord er utviklet for å utgjøre basis for tekstsøk. Dersom et nøkkelord gjenfinnes i teksten registreres hvilket nøkkelord samt dets posisjon.
4. Oppbygging av nøkkelbegreper på basis av strenger med nøkkelord. Komplette diagnoser lages på basis av ev enkeltord, for eksempel "apoplexia cerebri" fra apoplexia (diagnoseterm) + cerebri (lokasjonsterm).
5. Karakterisering av setninger i forhold til innholdet av nøkkelord
6. Oppbygging av tekststrenger som kan bestå av:
 - a. diagnosetermer som kan forekomme i utvalgte tekstavsnitt
 - b. tiltakstermer som kan forekomme i utvalgte tekstavsnitt
 - c. forekommende legemidler med dosering,
 - d. forekommende laboratorieundersøkelser
 - e. genererte korttekster på basis av forekomst av nøkkelbegreper i enkeltsetninger, jfr pkt 5. Tekst kan bygges opp utelukkende på basis av setninger med diagnostiske termer, tiltak, medikamenter osv.

Det er laget ulike grensesnitt for demonstrere konsekvenser av prinsippene, men disse tjener bare som illustrasjon. Fra et grensesnitt som viser basale pasientdata kan en f.eks klikke seg fram til korttekster eller hel tekst.

Tekstursnittene ble evaluert mht

Tid for å få oversikt over klinisk problem. Testpersoner ble presentert for alternativene:

1. diagnosetekststrenger, evt med tillegg av
2. korttekst basert på diagnose- og tiltakstermer, evt med tillegg av
3. full tekst

eller (randomisert):

4. bare full tekst

Vei pkt 1-3 tok i gjennomsnitt ca 30% kortere tid for å gi oversikt over det aktuelle kliniske problemet i forhold til vei pkt 4. I ca 70% av tilfellene nøyde testpersonen seg med basale nøkkelord pluss evt korttekst før korrekt vurdering kunne avgis.

Kompletthet av korttekstalternativ. Åtte klinikere fikk hver presentert ca 10 komplette tekster (randomisert utvalg) samt de relevante listene over de forekommende nøkkelbegrepene pluss korttekster generert på basis av de forekommende diagnose- og tiltakstermene. Vurderinger gitt på en femdelt skala fra *Svært mangelfull* til *Helt dekkende* ga generelt *Mangelfull* som svar. Kommentarer til vurderingene viste at detaljopplysninger, som i mange tilfelle var knyttet til en mangel i katalogen over medikamenter, og/eller at opplysninger av symptomkarakter, som var utelatt, manglet. Nye korttekster ble generert på basis av en oppdatert medikamentliste og med inkludering av symptomtermer. En ny begrenset evaluering av disse ga som resultat en vurdert "kompletthet" på ca 90%.

Denne studien har vist at det er mulig å bruke nøkkelordkataloger til å karakterisere innholdet av medisinske tekster. På basis av lister av identifiserte nøkkeltermer kan man bygge opp ulike måter å representere innholdet i teksten. I dette prosjektet er det i anvendelsessammenheng bare utprøvd noen muligheter.

VISUAL BASIC PROGRAM

```
Dim ParName(20) As String           ' paragraph name, according to list
Dim ParPos(20) As Integer           ' startpos paragraph
Dim Sent(200) As String             ' sentences
Dim SentClass(200) As String        ' REL = relevant; "" = irrelevant; QED =
question; NEG = negative
Dim SentPar(200) As String          ' paragraph related to sentencen
Dim SentPos(200) As Integer         ' startpos sentence
Dim strWord(1000) As String         ' word
Dim WordPos(1000) As Integer        ' startpos, ord
Dim KeyWord(200) As String          ' keywords
Dim KeyWordPos(200) As Integer      ' startpos, keyword
Dim KeyWordClass(200) As Variant
Dim AntKeyWords As Integer          ' # keywords
Dim AntPar As Integer, AntSent As Integer 'antall avsnitt, setninger og ord
Dim AntWords As Integer
Dim AntKeys As Integer, lastKeyword As Integer
Dim strCompress_D As String, strCompress_T As String, strCompress_L As String
Option Explicit
```

```
-----
Private Sub cmdSave_Click()
' to transfer new data into datase
'On Error GoTo Error
'update database fields:
Me.txtHelTekst = rchTekst.TextRTF
Me.txtProbl = Me.txtProblems.Text
Me.txtTiltak = Me.txtActions.Text
Me.txtLabverdier = Me.txtLaboratory
Me.txtLegemidl = Me.txtMedications
Me.txtShort = Me.txtShort_text

Me.adoData.Recordset.Update

Unload frmShortText ' if open
mnuNyPasient_Click ' new patient
Exit Sub

Error: MsgBox "error: allerede lagret?"

End Sub
```

```
Private Sub Form_Initialize()
lastKeyword = 0
End Sub
```

```
-----
Private Sub Form_Load()
Dim strQuery As String
' select correct database record
strQuery = "Select * from Løpende_journal Where RecordID=" & _
& frmNavn.txtRecordID
With adoData
.RecordSource = strQuery
On Error Resume Next
.Refresh
End With
'add new database record:
Me.adoData.Recordset.Delete
Me.adoData.Recordset.AddNew

Me.txtPasID = frmNavn.txtPasID ' bind to
```

```

Me.txtRecordID = frmNavn.txtRecordID
'Close #3
'sample necessary data after random-conversion
AntKeys = frmNavn.iAntKeys ' number in random list of keywords
lastKeyword = 0
' position form
With Me
    .Width = 14000
    .Height = 9500
    .Top = 0
    .Left = 0
End With
Open "C:/demo/res.txt" For Output As #3
filTekst_Click
End Sub
-----
Private Sub filTekst_Click()
    Dim strInfil As String, strLine As String, strQuery As String
    Dim i As Integer, iWhere As Integer, char As String
    rchTekst.Text = "" 'blanking

    strQuery = "Select Fil_HelTekst from Kontakter where RecordID=" & _
    & frmNavn.txtRecordID
    'Text1 er koplet til denne ADO'en; filnavnet på, tekstfila står nå der
    With adoFil
        .RecordSource = strQuery
        On Error Resume Next
        .Refresh
    End With
    Me.txtKontaktType = frmNavn.txtKontaktType
    strInfil = Text1
    rchTekst.SelColor = &H80000007
    'read and display text:
    Open strInfil For Input As #1
        Do While Not EOF(1)
            Line Input #1, strLine
            If strLine <> "" Then
                'replace tabs:
                10 iWhere = InStr(1, strLine, Chr$(9), 0)
                    If iWhere > 0 Then
                        strLine = Left$(strLine, iWhere - 1) & " " & Right$(strLine,
Len(strLine) - iWhere)
                        GoTo 10 ' CHECK IF MORE
                    End If
                    If Right$(rchTekst.Text, 1) = "-" Then
                        rchTekst.Text = Left$(rchTekst.Text, Len(rchTekst.Text) - 1) '
kapper -
                    Else
                        rchTekst.Text = rchTekst.Text + " " + vbCrLf ' ordskille
                    End If
                    ' add dot to text if not present:
                    If Trim$(rchTekst.Text) <> "" _
                    And Right$(Trim$(rchTekst.Text), 1) <> "." _
                    And Right$(Trim$(rchTekst.Text), 1) <> ":" Then
                        If (Left$(strLine, 1) >= "A" _
                        And Left$(strLine, 1) <= "Z") _
                        Or Left$(strLine, 1) = "Æ" _
                        Or Left$(strLine, 1) = "Ø" _
                        Or Left$(strLine, 1) = "Å" _
                        Or Left$(strLine, 2) = vbCrLf Then
                            rchTekst.Text = Trim$(rchTekst.Text) + ". "

