Institutt for elektronikk og telekommunikasjon

Eksamensoppgave i TTT4185 Taleteknologi

Faglig kontakt under eksamen: Tor André Myrvoll
Tlf.: +47 95 14 80 14

Eksamensdato: Fredag 9. desember 2016

Eksamenstid (fra - til): 09.00 - 13.00

Hjelpemiddelkode/tillatte hjelpemidler: **C** – Spesifiserte trykte og håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.

Annen informasjon:

- Eksamen består av 3 oppgaver der
  - oppgave 1 omhandler taleanalyse
  - oppgave 2 omhandler talegjenkjenning
  - oppgave 3 omhandler dyp læring
- Alle deloppgaver teller likt
- Alle oppgavene skal besvares
- Sensurfrist er 3 uker etter eksamensdato.

Målform/språk: Norsk - bokmål

Totalt antall sider: 10

Herav, antall vedleggsider:

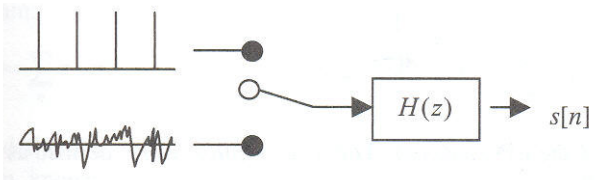
Kontrollert av:

Dato Signatur

Merk! Studenter finner sensur i Studentweb. Har du spørsmål om din sensur må du kontakte instituttet. Eksamenskontoret vil ikke kunne svare på slike spørsmål.

Oppgave 1

1a) Skisser kilde/filter-modellen og forklar hvorfor den er en rimelig god modell for tale. Hvilke typer fonemer er den spesielt godt egnet for, og hvilke er den mindre egnet for?



Skissen over er en kilde-filter modell. Til venstre finner vi eksitasjonen, som enten er i form av et pulstog fra stemmebåndene eller turbulent luft. Til høyre finner vi filteret, som er en representasjon av taleorganet.

Modellen er egnet for lyder som er kort-tids-stasjonære. Dette inkluderer vokaler, frikativer og diftonger. Stopp-lyder er imidlertid ikke i denne klassen av lyder.

1b) I mange anvendelser, for eksempel ved uttrekking av egenskaps-vektorer, ønsker man å separere kilden fra filteret. Beskriv kort hvordan dette kan gjøres ved hjelp av henholdsvis lineær prediksjon og signalets cepstrum.

Basert på kilde-filter modellen antar vi at signalet $x(n)$ er av formen

$$x(n) = \sum_{i=1}^{p} h_i x(n-i) + e(n) \tag{1}$$

der h_i er filteret og $e(n)$ er eksitasjonen. Et lineært predisjonsfilter estimerer h_i ved å minimaliserer predisjonsfeilen

$$\hat{h} = \underset{h}{\operatorname{argmax}}\, E = \underset{h}{\operatorname{argmax}} \sum_{n} \hat{e}^2(n) \tag{2}$$

