Einführung in die statistische Sprachverarbeitung

SS07
Seminar für Computerlinguistik

Mitschrift aus der Vorlesung erstellt von:
Mateusz Dworaczek, Andreas Dörr

19. Juli 2007

1 Grundbegriffe der Wahrscheinlichkeitstheorie und Statistik

1.1 Korpora

Definition 1.1 (Korpus). Sei $\mathcal{X}$ eine abzählbare Menge. Eine reellwertige Funktion $f : \mathcal{X} \mapsto \mathbb{R}$ wird **Korpus** *(engl. corpus)* gennant, falls:

$$f(x) \geq 0 \; \forall \; x \in \mathcal{X}.$$

Jedes $x \in \mathcal{X}$ heißt **Typ** *(engl. type)*. Jeder Funktionswert $f(x)$ heißt **Häufigkeit** oder Frequenz *(engl. frequency)*. Die **Korpusgröße** *(engl. corpus size)* ist definiert als:

$$|f| = \sum_{x \in \mathcal{X}} f(x).$$

Der Korpus ist **nicht-leer** *(engl. non-empty)* und **endlich** *(engl. finite)*, falls:

$$0 < |f| < \infty.$$

Bemerkung.
Für Statistiker ist es nicht wichtig, dass ein Korpus normalerweise als Folge von Entitäten (z.B. Wörtern, Sätzen usw.) angesehen wird. Wichtig hingegen ist die Vorkommenshäufigkeit im Korpus, nicht die spezielle Position einer einzelnen Entität. Man definiert hier die Häufigkeit als reelle Zahl, um später auch mit sogenannten Erwartungswerten, welche die erwartete Häufigkeit eines Typs im Mittel angeben, umgehen zu können.

Definition 1.2 (Vorkommenshäufigkeit von Token). Sei $x_1 \dots x_n$ eine endliche Folge von Typ-Instanzen aus $\mathcal{X}$. Dann heißt jedes x_i ein **Token**. Die **Vorkommenshäufigkeit** *(engl. occurrence frequency)* eines Typs x ist wie folgt definiert:

$$f(x) = |\{i | x_i = x\}| \, .$$

Beispiel.
Korpus: „Das Auto, das nicht fährt“;

$\mathcal{X} = \{das, Auto, nicht, fährt, \text{‘,‘}, \text{‘ ‘}, \text{‘.‘}\}$;
$f(Auto) = f(nicht) = f(fährt) = 1$;
$f(das) = 2$.

Die zwei Vorkommen des Wortes „das“ sind jeweils zwei verschiedene Token, d.h. der Token-Begriff berücksichtigt insbesondere die Position einer Entität innerhalb des Korpus.

Bemerkungen.

1. Obige Definition nennt man auch den „herkömmlichen Korpus-Begriff“, da dieser auf einem Folgen-Verständnis von Token basiert.

2. f, wie in Definition 1.2 ist ein Korpus gemäß Definition 1.1, denn:

$$f(x) \geq 0 \quad \text{und}$$

$$|f| = \sum_{i=1}^{n} |\{i|x_i = x\}| = n.$$

3. Achtung: Bei der Auswahl zwischen verschiedenen Entitäten ist stets darauf zu achten, dass diese genügend Informationsgehalt in sich tragen, um den gewünschten Bereich der statistischen Modellierung ausreichend abzudecken.
   Bspw.:

   - Syntax: Korpus von Sätzen
   - POS-Tagging: Korpus von Bi- oder Trigrammen
   - Morphologie: Korpus von Worten/Morphemen
   - Anaphern-Resolution: Korpus von Paragraphen usw.

1.2 Wahrscheinlichkeitsverteilungen

Definition 1.3 (Wahrscheinlichkeitsverteilung). Sei $\mathcal{X}$ eine abzählbare Menge. Eine reellwertige Funktion $p : \mathcal{X} \mapsto \mathbb{R}$ heißt **Wahrscheinlichkeitsverteilung** *(engl. probability distribution)* über $\mathcal{X}$, falls

$$p(x) \geq 0 \; \forall \; x \in \mathcal{X} \qquad \text{und} \qquad \sum_{x \in \mathcal{X}} p(x) = 1.$$

Bemerkungen.

1. Aus der Definition folgt sofort: $0 \leq p(\mathcal{X}) \leq 1 \; \forall \; x \in \mathcal{X}$.

2. Eigentlich ist obige Definition die einer *probability mass function*. Normalerweise basiert Wahrscheinlichkeitstheorie auf drei Axiomen:

   a) $p(A) \geq 0 \; \forall \; A \subseteq \mathcal{X}$
   b) $p(\mathcal{X}) = 1$
   c) $p(\bigcup A_i) = \sum p(A_i)$, falls $A_1 \ldots A_n$ paarweise disjunkt.

   Dabei geht man von einer Menge von möglichen Ereignissen $A \subseteq \mathcal{X}$ aus, denen jeweils eine bestimmte Wahrscheinlichkeit zugeordnet wird.

   Beispiel.
   Würfel; Menge der möglichen Elementarereignisse: $\mathcal{X} = \{1 \ldots 6\}$.

$$A_1 = \{2, 4, 6\} \Rightarrow p(A_1) = \frac{1}{2};$$

$$A_2 = \{1, 2\}, A_3 = \{5, 6\}$$

$$\Rightarrow p(A_2 \cup A_3) = p(\{1, 2, 5, 6\}) = p(A_2) + p(A_3) = \frac{2}{3}.$$

3. Diese Art von Wahrscheinlichkeitstheorie folgt übrigens aus der angegebenen Definition, wenn man setzt:

$$p(A) = \sum_{x \in A} p(x) \ \forall \ A \subseteq \mathcal{X}.$$

   Bspw. ist Axiom 2b erfüllt:

$$\text{Sei} \quad \mathcal{X} = \{x_1, x_2, \dots\}.$$

$$\Rightarrow p(\mathcal{X}) = \sum_i p(x_i) = \sum_{x \in \mathcal{X}} p(x) = 1.$$

4. Definition 1.3 erleichtert den Umgang mit abzählbaren Mengen. Da man sich in der Computerlinguistik hauptsächlich mit solchen Mengen beschäftigt, ist diese Definition naturgemäß völlig ausreichend.

1.3 Relative Häufigkeit

Definition 1.4 (Relative Häufigkeit). Sei f ein nicht-leerer, endlicher Korpus. Die Wahrscheinlichkeitsverteilung

$$\widetilde{p} : \mathcal{X} \mapsto [0, 1], \quad x \mapsto \frac{f(x)}{|f|} \quad \text{bzw.}$$

$$\widetilde{p}(x) = \frac{f(x)}{|f|} \ \forall \ x \in \mathcal{X}$$

heißt **relative Häufigkeit** *(engl. relative-frequency estimate, Abk.: RFE)* oder **empirische Verteilung** *(engl. empirical distribution).*

Bemerkungen.

1. Die Division durch |f| verlangt, dass der Korpus nicht-leer und endlich ist, andernfalls dividiert man durch 0 (wenn Korpus leer) bzw. durch unendlich (wenn Korpus unendlich groß).

2. $\widetilde{p}$ ist eine Wahrscheinlichkeitsverteilung, denn:

$$\widetilde{p}(x) \geq 0 \ \forall \ x \quad \text{und}$$

$$\sum_x \widetilde{p}(x) = \sum_x \frac{f(x)}{|f|} = \frac{1}{|f|} \sum_x f(x) = \frac{1}{|f|} \cdot |f| = 1.$$

   Damit sind beide Bedingungen, die an eine Wahrscheinlichkeitsverteilung gestellt werden, erfüllt.

1.4 Wahrscheinlichkeitsmodelle

Definition 1.5 (Wahrscheinlichkeitsmodell). Eine nicht-leere Menge $\mathcal{M}$ von Wahrscheinlichkeitsverteilungen auf einer Menge $\mathcal{X}$ von Typen heißt **Wahrscheinlichkeitsmodell auf** $\mathcal{X}$ *(engl. probability model)*. Die Elemente von $\mathcal{M}$ werden **Instanzen** *(engl. instance)* des Modells $\mathcal{M}$ genannt. Das **unbeschränkte** *(engl. unrestricted)* Wahrscheinlichkeitsmodell ist die Menge $\mathcal{M}(\mathcal{X})$ *aller* Wahrscheinlichkeitsverteilungen auf $\mathcal{X}$.

$$\mathcal{M}(\mathcal{X}) = \left\{ p : \mathcal{X} \mapsto [0,1] \,\middle|\, \sum_{x \in \mathcal{X}} p(x) = 1 \right\}.$$

In allen anderen Fällen heißt $\mathcal{M}$ beschränkt *(engl. restricted)*:

$$\mathcal{M} \subseteq \mathcal{M}(\mathcal{X}) \wedge \mathcal{M} \neq \mathcal{M}(\mathcal{X}).$$

1.5 Korpuswahrscheinlichkeit

Definition 1.6 (Korpuswahrscheinlichkeit). Sei f ein nicht-leerer, endlicher Korpus. Sei $\mathcal{M}$ ein Wahrscheinlichkeitsmodell. Die Korpuswahrscheinlichkeit, die dem Korpus durch eine Instanz $p \in \mathcal{M}$ zugewiesen wird, ist:

$$L(f, p) = \prod_{x \in \mathcal{X}} p(x)^{f(x)}.$$

Bemerkung.
Die Korpuswahrscheinlichkeit eines Korpus wird im Allgemeinen als diejenige Wahrscheinlichkeit interpretiert, mit der ein gegebener Korpus *generiert* würde.

Beispiel.
Gegeben sei ein Korpus $x_1 \dots x_n$ mit $x_i \in \mathcal{X}$. Dann gilt:

$$L(x_1 \dots x_n, p) = p(x_1) \cdot p(x_2) \dots p(x_n) = \prod_{i=1}^{n} p(x_i).$$

D.h. die Wahrscheinlichkeit eines Korpus ist das Produkt seiner Token-Wahrscheinlichkeiten. Ist nun bspw. $n = 5, x_1 = x_3 = a \quad \text{und} \quad x_2 = x_4 = x_5 = b$ folgt:

$$L(x_1 \dots x_5, p) = p(a) \cdot p(b) \cdot p(a) \cdot p(b) \cdot p(b) = p(a)^2 \cdot p(b)^3 = p(a)^{f(a)} \cdot p(a)^{f(b)} = L(f, p),$$

weil $f(a) = 2$ und $f(b) = 3$ in diesem Beispiel.