```

```

        End If
    End If
    ' finally add the new line
    rchTekst.Text = rchTekst.Text + Trim$(strLine)
End If
Loop
Close #1
rchTekst.Text = rchTekst.Text & ". " ' add final dot
' zeroing:
AntPar = 0: AntSent = 0: AntWords = 0:
Paragraphs
Words
End Sub

```

```

-----
Private Sub Paragraphs()
    Dim iPar As Integer, iStart As Integer, iBlank As Integer
    Dim i As Integer, iWhere, iTemp As Integer, strTemp As String, sign As
String
    Dim Swap As Boolean
    'denne rutinen markere ut faste tekstavsnitt.
    'Tekstavsnittene markeres i hht malen for vedk. journalsystem. OBS at dette
kan variere.
    Dim Para As Variant
    Para = Array("Innl.diag", "Diagn", "Innleggelsesdiagn", _
        "Journal ved", "Årsak til innl", _
        "Fam", "Sos", "Anamn", _
        "Gyn/obst anamn", "Tidligere sykd", _
        "Aktuelt", "Aktuell sykehist", "Medikament", "Allergi",
"Stimulant", _
        "Funn", "Naturlige funksj", _
        "Status pres", "Status lo", "Vurdering", _
        "Resym", ";")

    AntPar = 0
    'ParPos (i) = startposisjonen for avsnittet
    ParPos(0) = 1
    For i = 0 To 21 ' the number of arguments in parameter array (minus ";")-
must be correct!
        ' Find string in text.
        iWhere = InStr(1, rchTekst.Text, Para(i), 0)
        If iWhere > 0 Then
            iPar = iPar + 1                'paragraph is present
            'ParName(iPar) = Para(i)
            ' sett inn kolon etter avsnitts-overskriften om dette mangler:
            iBlank = InStr(iWhere + Len(Para(i)) - 1, _
                rchTekst.Text, " ", 1) ' first blank after paragraph
            ParName(iPar) = Mid$(rchTekst.Text, iWhere, iBlank - iWhere)
            If Right$(ParName(iPar), 1) <> ":" Then
                ParName(iPar) = ParName(iPar) & ":"
            ElseIf Right$(ParName(iPar), 1) = "." Then
                ParName(iPar) = Left$(ParName(iPar), Len(ParName(iPar)) - 1) &
"; "
            End If
            If Left$(ParName(iPar), 4) = vbCrLf + vbCrLf Then
                ParName(iPar) = Right$(ParName(iPar), Len(ParName(iPar)) - 2)
            End If
            With Me.rchTekst
                ' If found
                .SetFocus
                .SelStart = iWhere - 1
                .SelText = vbCrLf
                .SelStart = iWhere + 1
            End With
        End If
    Next i
End Sub

```

```

        .SelLength = iBlank - iWhere
        .SelColor = RGB(0, 0, 200)           'marker avsnitt med blålig
farge
        .SelText = ParName(iPar) & " "
    End With
    ParPos(iPar) = iWhere ' start of paragraph
    If ParPos(iPar) < ParPos(iPar - 1) Then
        ' swap
        strTemp = ParName(iPar - 1)
        iTemp = ParPos(iPar - 1)
        ParName(iPar - 1) = ParName(iPar)
        ParPos(iPar - 1) = ParPos(iPar)
        ParName(iPar) = strTemp
        ParPos(iPar) = iTemp
    End If
End If
Next i
AntPar = iPar 'total # of paragraphs
ParPos(AntPar + 1) = Len(rchTekst.Text) 'include end-point
rchTekst.SelStart = Len(rchTekst.Text)
200 'rearrange the paragraph sequence if necessary
Swap = False
For i = 1 To AntPar
    iTemp = ParPos(i)
    strTemp = ParName(i)
    If ParPos(i + 1) < ParPos(i) Then
        Swap = True
        ParPos(i) = ParPos(i + 1)
        ParPos(i + 1) = iTemp
        ParName(i) = ParName(i + 1)
        ParName(i + 1) = strTemp
    End If
    'Print #3, i, ParPos(i), ParName(i)
Next i
' iterate if necessary
If Swap Then GoTo 200
Close #3
End Sub

