6.5 Statistische Spracherkennung

6.5.1 Spracherkennung mit MAP Regel

MAP – Regel (Maximum-a-posteriori-Regel

$$\hat{W} = \underset{W \in V^*}{\operatorname{argmax}} P(W \mid \mathbf{X})$$

optimale Wortfolge

$\hat{W} = w_1 w_2 \ldots w_K \quad w_i \in V$

Wortfolge

Merkmalssequenz

$\mathbf{X} = \mathbf{x}_1 \mathbf{x}_2 \ldots \mathbf{x}_T$

$$P(W \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid W) P(W)}{P(\mathbf{X})}$$

$$\hat{W} = \underset{W \in V^*}{\operatorname{argmax}} P(\mathbf{X} \mid W) \cdot P(W)$$

akustisches Modell
z.B.: HMM

Sprachmodell

6.5.2 Modellierung von Merkmalssequenzen

Merkmalssequenzen

$$\mathbf{X} = \mathbf{x}_1 \mathbf{x}_2 \ldots \mathbf{x}_T$$

- Sprachsignale mit derselben Aussage können sehr unterschiedlich sein
- es variiert die Länge der Merkmalssequenz
- und die Werte der Merkmale an der Stelle t
- Merkmale stetig oder kontinuierlich (Gauß – Mischverteilung)
- Merkmalssequenz – HMM (Länge kann variieren)

6.5.3 Akustische Modelle für Wörter

Akustisches Modell

akustisches Modell $P(\mathbf{X} \mid v_j)$

Merkmalssequenz $\mathbf{X} = \mathbf{x}_1 \mathbf{x}_2 \ldots \mathbf{x}_T$

Wortform

Wörter des Erkennervokabulars $V = \{v_1, v_2, \ldots, v_{|V|}\}$

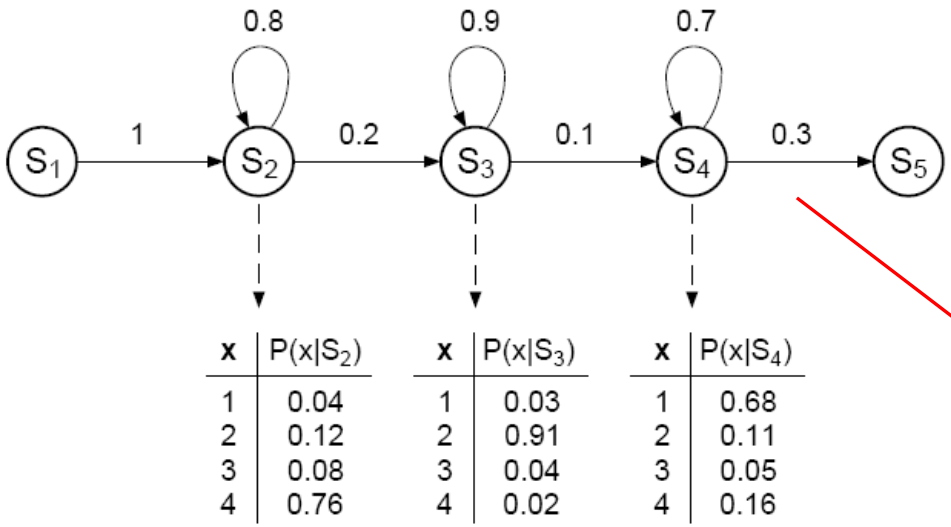
HMM als Wortmodell

$$V = \{v_1, v_2, \ldots, v_{|V|}\}$$

$$v_1 \rightarrow \lambda_1$$
$$v_2 \rightarrow \lambda_2$$
$$\ldots$$
$$v_{|V|} \rightarrow \lambda_{|V|}$$

Für jedes Wort (Wortform) wird ein zugehöriges HMM ermittelt.

Topologie der HMM



x	P(x|S2)
1	0.04
2	0.12
3	0.08
4	0.76

x	P(x|S3)
1	0.03
2	0.91
3	0.04
4	0.02

x	P(x|S4)
1	0.68
2	0.11
3	0.05
4	0.16

lineare HMM

Anzahl der Zustände: $N_{\max} = 5n_L + 2$

$N_{\min} = n_L + 2$

Anzahl der Laute (Phoneme) des Wortes

2 oder 3 Zustände pro Laut sind günstig

Erzeugen von Wortmodellen

- für das Training müssen mehrere Merkmalssequenzen eingesetzt werden
- mind 10 Sprachdaten pro freien Parameter
- sprecherunabhängigkeit
  - 200 Sprecher
  - größere Sprachregion mit verschiedenen Dialekteinflüssen (5000 Sprecher)
- internationale Projekte beschäftigen sich mit dem Aufbau solcher Sprachdatenbanken für alle wichtigen Sprachen

Initial HMM

$$a_{ii} = 0.7 \qquad a_{ii+1} = 0.3$$

Beobachtungswahrscheinlichkeiten:

DDHMM - Gleichverteilung

Bei CDHMM wird über alle Trainingsdaten, die für das Training des Wortmodells eingesetzt werden sollen, der Mittelwertvektor und die Kovarianzmatrix ermittelt und diese Werte als Initialwerte für die Beobachtungswahrscheinlichkeitsverteilungen aller Zustände verwendet. Das Initial-CDHMM hat demzufolge nur eine Mischkomponente.

Nicht zum Vokabular des Erkenners gehörige Wörter

- Es ist wichtig zu erkennen, ob die Eingabe des Benutzers zum Vokabular des Erkenners gehört.
- Sonst wird die Eingabe dem ähnlichsten Wort zugeordnet.

Mindestwert für: $P(\mathbf{X} \mid \lambda_i)$ nicht möglich

wird mit zunehmender Länge von **X** kleiner

Mindestwert für: $P(\lambda_i \mid \mathbf{X})$

$$P(W \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid W)P(W)}{P(\mathbf{X})}$$

Berechnung schwierig wegen P(X)

Rückweisungsmodell

Rückweisungsmodell: λ_R

$$P(\mathbf{X} \mid \lambda_R) \leq P(\mathbf{X} \mid \lambda_w) \quad w \in V$$

$$P(\mathbf{X} \mid \lambda_R) \geq P(\mathbf{X} \mid \lambda_w) \quad w \notin V$$

Das Rückweisungsmodell soll also genau dann die höchste Produktionswahrscheinlichkeit liefern, wenn die Merkmalssequenz **X** von einem Wort stammt, das nicht zum Vokabular *V* gehört.