1.6 Maximum-Likelihood Estimierung

Definition 1.7. (MLE) Eine Instanz $\hat{p}$ des Modells $\mathcal{M}$ heißt **Maximum-Likelihood Schätzung**(engl. Maximum-Likelihood estimate) von $\mathcal{M}$ auf den Korpus f genau dann, wenn:

$$L(f, \hat{p}) = \max_{p \in \mathcal{M}} L(f, p)$$

Interpretation: Diejenige Instanz, die den gegebenen Korpus die höchste Wahrscheinlichkeit zuweist, ist diejenige Instanz die den Korpus erzeugte.

Beispiel:

p	$L(f,p)$	0.07
p'	$L(f,p')$	0.01
p''	$L(f,p'')$	**0.12**
p^n	⋮	⋮

Hier wäre p'' der ML-Schätzer

Diese Interpretation kann wahr oder falsch sein!
D.h. es kann Korpora geben, die durch eine Verteilung p erzeugt werden, die aber nicht ML Schätzung eines Modells auf dem Korpus sein muß.

Bemerkung:

ML-Estimierung ist ein kompliziertes Optimierungsproblem und hat folgende Probleme:

Existenz: Gibt es mindestens eine Lösung?

Eindeutigkeit: Gibt es mehr als eine Lösung?

Berechenbarkeit: Ist eine Lösung einfach zu berechnen?

Die folgenden Abbildungen illustrieren diese Probleme.

1. Abbildung 4 zeigt einen Fall, in dem kein MLE existiert.
2. Abbildung 5 illustiert dieses Problem etwas genauer.
3. Abbildungen 2 und 3 zeigen Fälle in denen MLE nicht eindeutig ist, wärend
4. in Abbildung 1 ein Fall diskutiert wird, in dem es genau ein MLE gibt.

„Common Wisdom“

RFE = MLE

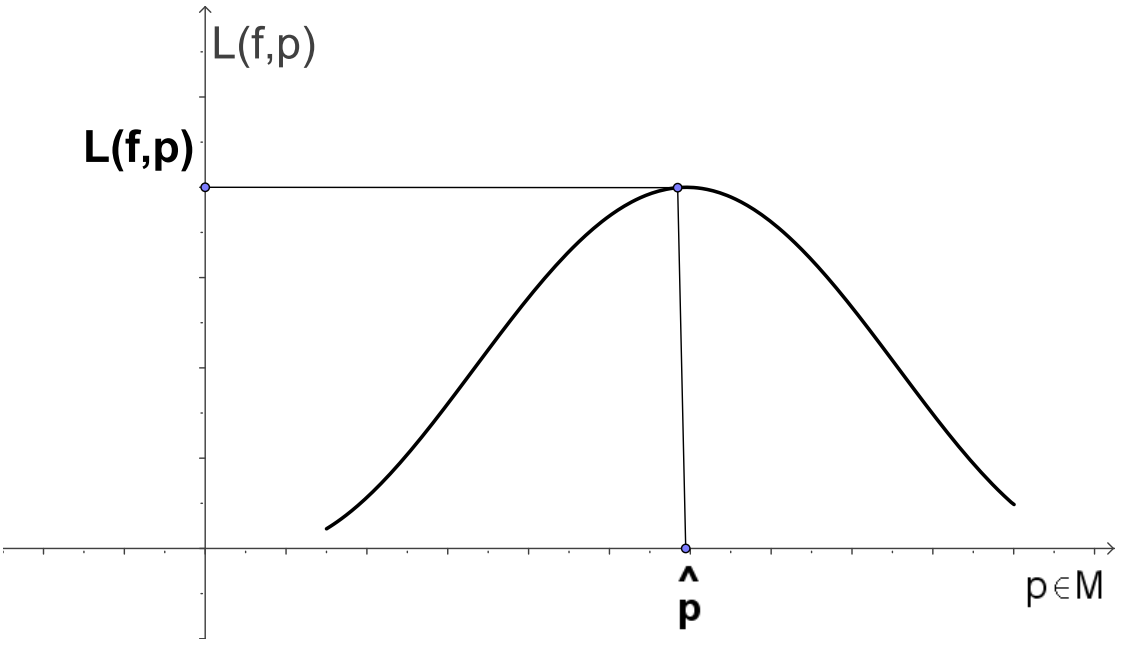


Abbildung 1: Beispiel, wo ML-Estimierung genau eine Lösung hat.

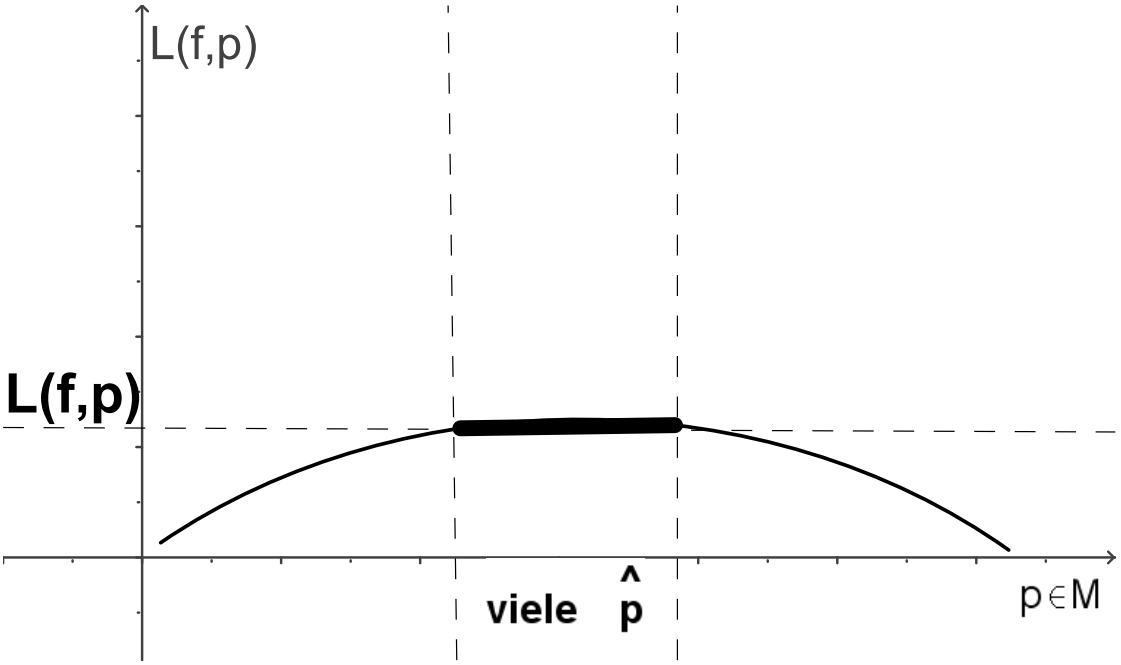


Abbildung 2: Beispiel, in dem ML-Estimierung mehr als eine Lösung hat (unendlich viele sogar)