```

```

Private Sub Words()
'Rutine for å definere ord og setninger i teksten
Dim i As Long, iWord As Integer, iNumWord As Integer, iSent As Integer
Dim iWhere As Long, iWhere1 As Long, iWhere2 As Long
Dim NewSent As Boolean: NewSent = False
Dim LastBreak As String

'zeroing:
iWord = 0: WordPos(0) = 1: WordPos(1) = 1: lastKeyword = 0
SentPos(0) = 1
' define words
For i = 1 To Len(rchTekst.Text) - 2
    ' find blank or tab or / separating words, in pos >= 1 from last
position
10    iWhere1 = InStr(i + 1, rchTekst.Text, " ", 0)
        iWhere2 = InStr(i + 1, rchTekst.Text, "/", 0)
        If iWhere2 > 0 Then
            iWhere = Min(iWhere1, iWhere2)
        Else
            iWhere = iWhere1
        End If
        If iWhere <> 0 Then

```

```

        iWord = iWord + 1                                ' antall
ord
    ' ordets startposisjon:
    WordPos(iWord) = iWhere '
    strWord(iWord) = Trim$(Mid$(rchTekst.Text, WordPos(iWord - 1), _
    WordPos(iWord) - WordPos(iWord - 1)))
    ' skip empty words:
    If strWord(iWord) = "" Or strWord(iWord) = " " Or strWord(iWord) =
vbCrLf Then
        iWord = iWord - 1
        i = i + 1
        GoTo 10
    End If
    ' hopp til slutten av ordet:
100    i = WordPos(iWord)    ' jump to next start position
    End If
Next i
AntWords = iWord    ' number of words

' =====
' define sentences:
iSent = 1: SentPos(1) = 1
For iWord = 1 To AntWords
    ' necessary correction because WordPos now is at the word end
    WordPos(iWord) = WordPos(iWord) - Len(strWord(iWord))
    'print #3, iWord, WordPos(iWord), strWord(iWord)
    ' check if start of new sentence :
    If Left$(strWord(iWord), 2) = Chr(13) + Chr(10) Then ' _
        'Or (Left$(strWord(iWord), 1) >= "A" _
        'And Left$(strWord(iWord), 1) <= "Z" _
        NewSent = True
    End If
    If Right$(Trim$(strWord(iWord - 1)), 1) = "." _
    Or Right$(Trim$(strWord(iWord - 1)), 1) = ":" Then '
        If (Left$(strWord(iWord), 1) >= "A" _
        And Left$(strWord(iWord), 1) <= "Z") _
        Or Left$(strWord(iWord), 1) = "E" _
        Or Left$(strWord(iWord), 1) = "Ø" _
        Or Left$(strWord(iWord), 1) = "Å" _
        Or Left$(strWord(iWord), 2) = vbCrLf _
        Or InStr(1, strWord(iWord), "#", 0) Then
            NewSent = True
        End If
    ' special cases:
    If InStr(1, strWord(iWord - 1), "Resym", 0) Then NewSent = True
    If InStr(1, strWord(iWord - 1), "Vurder", 0) Then NewSent = True

    ' break at "," or "dvs" or "men" but avoid repetitions:
    ElseIf Right$(strWord(iWord - 1), 1) = "," _
    Or Left$(strWord(iWord), 3) = "dvs" _
    Or Left$(strWord(iWord), 3) = "men" _
    Or InStr(1, (strWord(iWord) + " " + strWord(iWord + 1)), "selv om", 0) _
    Or InStr(1, (strWord(iWord) + " " + strWord(iWord + 1)), "slik at", 0) _
Then
        If (Left$(strWord(iWord), 1) >= "a" _
        And Left$(strWord(iWord), 1) <= "z") _
        And Not (Right$(strWord(iWord), 1) = "," _
        Or Right$(strWord(iWord + 1), 1) = "," _
        Or Right$(strWord(iWord - 2), 1) = "," _
        Or Right$(strWord(iWord - 1), 1) = ":") Then

```

```

        NewSent = True
    End If
End If
' correction:
If Right$(strWord(iWord - 1), 1) = ":"
And InStr(1, strWord(iWord), "U.a", 0) Then
    NewSent = False
End If
If NewSent Then
    iSent = iSent + 1
    SentPos(iSent) = WordPos(iWord)
    ' content previous sentence:
    Sent(iSent - 1) = Trim$(Mid$(rchTekst.Text, SentPos(iSent - 1), _
    SentPos(iSent) - SentPos(iSent - 1)))
    NewSent = False
End If
' remove CRLF from start of words
If Left$(strWord(iWord), 2) = vbCrLf Then
    strWord(iWord) = Right$(strWord(iWord), Len(strWord(iWord)) - 2)
End If

Next iWord

' last sentence:
AntSent = iSent + 1
'SentPos(AntSent) = SentPos(AntSent - 1) + Len(Sent(AntSent - 1))
Sent(AntSent - 1) = Trim$(Mid$(rchTekst.Text, SentPos(AntSent - 1),
Len(rchTekst.Text) - SentPos(AntSent - 1)))
Sent(AntSent) = ""
SentPos(AntSent) = Len(rchTekst.Text)
Sent(AntSent - 1) = Sent(AntSent - 1) ' for debugging
For i = 1 To AntSent
    SentPar(i) = Parag(SentPos(i))
Next i
End Sub

-----
Sub KeyWords()
' routine to identify keywords (according to file) in the text
Dim i As Integer, iLen As Integer, iStart As Integer
Dim j As Integer: j = 0 'running n.
Dim SeekWord As String, FoundWord As String, Class As Variant
Dim Mark As Boolean

'zeroing
AntKeyWords = 0
txtKeyWords = ""
For i = 1 To AntWords
    SeekWord = Trim$(LCase$(strWord(i)))
    ' exclude , or . etc at end of word
10 If SeekWord <> "" Then
        If Right$(SeekWord, 1) < "A" Then
            SeekWord = Left$(SeekWord, Len(SeekWord) - 1)
            'repeat if necessary
            GoTo 10
        End If
    End If
    Ordstamme SeekWord, FoundWord, Class ' slow sequential search, limited
file
    If FoundWord <> "" Then
        FoundWord = FoundWord ' for debugging
    End If