der

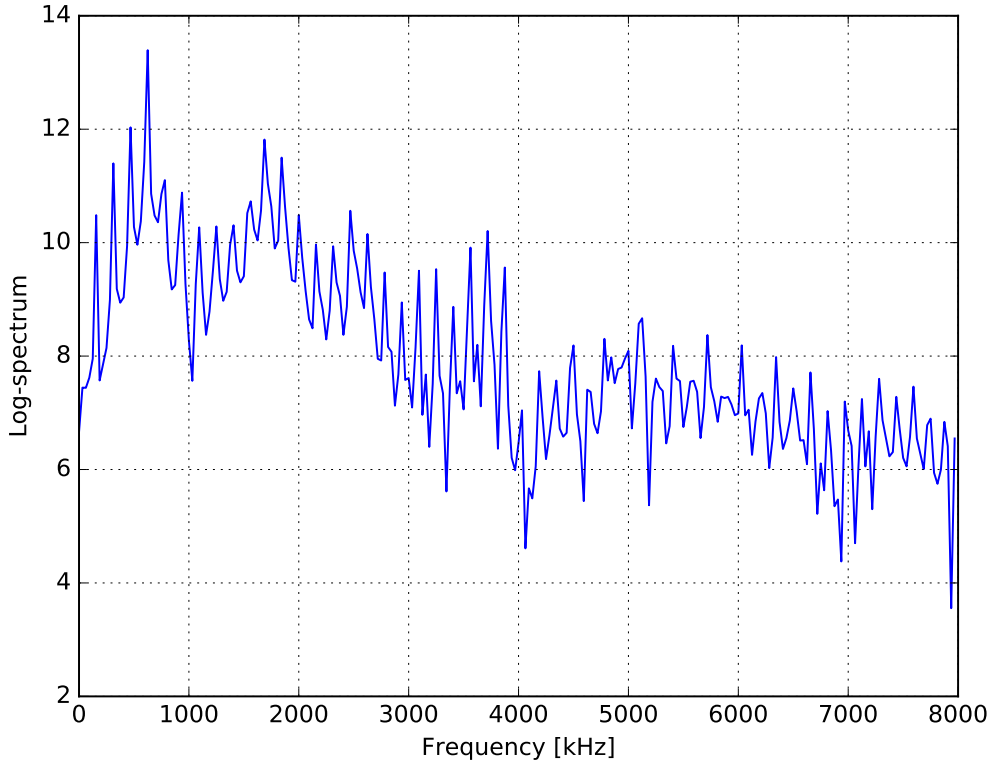
$$\begin{aligned} \hat{e}(n) &= x(n) - \hat{x}(n) \\ &= x(n) - \sum_{i=1}^{p} h_i x(n-i) \end{aligned} \tag{3}$$

I cepstral-domenet er kanalen additiv til signalet. Signalene er separable siden kanalen stot sett er glatt og saktevarierende og derfor representert av $c(k)$ for lave k, mens eksitasjonen x er hurtigvarierende og representeres av høye k. Ved å sette

$$\hat{c}(k) = \begin{cases} c(k), & k < K \\ 0, & K \leq k \end{cases} \tag{4}$$

og deretter gjenvinne log-spekteret fra $\hat{c}(k)$ kan vi finne omhyllingskurven.

1c) Forklar kort begrepene *fundemental frekvens* (F_0) og *formant* ($F_1, F_2, \ldots$). Estimer F_0, F_1 og F_2, altså den fundamentale frekvensen og de to første formantene, fra log-spektrumet i figuren under.



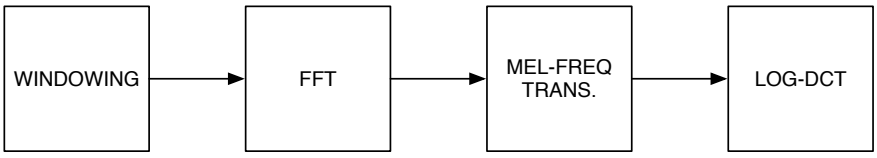
Den fundamentale frekvensen er relatert til et stemt talesignal, og er gitt ved tidsavstanden mellom pulsene som genereres av stemmebåndene. Hvis avstanden er T_0, så er den fundamentale frekvensen $F_0 = 1/T_0$

Formanter er spektrale topper i talespekteret gitt av akustiske resonanser i taleorganet. Forskjellige konfigurasjoner av taleorganet (plassering av tungen o.l.), gir distinkte formanter.

Den fundamentale frekvensen: Et pulstog med pulsavstand T_0 gir et pulstog med pulsavstand $F_0 = 1/T_0$ i frekvensplanet. Her har vi 19 pulser på 3000 Hz, mao. $F_0 \approx 157$ Hz.

Øyemål gir $F_1 \approx 600$ Hz, $F_2 \approx 1700$

1d) Den mest brukte egenskaps-vektoren for talegjenkjenning er basert på såkalte Mel-frequency-cepstral-coefficients (MFCC). Beskriv steg for steg hvordan disse egenskapsvektorene beregnes basert på et punktprøvet talesignal.



Prosesseringen skjer som følger:

- WINDOWING: Talesignalet deles opp i små, overlappende blokker som pålegges et vindu (typisk Hamming).

- FFT: Blokken transformeres til frekvensdomenet
- MEL-FREQ: Energien i overlappende delbånd av økende bredde beregnes. Delbåndenes posisjon og bredde er gitt av Mel-skalaen, en perseptuell frekvensskala.

