Rheinisch Westfälische Technische Hochschule Aachen
Lehrstuhl für Informatik VI
Prof. Dr.-Ing. Hermann Ney

Seminar “Speech Recognition and Language Processing” im SS 2003

Diskriminatives Training

Lars Haferkamp

6. August 2003

Literatur

[1] W. Macherey. Implementierung und Vergleich diskriminativer Verfahren für Spracherkennung bei kleinem Vokabular. Master's thesis, RWTH Aachen, 1998.

[2] Y. Normandin. Maximum Mutual Information Estimation of Hidden Markov Models. In K. K. Paliwal C.H. Lee, F. K. Soong, editor, *Automatic Speech and Speaker Recognition*, pages 57–81. Kluwer Academic Publishers, Norwell, MA, 1996.

[3] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young. MMIE Training of Large Vocabulary Recognition Systems. In *Speech Communication*, volume 22, No. 4, pages 303–314, September 1997.

[4] P. C. Woodland and D. Povey. Large Scale Discriminative Training of Hidden Markov Models for Speech Recognition. In *Computer Speech and Language*, volume 16, No. 1, pages 25–47, Januar 2002.

Diskriminatives Training

- Einführung und Motivation
- Optimierung der Zielfunktion
  - Berechnung der speziellen Forward-Backward (FB)-Wahrscheinlichkeit
  - Berechnung der generalisierten FB-Wahrscheinlichkeit
  - Neuschätzen der Parameter
- Experimentelle Ergebnisse
- Zusammenfassung

Motivation

- Training = Schätzung der Parameter des akustischen Modells (HMM) mit Hilfe der Trainingsdaten.
- konventionelle Methode: Maximum-Likelihood (ML) Training.
- hier: Diskriminatives Training.

Notation

- (X_r, W_r) Trainingsäußerung, $r = 1$, ..., R.
- W_r gesprochene Wortfolge w_{r1}, ..., w_{rN_r}.
- X_r Folge von Beobachtungsvektoren x_{r1}, ..., x_{rT_r}.
- $P(W)$ Sprachmodell (a-priori) Wahrscheinlichkeit.
- $P(X|W)$ Emmissionswahrscheinlichkeit.
- $P(W|X)$ a-posteriori-Wahrscheinlichkeit.
- λ Menge der aktuellen Parameter.
- $\mathcal{M}_r$ Menge der zu W_r konkurrierenden Wortfolgen.
- α Wichtungsexponent, s Zustand, l Mischverteilungsindex.

Bayessche Entscheidungsregel

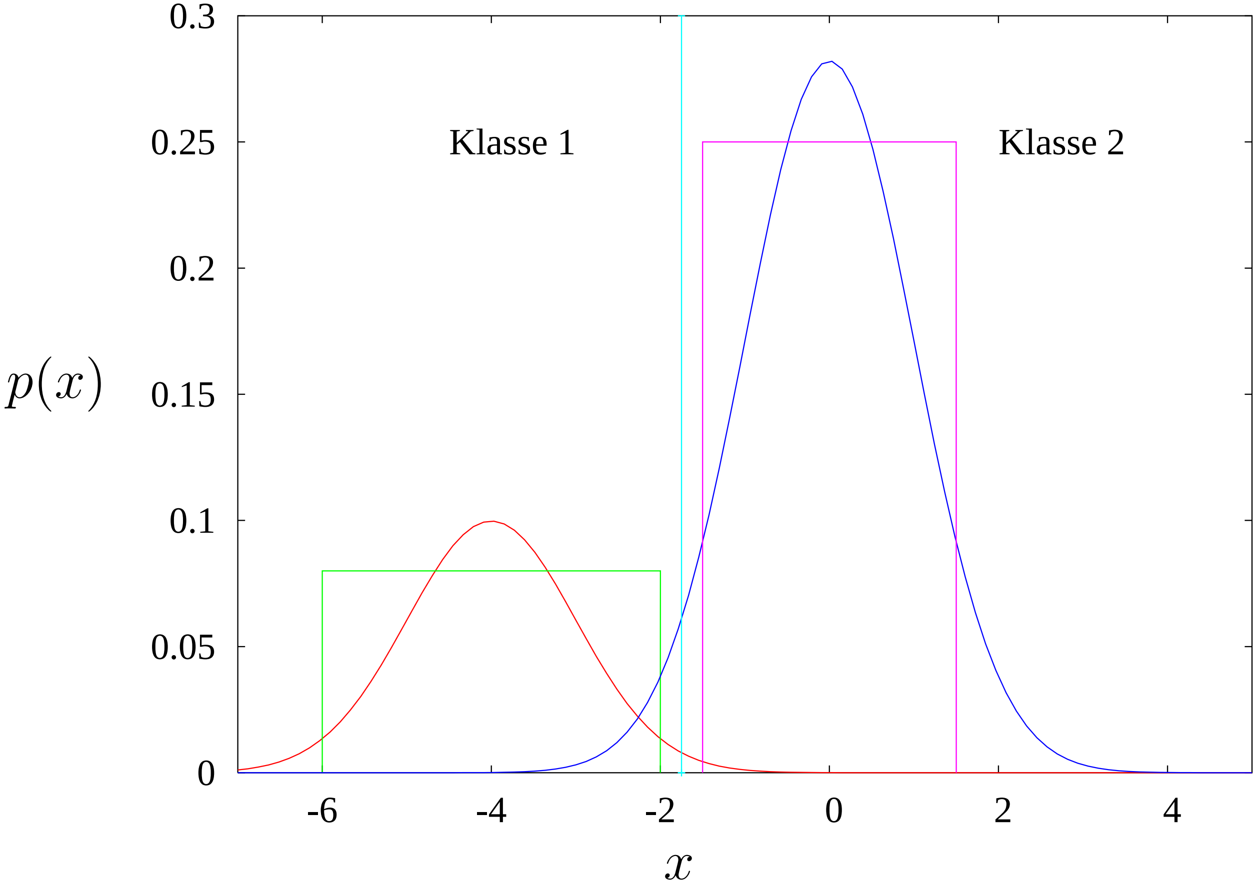
Die Bayessche Entscheidungsregel liefert uns

$$\begin{aligned} W_{optimal} &= \arg\max_W \; pr(W|X) \\ &= \arg\max_W \; [\, pr(X|W)\, pr(W) \;/\; pr(X)] \\ &\approx \arg\max_W \; [\, p_\lambda(X|W)\, p(W) \;/\; p_\lambda(X)] \end{aligned}$$