```



```

If Class = "" Then
    SeekKeyWord SeekWord, FoundWord, Class      ' fast routine
End If

'Print #3, SeekWord, FoundWord, Class
' show keywords in the text:
If Class = "D" _
Or Class = "T" _
Or Class = "L" _
Or Class = "S" _
Or Class = "R" _
Or Class = "N" _
Or Class = "A" _
Then
    j = j + 1
    KeyWord(j) = FoundWord
    ' alternatively:
    'KeyWord(j) = SeekWord
    KeyWordClass(j) = Class
    KeyWordPos(j) = WordPos(i)
    '
    ' IMPORTANT PART:
    ' combine related keywords in sentences if last word was
    ' also a keyword: (R+L; T+L; D/S+L; R+S;
    If LCase$(strWord(i - 1)) = KeyWord(j - 1) And _
        WhichSent(KeyWordPos(j - 1)) = WhichSent(KeyWordPos(j)) Then
        If KeyWordClass(j - 1) = KeyWordClass(j) Or _
            KeyWordClass(j - 1) = "R" And KeyWordClass(j) = "L" Or _
            KeyWordClass(j - 1) = "R" And KeyWordClass(j) = "S" Or _
            KeyWordClass(j - 1) = "L" And KeyWordClass(j) = "S" Or _
            KeyWordClass(j - 1) = "L" And KeyWordClass(j) = "D" Then
            KeyWord(j - 1) = KeyWord(j - 1) + " " + KeyWord(j)
            KeyWordClass(j - 1) = KeyWordClass(j)
            strWord(i) = KeyWord(j - 1)
            j = j - 1      ' delete last keyword
        ' combine keywords but classify according to the first: (D,T+L)
    ElseIf _
        KeyWordClass(j - 1) = "D" And KeyWordClass(j) = "L" Or _
        KeyWordClass(j - 1) = "D" And KeyWordClass(j) = "R" Or _
        KeyWordClass(j - 1) = "T" And KeyWordClass(j) = "L" Then
            KeyWord(j - 1) = KeyWord(j - 1) + " " + KeyWord(j)
            KeyWordClass(j - 1) = KeyWordClass(j - 1) ' to be explicit
            strWord(i) = KeyWord(j - 1)
            j = j - 1      ' delete last keyword
        End If
    End If
End If

End If

'show:

If j > 0 And Class <> "" Then
    iLen = 0: iStart = 0: Mark = False
    ' red mark significant keywords
    ' mark keyword combinations
    If _
        (KeyWordClass(j - 1) = "D" And KeyWordClass(j) = "L") Or _
        (KeyWordClass(j - 1) = "S" And KeyWordClass(j) = "L") Or _
        (KeyWordClass(j - 1) = "N" And KeyWordClass(j) = "D") Or _
        (KeyWordClass(j - 1) = "N" And KeyWordClass(j) = "S") Or _

```

```

(KeyWordClass(j - 1) = "N" And KeyWordClass(j) = "R") Then
If WhichSent(KeyWordPos(j - 1)) = WhichSent(KeyWordPos(j)) Then
    iLen = KeyWordPos(j) + Len(KeyWord(j)) - KeyWordPos(j - 1)
    iStart = KeyWordPos(j - 1) - 1
    Mark = True
End If
' or mark single words
ElseIf KeyWordClass(j) = "D" Or _
KeyWordClass(j) = "S" Or _
KeyWordClass(j) = "T" Then
    iLen = Len(KeyWord(j))
    iStart = KeyWordPos(j) - 1
    Mark = True
End If
If Mark Then
    With Me!rchTekst
        ' If found.
        '.SetFocus
        .SelStart = iStart          ' startposisjon
        .SelLength = iLen          ' markeringslengde
        .SelColor = RGB(200, 0, 0) ' rød uthevelse
    End With
End If
End If
Next i
AntKeyWords = j
For j = 1 To AntKeyWords
    txtKeyWords = txtKeyWords & KeyWord(j) & "/" & KeyWordClass(j) _
        & Str$(j) & "/" & Str$(KeyWordPos(j)) & vbCrLf
Next j
Me!rchTekst.SelStart = Len(rchTekst.Text)
'Close #3
End Sub

```

```

-----
Sub KeyWord_Display()
' to analyse the keywords
Dim iSent As Integer, dummy As String
Dim KeyWordList_D As String, KeyWordList_T As String, KeyWordList_L As
String

'zeroing
txtSent.Text = "" ' is updated with class
txtActions = "": txtLaboratory = "": txtMedications = ""
strCompress_D = "": strCompress_T = "": strCompress_L = ""
txtProblems = ""

For iSent = 2 To AntSent
    'Sent(iSent) = Sent(iSent) ' for debugging
    'dummy = Parag(SentPos(iSent))
    ' always start with text in diagnosis:
    If InStr(1, Parag(SentPos(iSent)), "diagn", 1) And Not InStr(1,
Sent(iSent), "diagn", 1) Then
        txtProblems.Text = txtProblems.Text & Sent(iSent)
    End If
    SentClass(iSent) = ""
    Classify iSent
    'SentClass(iSent) = SentClass(iSent) ' for debugging
    If SentClass(iSent) <> "" Then
        If InStr(1, SentPar(iSent), "Innl", 0) > 0 Or _
            InStr(1, SentPar(iSent), "Diagn", 0) > 0 Or _
            InStr(1, SentPar(iSent), "Årsak til innl", 0) > 0 Or _