LOG-DCT: Energien i delbåndene representeres ved sin log-verdi og transformeres vha. en diskret cosinus transform. Dette er ment å dekorrelere egenskapsvektoren.

Oppgave 2

2a) Når man beskriver forskjellige basisenheter for talegjenkjenning bruker man ofte begrepene *accuracy*, *trainability* og *generalizability*. Forklar hva man mener med disse begrepene, og diskuter hvordan basisenhetene ord, fonemer og kontekstavhengige fonemer oppfyller dem.

There are mainly three characteristics we want a speech unit to have: accuracy, trainability and generalizability.

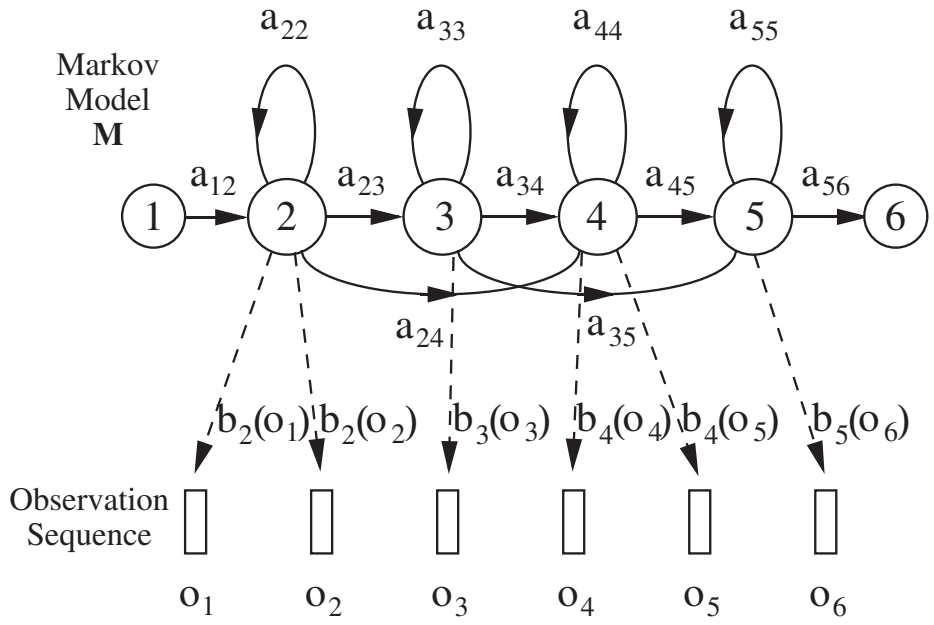
- Words
  - High accuracy, since the all coarticulation effects are internal to the word.
  - Low trainability since the number of words are is very high.
  - Low generalizability as a new data needs to be collected and models trained if new words are to added
  - Suited for tasks with small vocabularies
- Phones are the most common speech unit in current systems, and can be further divided into two types:
  - Context independent phones
    * Low accuracy since coarticulation effects are not modeled.
    * Very high trainability since the number of phonemes is low (40-50).
    * Very high generalizability.
    * Suited for tasks with little available training data and little information about the kinds of input the system will receive.
  - Context dependent phones
    * High accuracy since coarticulation effects over triphones are modelled.
    * Low trainability since the number is very high. (However, clustering states where different contexts give rise to similar coarticulation effects can reduce this number considerably.)
    * Good generalizability.
    * Well suited for all tasks where there is sufficient data available to train them.

2b) Det skal lages en generell små-vokabular talegjenkjenner for industriell tale-kontroll. Vokabularet kan være opptil 50 ord, men forskjellige anvendelser hos bedriftene trenger forskjellige vokabularer. Diskuter fordeler og ulemper ved å bruke basisenhetene ord, fonemer og kontekstavhengige fonemer for en slik gjenkjenner.

Ordmodeller er et mulig alternativ for små, faste vokabularer. Her er det ikke lagt opp til et fast vokabular, og nye anvendelser kan innebære at man må trene nye modeller for hvert tilfelle.