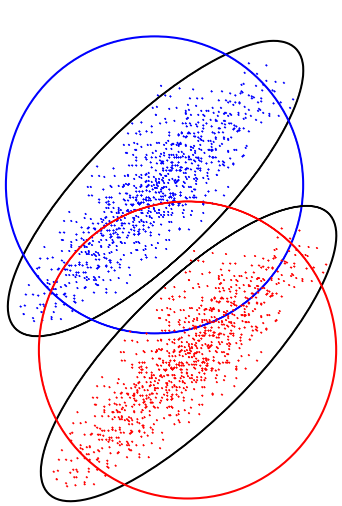
Gesucht: Schätzung der HMM-Parameter $\lambda \rightarrow$ Training.

Mögliche Ursachen für Erkennungsfehler

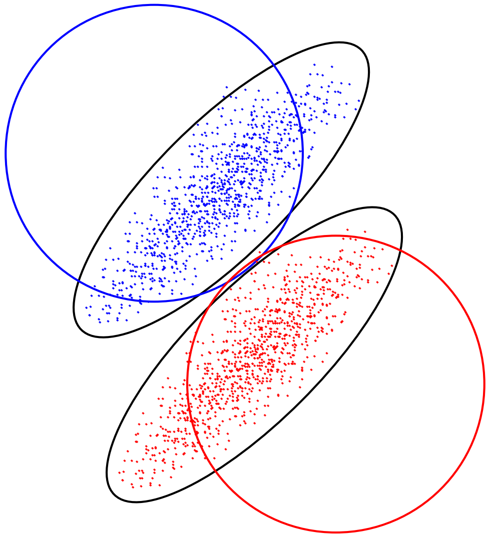
Verwendung einer „falschen“ Wahrscheinlichkeitsdichtefunktion:



Überlappungsproblem



Klassentrennung im ML-Training



Idee: Diskriminatives Training, Optimierung der Klassengrenzen.

ML Training

- **Maximiere:**

$$\mathcal{F}_{ML}(\lambda) = \sum_{r=1}^{R} \log p_\lambda(X_r|W_r),$$

- **Problem:**

Was ist $p_\lambda(X_r|W)$ für $W \neq W_r$?

$\rightarrow$ ML trainiert nur klassenindividuell.

Ableitung nach einem zustandsabhängigen Parameter $\theta_{s,l}$:

$$\partial F_{ML}(\lambda)/\partial\theta_{s,l} = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_{r,t}(s|W_r)\ \delta(l, l(x_{r,t}, s))\ \partial \log p(x_{r,t}|\theta_{s,l})/\partial\theta_{s,l}$$

EM-Algorithmus

1. Initialisierung (lineare Segmentierung,...)

2. ML-Training:

   Wiederhole bis Konvergenz der ML-Zielfunktion erreicht ist

   (a) Für jede Trainingsäußerung $r = 1, \cdots, R$ aus den Trainingsdaten

      i. Bestimme Anzahl der Zustände.

      ii. Forced Time Alignment der gesprochenen Wortfolge.

      Akkumuliere: $\sum_{t=1}^{T_r} \gamma_{r,t}(s|W)$,

      $\sum_{t=1}^{T_r} \gamma_{r,t}(s|W) x_{r,t}$,

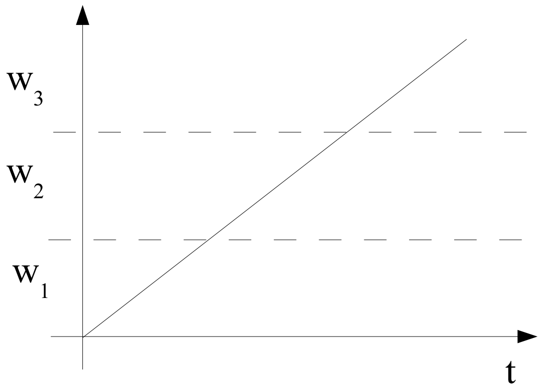
      $\sum_{t=1}^{T_r} \gamma_{r,t}(s|W) x_{r,t}^2$.

   (b) Schätze die Parameter neu ($\mu_{s,l},\ \sigma_{s,l}^2,\ c_{s,l}$).

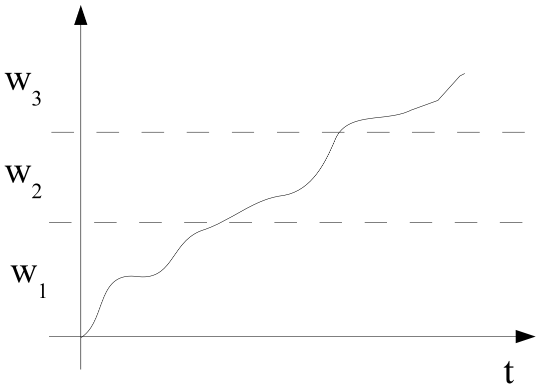
   (c) Splitten der Dichten (optional).