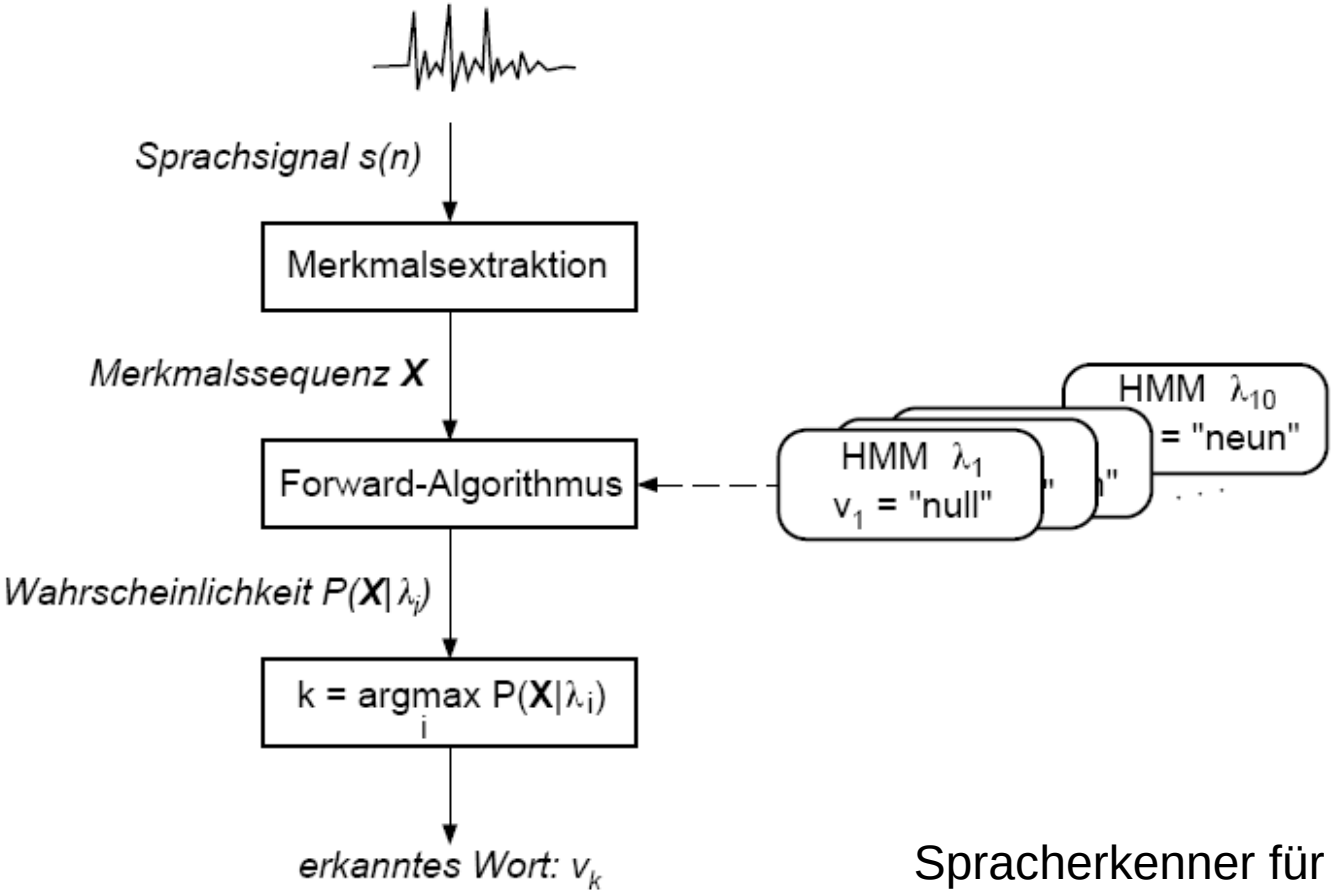
Ein einfaches Vorgehen ist, das Rückweisungsmodell mit den Trainingsdaten aller Wörter des Vokabulars zu trainieren. Dadurch entsteht ein Modell mit sehr flachen Beobachtungswahrscheinlichkeitsverteilungen.

Modelle für Geräusche und Pausen

- Hier sind ebenfalls Modelle für Geräusche (Hintergrund, Räuspern, Atmen, …) erforderlich.
- Gewöhnlich werden für die Modellierung von Geräuschen vollverbundene HMM eingesetzt, weil die zeitliche Abfolge der aus den Geräuschsignalen extrahierten Merkmale in diesem Zusammenhang nicht interessiert.
- Training kann analog zum Wort-HMM erfolgen (typische Geräuschsignale sind vorhanden)
- In der Spracherkennung ist oft auch ein Pausenmodell nützlich.
- Während Pausen sind aber stets noch mehr oder weniger leise Hintergrundgeräusche vorhanden.
- Ein solches Pausenmodell hat die gleiche Topologie wie ein Geräuschmodell, es wird jedoch mit Signalen von Pausen trainiert.

6.5.4 Spracherkennung mit Wort – HMM

Einzelworterkennung



Spracherkenner für einzeln gesprochene deutsche Ziffern

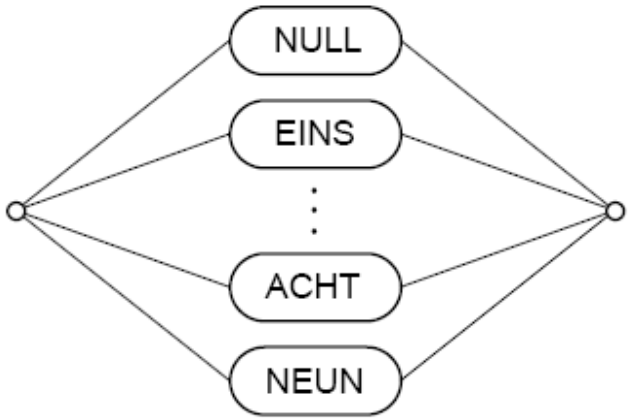
Vergleich mit MAP – Regel

$$\hat{W} = \underset{W \in V^*}{\operatorname{argmax}} P(\mathbf{X} \mid W) \cdot P(W)$$

Annahme: alle Ziffern gleichwahrscheinlich

$$k = \underset{i}{\operatorname{argmax}} P(\mathbf{X} \mid \lambda_i)$$

Worterkenner mit Erkennungsnetzwerk

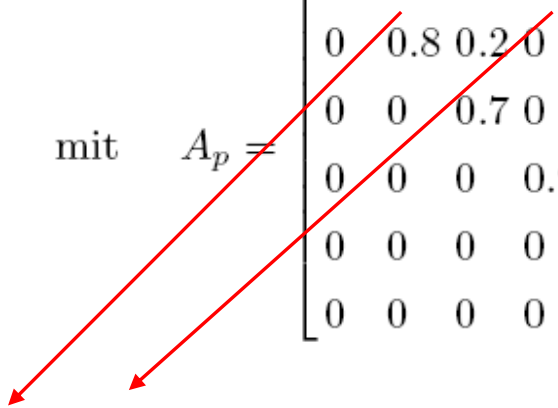


Dabei werden die Anfangszustände der Ziffern-HMM zu einem gemeinsamen Anfangszustand zusammengefasst und gleichermaßen auch die Endzustände. Wir bezeichnen dies als Parallelschaltung der HMM.