Fonemer og kontekstavhengige fonemer har begge god *generalizability*. Kontekstavhengige fonemer vil ha bedre nøyaktighet, men krever mer treningsdata. For små vokabularer kan imidlertid fonemer være nøyaktige nok da antall kontekster er begrenset.

2c) En prototype av talegjenkjenneren blir utviklet basert på ord som basisenhet. Den akustiske modellen er en skjult Markov-modell (HMM) med kontinuerlige observasjoner. Beskriv en slik modell og dens parametre der observasjons-sannsynlighetene er gitt ved en Gaussisk blandingsfordeling (GMM-HMM). Forklar hvordan man med slike modeller kan beregne sannsynlighetene til observasjons-sekvenser av vilkårlig lengde.



Et eksempel på en HMM med fire tilstander er gitt over (den første og siste tilstanden kalles hhv. start- og stopp-tilstander, men er strengt ikke nødvendig å inkludere i figuren).

En HMM baserer seg på en Markov-kjede, dvs. en serie med tilfeldige diskrete tilstander der sannsynligheten for neste tilstand kun avhenger av den nåværende. Disse tilstandene observeres ikke direkte, men gjennom er en *observasjonsvektor* som trekkes fra en sannsynlighetsfordeling avhengig av tilstanden man står i.

Modellen har følgende parametre:

Transisjonssannsynlighetene bestemmer oppførelsen til Markov-kjeden.

En Gaussisk blandingsfordeling er gitt som

$$P(o;\Theta) = \sum_{k=1}^{K} c_k \mathcal{N}(o;\mu_k,\Sigma_k) \tag{5}$$

der $\{c_k, \mu_k, \Sigma_k\}$ er hhv. blandingskoeffisienten, middelverdivektoren og kovariansmatrisen til kompenent k.

En ser fra figuren at en skjult markovmodell kan generere et vilkårlig antall tilstander, $S = \{s_1, \ldots, s_T\}$, der hver tilstand genererer en observasjon. Gitt en serie observasjoner $O = \{o_1, \ldots, o_T\}$ så kan vi skrive

$$P(O,S) = P(O|S)P(S) \tag{6}$$

der

$$P(O|S) = b_{s_1}(o_1)b_{s_2}(o_2)\ldots b_{s_T}(o_T) \tag{7}$$

$$P(S) = a_{s_0 s_1} a_{s_1 s_2} \ldots a_{s_{T-1} s_T} \tag{8}$$

noe som igjen gir oss sannsynligheten for O ved å summere over alle lovlige tilstandssekvenser

$$P(O) = \sum_S P(O, S) \tag{9}$$

2d) Forover-algoritmen er basert på at man ved hjelp av rekursjon effektivt kan beregne

$$\alpha_t(i) = P(X_1^t, s_t = i; \Theta), \tag{10}$$

for alle t og i, der X_1^t er observasjonsvektorer, s_t er HMM-tilstanden ved tidspunkt t og Θ er HMM-parametrene. Forklar hvordan dette kan brukes til å beregne $P(X_1^T; \Theta)$.

Siden alle $\alpha_t(i)$ kan beregnes effektivt så betyr det at $\alpha_T(i) = P(X_1^T, s_t = i; \Theta)$ kan beregnes effektivt for alle i. Da kan vi enkelt finne $P(X_1^T; \Theta)$ ved

$$P(X_1^T; \Theta) = \sum_i P(X_1^T, s_t = i; \Theta) = \sum_i \alpha_T(i) \tag{11}$$

2e) Gitt at brukeren kun benytter ytringer bestående av enkeltord. Anta at a priori sannsynligheten for de enkelte ordene i vokabularet er gitt ved $P(w)$. Forklar hvordan forover-algoritmen i dette tilfellet kan brukes til å lage en enkel talegjenkjenner. Hvorfor vil ikke en slik talegjenkjenner være brukbar i praksis hvis vi tillater korte setninger i stedet for enkelt-ord?

En talegjenkjenner forsøker å finne den mest sannsynlige setningen W gitt et akustisk signal O,

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(W|O) = \underset{W}{\operatorname{argmax}}\, P(O|W)P(W) \tag{12}$$

der $P(O|W)$ er den akustiske modellen og $P(W)$ er språkmodellen.