Forced Time Alignment

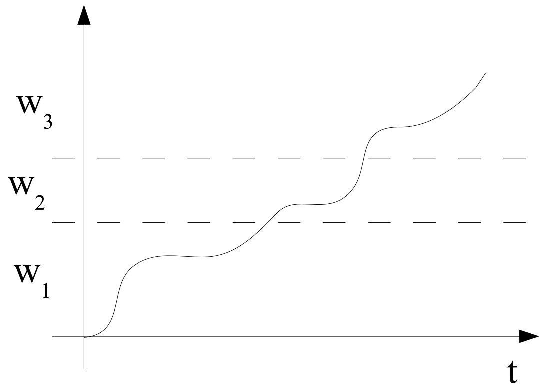
(für gegebene Wortfolge W)



1. Iteration
Lineare Segmentierung



2. Iteration



3. Iteration

Diskriminatives Training

- **Ziel:** So viele Trainingsäußerungen wie möglich richtig erkennen.
- **Maximiere:**

$$\begin{aligned} p_\lambda(W|X) &= \frac{p_\lambda(X|W)\; p(W)}{p_\lambda(X)} \\ \mathcal{F}_{MMI}(\lambda) &= \sum_{r=1}^{R} \log \frac{p_\lambda(X_r|W_r)\; p(W_r)}{\sum\limits_{W \in \mathcal{M}_r} p_\lambda(X_r|W)\; p(W)}. \end{aligned}$$

$\rightarrow$ **Vorteil:** klassenfremde Trainingsdaten werden berücksichtigt.

Diskriminative Trainingskriterien

Allgemeine Form der Zielfunktion:

$$F_D(\lambda) = \sum_{r=1}^{R} f\left(\log \frac{p_\lambda^\alpha(X_r|W)\ p(W)^\alpha}{\sum\limits_{W \in \mathcal{M}_r} p_\lambda^\alpha(X_r|W)\ p(W)^\alpha}\right)$$

diskrim. Kriterium	Glättungsfkt. $f(z)$	Wortfolgen in $\mathcal{M}_r$	Gewicht α
MMI	id	alle konk. Wortfolgen (*einschließlich* W_r)	1
MCE	$1/(1+e^{2\beta z})$	alle konk. Wortfolgen *ohne* W_r	≥ 1

Optimierung der MMI-Zielfunktion

Ableiten nach einem zustandsabhängigen Parameter ergibt:

$$\frac{\partial F_{MMI}(\lambda)}{\partial \theta_{s,l}} = \sum_{r=1}^{R} \sum_{t=1}^{T_r} [\gamma_{r,t}(s|W_r) - \gamma_{r,t}(s)] \cdot \delta(l, l(x_{r,t}, s)) \frac{\partial}{\partial \theta_{s,l}} \log p(x_{r,t}|\theta_{s,l})$$

$\gamma_{r,t}(s|W) := p_\lambda(s_t = s|X_r, W).$

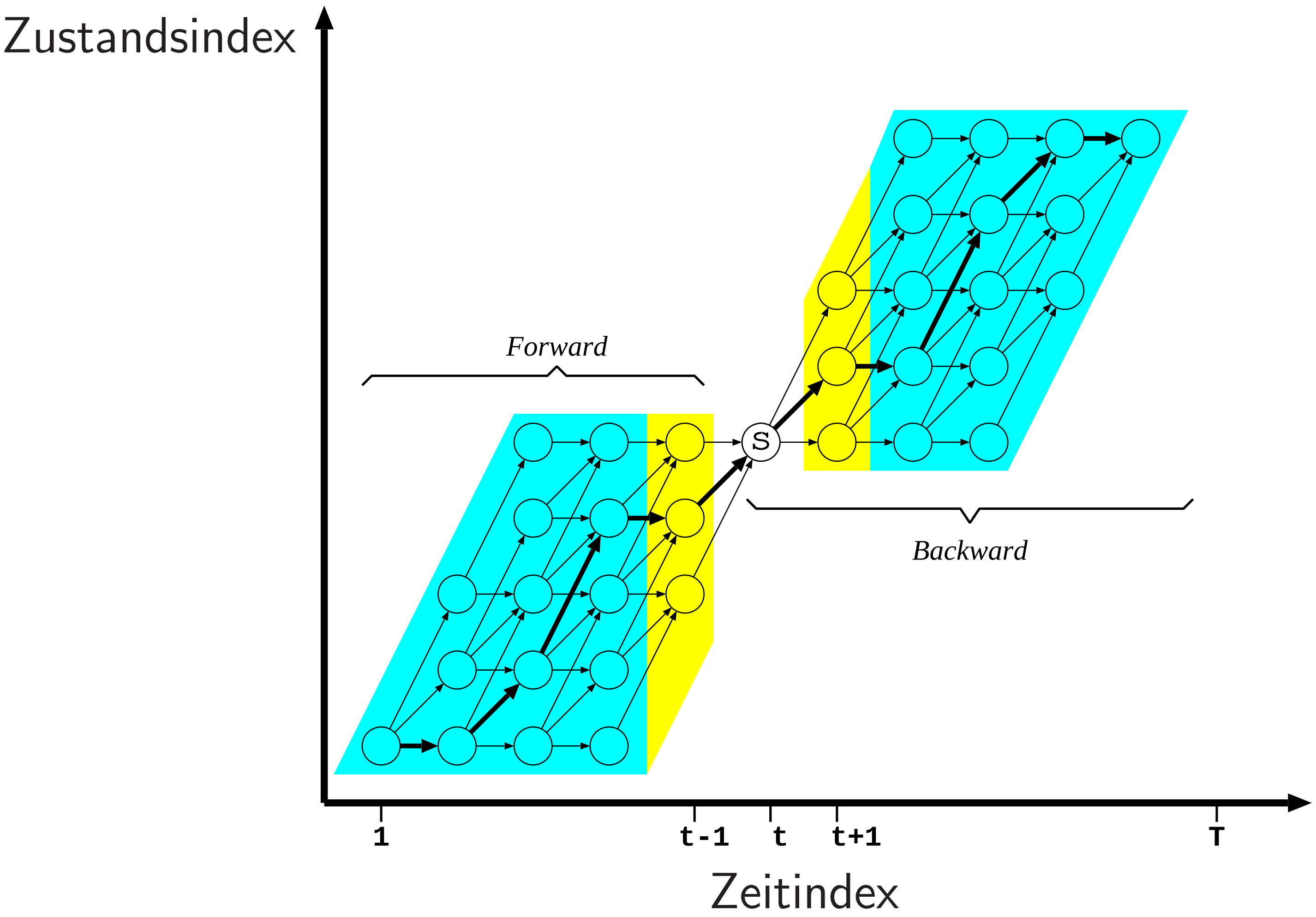
$\gamma_{r,t}(s) := p_\lambda(s_t = s|X_r).$