Parallelschaltung – Beispiel

$$\lambda_1 = (A_1, B_1) \quad \text{mit} \quad A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0.8 & 0.2 & 0 \\ 0 & 0 & 0.7 & 0.3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \lambda_2 = (A_2, B_2) \quad \text{mit} \quad A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0.9 & 0.1 & 0 \\ 0 & 0 & 0.6 & 0.4 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\lambda_p = (A_p, B_p) \quad \text{mit} \quad A_p = \begin{bmatrix} 0 & 0.5 & 0 & 0.5 & 0 & 0 \\ 0 & 0.8 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0 & 0 & 0.3 \\ 0 & 0 & 0 & 0.9 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.6 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{und} \quad B_p = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}.$$



$$P(\lambda_1) = P(\lambda_2) = 0.5$$

Erkennungsnetzwerk

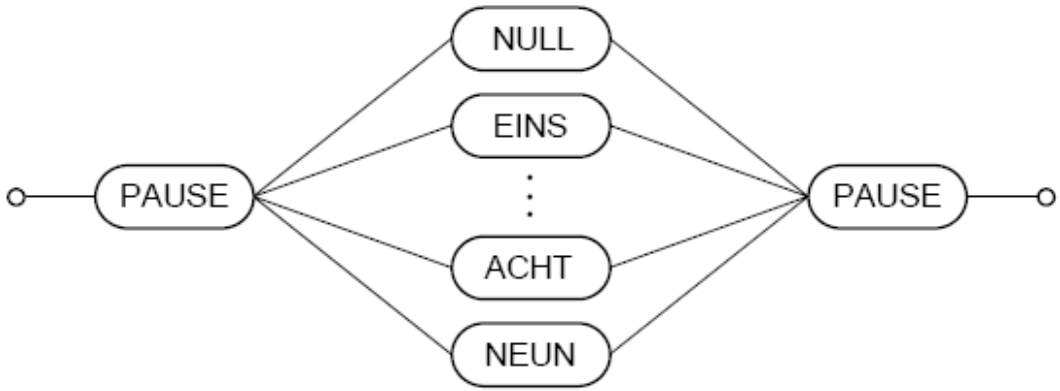
- Das Erkennungsnetzwerk ist also nichts anderes als ein HMM für die ganze Erkennungsaufgabe.
- Wir bezeichnen es mit λ_{net}.
- Mit dem Viterbi-Algorithmus kann sodann für die gegebene Merkmalssequenz **X** die optimale Zustandssequenz s* im HMM λ_{net} ermittelt werden.
- Da bekannt ist, zu welchen Ziffern-HMM die einzelnen Zustände des HMM λ_{net} gehören, kann aus der optimalen Zustandssequenz auf die erkannte Ziffer geschlossen werden.
- Logischerweise kann bei der Spracherkennung mit einem Erkennungsnetzwerk nur der Viterbi-Algorithmus eingesetzt werden
- Mit dem Forward-Algorithmus könnte man die Produktionswahrscheinlichkeit $P(\mathbf{X}|\lambda_{net})$ ermitteln, diese erlaubt jedoch keinerlei Rückschlüsse auf das geäußerte Wort.

Serienschaltung

$$\lambda_s = (A_s, B_s) \quad \text{mit} \quad A_s = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0.8 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 & 0 & 0 \\ 0 & 0 & 0 & 0.9 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.6 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{und} \quad B_s = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

Zu bemerken ist, dass bei der Serienschaltung der Endzustand von λ_1 und der Anfangszustand von λ_2 wegfallen.

Pausen



Mit den Pausen-HMM kann der Erkenner gleichzeitig auch den Anfangs- und den Endpunkt der Ziffer im Signal detektieren.

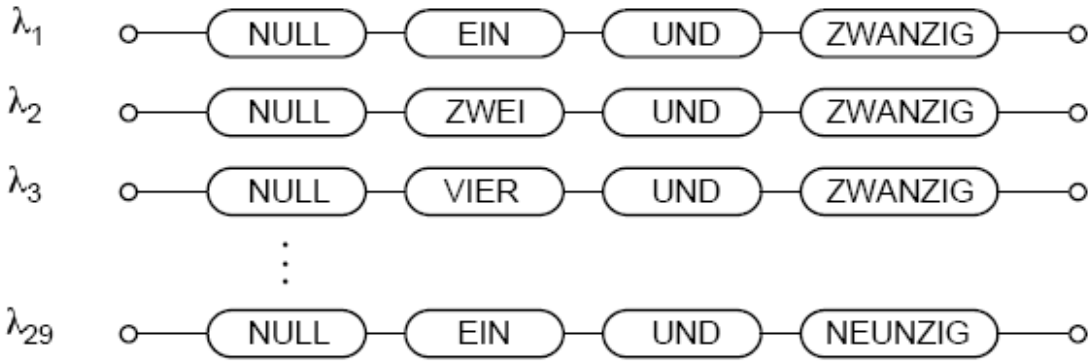
Schlüsselworterkennung

- Ein Erkenner für in Äußerungen eingebettete Schlüsselwörter (Keyword-Spotter) ist ähnlich aufgebaut wie der Ziffernerkenner.
- Statt der Pausenmodelle sind jedoch Modelle für beliebige (und insbesondere beliebig lange) Sprachsignale vorhanden.

Verbundworterkennung

- Um nicht bloß einzeln gesprochene Ziffern erkennen zu können, sondern auch Folgen von Ziffern, z.B. Kreditkartennummern oder Telefonnummern, kann man das Konzept der Erkennungsnetzwerke erweitern.