```

```

InStr(1, SentPar(iSent), "Aktuel", 0) > 0 Or _
InStr(1, SentPar(iSent), "Resym", 0) > 0 Or _
InStr(1, SentPar(iSent), "Vurder", 0) > 0 Or _
InStr(1, Me.txtKontaktType.Text, "notat", 1) > 0 Then
    ' include diagnosis always

        ' collect keywords in sentence; strCompress'es are in the
common memory
        Collect SentPos(iSent), SentPos(iSent + 1), SentClass(iSent)
    End If
End If
strCompress_D = strCompress_D ' for debugging
Next iSent
' display:
txtProblems = txtProblems & "; " & Trim$(strCompress_D)
'If txtProblems = "" Then txtProblems = "ingen alv. probl."
txtActions = txtActions & Trim$(strCompress_T)
If txtActions = "" Then txtActions = "ingen anført"
End Sub
-----
Private Sub Short_Text(show As Boolean)
    ' to display relevant sentences
    Dim iSent As Integer, iWhere As Integer, i As Integer, char As String
    Dim LabText As String, ShortText As String, AddText As String
    If InStr(1, txtKontaktType, "innl", 0) Then
        ShortText = "Diagnose: " 'start
    End If
    For iSent = 1 To AntSent
        'Sent(iSent) = Sent(iSent)
        If Sent(iSent) <> "" Then
            ' always start with text in diagnosis:
            If InStr(1, Parag(SentPos(iSent)), "diagn", 1) And Not InStr(1,
Sent(iSent), "diagn", 1) Then
                ShortText = ShortText & Sent(iSent)
            End If
            If InStr(1, SentClass(iSent), "REL", 0) > 0 _
And InStr(1, SentClass(iSent), "NEG", 0) = 0 Then
                ' skip first words like "men", mens", "fortsatt" a.o.
                If InStr(1, Sent(iSent), "men ", 1) _
Or InStr(1, Sent(iSent), "mens ", 1) _
Or InStr(1, Sent(iSent), "dvs. ", 1) _
Or InStr(1, Sent(iSent), "samt ", 1) _
Or InStr(1, Sent(iSent), "fortsatt ", 1) Then
                    iWhere = InStr(1, Sent(iSent), " ", 0)
                    Sent(iSent) = Right$(Sent(iSent), Len(Sent(iSent)) - iWhere)
                End If
                ' skip CRLF as first characters:
                If Left$(Sent(iSent), 2) = vbCrLf Then
                    Sent(iSent) = Right$(Sent(iSent), Len(Sent(iSent)) - 2)
                End If

                If SentPar(iSent) <> "" And InStr(1, ShortText, SentPar(iSent),
0) = 0 Then
                    AddText = vbCrLf + SentPar(iSent) + " " +
UCase$(Left$(Trim$(Sent(iSent)), 1)) _
+ Right$(Trim$(Sent(iSent)), Len(Trim$(Sent(iSent))) - 1)
                Else
                    If Right$(ShortText, 1) = "." And Len(Trim$(Sent(iSent))) >
1 Then
                        AddText = " " + UCase$(Left$(Trim$(Sent(iSent)), 1)) _

```

```

+ Right$(Trim$(Sent(iSent)), Len(Trim$(Sent(iSent))) -
1)
    Else
        AddText = " " & Trim$(Sent(iSent))
    End If
End If
SentPar(iSent) = SentPar(iSent) ' debugging
ShortText = ShortText + AddText
End If
End If
Next iSent
Me.txtShort_text = ShortText + vbCrLf +
"-----" + vbCrLf
& "Problemstilling: " & txtProblems.Text & vbCrLf
& "Tiltak: " & txtActions.Text & vbCrLf
& "Legemidler: " & txtMedications.Text & vbCrLf
& "U.s. & Lab: " & txtLaboratory.Text
' final textpre"p:
If Left$(txtShort_text, 2) = vbCrLf Then
    txtShort_text.Text = Right$(txtShort_text, Len(txtShort_text) - 2)
End If
txtShort_text.Text = Trim$(txtShort_text.Text)
frmShortText.Text1 = txtShort_text.Text
show: Load frmShortText
frmShortText.txtShorttext.Text = Me.txtShort_text
End Sub

```

```

-----
Private Sub Classify(iSent As Integer)
' routine to classify the type of sentence;
' "REL" = relevant, i.e, contains T or D type keywords;
' "NEG": contains negations; "QUE" contains a clinical question

Dim iKeyWord As Integer, RelevantKeyWords As String, strLine As String
Dim iAntT As Integer, iAntD As Integer, iAntL As Integer
' first characterize sentences according to existing expressions:
SentClass(iSent) = SentClass(iSent) ' debug
If InStr(1, (Sent(iSent)), "ingen ", 1) Or _
InStr(1, Sent(iSent), "Ingen ", 0) Or _
Left$(Sent(iSent), 6) = "ingen " Or _
InStr(1, (Sent(iSent)), "intet ", 1) Or _
InStr(1, Sent(iSent), "Intet ", 0) Or _
Left$(Sent(iSent), 6) = "intet " Or _
InStr(1, (Sent(iSent)), "Ikke ", 0) Or _
InStr(1, (Sent(iSent)), " ikke ", 1) Or _
InStr(1, Sent(iSent), "Normal ", 0) Or _
InStr(1, (Sent(iSent)), "Normale ", 0) Or _
InStr(1, (Sent(iSent)), " normal ", 1) Or _
InStr(1, (Sent(iSent)), " normale ", 1) Or _
InStr(1, (Sent(iSent)), "U.a", 0) Or _
InStr(1, (Sent(iSent)), " u.a", 1) Or _
InStr(1, (Sent(iSent)), " hverken ", 1) Or _
(InStr(1, Sent(iSent), "Hverken ", 0)
And InStr(1, Sent(iSent), " eller ", 1)) Then
    SentClass(iSent) = "/NEG" ' negative clause
ElseIf InStr(1, LCase$(Sent(iSent)), "mulig ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "kanskje ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "mistanke ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "usikker ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "prøve ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "sjekke om ", 0) Or _
InStr(1, LCase$(Sent(iSent)), "spørsmål ", 0) Then