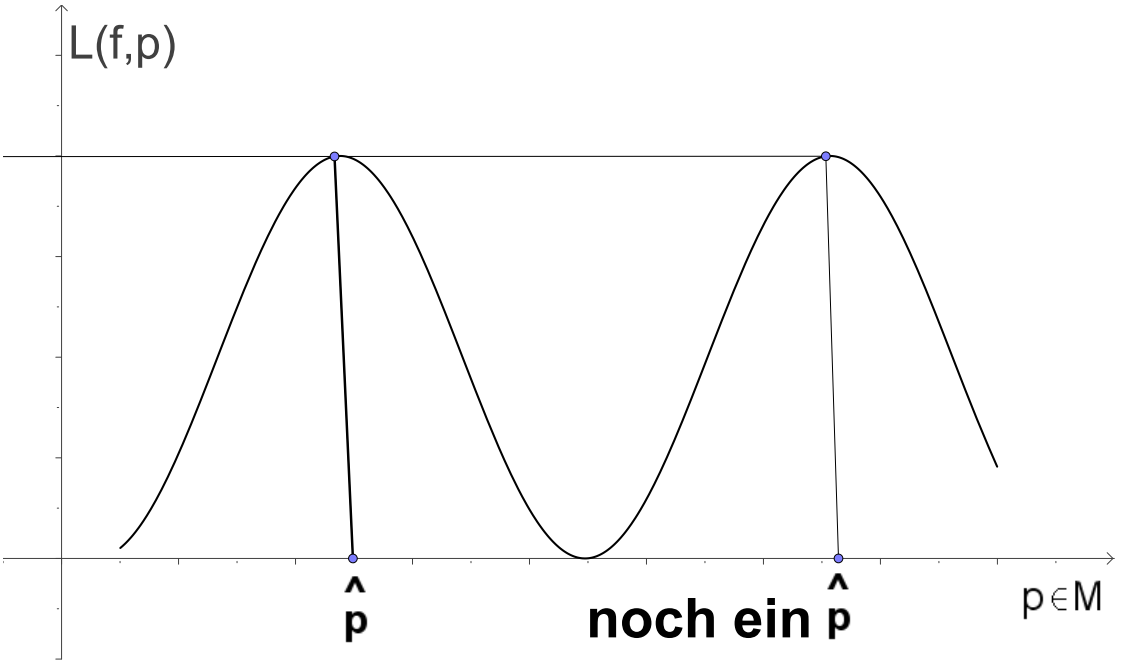


Abbildung 3: Noch ein Beispiel, in dem ML-Estimierung mehr als eine Lösung hat

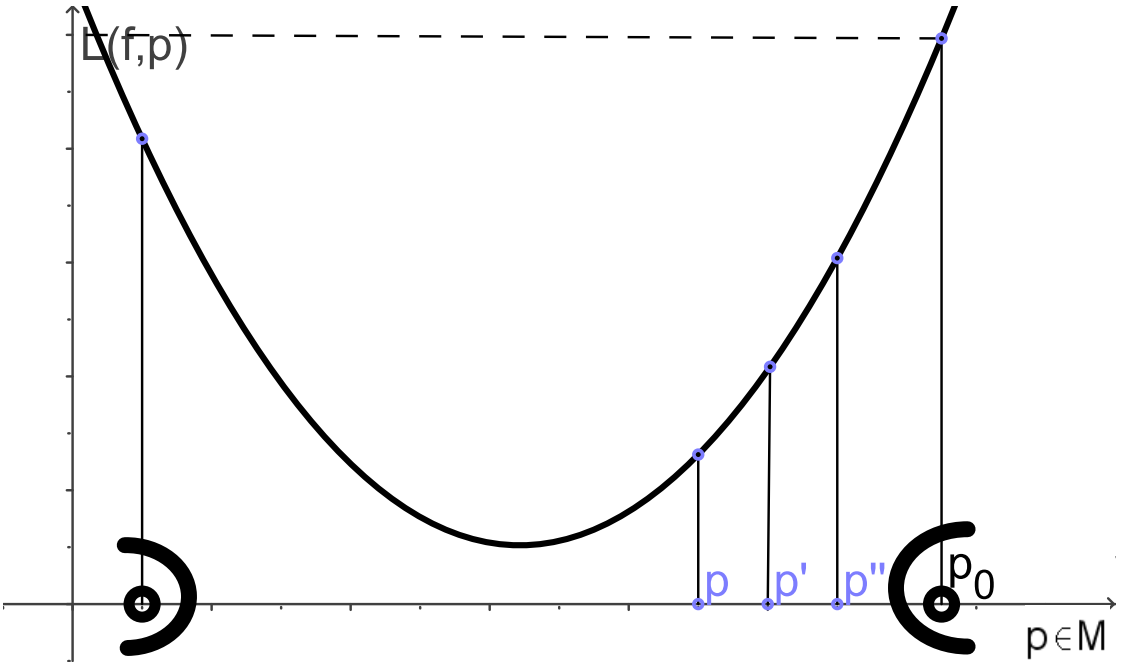


Abbildung 4: Die Randpunkte gehören nicht zu $\mathcal{M}$.$p_0 \notin \mathcal{M}$, was aber für ein ML estimate der Fall sein muss

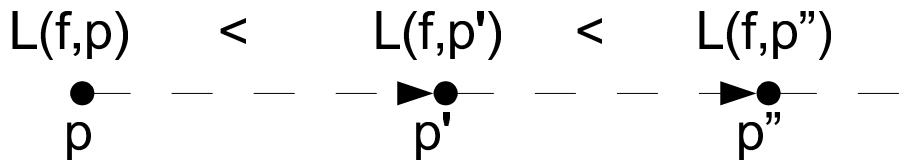


Abbildung 5: **typische Situation**: zu jedem $p \in \mathcal{M}$ findet man ein $p' \in \mathcal{M}$, sodaß $L(f,p) < L(f,p')$

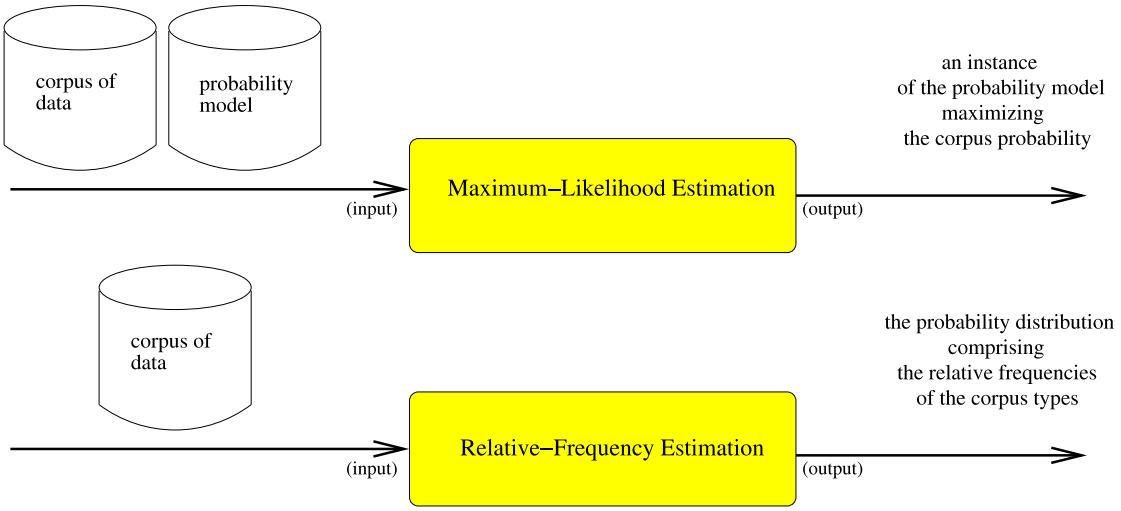


Abbildung 6: Vergleich

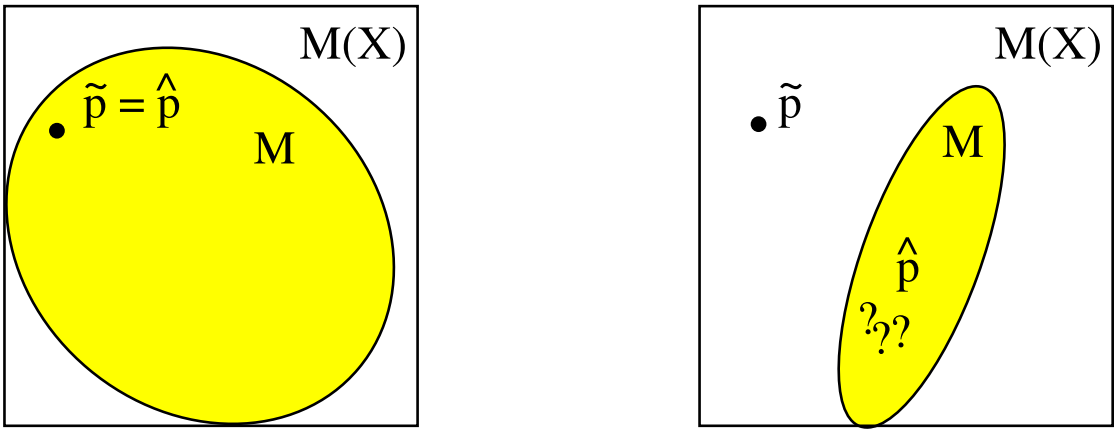


Abbildung 7: Theorem

Diese Vorlesung: Im folgenden werden wir zeigen, dass im allgemeinen $RFE \neq MLE$ gilt, und nur in wenigen (aber wichtigen) Fällen das MLE mit dem RFE übereinstimmt.

Beachte auch, dass Abbildung 6 schon anzeigt, dass MLE und RFE verschieden sind, da Input und Output beider Estimmierungsverfahren sich schon auf den ersten Blick stark voneinander unterscheiden.

Das folgende Theorem gibt an, wann MLE und RFE zusammenfallen bzw. verschieden sind. Die Abbildung 7 illustiert beide Fälle.

Theorem 1.1. Sei f ein nicht-leerer und endlicher Korpus auf $\mathcal{X}$. Dann gilt:

1. Das **relative frequency estimate** $\tilde{p}$ ist ein eindeutiges MLE des unbeschränken Modells $\mathcal{M}(\mathcal{X})$ auf f.
2. Das **relative frequency estimate** $\tilde{p}$ ist ein MLE eines (beschränkten oder unbeschränkten) Modells $\mathcal{M}$ auf f.

$$\Leftrightarrow \tilde{p} \in \mathcal{M}$$

In diesem Fall ist $\tilde{p}$ ein eindeutiges MLE von $\mathcal{M}$ auf f.

Beweis: mit Satz ?? und der Informationsungleichung der Informationstheorie. (Übung)

1.7 Relative Entropie

Definition 1.8 (Relative Entropie). Die **relative Entropie** $D(p\,\|q)$ einer Wahrscheinlichkeitsverteilung p bezüglich einer anderen Wahrscheinlichkeitsverteilung q ist definiert als:

$$D(p\,\|q) = \sum_{x\in\mathcal{X}} p(x)\cdot\log\frac{p(x)}{q(x)}.$$

Bemerkungen.

1. Hierbei benutzen wir die folgenden Konventionen:
   a) $0\cdot\log\frac{0}{q} = 0$
   b) $p\cdot\log\frac{p}{0} = \infty$
   c) $0\cdot\log\frac{0}{0} = 0$
   d) Mit log bezeichnen wir den Logarithmus zur Basis 2 ($\log_2$).
2. Die Konventionen 1a - 1d folgen aus Stetigkeitsgründen. Mathematisch „sauberer" müsste man sich eigentlich einer Grenzwertbildung bedienen. Für Konvention 1b bspw.:

$$\lim_{\epsilon\to 0}\left(p\cdot\log\frac{p}{\epsilon}\right) = \infty.$$

1.8 Die Informationsungleichung aus der Informationstheorie

Theorem 1.2. Sei $\widetilde{p}$ die relative Häufigkeitsschätzung auf einem nicht-leeren und endlichen Korpus. Sei $\mathcal{M}$ ein Wahrscheinlichkeitsmodell auf einer Menge von Typen $\mathcal{X}$. Dann gilt:
Eine Instanz $\widehat{p}$ des Modells $\mathcal{M}$ ist ein Maximum-Likelihood-Estimate auf f genau dann, wenn die relative Entropie von $\widetilde{p}$ bezüglich $\widehat{p}$ minimal ist, d.h., wenn

$$D(\widetilde{p}\,\|\widehat{p}) = \min_{p\in\mathcal{M}} D(\widetilde{p}\,\|p).$$

Beweis. Die relative Entropie ist die Differenz zweier weiterer Entropiewerte, nämlich der so genannten **Cross-Entropie:**

$$H(\widetilde{p},p) = -\sum_{x\in\mathcal{X}}\widetilde{p}\cdot\log p(x) \tag{1}$$

und der **(Korpus-)Entropie:**

$$H(\widetilde{p}) = -\sum_{x\in\mathcal{X}}\widetilde{p}\cdot\log\widetilde{p}(x), \tag{2}$$

d.h. es gilt:

$$D(\widetilde{p}\,\|\,p) = H(\widetilde{p},p) - H(\widetilde{p}). \tag{3}$$

Bemerkungen.