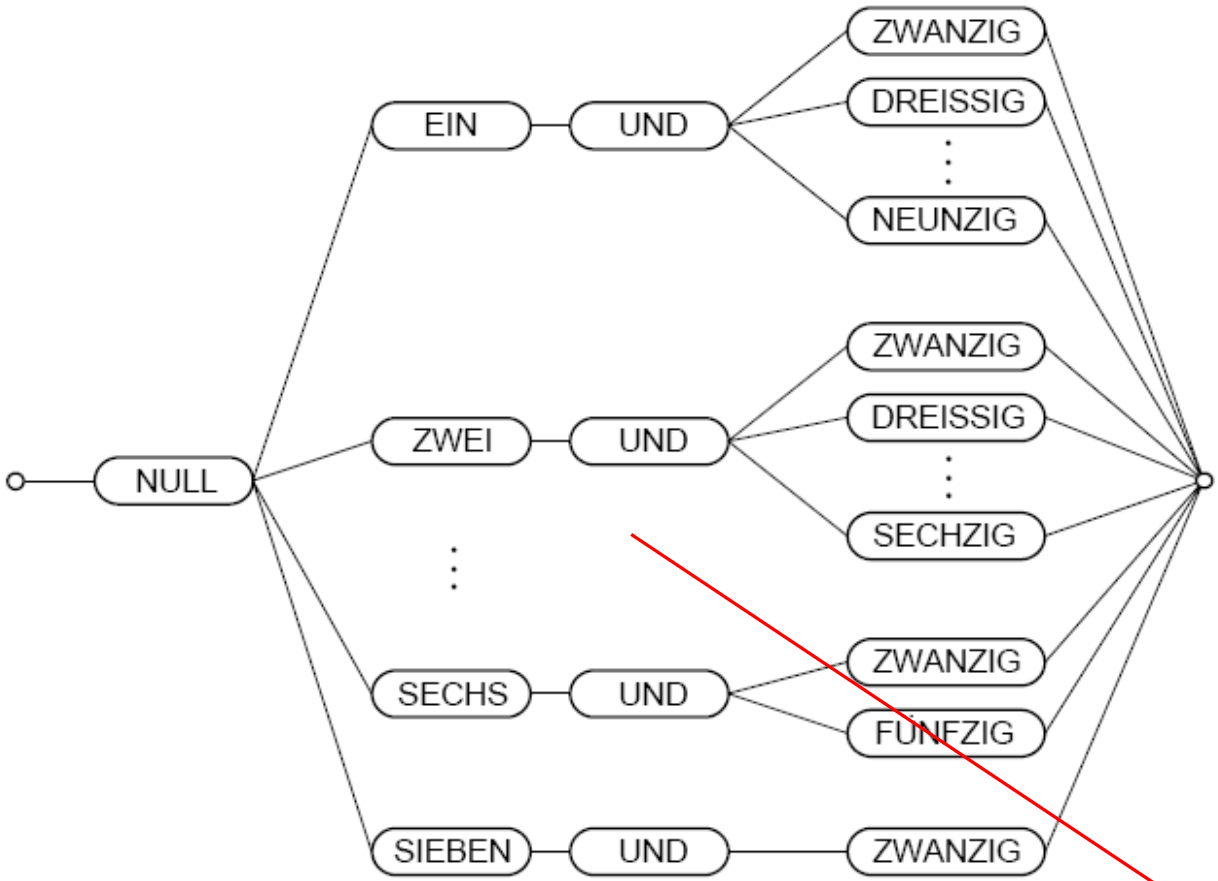
Erkennung von Vorwahlnummern



Für jede Nummer wird ein separates Verbund-HMM verwendet, das aus der Serienschaltung der betreffenden Wort-HMM entsteht.

Während dieses Vorgehen für Vorwahlnummern noch praktikabel ist, ist es für ganze Telefonnummern oder sogar für Kreditkartennummern ausgeschlossen.

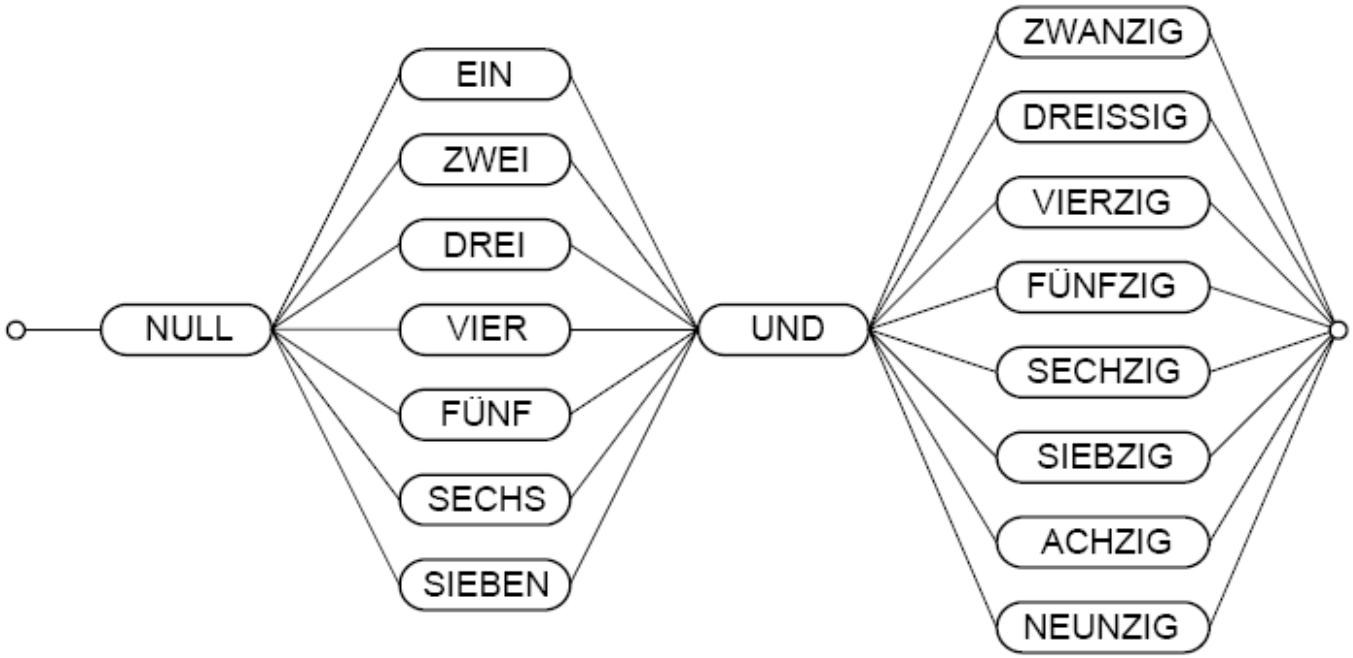
Erkennungsnetzwerk



Redundanz

Erkennen mit Viterbi – Algorithmus

Erkennungsnetzwerk



Mit diesem Erkennungsnetzwerk können nun jedoch Wortfolgen erkannt werden, die eigentlich nicht erlaubt sind, so zum Beispiel die in der Schweiz ungültige Vorwahlnummer 023.