For hvert enkelt ord w har vi en HMM Θ_w. Gitt en serie med observasjoner $O = \{o_1, \ldots, o_T\}$ kan vi altså bruke forward-algoritmen til å beregne $P(O|w) = P(O; \Theta_w)$ for hvert enkelt or w og plugge dette rett inn i

$$\hat{w} = \underset{w}{\operatorname{argmax}}\, P(w|O) = \underset{W}{\operatorname{argmax}}\, P(O|w)P(w) \tag{13}$$

Hvis vi tillater ordsekvenser $W = w_1, w_2, \ldots, w_N$, så må forward-algoritmen brukes til å beregne $P(O|W)$ for alle lovlige W. For selv små vokabularer er dette alt for komplekst da antall setninger øker eksponensielt i N.

2f) En utvidelse av systemet tillater bruk av hele setninger. Forklar hvordan man kan modellere sannsynligheten til en ord-sekvens ved hjelp av *N-gram modeller*.

The n-gram model models the probability of a given word in an utterance as a discrete probability given the $n-1$ previous words in the utterance. Example of a 3-gram:

$$\begin{aligned} &P(\text{I would like to listen to some Jazz}) \\ =&P(\text{I})P(\text{would}|\text{I})P(\text{like}|\text{would, I})P(\text{to}|\text{like, would})P(\text{listen}|\text{to, like})P(\text{to}|\text{listen, to}) \\ &\times P(\text{some}|\text{to, listen})P(\text{Jazz}|\text{some, to}) \end{aligned}$$

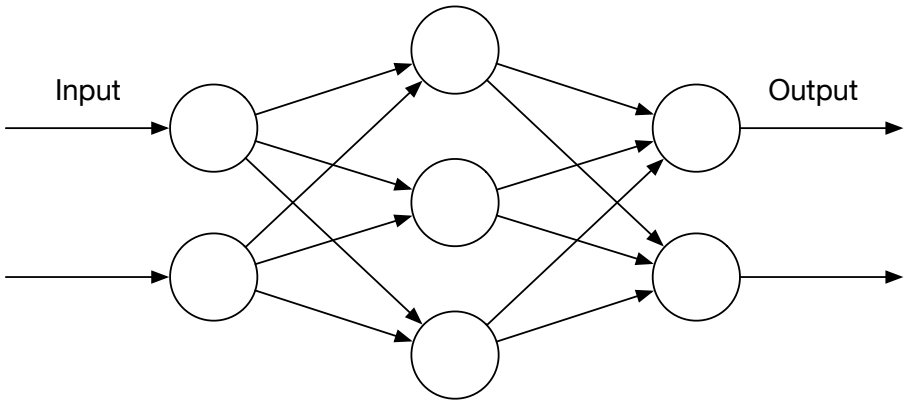
2g) Forklar hvorfor parameterestimering for N-gram med $N \geq 2$ er så vanskelig. Beskriv kort hvordan teknikkene *diskontering*, *backoff* og *interpolasjon* brukes i språkmodellering.

Discounting, *backoff* and *interpolation* are used to address the problem of estimating eg. 64 trillion parameters for a trigram (there is just not enough data in the world. Also, you wouldn't want to store all those parameters). In practice most trigrams would have zero examples in a limited training set, giving them a probability of zero, which would be a problem if that trigram was encountered during use.

- Discounting: Keep a small probability mass in reserve and distribute it evenly across all unseen trigrams.
- Backoff: If the trigram isn't found – use the bigram. If the bigram is not found – use the unigram.
- Interpolation: $P_I(w_3|w_2, w_1) = \alpha P(w_3|w_2, w_1) + \beta P(w_3|w_2) + (1 - \alpha - \beta)P(w_3)$ The weights $alpha, \beta$ are usually trained to maximize perplexity.

Oppgave 3

3a) Skisser et nevralt nettverk med ett skjult lag. Hva er fordelen med å ha flere skjulte lag; såkalte dype nevrale nettverk (DNN)?



Et nevralt nettverk med ett skjult lag kan i prinsippet approksimere enhver mapping (funksjon) mellom inngangen og utgangen av nettverket. Problemet er at antall noder i det skjulte laget må være enormt stort. Ved å bruke mange lag oppnår man samme effekt med mange færre noder i hvert lag.