```

```

    SentClass(iSent) = "/QUE"
End If
' then check keywords to pick out relevant sentences, _
' i.e. contains keywords of of class D or T
For iKeyWord = 1 To AntKeyWords
    ' any keyword in the sentence?
    If SentPos(iSent) <= KeyWordPos(iKeyWord) And _
        KeyWordPos(iKeyWord) < SentPos(iSent + 1) Then

        KeyWord(iKeyWord) = KeyWord(iKeyWord) ' for debug
        KeyWordClass(iKeyWord) = KeyWordClass(iKeyWord) ' for debug

        ' keyword of relevant class?
        If KeyWordClass(iKeyWord) = "D" Or KeyWordClass(iKeyWord) = "T" Then
            SentClass(iSent) = SentClass(iSent) + "/REL"
        ElseIf KeyWordClass(iKeyWord) = "L" Then
            SentClass(iSent) = SentClass(iSent) + "/LOC"
        End If
        ElseIf KeyWordPos(iKeyWord) > SentPos(iSent + 1) Then
            GoTo Slutt
        End If
    Next iKeyWord
    ' classifiactions occur also in the MED and LAB subroutines
Slutt:

    Close #4
    Close #5
End Sub

-----
Private Sub Collect(StartPos As Integer, EndPos As Integer, Class As String)
    'to collect all keywords within a sentence according to position StartPos
    from to EndPos of sentence
    Dim iKeyWord As Integer
    Dim Added As Boolean: Added = False
    Dim Found_D As Boolean: Found_D = False
    Dim Found_T As Boolean: Found_T = False
    Dim Found_L As Boolean: Found_L = False
    If lastKeyword = 0 Then lastKeyword = 1 ' initial value

    For iKeyWord = lastKeyword To AntKeyWords
        If KeyWordPos(iKeyWord) < StartPos - 1 Then ' allow a margin of one
position
            GoTo 10
        ElseIf KeyWordPos(iKeyWord) > EndPos Then
            lastKeyword = iKeyWord - 1
            GoTo Add_Sign
        End If
        If StartPos - 1 <= KeyWordPos(iKeyWord) And KeyWordPos(iKeyWord) <=
EndPos - 1 Then
            ' include only D,T or L type keywords if Class is REL
            If InStr(1, Class, "REL", 0) Or InStr(1, Class, "LOC", 0) Then
                If (KeyWordClass(iKeyWord) = "D" Or _
                    KeyWordClass(iKeyWord) = "T" Or _
                    KeyWordClass(iKeyWord) = "L") Then
                    ' combine keywords?:
                    ' If KeyWordClass(iKeyWord - 1) = "D" And
KeyWordClass(iKeyWord) = "L" Then 'Or KeyWordClass(iKeyWord - 1) = "S" And
KeyWordClass(iKeyWord) = "L"
                        ' Found_D = True
                        ' If InStr(1, strCompress_D, Left$(KeyWord(iKeyWord),
Int(Len(KeyWord(iKeyWord)) * 2 / 3)), 0) = 0 Then

```

```

''          strCompress_D = strCompress_D + KeyWord(iKeyWord)
+ " "
'          Added = True
'          End If
'          End If
If KeyWordClass(iKeyWord) = "D" Then
    Found_D = True
    ' add keyword, but avoid repetitions in keyword string;
    'OBS KeyWord minus 3 characters to avoid repetitions
because of differnt word endings
    If InStr(1, strCompress_D, Left$(KeyWord(iKeyWord),
Int(Len(KeyWord(iKeyWord)) * 2 / 3)), 0) = 0 Then
        strCompress_D = strCompress_D + KeyWord(iKeyWord) +
" "
        Added = True
    End If
End If
If KeyWordClass(iKeyWord) = "T" Then
    Found_T = True
    ' avoid repetitions in keyword string:
    If InStr(1, strCompress_T, KeyWord(iKeyWord), 0) = 0
Then
        strCompress_T = strCompress_T + KeyWord(iKeyWord) +
" "
        Added = True
    End If
End If
If KeyWordClass(iKeyWord) = "L" Then
    Found_L = True
    ' avoid repetitions in keyword string:
    If InStr(1, strCompress_L, KeyWord(iKeyWord), 0) = 0
Then
        strCompress_L = strCompress_L + KeyWord(iKeyWord) +
" "
        Added = True
    End If
End If
End If
End If
10 Next iKeyWord

Add_Sign: ' insert separation marks, and ?-mark if appropriate
    If Found_D Then
        If InStr(1, Class, "QUE", 0) > 0 And Right$(Trim$(strCompress_D), 1)
<> "?" Then
            If (Right$(Trim$(strCompress_D), 1) = ";" Then
                strCompress_D = Trim$(Left$(strCompress_D,
Len(strCompress_D) - 2)) + "? "
            Else
                strCompress_D = Trim$(Left$(strCompress_D,
Len(strCompress_D) - 1)) + "? "
            End If
            ElseIf Added And InStr(1, Class, "REL", 0) > 0 _
And (Right$(Trim$(strCompress_D), 1) <> ";" Or
Right$(Trim$(strCompress_D), 1) <> "?") Then
                strCompress_D = Trim$(strCompress_D) + "; "
            End If
            ElseIf Found_T Then
                If InStr(1, Class, "QUE", 0) > 0 And Right$(Trim$(strCompress_T), 1)
<> "?" Then

```

```

        If (Right$(Trim$(strCompress_T), 1) = ";") Then
            strCompress_T = Trim$(Left$(strCompress_T,
Len(strCompress_T) - 2)) + "? "
        Else
            strCompress_T = Trim$(Left$(strCompress_T,
Len(strCompress_T) - 1)) + "? "
        End If
        ElseIf Added And InStr(1, Class, "REL", 0) > 0 _
        And (Right$(Trim$(strCompress_T), 1) <> ";" Or
Right$(Trim$(strCompress_T), 1) <> "?") Then
            strCompress_T = Trim$(strCompress_T) + "; "
        End If

    End If
End Sub

```

```

-----
Function Parag(iPos As Integer)
Dim i As Integer, idum As Integer
For i = 1 To AntPar
    If ParPos(i) <= iPos And iPos < ParPos(i + 1) Then
        Parag = ParName(i)
    End If
Next i
End Function

```

```

Function WhichSent(iPos As Integer)
Dim i As Integer, idum As Integer
For i = 1 To AntSent
    If SentPos(i) <= iPos And iPos < SentPos(i + 1) Then
        WhichSent = SentPos(i)
    End If
Next i
End Function

```

```

Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y
As Single)
    If Button = 2 Then PopupMenu mnuPOP
End Sub

```

```

-----
Sub SeekKeyWord(SeekWord As String, FoundWord As String, Class As Variant)
' routine to seek a specific word in a random string of
' almost 16000 identified words
' in ICD-9 and ICD-10, ICPC, NCSP; etc classifications.
' Each Keyword is classified according to:
' diagnostic term : "D"; therapy related: "T";
' anatomy related (location) : "L"; administrative term : "0";
' agent: "A", unspecified "?"