Erkennung mit dem N-best Viterbi – Algorithmus

- In manchen Anwendungen kann das gesamte Wissen nicht zweckmäßig in das Erkennungsnetzwerk eingebaut werden.
- In solchen Fällen ist es erwünscht, dass der Spracherkenner nicht nur eine einzige Wortfolge ausgibt, sondern die *N* wahrscheinlichsten.
- Mit dem zusätzlichen, im Erkennungsnetz nicht berücksichtigten Wissen wird dann nachträglich entschieden, welche dieser *N* wahrscheinlichsten Lösungen die richtige ist.

Erkennung mit dem N-best Viterbi – Algorithmus

- Die *N* wahrscheinlichsten Lösungen erhält man mit dem N-best Viterbi – Algorithmus.
- Dies ist eine Erweiterung des gewöhnlichen Viterbi – Algorithmus so, dass am Ende nicht nur eine optimale Zustandssequenz resultiert, sondern *N*, mit der Nebenbedingung, dass alle Zustandssequenzen auch einer andern Wortfolge entsprechen müssen.

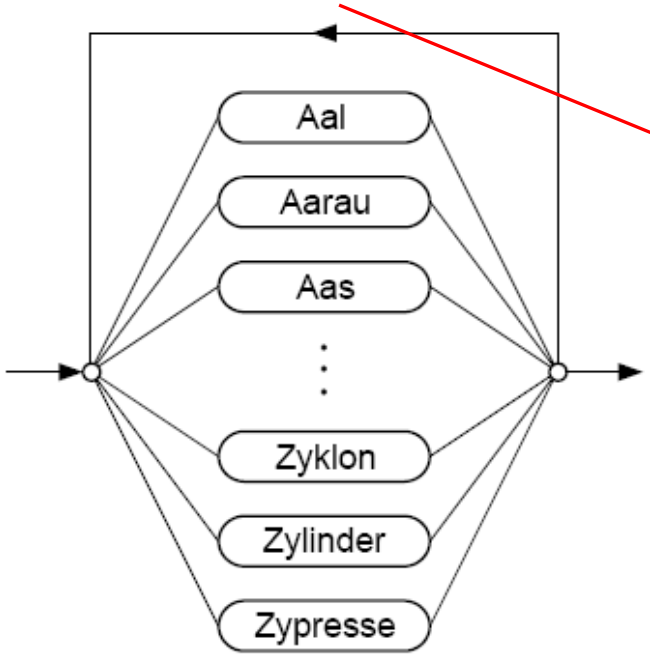
Anwendung

- Der Einsatz des N-best Viterbi – Algorithmus kann beispielsweise beim Erkennen von Kreditkartennummern sinnvoll sein.
- Von den 10^{16} möglichen Kombinationen der 16 Ziffern sind nur ein paar Millionen gültig.
- Es macht jedoch wenig Sinn, alle gültigen Kreditkartennummern explizit durch ein eigenes Verbund-HMM zu beschreiben.
- Dies wäre zu aufwändig, und die Liste müsste zudem für jede neue Kreditkartennummer erweitert werden.

Erkennung kontinuierlicher Sprache

- Das Ziel bei der Erkennung kontinuierlich gesprochener Sprache ist es, für eine gegebene Merkmalssequenz die wahrscheinlichste Wortfolge zu finden.
- Selbstverständlich können nicht alle möglichen Wortfolgen *W* aufgezählt werden.
- Da es unschön wäre, die maximale Länge der erlaubten Wortsequenzen von vornherein zu beschränken, ist diese Vorgehensweise für kontinuierliche Spracherkennung nicht praktikabel.

Erkennung kontinuierlicher Sprache



Der Zyklus im Erkennungsnetzwerk ermöglicht eine kompakte Darstellung aller möglichen Wortfolgen für das gegebene Vokabular.

Ein solches Netzwerk könnte beispielsweise in einem Diktiersystem verwendet werden.

Erkennung kontinuierlicher Sprache

- Die Erkennung kontinuierlicher Sprache mit Wort-HMM ist nur bei einem Vokabular mit relativ bescheidenem Umfang praktikabel.
- Für Sprachen mit vielen Wortformen ist diese Grenze relativ bald erreicht.
- Man setzt deshalb Wortteil-HMM ein, wie sie im Folgenden besprochen werden.

6.5.5 Akustische Modelle für Wortteile

Einführung

- Für Spracherkenner mit großem Vokabular ist der Einsatz von Wort-HMM unpraktisch, besonders dann, wenn der Spracherkenner sprecherunabhängig sein soll.
- Alle Wörter müssen von einigen hundert bis mehreren tausend Personen aufgenommen werden, was sehr aufwändig ist.
- Zudem wäre es praktisch unmöglich, das Vokabular eines solchen Spracherkenners zu erweitern.
- Deshalb werden für Spracherkenner mit großem Vokabular nicht Wort-HMM eingesetzt, sondern HMM für kleinere linguistische Einheiten.

Wahl der Grundelemente (Wortteile)

- Die Grundelemente können in zwei Gruppen eingeteilt werden:
- kontextunabhängige
  - für jedes Grundelement wird nur ein Modell verwendet, es wird also nicht berücksichtigt, dass ein Grundelement von verschiedenen Nachbarlauten unterschiedlich beeinflusst wird
- kontextabhängig
  - hier wird die Nachbarschaft berücksichtigt, indem für jeden auftretenden lautlichen Kontext ein separates Grundelementmodell trainiert wird

Kontextunabhängige Grundelemente

- Wörter
  - schlechte Trainierbarkeit, da für jedes Wort genügend Trainingsbeispiele vorhanden sein müssen
  - für große oder sogar offene Vokabulare sind deshalb Wörter als Grundelemente unbrauchbar
- Laute
  - für die Spracherkennung werden im Deutschen etwa 40–60 Laute unterschieden
  - ihre Präzision ist jedoch sehr begrenzt, weil benachbarte Laute sich gegenseitig beeinflussen