1. $H(\widetilde{p}, p)$ und $H(\widetilde{p})$ sind stets größer gleich 0.

2. $H(\widetilde{p}) = H(\widetilde{p}, \widetilde{p})$, d.h. (Korpus-)Entropie lässt sich als Spezialfall der Cross-Entropie darstellen. Damit folgt insbesondere, dass $D(\widetilde{p} \,\|\, \widetilde{p}) = 0$.

Die Aufgabe an dieser Stelle ist nun, ein $p \in \mathcal{M}$ so zu finden, dass $D(\widetilde{p} \,\|\, p)$ minimal wird, d.h. man muss $H(\widetilde{p}, p) - H(\widetilde{p})$ als Funktion von $p \in \mathcal{M}$ minimieren. Da $H(\widetilde{p})$ nicht abhängig von p und somit als konstanter Summand zu betrachten ist, genügt es, $H(\widetilde{p}, p)$ zu minimieren. Zu diesem Zweck bedienen wir uns der bereits in Definition 1.5 eingeführten Korpuswahrscheinlichkeit:

$$\begin{aligned} L(f,p) &= \prod_{x \in \mathcal{M}} p(x)^{f(x)} \\ \Leftrightarrow \log L(f,p) &= \sum_{x \in \mathcal{X}} log\Big(p(x)^{f(x)}\Big) \\ &= \sum_{x \in \mathcal{X}} f(x) \cdot \log p(x) \\ &= \frac{|f|}{|f|} \cdot \sum_{x \in \mathcal{X}} f(x) \cdot \log p(x) \\ &= |f| \cdot \sum_{x \in \mathcal{X}} fracf(x)|f| \cdot \log p(x) \\ &= |f| \cdot \sum_{x \in \mathcal{X}} \widetilde{p} \cdot \log p(x) \\ &= -H(\widetilde{p}, p) \cdot |f| \,. \end{aligned} \tag{4}$$

Somit gilt folgender Zusammenhang zwischen der Korpuswahrscheinlichkeit und der Cross-Entropie:

$$\begin{aligned} \log L(f,p) &= -H(\widetilde{p}, p) \cdot |f| \\ \Leftrightarrow H(\widetilde{p}, p) &= -\frac{1}{|f|} \cdot \log L(f,p). \end{aligned} \tag{5}$$

D.h. es gilt nun, $-\frac{1}{|f|} \cdot \log L(f,p)$ zu minimieren. Aufgrund des negativen Vorzeichens bedeutet dies wiederum, $\frac{1}{|f|} \cdot \log L(f,p)$ zu maximieren. $\frac{1}{|f|}$ ist aber konstant. Und da der Logarithmus eine monoton wachsende Funktion ist (aus $\log a > \log b$ folgt $a > b$), reicht es aus, $L(f,p)$ zu maximieren, sprich: ein Maximum-Likelihood-Estimate von $\mathcal{M}$ auf f zu finden. Somit ist Theorem 1.2 bewiesen. □

Theorem 1.3 (Informationsungleichung der Informationstheorie). Die relative Entropie $D(p \,\|q)$ zweier Wahrscheinlichkeitsverteilungen hat folgende Eigenschaften:

1. $D(p \,\|q) \geq 0$ für beliebige Wahrscheinlichkeitsverteilungen p und q.

2. $D(p\,\|q\,) = 0$ genau dann, wenn $p = q$.

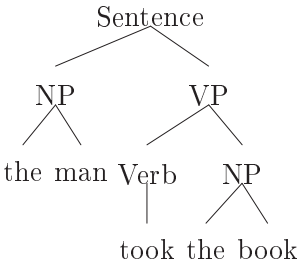
Bemerkung.
Im Beweis zu Theorem **??** beweist man mit Eigenschaft 1 die Existenz eines MLE, sowie mit Eigenschaft 2 die Eindeutigkeit dieses MLE.

2 Probabilistische kontextfreie Grammatiken *(engl. PCFG)*

Definition 2.1. (CFG) Eine kontexfreie Grammatik (engl. context-free grammar, CFG) besteht aus einer endlichen Menge von Regeln. Jede Regel besteht aus zwei Teilen, die durch ein spezielles Symbol „→“ getrennt werden. Die zwei Teile bestehen aus sogenannten **Terminal-** und **Nichtterminalsymbolen**:
Die linke Seite einer Regel besteht aus einem einzigen Nichtterminalsymbol, wärend die rechte Seite eine unendliche Folge von Terminal- und Nichtterminalsymbolen ist. Außerdem enthält die Menge der Nichtterminalsymbolen ein **Startsymbol**.

Bemerkung.
CFG werden auch Phrasenstruktur Grammatik genannt oder auch: Backus-Naur form (BNF 1959)

Beispiel.
Chomsky's Baum für folgenden Satz: „the man took the book“



Idee:

Sätze können in **Prasen** eingeteilt werden.
Phrasen sind Gruppen von Wörtern die eine Einheit bilden.
Phrasen können entdeckt werden, weil sie die Fähigkeit haben:

1. **allein zu stehen** (z.B. als Antwort auf eine Frage)
2. in **verschiedenen Satzpositionen** auftauchen
3. zu **expandieren** oder **substituiert** zu werden.
4. ...

Der obige Baum basiert auf der folgenden **kontextfreien Grammatik**:

$$\begin{array}{ccc} Sentence & \rightarrow & NP \\ NP & \rightarrow & the\ man \\ VP & \rightarrow & Verb\ NP \\ Verb & \rightarrow & took \\ NP & \rightarrow & the\ book \end{array}$$

Wobei:

Nichtterminal : {Sentence, NP, VP, Verb}

Terminal : {the, man, took, book}

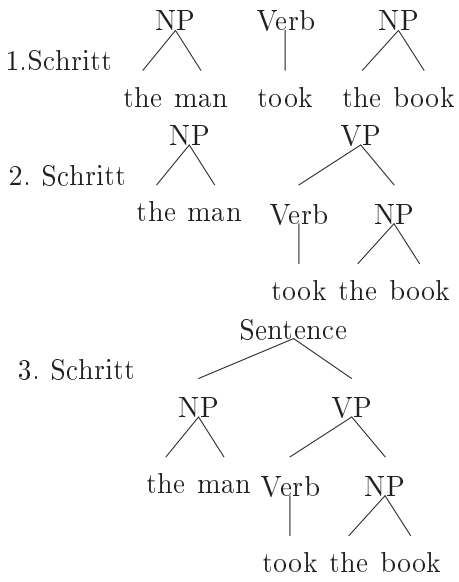
Startsymbol : Sentence

In der Computerlinguistik werden CFGs gewöhnlich auf zwei Arten benutzt:

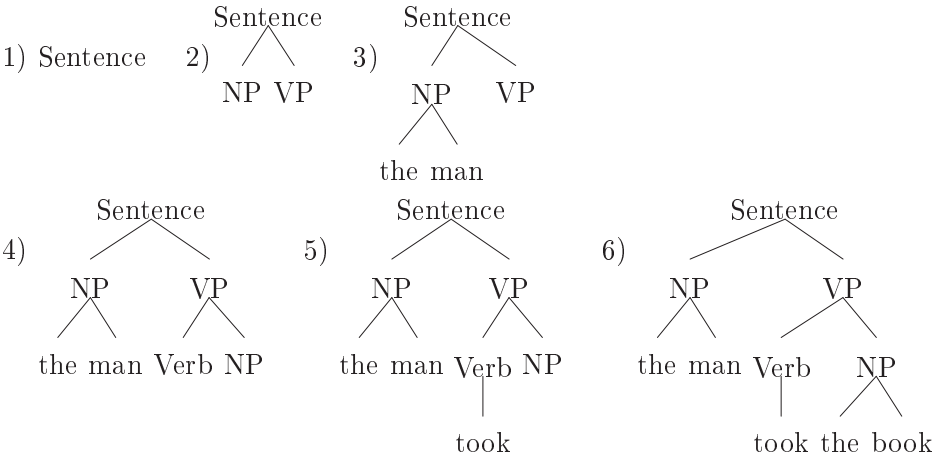
Generator

Parser

Beispiel.
CFG als Parser; Parsing ist der Prozess, der einem Eingabesatz eine (oder alle) Analysen zuweist.



Beispiel.
CFG als Generator; Generieren ist der Prozess, der aus dem Startsymbol einen (oder alle) Satzbaum generiert. (d.h. im Prinzip einen Satzt generiert)



Phänomen: Der Generierungsprozess ist **nicht deterministisch**.
Vergleiche Schritt 3) und 6). An diesen Stellen hätten wir die Wahl zwischen den Regeln:

$$\begin{aligned} NP &\rightarrow \textit{the man} \\ NP &\rightarrow \textit{the book} \end{aligned}$$

Wenn der Generator nur einen Output liefern soll, muss an diesen Stellen (3) und 6)) ausgewürfelt werden, ob die erste oder zweite Regel genommen werden soll. D.h. es müßen Wahrscheinlichkeiten i m Würfel festgelegt sein

$$\begin{aligned} 0 \leq\ & p(NP \rightarrow \textit{the man}) \leq 1 \\ 0 \leq\ & p(NP \rightarrow \textit{the book}) \leq 1 \end{aligned}$$

$$p(NP \rightarrow \textit{the man}) + p(NP \rightarrow \textit{the book}) = 1$$

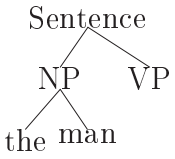
Dies führt zu einer probabilistischen kontextfreien Grammatik *(engl. PCFG)*:

$$\begin{aligned} p(\textit{Sentence} \rightarrow NP\ VP) &= 1 && (6) \\ p(NP \rightarrow \textit{the man}) &= x_1 && (7) \\ p(NP \rightarrow \textit{the book}) &= x_2 && (8) \\ p(VP \rightarrow V\ NP) &= 1 && (9) \\ p(V \rightarrow \textit{took}) &= 1 && (10) \end{aligned}$$

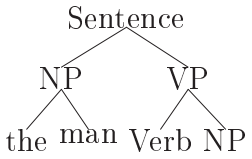
Beachte, dass (7) und (8):

$$x_1 + x_2 = 1$$

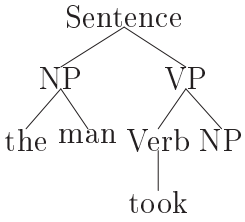
Frage: Wie häufig sind die partiellen Baumstrukturen im Generierungsprozess?