```

```

Dim MyRecord As Record
Dim Iter As Integer, iPos As Integer, iAntall As Integer, i As Integer, iLastPos
As Integer
Dim KeyWord As String, Compl As String
Dim iLow As Integer, iHigh As Integer
Dim lemma As Boolean

```

```

Open "C:/demo/rand-med-ord.txt" For Random As #2 Len = Len(MyRecord)
iHigh = AntKeys          ' antall nøkkelord i tabell
iLow = 0                 ' ca midt i filen
'finne omtrentlig posisjon, nedenfor (større enn) det søkte ordet
iLastPos = 0

```

```

Start_Loop:
Do While Abs(iPos - iLastPos) >= 0 And Iter < 15
    iPos = CInt((iHigh + iLow) / 2): If iPos = 0 Then GoTo 50

```

```

'If Abs(iPos - iLastPos) <= 1 Then GoTo 100
Get #2, iPos, MyRecord.Name
iLastPos = iPos
FoundWord = Left$(MyRecord.Name, Len(Trim$(MyRecord.Name)) - 5) '
nøkkelord minus hale
Class = Right$(Trim$(MyRecord.Name), 1) ' type nøkkelord
Compl = Mid$(Trim$(MyRecord.Name), Len(Trim$(MyRecord.Name)) - 3, 1)
If StrComp(SeekWord, FoundWord, vbTextCompare) = 1 Then
    iLow = iPos
ElseIf StrComp(SeekWord, FoundWord, vbTextCompare) = -1 Then
    iHigh = iPos
    ' perfect match:
ElseIf StrComp(SeekWord, FoundWord, vbTextCompare) = 0 Then
    GoTo Found ' found, success
End If
Iter = Iter + 1
' Print #3, Iter, iPos, FoundWord + " / " + SeekWord
Loop
'If Left$(SeekWord, 3) = Left$(FoundWord, 3) Then
''    GoTo 50
'Else
'    GoTo 100 ' ikke funnet, forsøk lemmatisering
'End If
50 If lemma Then GoTo Not_Found
    ' no success so far
100    ' forsøker lemmatisering
    'bestemt form
    If Right$(SeekWord, 2) = "en" Or _
        Right$(SeekWord, 2) = "et" Then
        SeekWord = Left$(SeekWord, Len(SeekWord) - 2)
        lemma = True
    ' flertall, ubestemt form
    ElseIf Right$(SeekWord, 2) = "er" Then
        SeekWord = Left$(SeekWord, Len(SeekWord) - 1)
        lemma = True
    ' flertall, bestemt form
    ElseIf Right$(SeekWord, 3) = "ene" _
        Or Right$(SeekWord, 3) = "ens" _
        Or Right$(SeekWord, 3) = "ets" Then
        SeekWord = Left$(SeekWord, Len(SeekWord) - 2)
        lemma = True
    Else
        lemma = False
    End If
    If lemma Then
        'seek around last position
        iHigh = iPos + 50
        ' AVOID NEGATIVE POSITIONS
        If iLow > 50 Then
            iLow = iPos - 50
        Else
            iLow = 1
        End If
        iLastPos = 0 ' resetting
        Iter = 0
        GoTo Start_Loop
    End If

Not_Found:
    If Not lemma And Len(SeekWord) > 5 Then
        ' handle abridged words

```



```

    If (Compl = "A" Or Compl = "S") And _
        (Class = "D" Or Class = "T" Or Class = "S") Then
        If Left$(SeekWord, Len(FoundWord)) = FoundWord Then
            FoundWord = SeekWord
            GoTo Found
        End If
    End If
End If
'no success at all
FoundWord = ""
Class = ""
GoTo 500
Found: ' success
'Print #3, Iter, iPos, SeekWord + " / " + FoundWord + "/" + Class
500 Close #2
End Sub

```

```
-----
Private Sub Medikamenter()
```

```

Dim iSent As Integer, iWhere As Integer
Dim strSumMed As String, MedOrd As String
Dim KeyWordList As String, Class As String, strSent As String

```

```

' MEDIKAMENT-MODUL
'=====

```

```

For iSent = 1 To AntSent
    If InStr(1, SentPar(iSent), "Medik", 0) _
        Or InStr(1, Me.txtKontaktType.Text, "notat", 1) Then
        ' fil med relevante begreper
        Open "c:\demo\legemidler.txt" For Input As #1
        Do While Not EOF(1)
            Line Input #1, MedOrd ' sekvensielt søk
            If Trim(MedOrd) <> "" Then
                MedOrd = " " + Left$(MedOrd, Len(MedOrd)) ' adderer
blanks og fjerner halen
                If Len(MedOrd) < 3 Then MedOrd = MedOrd + " " ' add
blank, otherwise...
                'Class = Right$(MedOrd, 1)
                'MedOrd in sentence? add blank as first! otherwise the
first word my be missing
                iWhere = InStr(1, " " + Sent(iSent), MedOrd, 0)
                If iWhere Then
                    strSent = Mid$(Sent(iSent), iWhere, Len(Sent(iSent))
- iWhere + 1)
                    SentClass(iSent) = SentClass(iSent) + "/MED"
                    If InStr(1, strSumMed, strSent, 1) = 0 Then
                        strSumMed = strSumMed + strSent + vbCrLf
                    End If
                End If
            End If
        Loop
        Close #1
    End If
Next iSent
If strSumMed = "" Then
    strSumMed = "Ingen iflg. off.liste"
End If
txtMedications = strSumMed

```

End Sub

Private Sub Labdata()

' utfører identifisering og sammentrekning av lab/rtg relaterte data

Dim LabOrd As String, RtgOrd As String, Class As String, Avsn(200) As Integer
Dim iWord As Integer, iNkl As Integer, iStart As Integer, iAktAvsn As Integer
Dim LabOrdBlank As String, LabOrdColon As String, sign As String, TestSent As String