Der MMI-Trainingsalgorithmus

1. Initialisierung.
2. ML-Training.
3. Generiere konkurrierende Wortfolgen (z.B. durch Wortgraphen).
4. Wiederhole bis Konvergenz der MMI-Zielfunktion erreicht ist
   (a) Für jede Trainingsäußerung $r = 1, \cdots, R$ aus den Trainingsdaten
      i. Forced Time Alignment der gesprochenen Wortfolge.
         Akkumuliere Zählerstatistiken: siehe ML-Training ($\sum_{t=1}^{T_r} \gamma_{r,t}(s|W)$,...).
      ii. Forced Time Alignment aller konkurrierenden Wortfolgen.
         Akkumuliere Nennerstatistiken:
         $\sum_{t=1}^{T_r} \gamma_{r,t}(s)$, $\sum_{t=1}^{T_r} \gamma_{r,t}(s) x_{r,t}$, $\sum_{t=1}^{T_r} \gamma_{r,t}(s) x_{r,t}^2$
   (b) Schätze die Parameter neu (EB oder GP).
   (c) Splitten der Dichten (optional).

Prinzip zur Berechnung der speziellen FB-Wahrscheinlichkeit

$$\gamma_{r,t}(s|W) = \sum_{\sigma} \gamma_{r,t}(s,\sigma|W)$$



Baum-Welch oder Viterbi

- Die Baum-Welch-Rekursion:

$$\gamma_{r,t}(s|W) = \sum_{\sigma} \frac{a_{r,t-1}(\sigma, W) \cdot p(s|\sigma) \cdot b_{r,t}(s, W)}{p_\lambda(X_r|W)}$$

Forward-Wahrscheinlichkeit: $a_{r,t-1}(\sigma, W)$.

Backward-Wahrscheinlichkeit: $b_{r,t}(s, W)$.

- Die Viterbi-Nährung:

$$\gamma_{r,t}(s|W) \overset{Viterbi}{\approx} \delta(s, s_t(X_r, W)).$$

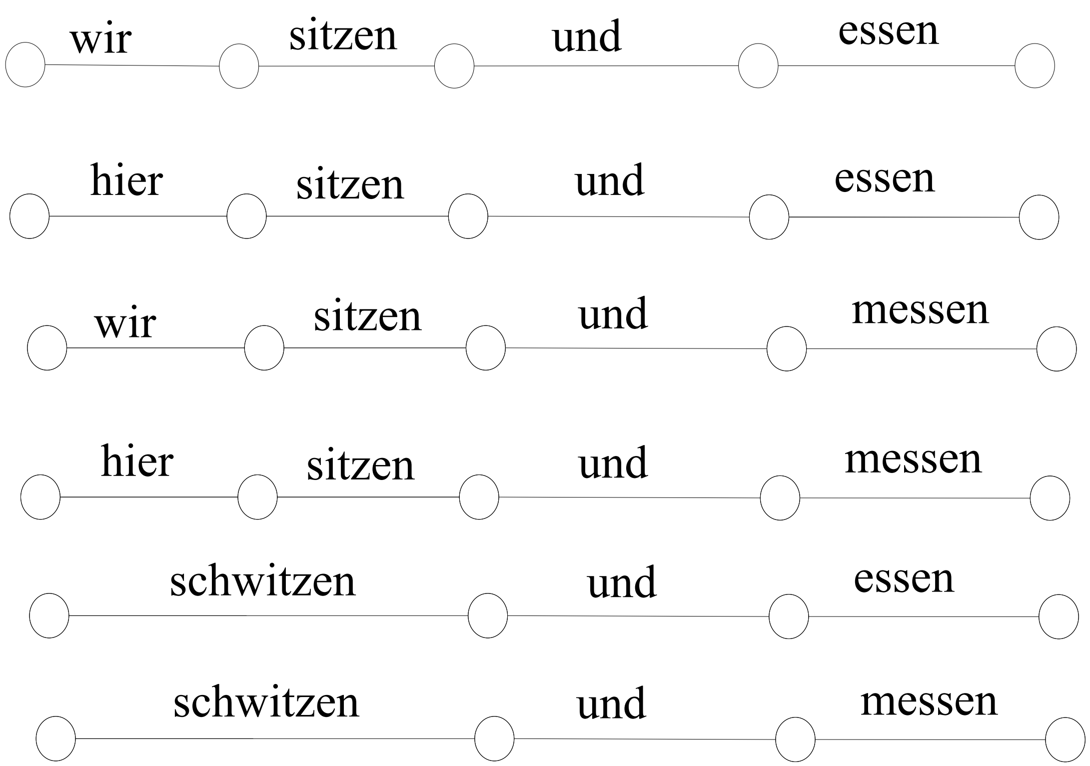
Berechnung der generalisierten FB-Wahrscheinlichkeit

$$\gamma_{r,t}(s) \;:=\; \sum_{W \in \mathcal{M}_r} \frac{p_\lambda(X_r, W)}{p_\lambda(X_r)} \, \gamma_{r,t}(s|W) \quad , \text{ für } \alpha = 1.$$

Problem: $\mathcal{M}_r$ muss eingeschränkt werden.

Idee: N-Besten-Listen, Wortgraphen.