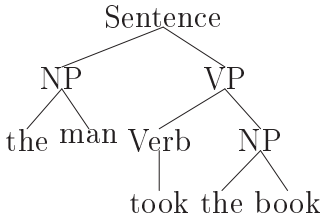
$p(NP \rightarrow the\ man) * 100\%$



$p(NP \rightarrow the\ man) * 100\%$



$p(NP \rightarrow the\ man) * 100\%$



$p(NP \rightarrow the\ man) * p(NP \rightarrow the\ book) * 100\%$

Weil alle anderen Regelwahrscheinlichkeiten 1 sind, erhalten wir:

$$p(t) = p(Sentence \rightarrow NP\ VP) * p(NP \rightarrow the\ man) *$$
$$*p(VP \rightarrow Verb\ NP) * p(Verb \rightarrow took) * p(NP \rightarrow the\ book)$$

d.h. Baumwahrscheinlichkeit ist der Produkt aller Regelwahrscheinlichkeiten (wobei jede Regel mit ihr Vorkommenswahrscheinlichkeiten gezählt werden muss)

Definition 2.2. (PCFG) Ein Paar $\langle G, p \rangle$, das aus einer kontextfreien Grammatik G und einer Wahrscheinlichkeitsverteilung $p : \mathcal{X} \rightarrow [0, 1]$ auf der Menge $\mathcal{X}$ aller vollen Parsebäumen von G besteht heißt **probabilistische kontextfreie Grammatik** *(engl. PCFG)*, falls für alle Parsebäume $x \in \mathcal{X}$ gilt:

$$p(x) = \prod_{r=G} p(r)^{fr(x)}$$

Hierbei ist fr(x) die Vorkommenshäufigkeit der Regel r im Parsebaum x, und p(r) ist eine Wahrscheinlichkeit für die Regel r, sodaß für alle Nichtterminalsymbole A

gilt:

$$\sum_{r \in G_A} p(r) = 1$$

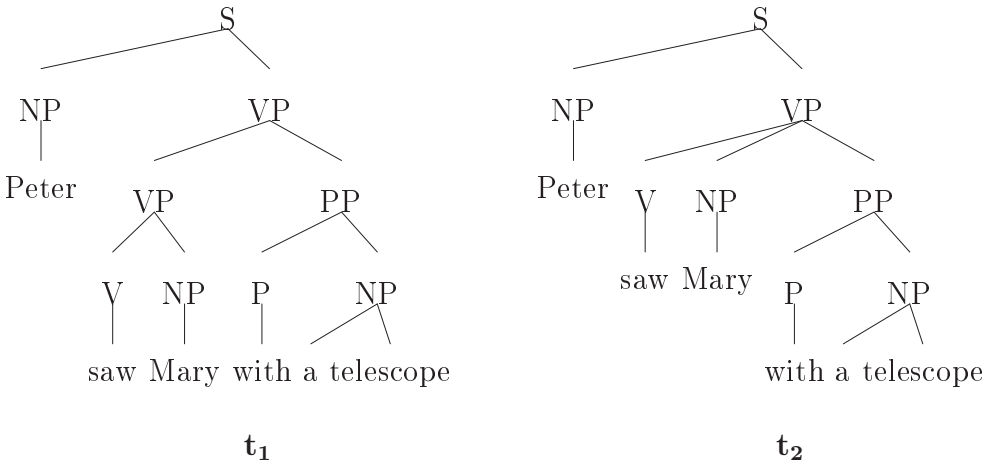
Hierbei ist G_A das Grammatikfragment, der aus allen Regeln mit der linken Seite A besteht, d.h.:

$$G_A = \{r = G | lhs(r) = A\}$$

Hierbei ist lhs(r) die linke Seite *(engl. left-hand-side)* der Regel r.

2.1 Auflösung von Ambiguitäten mit PCFGs

1. **Ambiguität durch PP-Attachment**
   Beispielsatz: „Peter saw Mary with a telescope“



t_1 **t_2**

Der semantische Unterschied zwischen beiden ergibt sich dadurch, dass Die Baumwahrscheinlichkeit für die Bäume t_1 und t_2 ergibt sich nun durch das Produkt der Regelwahrscheinlichkeiten der zur Erzeugung des Baumes ange-

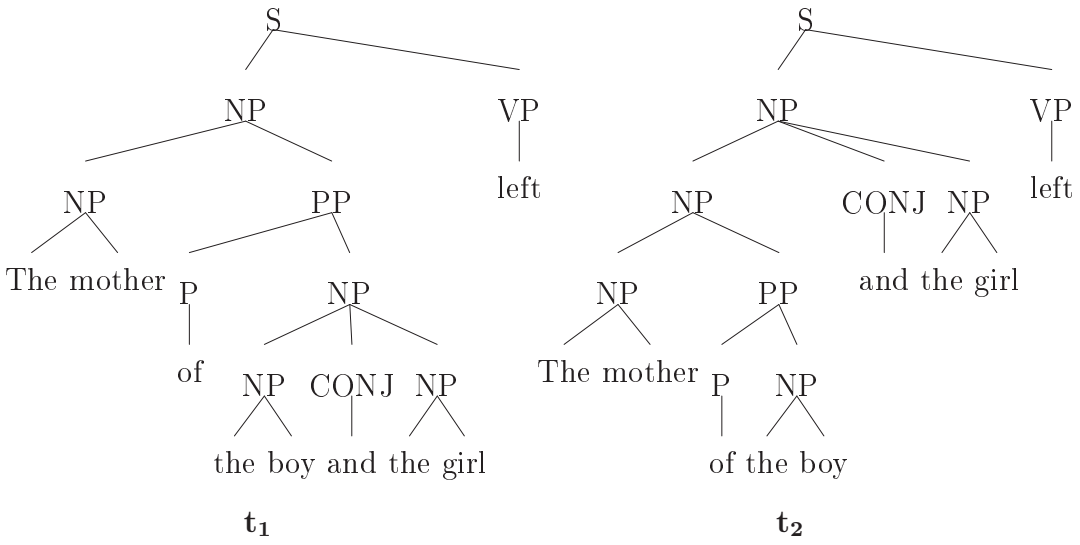
wandten Regeln:

$$
\begin{aligned}
p(t_1) = p(S \to NP\ VP) \cdot\ & p(VP \to VP\ PP) \cdot \\
& p(VP \to V\ NP) \cdot p(PP \to P\ NP) \cdot \\
& p(NP \to \text{Peter}) \cdot p(V \to \text{saw}) \cdot \\
& p(NP \to \text{Mary}) \cdot p(P \to \text{with}) \cdot \\
& p(NP \to \text{a telescope}). \\
p(t_2) = p(S \to NP\ VP) \cdot\ & p(VP \to V\ NP\ PP) \cdot \\
& p(PP \to P\ NP) \cdot p(NP \to \text{Peter}) \cdot \\
& p(V \to \text{saw}) \cdot p(NP \to \text{Mary}) \cdot \\
& p(P \to \text{with}) \cdot p(NP \to \text{a telescope}).
\end{aligned}
$$

Die beiden Baumwahrscheinlichkeiten $p(t_1)$ und $p(t_2)$ unterscheiden sich lediglich durch die oben rot markierten Regeln, welche gerade die eigentliche Ambiguität verursachen. Wollte man nun bspw. die von Baum $p(t_2)$ dargestellte Lesart bevorzugen, müsste man die Wahrscheinlichkeiten der Regeln, die ihn von Baum $p(t_1)$ unterscheiden, erhöhen, denn:

$$p(t_2) > p(t_1) \Leftrightarrow p(VP \to V\ NP\ PP) > p(VP \to VP\ PP) \cdot p(VP \to V\ NP).$$

2. **Durch Konjunktionen verursachte Ambiguität**
   Beispielsatz: „The mother of the boy and the girl left“



Hierbei ergibt sich nun folgendes Problem: Es gibt kontextfreie Grammatiken, so dass keine probabilistische Version dieser Grammatik in der Lage

ist, verschiedene Syntaxanalysen zu disambiguieren. Mögliche Lösungsansätze können sein:

a) Wahrscheinlichkeitsmodell verändern: z.B. könnte die Regelreihenfolge oder die Einbettungstiefe einer Regel im Baum mitberücksichtigt werden.

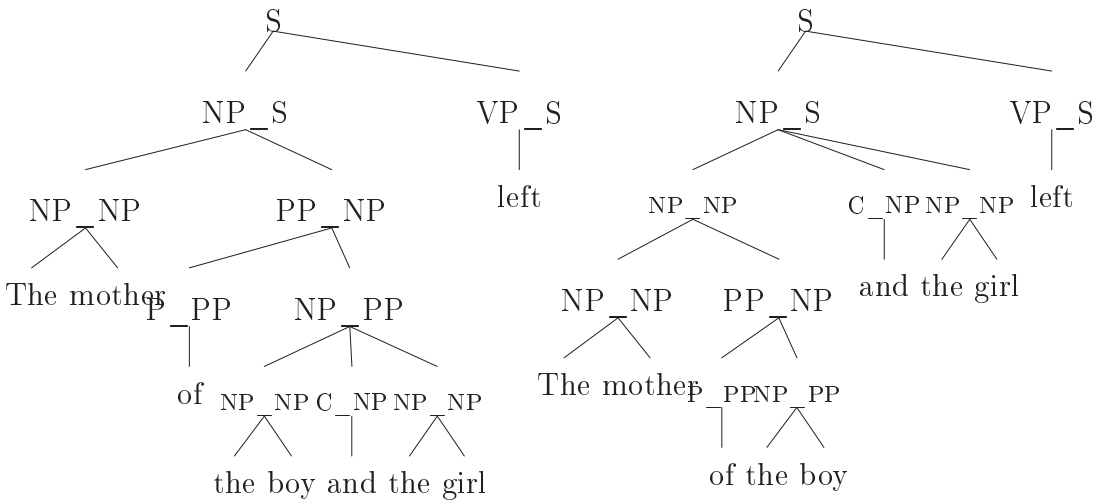
b) Grammatik verändern: in der Regel durch geeignete Grammatiktransformation.