Kontextabhängige Grundelemente

- Laute werden mehr oder weniger stark durch ihre Nachbarlaute mitgeprägt
- Triphone
  - Lautmodell, das nur je einen Laut auf der linken und rechten Seite als Kontext mitberücksichtigt
  - für einen bestimmten Kernlaut werden so viele Triphon-Modelle angesetzt wie verschiedene Kombinationen von linkem und rechtem Kontextlaut zu diesem Kernlaut existieren
  - auch wenn bei weitem nicht alle möglichen Triphone wirklich in der Sprache vorkommen, so ist die Zahl der zu modellierenden Grundelemente doch sehr hoch
- Generalisierte Triphone
  - da viele Triphone selten sind, findet man auch in einem sehr großen Trainingsset für viele Triphone zu wenig oder überhaupt keine Daten
  - unter der Annahme, dass ein Laut in ähnlichen Kontexten auch ähnlich realisiert wird, können Kontexte in gröbere Klassen zusammengefasst werden

Erzeugen von Grundelementmodellen

- Aufnahme von Sprachsignalen
  - hier werden nicht nur einzelne Wörter aufgenommen, sondern auch ganze Sätze
- Training
  - Wir haben die Wort-HMM voneinander unabhängig trainiert.
  - Im Gegensatz dazu werden für die kontinuierliche Spracherkennung ganze Sätze für das Training aufgenommen.
  - In den Trainingsdaten sind aber die Segmentgrenzen der Grundelemente nach wie vor unbekannt, und eine manuelle Segmentierung der umfangreichen Trainingsdaten wäre viel zu aufwändig.
  - die Trainingsdaten sollen möglichst automatisch in die gewünschten Grundelemente segmentiert werden, ausgehend von den Sprachsignalen und der phonetischen Transkription.
  - Dazu existiert ein Verfahren, welches die Sprachsignale segmentiert und gleichzeitig alle Grundelemente trainiert.
  - Dieses Verfahren heißt ***Embedded Training.***

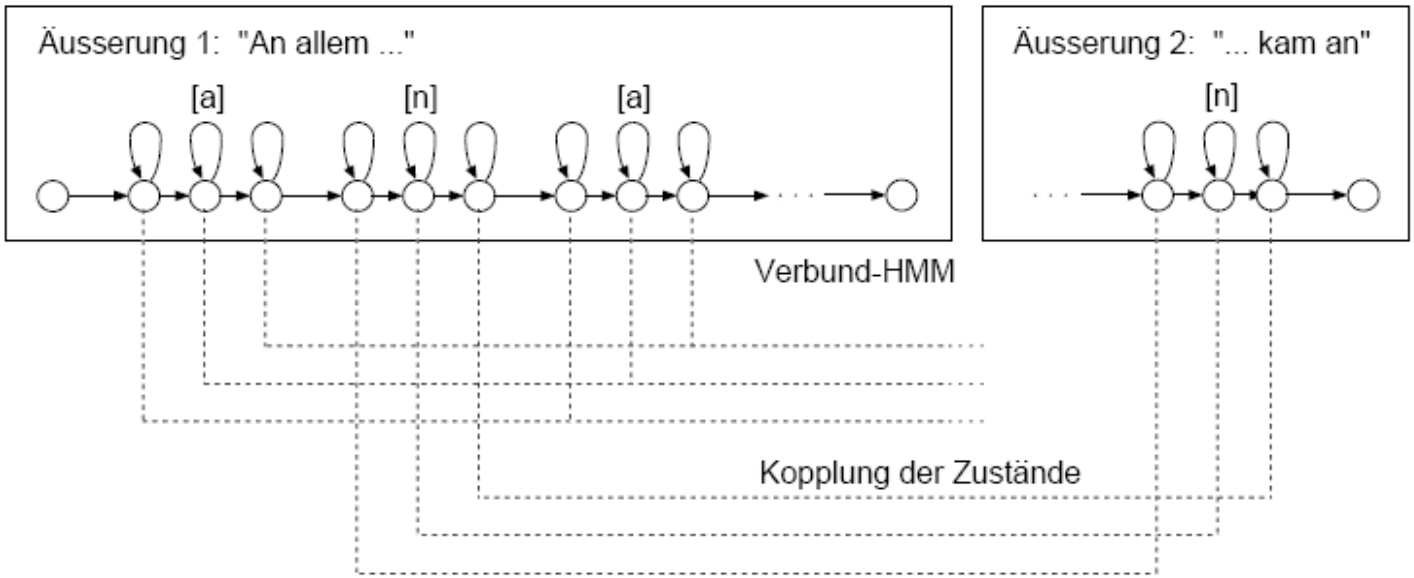
Initial – HMM für Grundelemente

- Die Topologie und die Anzahl der Zustände *N* muss festgelegt werden.
- lineare HMM
- 2 bis 5 Zustände pro Laut
- häufig drei emittierende Zustände
- Initialwerte der Beobachtungswahrscheinlichkeitsverteilungen
- je nach vorhandenen Daten unterschiedlich
- segmentierte Daten
  - durch Experten segmentiert (Lautgrenzen)
  - Segmentgrenzen innerhalb des Lautes (Merkmalssequenz eines Lautes wird gleichmäßig auf die entsprechenden Zustände aufgeteilt
  - nur je eine Mischkomponente
- nichtsegmentierte Daten (2 Möglichkeiten)
  - bei der ersten werden die Merkmalssequenzen gleichförmig segmentiert
  - Die zweite Möglichkeit ist, für alle Zustände die gleichen Initialwerte der Beobachtungswahrscheinlichkeitsverteilungen zu verwenden, nämlich der Mittelwertvektor und die Kovarianzmatrix über alle Trainingsdaten.

Embedded Training

- Für jede Trainingsäußerung, z.B. ein Satz oder ein Wort, wird entsprechend der gegebenen phonetischen Transkription aus Grundelement-HMM ein Verbund-HMM für die gesamte Äußerung zusammengesetzt.
- Zudem werden alle Zustände, die zum gleichen Grundelementzustand gehören, innerhalb und zwischen den Verbund-HMM gekoppelt.
- Wenn Zustände gekoppelt sind, so bedeutet das, dass die Parameter aller dieser Zustände gleich sind.
- Die Grundelemente werden gleichzeitig trainiert, indem in jeder Iteration über alle Äußerungen summiert wird.