3b) Anta et DNN med tre skjulte lag, der hvert av lagene har 200 prosesseringsnoder, observasjonsvektoren har dimensjon 10 og utgangen dimensjon to. Hvor mange frie parametre må estimeres i et slikt nettverk? Hvilken aktiveringsfunksjon bør vi bruke på utgangen av nettverket hvis vi ønsker en klassifiserer?

Fra pensumboken: La v^{l-1} være vektoren av utgangs verdiene fra lag $l-1$ og la z^l være vectoren av inngangsverdier til lag l. Da kan vi skrive

$$z^l = W v^{l-1} + b. \tag{14}$$

La antall noder i lag l være N_l. Da har W $N_{l-1} \times N_l$ parametre og b har N_l parametre.

La inngangen være lag 0 og utgangen lag 4. Da får vi

- Lag 0→1:$10 \times 200 + 200 = 2200$ parametre
- Lag 1→2:$200 \times 200 + 200 = 40200$ parametre
- Lag 2→3:$200 \times 200 + 200 = 40200$ parametre
- Lag 3→4:$200 \times 2 + 2 = 440$ parametre

Tilsammen 83040 frie parametre.

Hvis man ønsker å bruke nettverket som en klassifiserer må man bruke *soft-max* som aktiveringsfunksjon.

3c) Beskriv kort treningsstrategiene batch, mini-batch og stochastic gradient descent (SGD), samt deres fordeler og ulemper. Hva menes med global normalisering av egenskapsvektorer og hvorfor er dette viktig før trening?

- Batch: Gradienten beregnes ut fra alle eksemplene i treningssettet.
  - Fordel: Dette er den korrekte gradienten ut fra ønsket om å optimalisere kostfunksjonen på treningssettet.
  - Ulempe: Tar lang tid å beregne. Lite effektivt.
- SGD: Gradienten beregnes ut fra et eneste eksempel fra treningssettet
  - Fordel: Meget enkel og effektiv å beregne. Konvergerer raskere enn batch på store datasett
  - Ulempe: Vanskelig å parallelisere
- Minibatch: Beregn gradient vha. et lite subset av treningsdataene.
  - Fordel: Som SGD – mer effektiv enn batch for store datasett. Kan enkelt parallelliseres.
  - Ulempe: Ingen signifikante

Gitt et treningssett bestående av N traningsvektorer x_n. Global normalisering gir et treningssett der middelverdien av alle treningsvektorene er null, og der variansen til vektorkomponentene er én.

Formålet med dette er å unngå at gradienten blir tilnærmet lik null for mange parametre under trening. Dette kan skje hvis mange av inngangsverdiene er høye, noe som igjen vil få aktiveringsfunksjoner som sigmoider og hyperbolsk tangens til å gå i metning.

3d) Talegjenkjenning basert på DNN har de siste årene flyttet state-of-the-art ytelse betraktelig. Forklar hva som menes med *senoner*. Beskriv på overordnet nivå prinsippet bak den såkalte CD-DNN-HMM modellen (context-dependent deep neural network hidden Markov model).

Et *senon* er en tilstand i et trifon, og er ofte er delt mellom flere trifoner. Siden antall unike tilstander som trengs for å modellere en trifon-modell er svært høyt, kan man la flere trifoner dele en eller flere tilstander.

En CD-DNN-HMM er basert på et dypt nevralt nettverk som klassifiserer tale som senoner. Med andre ord – gitt en observasjonsvektor o_t, gir utgangen av et DNN oss sannsynlighetene, $P(s_t|o_t))$, for alle senonene i modellen.

En klassisk GMM-HMM baserer seg på observasjonssannsynlighetene $P(o_t|s_t)$, men disse kan skrives som

$$P(o_t|s_t) = \frac{P(s_t|o_t)P(o_t)}{P(s_t)} \tag{15}$$

Siden $P(o_t)$ er konstant for alle s_t kan den ignoreres, og vi kan bruke

$$\bar{P}(o_t|s_t) = \frac{P(s_t|o_t)}{P(s_t)} \tag{16}$$

direkte i vårt standard GMM-HMM rammeverk.