Dim strNklOrdTest(200) As String, TestOrd As String, iSent As Integer, iWhere As Integer, iWhere1 As Integer, iWhere2 As Integer

Dim strSumLab As String, strSumRtg As String, OrdAvsnType(200) As String, strLastString As String, iAntNkl As Integer

Dim KeyWordList_D As String, KeyWordList_T As String, KeyWordList_L As String, strSent As String

' Open "c:\demo\epikrise-res.txt" For Append As #4

' LAB-MODUL

'=====

```
For iSent = 1 To AntSent
  If InStr(1, SentClass(iSent), "NEG", 1) = 0 Then
    If InStr(1, SentPar(iSent), "Status ", 0)
      Or InStr(1, Me.txtKontaktType.Text, "notat", 1) Then
      ' labord in sentence? add blanks as first! otherwise the
first word may be missing
      TestSent = " " & Sent(iSent)
      ' fil med relevante begreper
      Open "c:\demo\lab-ord-ny.txt" For Input As #1
      Do While Not EOF(1)
        Line Input #1, LabOrd          ' sekvensielt søk
        If Trim(LabOrd) <> "" Then
          LabOrd = " " + Left$(LabOrd, Len(LabOrd) - 2) '
adder blanks og fjern halen
          If Len(LabOrd) <= 3 Then
            LabOrdBlank = LabOrd + " " ' add blank tail
            LabOrdColon = LabOrd + ":"
            iWhere1 = InStr(1, " " + TestSent, LabOrdBlank,
0) ' these will be alternatives
            iWhere2 = InStr(1, " " + TestSent, LabOrdColon,
0)
            If (iWhere1 + iWhere2) Then
              'SentClass(iSent) = SentClass(iSent) +
"/LAB"
              strSent = Mid$(TestSent, (iWhere1 +
iWhere2), Len(TestSent) - iWhere + 1)
              End If
            ElseIf Len(LabOrd) > 3 Then
              iWhere = InStr(1, " " + TestSent, LabOrd, 1)
              If iWhere > 0 Then
                strSent = Mid$(TestSent, iWhere,
Len(TestSent) - iWhere + 1)
              End If
            End If
          End If
          If strSent <> "" Then
            SentClass(iSent) = SentClass(iSent) & "\LAB"
            ' accept if not already registered
            If InStr(1, strSumLab, strSent, 1) = 0 Then
              strSumLab = strSumLab + " " + strSent + "; "
```

```

                End If
            End If
        End If
    Loop
    Close #1
End If
End If
Next iSent
'=====
' RTGMODUL
'=====
lastKeyword = 0 ' must be zeroed
For iSent = 1 To AntSent
    ' fil med relevante begreper
    Open "c:\demo\rtg-ny.txt" For Input As #1
    Do While Not EOF(1)
        Line Input #1, LabOrd          ' sekvensielt søk
        LabOrd = " " + Left$(LabOrd, Len(LabOrd) - 2) ' adder blanks og
fjern halen
        If Len(LabOrd) < 3 Then LabOrd = LabOrd + " " ' add blank,
otherwise...
        'Class = Right$(LabOrd, 1)
        'labord in sentence? add blank as first! otherwise the first
word my be missing
        If InStr(1, Sent(iSent), "rtg", 0) Then
            iWhere = iWhere                ' for debugging
        End If
        iWhere = InStr(1, " " + Sent(iSent), LabOrd, 0)
        ' show LabOrd and related L-class keywords
        If iWhere Then
            SentClass(iSent) = SentClass(iSent) + "/LAB"
            ' slett all tidligere info i strCompress_L
            strCompress_L = ""
            ' If InStr(1, SentClass(iSent), "/LOC", 0) Then
            Collect SentPos(iSent), SentPos(iSent + 1),
SentClass(iSent)
            ' add if not already present
            If InStr(1, strSumRtg, Trim$(LabOrd + " " +
strCompress_L), 1) = 0 Then
                If InStr(1, SentClass(iSent), "QUE", 0) Then
                    sign = "? "
                Else
                    sign = "; "
                End If
                strSumRtg = strSumRtg + Trim$(LabOrd + " " +
strCompress_L) + sign
            End If
        End If
    End If
    Loop
    Close #1
Next iSent

If strSumLab = "" Then strSumLab = "ingen prøver"
If strSumRtg = "" Then strSumRtg = "ingen prøver"

txtLaboratory = strSumLab _
& vbCrLf & "Rtg/UL: " & strSumRtg
End Sub
-----
Private Sub mnuKeyWords_Click()

```

```

Dim i As Integer
  lastKeyword = 0
  KeyWords
  Keyword_Display 'includes classify sentences and collect keywords in
relevant clauses
  lastKeyword = 0
  Labdata
  lastKeyword = 0
  Medikamenter

  For i = 1 To AntSent
    txtSent = txtSent & Sent(i) & SentClass(i) & vbCrLf
  Next i

  Short_Text 0
End Sub
-----
Function Min(Arg1 As Variant, Arg2 As Variant)
  If Arg1 <= Arg2 Then
    Min = Arg1
  Else
    Min = Arg2
  End If
End Function
-----
Private Sub Ordstamme(SeekWord As String, FoundWord As String, Class As Variant)
Dim infile As Integer, strOrdstamme As String

' routine to identify keywords based on general suffixes

  infile = FreeFile
  Open "c:\demo\ordstammer.txt" For Input As #infile
  Do While Not EOF(infile)
    ' read in sequence.
    Input #1, strOrdstamme
    If strOrdstamme <> "" Then
      If Right$(SeekWord, Len(strOrdstamme) - 5) = Left$(strOrdstamme,
Len(strOrdstamme) - 5) Then
        FoundWord = SeekWord
        Class = Right$(strOrdstamme, 1)
        Close #infile
        Exit Sub
      End If
    End If
  Loop
  ' no result:
  FoundWord = ""
  Class = ""
  Close #infile
End Sub

```