Embedded Training



6.5.6 Modelle für verschiedene akustische Ereignisse

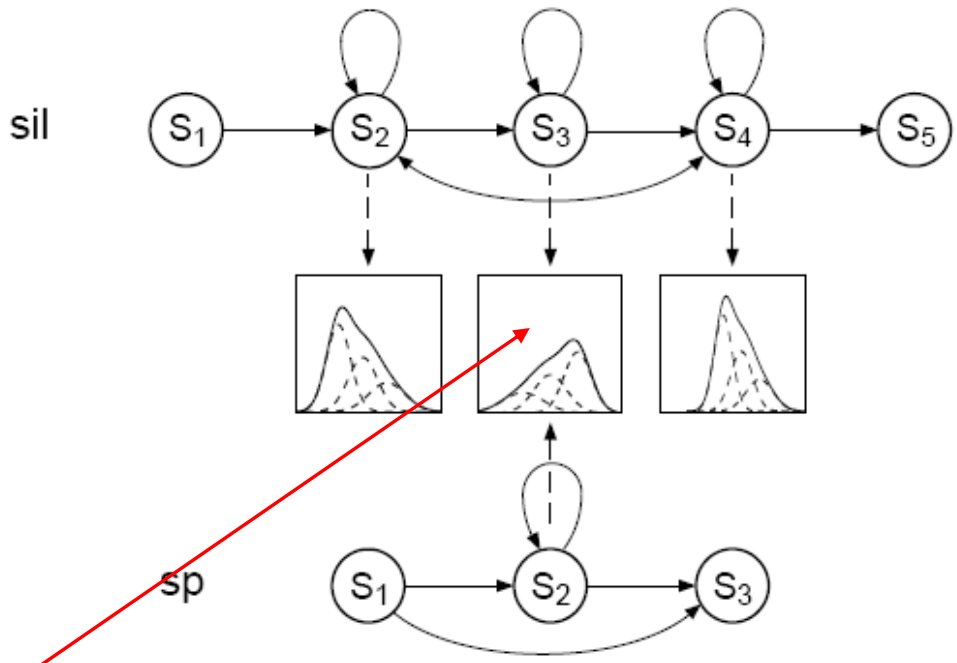
Einführung

- ein einsatztauglicher Spracherkenner muss nicht nur die zum Erkennervokabular gehörigen Wortformen erkennen
- Er sollte auch erkennen, wenn im Sprachsignal eine Wortform vorliegt, die nicht zum Erkennervokabular gehört oder überhaupt keine Sprache vorliegt.
- Dazu sollten entsprechende akustische Modelle vorhanden sein.
- Man benötigt also ein Pausenmodell, ein Geräuschmodell und ein Rückweisungsmodell für nicht zum Erkennervokabular gehörende Wörter, bzw. ein Modell für beliebige Sprache.

Modelle für Pausen

- sil – Modell (silence)
  - beschreibt beliebig lange Pausen vor oder nach der zu erkennenden Äußerung
  - drei emittierende Zustände
- sp – Modell (short pause)
  - beschreibt kurze optionale Sprechpausen zwischen den Wörtern
  - einen emittierenden Zustand
  - Weil Sprechpausen zwischen zwei Wörtern optional sind, hat das Modell zudem einen Zustandsübergang vom Start- in den Endzustand.
  - Es kann also direkt in den Endzustand übergehen, ohne eine Beobachtung zu erzeugen.
  - Das Modell wird bei der Erkennung jeweils zwischen zwei Wörtern eingefügt.

Modelle für Pausen



Die Beobachtungswahrscheinlichkeitsverteilungen
der mittleren Zustände der beiden Modelle sind gekoppelt.

Modelle für Geräusche

- hier werden die Geräuschmodelle nicht separat trainiert, sondern wie alle andern HMM ins Embedded Training einbezogen
- Dies geschieht, indem im Verbund-HMM einer Äußerung für die annotierten Geräusche die betreffenden Geräuschmodelle eingesetzt und mit den entsprechenden Geräuschmodellen in den andern Verbund-HMM gekoppelt werden.

Modell für beliebige Sprachsignale

- Rückweisungsmodelle (wie bei Wort-HMM)
- kleinere Grundelemente bieten jedoch eine weitere Möglichkeit, ein Rückweisungsmodell zu konstruieren
- es können Lautmodelle zu einem sogenannten *Phone-Loop* zusammengehängt werden
- dies ist ein Erkennungsnetzwerk mit Laut-HMM, in welchem auf jeden Laut jeder andere Laut folgen kann
- Dieses Rückweisungsmodell beschreibt also beliebig lange Sprachereignisse.
- Da die Produktionswahrscheinlichkeit des Phone-Loops aber immer mindestens gleich groß ist wie diejenige des wahrscheinlichsten Vokabularwortes, muss noch eine Gewichtung eingesetzt werden.

6.5.7 Spracherkennung mit Laut-HMM

Einführung

- Vorteil dieser Spracherkenner ist, dass das Erkennervokabular fast beliebig und sehr einfach erweitert werden kann.
- Um ein Verbund-HMM für ein neues Wort aus Laut-HMM zusammensetzen zu können, braucht man nur die phonetische Umschrift des Wortes zu kennen.
- Für Spracherkenner mit einem Vokabular, das mehr als ein paar hundert Wörter umfasst, ist der Einsatz von Laut-HMM deshalb äusserst vorteilhaft, insbesondere wenn der Spracherkenner dazu noch sprecherunabhängig sein soll.