Man präferiert (zum gegenwärtigen Zeitpunkt) den letzteren Ansatz, da man die mathematischen Grundlagen (Baumwahrscheinlichkeit ergibt sich aus dem Produkt der Regelwahrscheinlichkeiten) nicht verändern will.

2.2 Populäre Grammatiktransformationen

1. Lexikalisierung:
   Jedes Lexem (= Terminal) wird entlang gewisser Projektionslinien an Nichtterminale im Baum angefügt.

2. **Parent-Encoding:**
   An jede Tochter (Nichtterminal) wird die Mutter angefügt.



2.3 Baumbank-Training

Definition 2.3 (Baumbankgrammatik). Für eine gegebene **Baumbank**, d.h. für einen nicht-leeren und endlichen Korpus von Bäumen, wird die Baumbankgrammatik $< G, f >$ wie folgt definiert:

1. G ist diejenige kontextfreie Grammatik, die von der Baumbank abgelesen wird.

2. p ist Wahrscheinlichkeitsverteilung auf der Menge der Bäume von G, die durch folgende Regelwahrscheinlichkeiten definiert ist:

$$p(r) = \frac{f(r)}{\sum_{r \in G_A} f(r)} \text{ für alle Regeln } r \in G_A.$$

   Hierbei ist $f(r)$ die Vorkommenshäufigkeit der Regel r in der Baumbank und G_A das Grammatikfragment aller Regeln mit der linken Seite A.

Definition 2.4. Sei G eine kontextfreie Grammatik und sei $\mathcal{X}$ die Menge aller Parsebäume von G. Dann ist das Wahrscheinlichkeitsmodell von G definiert als

$$\mathcal{M}_G = \left\{ p \in \mathcal{M}(\mathcal{X}) \;\middle|\; p(x) = \prod_r p(r)^{f_r(x)} \right\} \forall \text{ Bäume } x \in \mathcal{X},$$

so dass $\sum_{r \in G_A} p(r) = 1$ für alle Grammatikfragmente G_A.

Bemerkungen.

1. Die Forderung, dass alle Regelwahrscheinlichkeiten eines Grammatikfragments von Regeln mit der gleichen Seite A zu 1 aufsummieren müssen nennt man auch *Standardnebenbedingung*.

2. An dieser Stelle stellt sich natürlich die Frage, ob $\widetilde{p}$ in $\mathcal{M}_G$ enthalten ist bzw. (was natürlich gleichbedeutend ist) ob $\mathcal{M}_G$ unbeschränkt ist.

Theorem 2.1. Sei G eine kontextfreie Grammatik und sei $\widetilde{p}$ die relative Häufigkeitsverteilung auf einem nicht-leeren und endlichen Korpus $f_{TB} : \mathcal{X} \mapsto \mathbb{R}$ von Parsebäumen von G. Dann gilt $\widetilde{p} \notin \mathcal{M}_G$, falls:

1. G kann von f_{TB} abgelesen werden.

2. G hat einen Parsebaum $x_\infty \in \mathcal{X}$, der nicht in f_{TB} ist.

Theorem 2.2. Sei f_{tb} ein nicht-leerer und endlicher Korpus von Bäumen. Sei $< G, p_{tb} >$ Charniaks Baumbank-Grammatik. Dann ist p_{tb} eine Maximum-Likelihood-Schätzung von $\mathcal{M}_G$ auf f_{tb}, d.h.:

$$L(f_{tb}, p_{tb}) = \max_{p \in \mathcal{M}_G} L(f_{tb}, p).$$

Außerdem gilt: p_{tb} ist eindeutige Maximum-Likelihood-Schätzung von $\mathcal{M}_G$ auf f_{tb}.

Beweis. Der Beweis zu Theorem 2.2 erfolgt in drei Schritten:

1. Im ersten Schritt ist zu beweisen, dass

   $$L(f_{tb}, p) = L(f, p).$$

   Man beweist also, dass die Korpuswahrscheinlichkeit der Baumbankgrammatik genauso groß ist wie die Korpuswahrscheinlichkeit, die sich aus dem

Korpus aller Regeln und ihrer Wahrscheinlichkeiten ergibt:

$$\begin{aligned} L(f_{tb}, p) &= \prod_{x \in \mathcal{X}} p(x)^{f_{tb}(x)} \\ &= \prod_{x \in \mathcal{X}} \left(\prod_{r} p(r)^{fr(x)} \right)^{f_{tb}(x)} \\ &= \prod_{x \in \mathcal{X}} \prod_{r} p(r)^{fr(x) \cdot f_{tb}(x)} \\ &= \prod_{r} \prod_{x \in \mathcal{X}} p(r)^{fr(x) \cdot f_{tb}(x)} \\ &= \prod_{r} p(r)^{\sum_{x \in \mathcal{X}} f_{tb}(x) \cdot fr(x)} \\ &= \prod_{r} p(r)^{f}(r) \\ &= L(f, p). \end{aligned} \tag{11}$$

2. In einem zweiten Schritt zeigt man nun, dass

$$L(f, p) = \prod_{A} L(f_A, p).$$

Hierbei sei f_A der Korpus aller Regeln, deren linke Seite aus dem Nichtterminal A besteht.

$$\begin{aligned} L(f, p) &= \prod_{r} p(r)^{f(r)} \\ &= \prod_{A} \left(\prod_{r \in G_A} p(r)^{f(r)} \right) \\ &= \prod_{A} \left(\prod_{r \in G_A} p(r)^{f_A(r)} \right) \\ &= \prod_{A} L(f_A, p). \end{aligned} \tag{12}$$

3. Die Kombination aus den Schritten 1 und 2 ergibt nun:

$$L(f_{tb}, p) = \prod_{A} L(f_A, p).$$

D.h. es gilt nun, diesen Term in p zu maximieren. Ein Produkt wird maximal, wenn alle Faktoren maximiert werden, sofern diese unabhängig voneinander

ein. Dies ist hier der Fall, obwohl man zwar stets das gleiche Symbol p für die Wahrscheinlichkeitsverteilungen über verschiedenen Grammatikfragmenten benutzt wird, sind diese jeweils voneinanander unabhängig. Denn auf Grund der Standardnebenbedingung müssen sich alle Regeln innerhalb eines Grammatikfragments mit der gleichen Seite A auf 1 aufsummieren. Somit sind auch die einzelnen Faktoren voneinander unabhängig und das „große" MLE-Problem von f_{tb} auf viele „kleine" Probleme reduziert worden.
Die Frage also: Ist $\mathcal{M}(G_A)$ unbeschränkt?
$\mathcal{M}(G_A)$ ist natürlich unbeschränkt, da uns niemand verbietet, für Regeln innerhalb eines Grammtikfragments beliebige Wahrscheinlichkeiten anzusetzen, um bspw. die Anwendung einer Regel nach unserem Belieben zu bevorzugen. Folglich ist das MLE auf $\mathcal{M}(G_A)$ natürlich RFE, sprich:

$$\hat{p}_A(r) = \widetilde{p_A} = \frac{f_A(r)}{|f_A|}.$$

□

2.4 Unüberwachtes Training*(engl. unsupervised training)* von PCFGs

Bisher: „Überwachtes Training" *(engl. supervised training)* von PCFGs

Kennzeichen des überwachten Trainings:
nutzt ein (in aller Regel manuell) anotiertes Korpus(bisher Baumbank) aus.

Für unüberwachtes Training **gegeben:**

- Korpus von Sätzen
- CFG (in aller Regel **manuell geschrieben**)

Gesucht: PCFG! (trainierte PCFG)

Beispiel.
Gegeben sei ein Korpus

$f(y)$	y
5	y_1
10	y_2

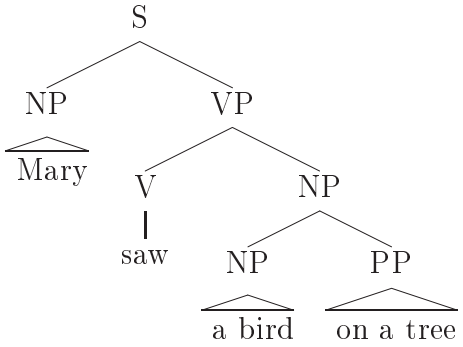
$y_1 =$ „*Mary saw a bird on a tree*" $y_2 =$ „*A bird on a tree saw a worm*"

Ferner sei gegeben:

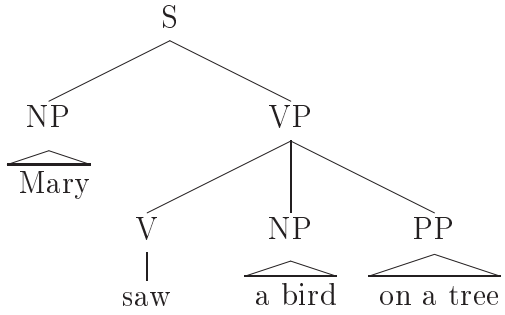
$$
\begin{array}{rcl}
S & \rightarrow & NP\ VP \\
VP & \rightarrow & V\ NP \\
VP & \rightarrow & V\ NP\ PP \\
NP & \rightarrow & NP\ PP \\
NP & \rightarrow & Mary \\
NP & \rightarrow & a\ bird \\
NP & \rightarrow & a\ worm \\
PP & \rightarrow & on\ a\ tree \\
V & \rightarrow & saw
\end{array}
$$

Grobe Idee: Wir stellen uns nun eine Baumbank her!