Verwendung von N-Besten Listen

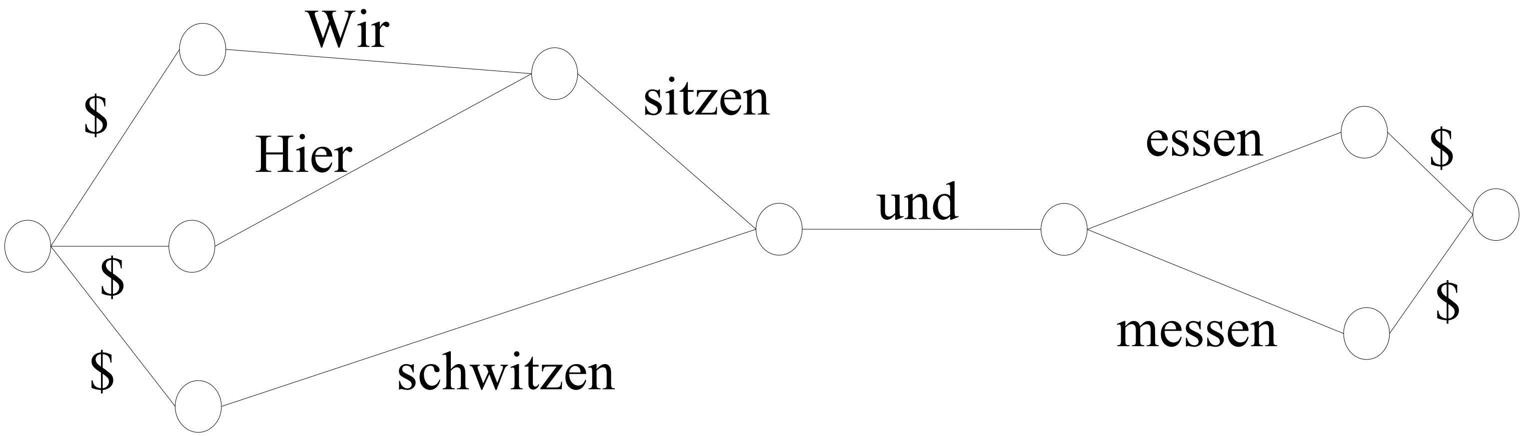


Verwendung von N-Besten Listen

- Manche Hypothesen unterscheiden sich nur minimal
- Für jede Wortfolge muss $p_\lambda(X_r, W)/p_\lambda(X_r)$ berechnet werden.
- Große Listen notwendig, um die Trainingsdaten gut auszuschöpfen.

$\rightarrow$ hohes Rechenzeitaufkommen.

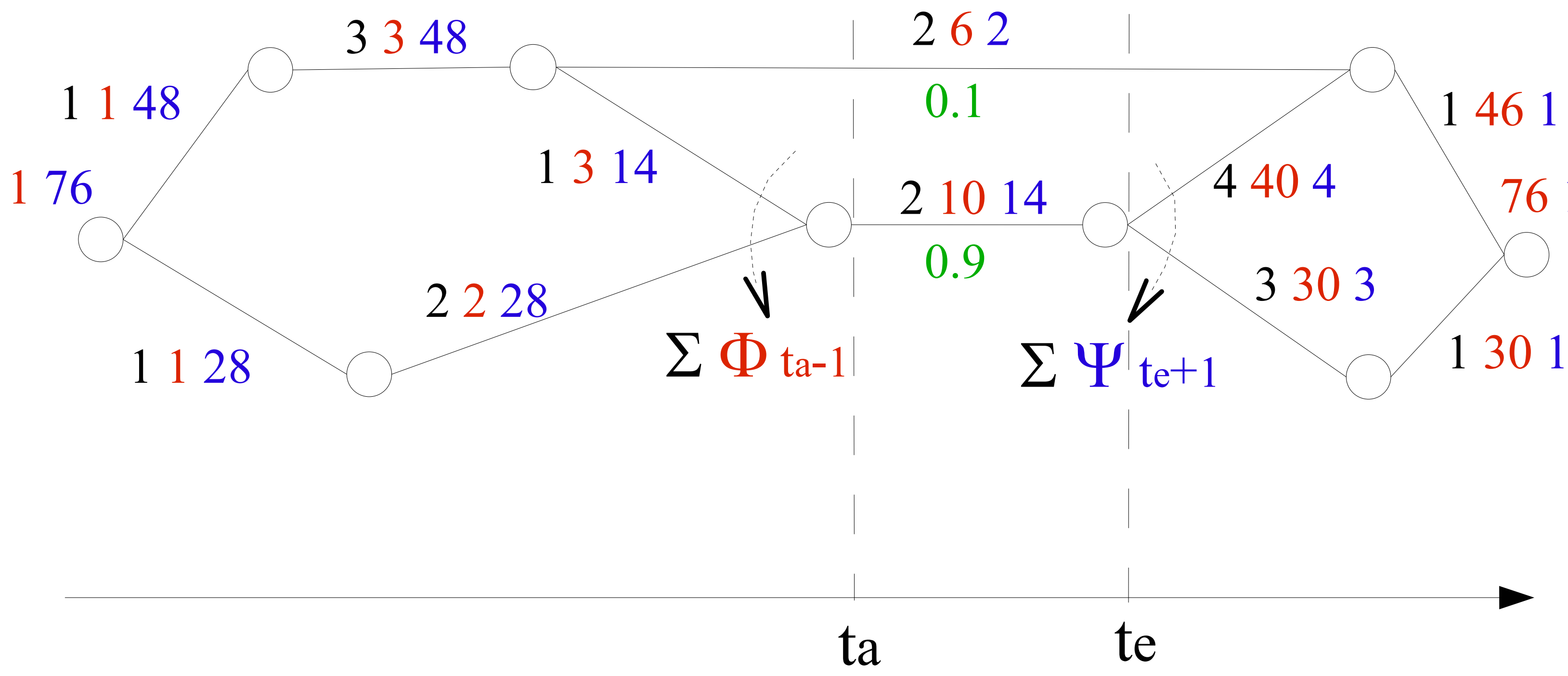
Verwendung von Wortgraphen



Berechnung der Wortwahrscheinlichkeit

$$P_{[ta,te]}(w \mid X_r) = \frac{\text{Forward} * \text{Backward}}{\text{Score} * \text{Normierung}}$$