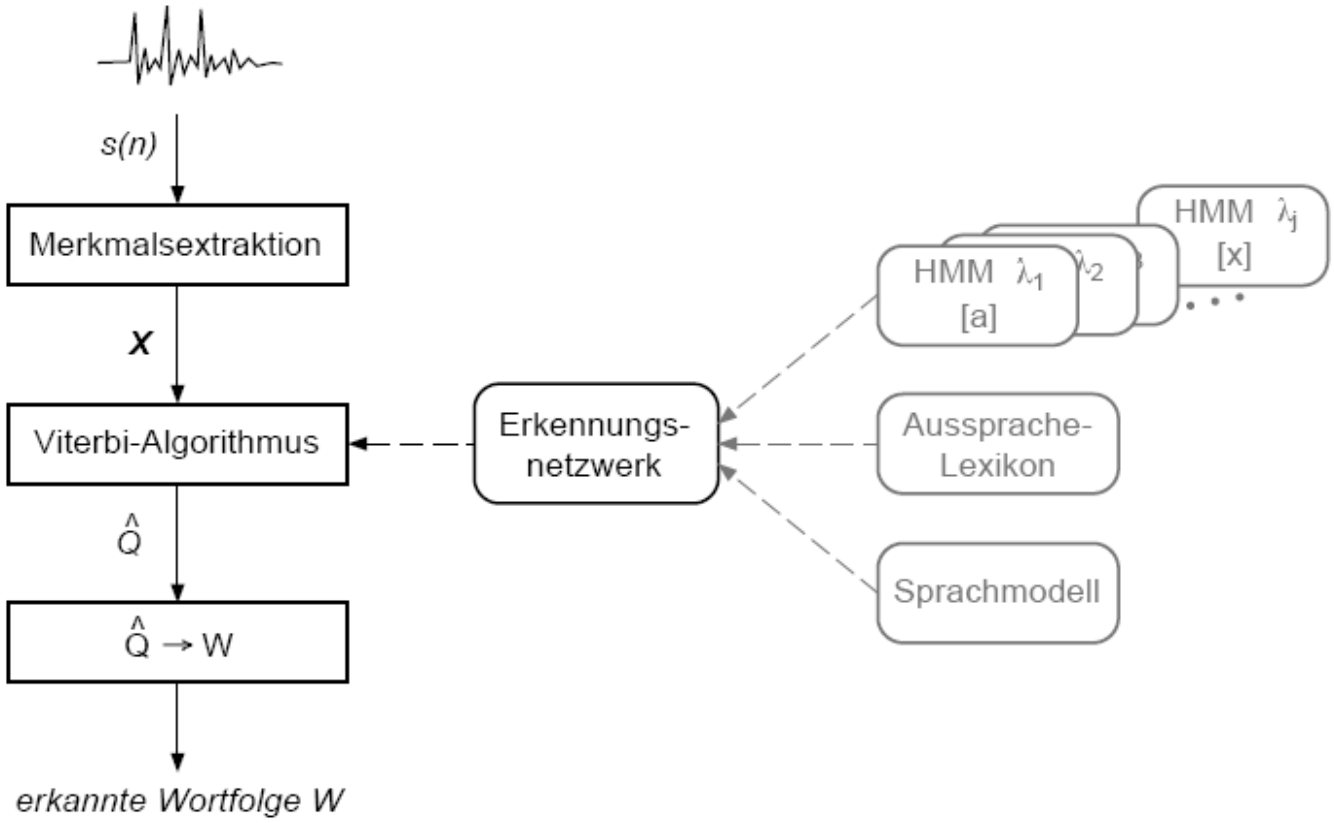
Erkennung einzeln gesprochener Wörter

- Ein auf Laut-HMM basierender Spracherkenner für die Erkennung einzeln gesprochener Wörter kann grundsätzlich so aufgebaut werden wie der Einzelworterkenner bei Wort-HMM.
- Dazu ist für jedes Wort des Erkennervokabulars ein entsprechendes Verbund-HMM zusammenzusetzen.
- auch Erkennungsnetzwerke und für die Erkennung der Viterbi-Algorithmus können verwendet werden

Erkennung kontinuierlicher Sprache

- Das Zusammenhängen von HMM zu Erkennungsnetzwerken ist nicht nur mit Wort-HMM möglich, sondern mit beliebigen Grundelement-HMM, insbesondere auch mit Laut-HMM.
- Um aus den Laut-HMM ein Erkennungsnetzwerk zusammenzustellen, benötigt der Spracherkenner die beiden folgenden Komponenten:
  - Ein Aussprache-Lexikon, in welchem alle zu erkennenden Wörter zusammen mit ihrer Aussprache (d.h. mit der phonetischen Umschrift) verzeichnet sind. So lässt sich für jedes Wort ein Verbund-HMM zusammenstellen, indem die der phonetischen Beschreibung entsprechenden Laut-HMM in Serie geschaltet werden.
  - Ein Sprachmodell (siehe Kapitel 7), das berücksichtigt, dass in einer gegebenen Anwendung nicht alle Wortfolgen gleich plausibel sind.
- Da das Erkennungsnetzwerk im Prinzip ja nichts anderes als ein sehr großes HMM ist (d.h. ein HMM mit sehr vielen Zuständen), kann für die eigentliche Erkennung wiederum der Viterbi-Algorithmus eingesetzt werden.
- Die resultierende optimale Zustandssequenz beschreibt nun einen Pfad durch das Netzwerk.
- Es gelten diejenigen Wörter als erkannt, deren HMM im Netzwerk nacheinander vom optimalen Pfad durchlaufen werden.

Erkennung kontinuierlicher Sprache



Reduktion des Rechenaufwands

- Je nach Art der Erkennungsaufgabe kann mit der Organisation des Erkennungsnetzwerks die Zahl der möglichen Hypothesen unterschiedlich stark eingeschränkt werden.
- Dies reicht aber meistens noch nicht aus.
- Um den Rechenaufwand auf ein realistisches Maß zu reduzieren, braucht es zusätzlich zu dieser Maßnahme ein Verfahren, um den Suchraum dynamisch während der Decodierung einzuschränken.
- Die Idee besteht darin, im Viterbi-Algorithmus wenig aussichtsreiche Hypothesen frühzeitig zu verwerfen.
- Dabei werden Teilpfade, deren Bewertung unter einen dynamischen Schwellwert fallen, nicht mehr weiter berücksichtigt, sondern abgeschnitten, was als *Pruning* bezeichnet wird.

6.5.8 Stärken und Schwächen von HMM

Stärken

- HMM erweisen sich als gute Modellierung von Sprachsignalen
- Existenz von relativ einfachen Algorithmen für das Training dieser Modelle
- Die Flexibilität in der Modellierung von Sprachsignalen mit HMM besteht auch darin, dass die Modelltopologie und Beobachtungswahrscheinlichkeitsverteilungen so gewählt werden können, dass sie dem Erkennungsproblem am besten angepasst sind.
- Beispielsweise können fast beliebige Parameter der HMM gekoppelt werden und Modelle für größere Einheiten können aus den HMM kleinerer Einheiten zusammengesetzt werden.
- Diese Möglichkeiten erlauben es dem Systementwickler, Vorwissen auf verschiedenen linguistischen Ebenen in die Modellierung mit einzubeziehen.
- Gleichzeitig zeigt sich aber hier auch die Grenze der HMM für praktische Anwendungen.

Schwächen

- Da die wahren Wahrscheinlichkeitsverteilungen $P(W)$ und $P(\mathbf{X}|W)$ durch Modelle geschätzt werden müssen, die mit limitierten Daten trainiert worden sind, weichen diese Schätzungen stets mehr oder weniger stark von den wahren Verteilungen ab.
- Die Markov Annahmen und die in den Zustandsübergängen implizierte Lautdauermodellierung treffen bei Sprachsignalen nur schlecht zu.

Zusammenfassung

- Trotz dieser Vorbehalte haben sich die HMM in der Sprachverarbeitung weitgehend durchgesetzt.
- Der größte Teil der heutigen Spracherkennungssysteme basiert auf HMM.
- Daneben werden HMM auch in anderen Bereichen der Sprachverarbeitung (z.B. in der Sprechererkennung) und auch zur Lösung anderer Probleme (z.B. Handschrifterkennung) mit Erfolg eingesetzt.