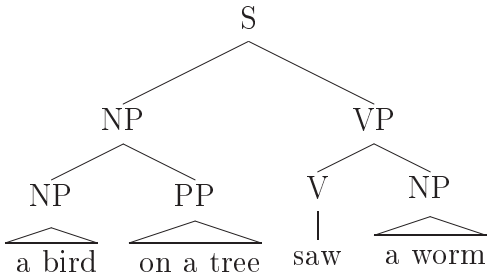
x_1 :



x_2 :



x_3 :



Problem: Wie oft soll jeder Baum in der zu schaffender Baumbank vorkommen?

1. Vorschlag:

$$\text{`}\textit{Initialer Split}\text{`} \qquad \begin{array}{c|c} f(x) & x \\ \hline i_1 & x_1 \\ i_2 & x_2 \\ 10 & x_3 \end{array} \qquad i_1 + i_2 = 5$$

2. Gewöhnliches Baumbank-Training machen

$$\rightarrow \; PCFG$$

3. Mit dieser PCFG werden nun die Wahrscheinlichkeiten der Analysen x_1, x_2 und x_3 gerechnet.

$$\left.\begin{array}{ll} p(x_1) = \ldots z.B. & 0.005 \\ p(x_2) = \ldots & 0.002 \\ p(x_3) = \ldots & 0.01 \end{array}\right\} 5 \quad 0.05 : 0.02$$

   Diese Wahrscheinlichkeiten werden benutzt, um einen neuen Split zu berechnen.

4. neue Baumbank ablesen

5. neues Baumbank-Training, usw.

Grundlegendes Prinzip: („Henne und Ei Problem“) „EM-Algorithmus für PCFGs“

- Man weiss, wie man aus einer PCFG und einem Korpus von Sätzen eine Baumbank gewinnen kann
- Man weiss, wie man aus einer Baumbank eine PCFG rechnen kann.

Lösung: Man erstellt sich eins von beiden mit Gewalt her.(durch Auswürfeln)

3 Der Erwartungswert-Maximierungs-Algorithmus(EM-Alg.)

3.1 Symbolische und statistische Analysemethoden

Definition 3.1. Seien $\mathcal{X}$ und $\mathcal{Y}$ nicht leere und abzählbare Mengen. Eine Funktion

$$\mathcal{A} : \mathcal{Y} \rightarrow 2^{\mathcal{X}}$$

heißt **symbolische Analysekomponente** ***(engl. symbolic analyzer)***, falls die Analysen $\mathcal{A}(y) \subseteq \mathcal{X}$ **paarweise disjunkt** sind, und wenn ihre Vereinigung **komplett** ist, d.h.:

$$\mathcal{X} = \sum_{y \in \mathcal{Y}} \mathcal{A}(y)$$

In diesem Fall heißt $\mathcal{Y}$ die Menge der **unvollständigen Daten** ***(engl. incomplete data)*** und $\mathcal{X}$ heißt die Menge der **vollständigen Daten** ***(engl. complete data)*** In anderen Worten, die Analysen $\mathcal{A}(y)$ formen eine Partition der vollständigen Daten $\mathcal{X}$. Deshalb gibt es für jeden vollständigen Datentyp $x \in \mathcal{X}$ ein eindeutiges $y \in \mathcal{Y}$ mit $x \in \mathcal{A}(y)$. Dieser vollständige Datentyp wird auch yield(x) genannt.

Beispiel.
Symbolisches Parsing

1. **symbolische Analysekomponente $\mathcal{A}$:** Parser (ordnet den Sätzen ihre Analysen zu)
2. **unvollständige Datentypen $\mathcal{Y}$:** die Sätze
3. **vollständige Datentypen $\mathcal{X}$:** die Bäume

In diesem Beispiel:

$$\begin{array}{ll} \mathcal{A} : \mathcal{Y} \to \mathcal{X} & (\mathcal{A} : \textit{Menge der Sätze} \to \textit{Menge der Bäume}) \\ y \to \mathcal{A}(y) & (\textit{Satz} \to \textit{Analysen des Satzes}) \end{array}$$

Zwei Bedingungen:

1. **Disjunktheit**

$$\mathcal{A}(y_1) \cap \mathcal{A}(y_2) = \varnothing \quad \textit{wann immer } y_1 \neq y_2$$

„Kein Baum ist Analyse zweier verschiedenen Sätze“
„Verschiedene Sätze haben verschiedene Analysen“

2. **Komplettheit:**

$$\mathcal{X} = \bigcup_{y \in \mathcal{Y}} \mathcal{A}(y)$$

„Jede Analyse ist Analyse eines Satzes“
„Wir betrachten keine Grammatiken mit überflüßigen Analysen“

Beispiel.
Nummer 2: POS-Tagging

1.**symbolische Analysekomponente $\mathcal{A}$:** Tagger

2.**unvollständige Daten:** Wörter

3.**vollständige Daten:** Wort-Tag Tupel

Disjunktheit:

$wort_1 - tag_1 \quad wort_2 - tag_2 \Longrightarrow (weil \quad wort_1 \neq wort_2) \quad wort_1 - tag_1 \neq wort_2 - tag_2$

Kompletheit: keine nutzlose Wort-Tag Parse benutzt.

Begrifflichkeit:

vollständig	unvollständig
Baum	Satz
Wort-Tag Paar	Wort

$\leftarrow$*Teil von Relation*

3.2 Statistische Analysekomponente

Definition 3.2. (Statistische Analysekomponente) Ein Paar $< \mathcal{A}, p >$ aus einer symbolischen Analysekomponente $\mathcal{A}$ und einer Wahrscheinlichkeitsverteilung p auf den vollständigen Datentypen $\mathcal{X}$ heißt **statistische Analysekomponente**.

- Wir benutzen eine statistische Analysekomponente um **Warscheinlichkeiten für die unvollständigen Daten** $\mathcal{Y}$ wie folgt zu **induzieren**:

$$p(y) = \sum_{x \in \mathcal{A}(y)} p(x) \qquad \forall y \in \mathcal{Y}$$

- Noch wichtiger, wir benutzen eine statistische Analysekomponente, um die Ambiguität eines unvollständigen Datentyp $y \in \mathcal{Y}$ aufzulösen, indem wir die **bedingten Wahrscheinlichkeiten der Analysen** $x \in \mathcal{A}(y)$ heranziehen.

$$p(x|y) =_{def} \frac{p(x)}{p(y)} \qquad y = yield(x) \ (bzw.\ x \in \mathcal{A}(y))$$

Beispiel.
Parsing

$x = Baum/Analyse$	$y = Satz$
$\mathcal{X} = Analysen$	$p(x) = Baum - /Analysewahrscheinlichkeit$
$\mathcal{Y} = Sätze$	$p(y) = Satzwahrscheinlichkeit$

Neu: $p(y)$=Summe aller Analysen-/Baumwahrscheinlichkeiten
(Wobei nur Baumwahrscheinlichkeiten gezählt werden, die Analysen eines Satzes sind $y = yield(x)$)

Frage: Warum **„Summe“** und nicht **„Produkt“**

Interpretation:

Beim Korpus und Baum : Unabhängige Elementarereignisse, die ein „Ganzes“ formen. ($\rightarrow$ „*Konjunktion*“)

Beim Satz: Abhängige Elementarereignisse , von denen jedes Einzelne schon „das Ganze“ formt. ($\rightarrow$ „*Alternativen*“)

Von der Qualität: Menge gerader Augenzahlen beim Würfeln

$$p(\{2,4,6\}) = p(2) + p(4) + p(6)$$

Bemerkung.
Wir prüfen, ob beide Wahrscheinlichkeitsdefinitionen auch wirklich Wahrscheinlichkeiten sind. $p(y)$ ist als Summe von Wahrscheinlichkeiten nicht negativ. Außerdem gilt:

1.

$$\sum_{y\in\mathcal{Y}} p(y) = \sum_{y\in\mathcal{Y}} \sum_{x\in\mathcal{A}(y)} p(x) = \mathcal{X}=\sum_{y\in\mathcal{Y}} \mathcal{A}(y) \sum_{x\in\mathcal{X}} p(x) = 1$$

$$\textit{weil} \qquad p : \mathcal{X} \rightarrow [0,1] \qquad x \rightarrow p(x)$$
$$\textit{eine Wahrscheinlichkeitsverteilung ist } (3.2)$$

2.Annahme: y - fest

$$\sum_{x\in\mathcal{A}(y)} p(x|y) = \sum_{x\in\mathcal{A}(y)} \frac{p(x)}{p(y)} = \frac{1}{p(y)} \sum_{x\in\mathcal{A}(y)} p(x) = \frac{1}{p(y)} * p(y) = 1$$

Definition 3.3. (Input des EM-Algorithmus). Input des EM-Algorithmus ist:

1. eine **symbolische Analysekomponente** $\mathcal{A}$
2. ein nicht-leeres und endliches **Korpus von unvollständigen Daten**
$$f : \mathcal{Y} \rightarrow \mathbb{R}$$
3. Ein **Wahrscheinlichkeitsmodell** $\mathcal{M} \subseteq \mathcal{M}(\mathcal{X})$ für die **vollständigen Daten**. D.h. jedes $p \in \mathcal{M}$ ist von der Gestalt:
$$p : \mathcal{X} \rightarrow [0,1] \quad \textit{und} \quad \sum_{x\in\mathcal{X}} p(x) = 1$$

4. **implizit noch gegeben**: $\mathcal{M} \subseteq \mathcal{M}(\mathcal{Y})$, **Modell auf unvollständigen Daten**, durch die Def.3.2

$$p(y) = \sum_{x \in \mathcal{A}(y)} p(x)$$

5. **Start-Instanz** $p_0 \in \mathcal{M}$

Definition 3.4. (Die Prozedur des EM-Algorithmus) ist:

(1) for each $i = 1,\ 2,\ 3,\ \ldots$ do
(2) $q := p_{i-1}$
(3) **E-step:** berechne das **von** q **erwartete Korpus** $f_q \colon \mathcal{X} \to \mathbb{R}$ mittels

$$f_q(x) \ := f(y) \cdot q(x|y) \qquad \text{wobei } y = \text{yield}(x)$$

(4) **M-step:** berechne ein maximum-likelihood estimate $\hat{p}$ von $\mathcal{M}$ auf f_q

$$L(f_q, \hat{p}) = \max_{p \in \mathcal{M}} L(f_q, p)$$

(5) $p_i := \hat{p}$
(6) end // for each i
(7) print $p_0, p_1, p_2, p_3, \ldots$

Bemerkungen.