Hier: Normierung = 76



Verwendung von Wortgraphen

Berechnung der generalisierten FB-Wahrscheinlichkeit:

$$\gamma_{r,t}(s) = \sum_{\{w|t_a(w)\le t\le t_e(w)\}} p_\lambda(s_t = s|w, x_{r t_a}^{t_e}) \cdot \; p_{[t_a,t_e]}(w|X_r)$$

Berechnung der Wortwahrscheinlichkeit:

$$p_{[t_a,t_e]}(w|X_r) = \sum_{h_2^{m-1}} \sum_{f_1^{m-2}} \frac{\Phi_{r,t_e}(h_2^{m-1}, w) \cdot \tilde{\Psi}_{r,t_a}(w, f_1^{m-2})}{\bar{p}_\lambda(X_r) \cdot p_\lambda^\alpha(x_{t_a}^{t_e}|w)} \cdot \prod_{i=1}^{m-2} p^\alpha(f_i|h_{i+1}^{m-1}, w, f_1^{i-1})$$

Ausschnitt aus dem Trainingsalgorithmus

Wiederhole bis Konvergenz der Zielfunktion erreicht ist

1. Für jeden Satz aus den Trainingsdaten
   (a) Bestimme Anzahl der Zustände.
   (b) Forced Time Alignment der gesprochenen Wortfolge.
   Akkumuliere Zählerstatistiken:
   $\sum_{t=1}^{T_r} \gamma_{r,t}(s|W),\ \sum_{t=1}^{T_r} \gamma_{r,t}(s|W) x_{r,t},\ \sum_{t=1}^{T_r} \gamma_{r,t}(s|W) x_{r,t}^2$
   (c) Forced Time Alignment aller konkurrierenden Wortfolgen.
   Akkumuliere Nennerstatistiken:
   $\sum_{t=1}^{T_r} \gamma_{r,t}(s),\ \sum_{t=1}^{T_r} \gamma_{r,t}(s) x_{r,t},\ \sum_{t=1}^{T_r} \gamma_{r,t}(s) x_{r,t}^2$
2. Schätze die Parameter neu:
   EB-Algorithmus oder Gradientenabstiegsverfahren.

Gradientenabstiegsverfahren oder EB-Algorithmus

- Gradientenabstiegsverfahren:

$$\text{neuer Mittelwert} \quad \hat{\mu}_{s,l} = \mu_{s,l} + \epsilon_{\mu_{s,l}} \frac{\partial F(\lambda)}{\partial \mu_{s,l}}$$

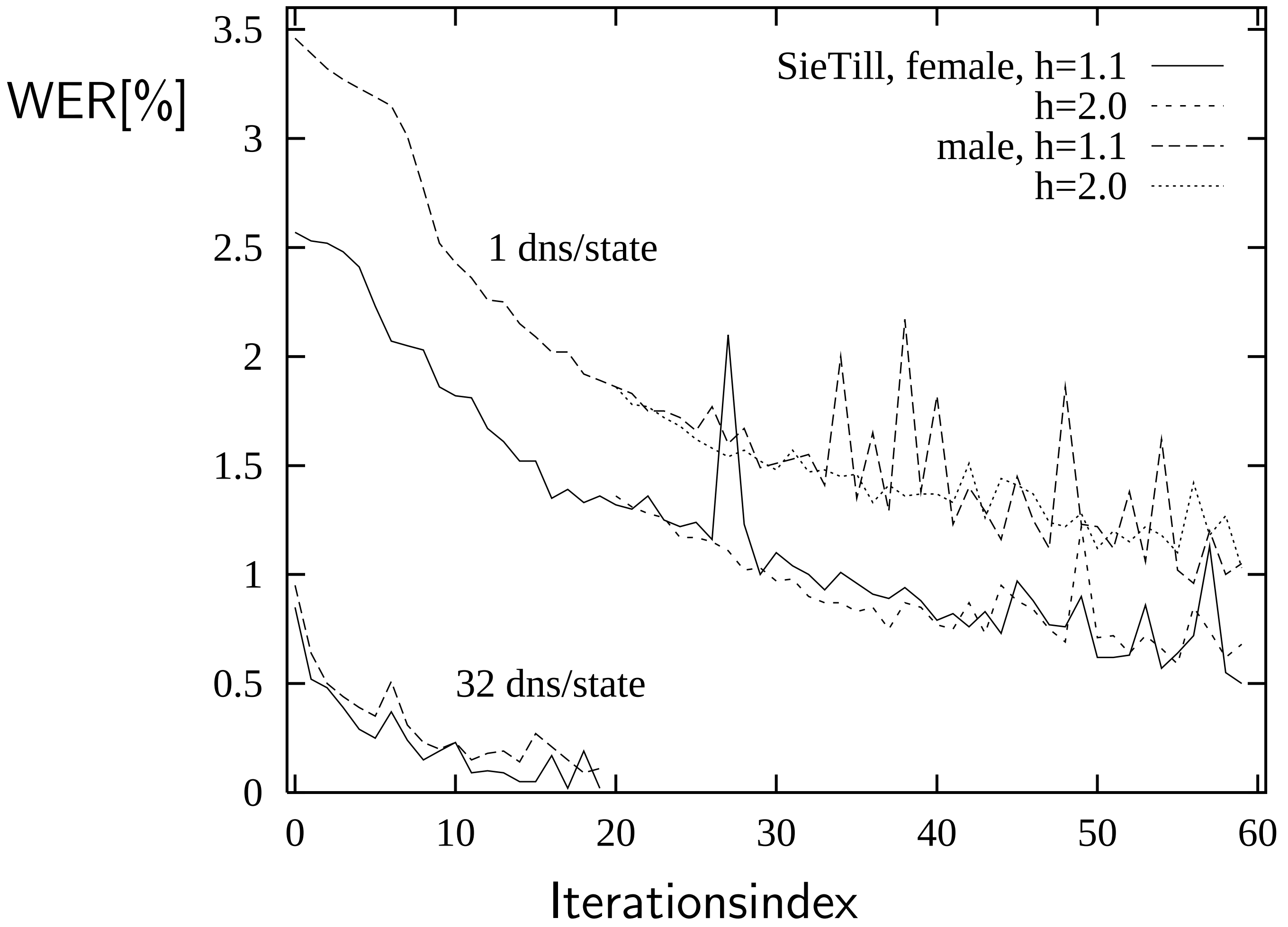
- EB-Algorithmus:

$$\text{neuer Mittelwert} \quad \hat{\mu}_{s,l} = \frac{\Gamma_{s,l}(x) + D\mu_{s,l}}{\Gamma_{s,l}(1) + D}$$

mit

$$\Gamma_{s,l}(g(x)) := \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_{r,t}(s|W_r) g(x_{r,t}) - \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_{r,t}(s) g(x_{r,t})$$

Wortfehlerrate bei der Wahl unterschiedlicher Schrittweiten



Bestimmung der Schrittweite/Iterationskonstanten

Problem: richtige Wahl der Schrittweite/Iterationskonstanten

- kleine Schrittweite $\longrightarrow$ „sichere“ aber langsame Konvergenz.
- grosse Schrittweite $\longrightarrow$ schnellere Konvergenz aber „gefährlich“.
- globale (D) *vs.* dichteabhängige $(D_{s,l})$ Iterationskonstante.

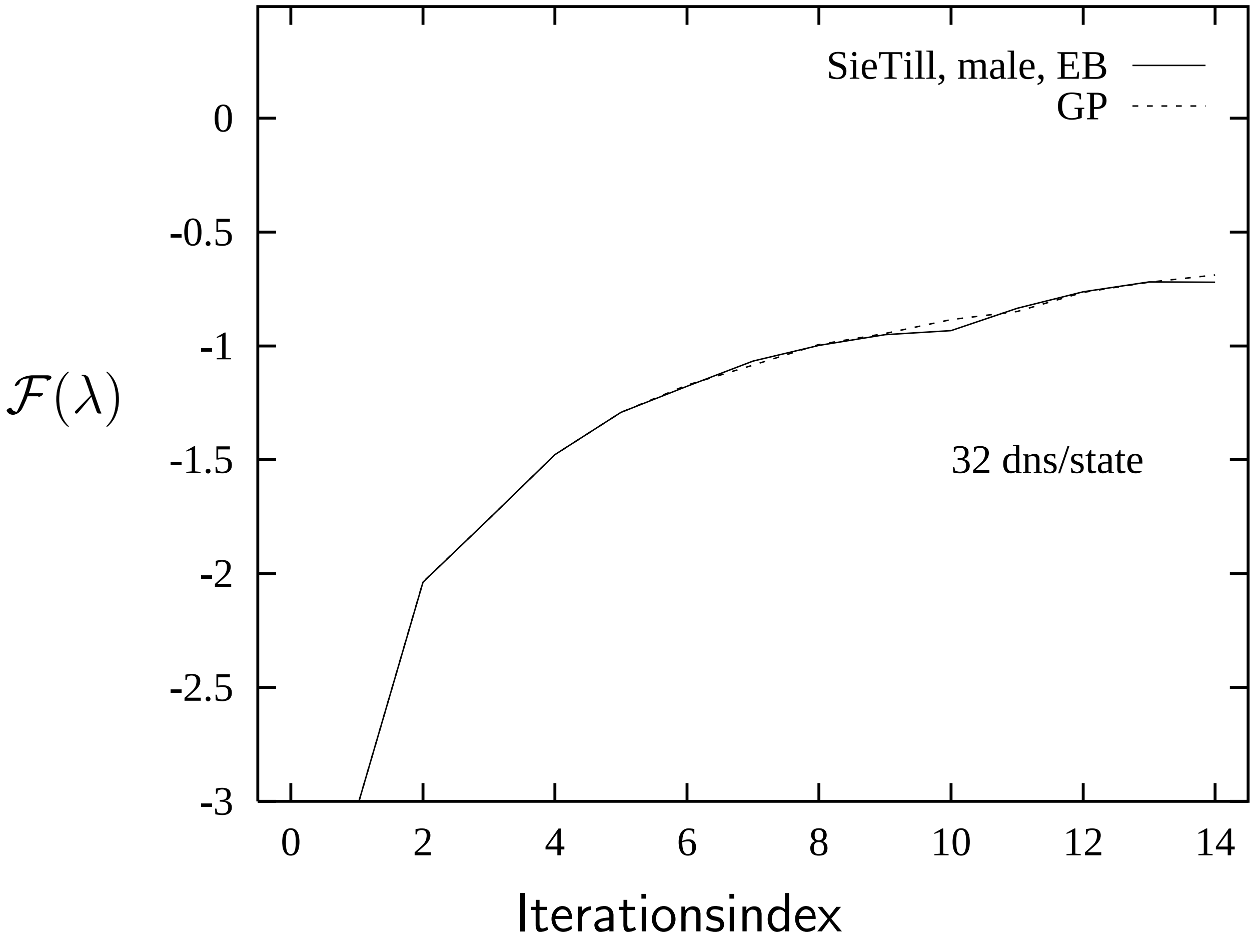
SieTill Korpusbeschreibung

Korpus-Statistiken								
Korpus	Aufnahmekanal/ Sprache	Train./ Test	male		female		gesamt	
			Sätze	Ziffern	Sätze	Ziffern	Sätze	Ziffern
SieTill	Telefon/ deutsch	Train.	6886	22631	6150	20226	13036	42857
		Test	6938	22881	6176	20205	13114	43086

SieTill Erkennungssystem

- 11 deutsche Ziffern, inklusive „zwo“
- geschlechtsabhängige Ganzwortmodelle
- Gaußsche Mischverteilungen
- 25-komponentiger Merkmalsvektor

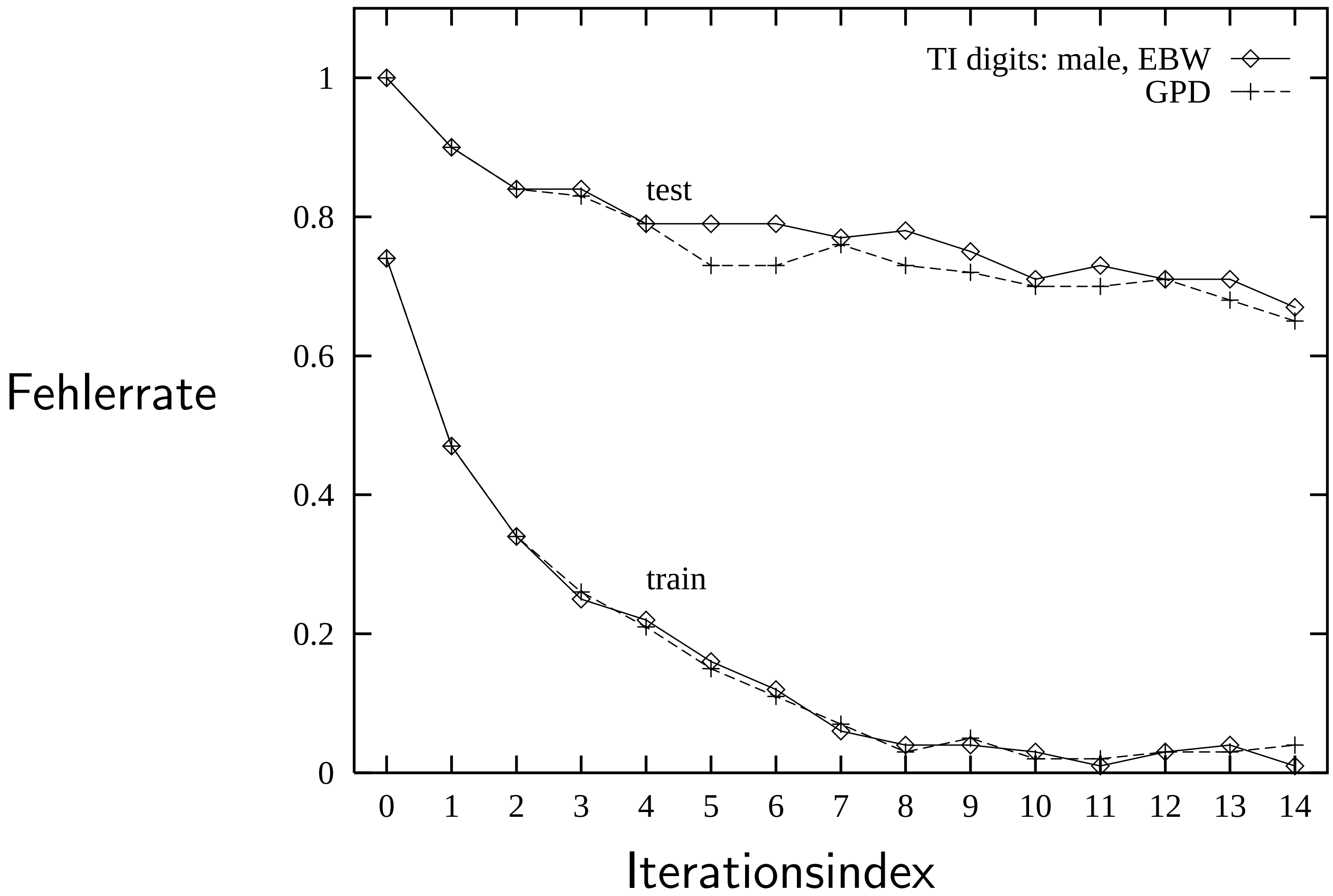
Vergleich EB-Algorithmus, Gradientenabstiegsverfahren



Die verschiedenen Trainingskriterien im Vergleich

SieTill: Kriterien						
Korpus	LDA	Kriterium	Dichten m+f	del - ins - sub	WER[%]	SER[%]
Test	ja	ML	215+215	304-270-1053	3.78	9.74
			6862+6839	198-201-450	1.97	5.31
			13K + 13K	**198-162-419**	**1.81**	**4.93**
		MCE	215+215	315-176-657	2.60	6.73
			13472+13397	**183-148-398**	**1.69**	**4.64**
		MMI	215+215	349-175-684	2.81	7.13
			6858+6825	**182-161-407**	**1.74**	**4.80**

Ein Vergleich zwischen der Fehlerrate auf den Trainingsdaten und der Fehlerrate auf den Testdaten



Zusammenfassung

- Berücksichtigung klassenfremder Trainingsdaten durch Maximierung von $p(W|X)$.
- Berechnung der speziellen und der generalisierten FB-Wahrscheinlichkeit.
- Modellierung der konkurrierenden Wortfolgen durch Verwendung von Wortgraphen.
- Neuschätzung der Parameter nach dem EB-Algorithmus bzw. Gradientenabstiegsverfahren.
- Diskriminatives Training verbessert die Erkennung gegenüber ML-trainierten Systemen sichtlich.
- Gewinn/Vorteil wird geringer bei großer Anzahl von Parametern $\rightarrow$ Generalisierungsproblem.