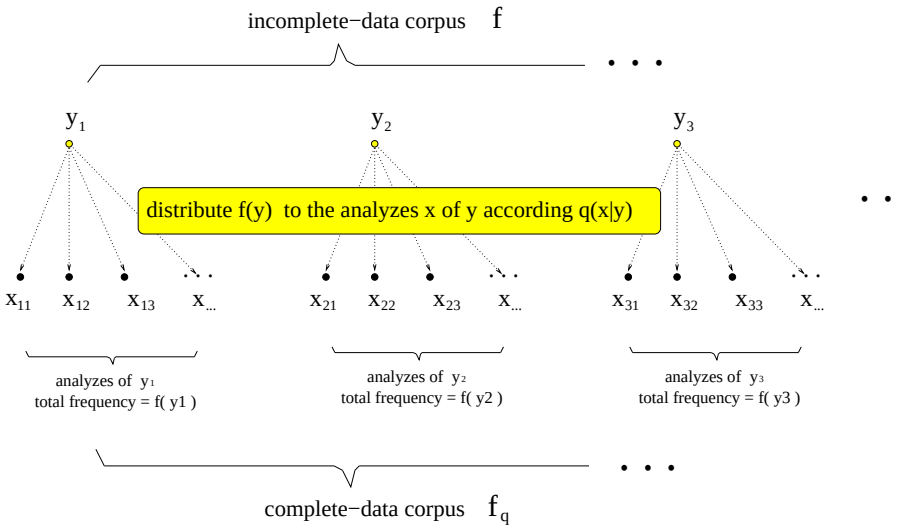


Abbildung 8: E step vom EM-Algorithmus

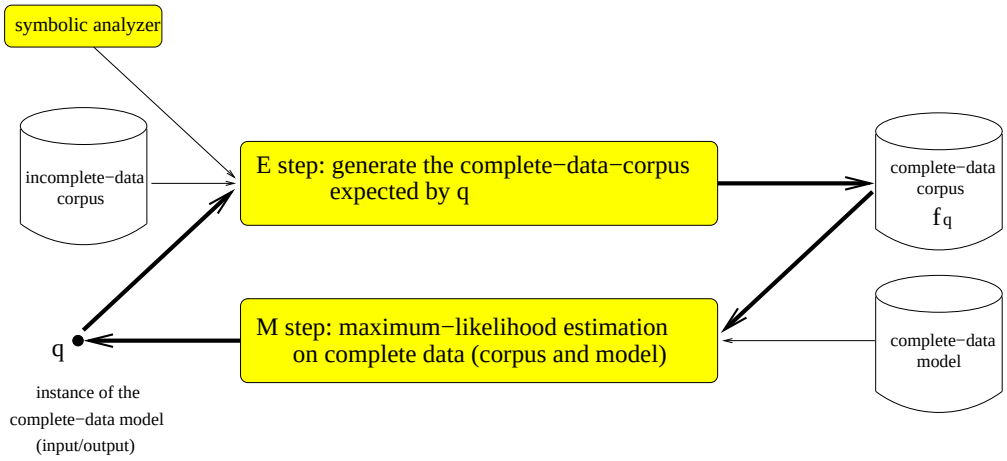


Abbildung 9: Prozedur vom EM-Algorithmus.

Zwei Fragen:

1. (y-fest)

$$\sum_{x\in\mathcal{A}(y)} f_q(x) = \sum_{x\in\mathcal{A}(y)} f(y) * q(x|y) = f(x)\sum_{x\in\mathcal{A}(y)} q(x|y) = f(y)\sum_{x\in\mathcal{A}(y)} \frac{q(x)}{q(y)} =$$

$$= f(y) * \frac{1}{q(y)} \sum_{x\in\mathcal{A}(y)} q(x) = f(y) * \frac{1}{q(y)} * q(y) = f(y)$$

2.

$$\sum_{x\in\mathcal{X}} f_q(x) =_{weil\ Partition(Vertauschen\ der\ Summationsreihenfolge)} \sum_{y\in\mathcal{Y}}\sum_{x\in\mathcal{A}(y)} f_q(x) =$$

$$=_{Antwort\ auf\ 1.Frage} \sum_{y\in\mathcal{Y}} f(y) = |f|$$

D.h. jedes $y \in \mathcal{Y}$ und jedes $x \in \mathcal{A}(y)$ ist für die Frequenzverteilung wichtig. Es wird nichts hinzugefügt, oder entfernt um das erwartete Korpus von vollständigen Daten zu bilden/generieren!

3.3 Output des EM-Algorithmus

Theorem 3.1. Der Output des EM-Algorithmus ist eine Sequenz von Instanzen des Modells $\mathcal{M} \subseteq \mathcal{M}(\mathcal{X})$, den sogennaten EM re-estimates,

$$p_0, p_1, p_2, p_3, ...$$

sodaß die Wahrscheinlichkeiten des Unvollständigen-Daten-Korpus f,(die diesen von jeweiligen Instanzen zugewiesen wird) monoton wachsend ist:

$$L(f, p_0) \leq L(f, p_1) \leq L(f, p_2) \leq ...$$

Bemerkung.
Im allgemeinen gilt, dass der EM-Algorithmus zu einem MLE führen könnte, aber nicht unbedingt führen muss!

Beispiel.
Pathologische Fälle:

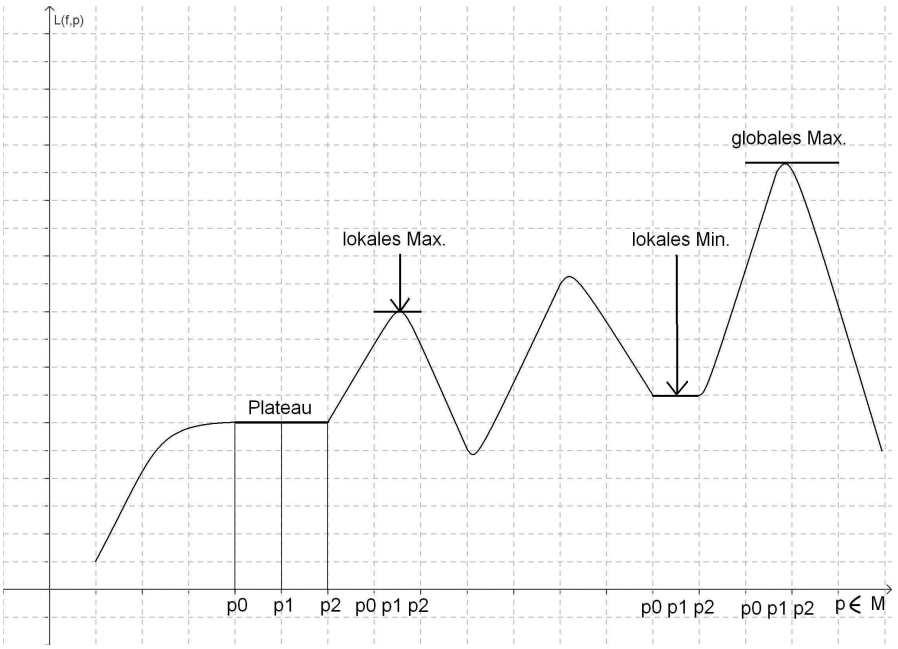


Abbildung 10: Pathologische Fälle

„Abhilfe“: Vielmaliges Wiederholen der gesamten EM-Prozedur mit jeweils verschiedenen Startparametern.(teuer!)

Kosten im Vergleich zu MLE auf vollständigen Daten:

1. ein M-Schritt $\approx$ eine „normale“ MLE
2. ein E-Schritt : Kosten vernachlässigen wir auch (in der Praxis: meist Vergleichbar von E- und M-Schritt mittels dynamischer Programmierung)

$\Longrightarrow$ **Kosten vom EM:** Anzahl Iterationen * Anzahl Startparametern * Kosten einer normalen MLE.

Frage: Warum/Wann macht man dann EM?

- Baumbank-Herstellung teuer (Herstellung vollständigen Daten ist teuer)

Gegenwart: Grammatik-Schreiben ist ziemlich teuer!(Bsp.:CFG in Stuttgart,HPSG in Saarbrücken)

- Alternative zu EM
  - Grammatik
  - Korpus von Sätzen

  $\Longrightarrow$ im Prinzip könnten wir auch MLE eines Grammatik-Modells auf einem Korpus von Sätzen machen.

Frage: Was ist die Wahrscheinlichkeit des Korpus der vollständigen Daten bezüglich einer Instanz $p \in \mathcal{M} \subseteq \mathcal{M}(\mathcal{Y})$

- **gegeben:** $f : y \to \mathbb{N}$
- **gesucht:** $L(f, p) \leftarrow (Bem. : \ dieselben\ Wahrscheinlichkeiten, wie\ im\ letzten\ Theorem)$

$$L(f,p) =_{def} \prod_{y \in \mathcal{Y}} p(y)^{f(y)} = \prod_{y \in \mathcal{Y}} (\sum_{x \in \mathcal{A}(x)} p(x))^{f(y)}$$

Was ist also MLE hier in diesem Fall?

Antwort:

$$\hat{p} = argmax_{p \in \mathcal{M}} \prod_{y \in \mathcal{Y}} (\sum_{x \in \mathcal{A}(y)} p(x))^{f(y)}$$

Falls ihr dies lösen könnt, tut es! Benutzt nicht EM. Die Minimierung eines Produkts von Summen ist in der Regel sehr schwer;

$\Longrightarrow$ man muss in der Regel EM wählen!

Rezept: Falls MLE auf unvollständigen Daten möglich ist, dann macht man iterierte MLE auf vollständigen Daten, d.h. EM.