Regel-basiertes Tagging

Morphologieanalyse und Lexikonaufbau (7. Vorlesung)

Dozent: Gerold Schneider

Übersicht

Regel-basiertes Tagging (Brill-Tagging)

Lit.: [Brill 92], [Brill 94]

Die CL-Gruppe in Zürich hat den Brill-Tagger für das Deutsche trainiert.

Vorteile von regelbasierten Systemen:

- weniger Information zu verwalten
- Übersichtlichkeit der Regeln
- leichte Veränderbarkeit des Taggers (Publikation)
- bessere Portabilität

Brills Tagger:

- lernt Regeln selbständig
- kommt ohne externes Lexikon aus. (Baut eigenes Lexikon auf)
- basiert auf einem getaggten Corpus.

Lernphase des Taggers

1. Zuerst werden POS-Wahrscheinlichkeiten aus einem getaggten Korpus (hier Brown Corpus) ermittelt. Damit wird ein Vollformenlexikon aufgebaut. (z.B. die höchste Wahrscheinlichkeit für *run* ist Verb)
2. *Lexikalische Regeln*: Präfix- und Suffixwahrscheinlichkeiten werden aus dem getaggten Korpus automatisch ermittelt. (z.B. ein Wort auf -*ous* ist wahrscheinlich Adjektiv)
3. *Kontextregeln*: Der Algorithmus wendet seine gelernten POS-Wahrscheinlichkeiten an und vergleicht seine Tagging-Ergebnisse mit den im getaggten Corpus vorgegebenen und leitet Änderungsregeln zu vorgegebenen Regelmustern ab: ***Change tag `a' to tag `b' when:***
    1. The preceding (following) word is tagged `z'.
    2. The word two before (after) is tagged `z'.
    3. One of the two preceding (following) words is tagged `z'.
    4. One of the three preceding (following) words is tagged `z'.
    5. The preceding word is tagged `u' and the following word is tagged `z'.
    6. The preceding (following) word is tagged `u' and the word two before (after) is tagged `z'.
    7. The preceding (following) word is `w'.
    8. The word two before (after) is `w'.
    9. One of the two preceding (following) words is `w'.
    10. The current word is `v' and the preceding (following) word is `w'.
    11. The current word is `w' and the preceding (following) word is tagged `z'.

    Für jede Regelvariante `<tag_a, tag_b, Variantennummer>` wird berechnet, wie oft sie richtige und wie oft sie falsche Ergebnisse liefert. Die Differenz ergibt die Verbesserungssumme. Die Regelvariante mit der besten Verbesserungssumme wird angewendet.

Bsp. für Kontextregeln: Der Tagger markiert ursprünglich 159 Wörter als Verben, die Nomen sein sollten. Mit der Regel "Ändere Tag von Verb zu Nomen, falls eines der zwei vorhergehenden Wörter als Determiner getaggt ist." werden 98 Fälle (von den 159) korrigiert, aber es werden 18 andere Fehler erzeugt. Die Verbesserungssumme ist also (98-18=) 80.

Bsp. für zwei vom englischen System ermittelte Regeln:

```
1.   TO IN Next-tag AT
```

Ein mit `TO` (*to*-Infinitiv) getaggtes Wort wird mit `IN` (Präposition) getaggt, falls das nächste Wort mit `AT` (Artikel) getaggt ist.

```
2.   VBN VBD Prev-Word-is-cap Yes
```

Ein mit `VBN` (Past Part. Verb) getaggtes Wort wird mit `VBD` (Past Verb) getaggt, falls das vorhergehende Wort mit Grossbuchstaben beginnt (d.h. ein Eigenname ist).

Bsp. für eine vom deutschen System ermittelte Regel:

```
VVFIN VAFIN Next1or2or3tag VVPP
```

Diese Regel kommt bei Verben zum Einsatz, die gemäss Lexikon sowohl Voll- als auch Hilfsverben sein können, also vor allem bei *sein* und *haben*.. Sie transformiert ein finites Hauptverb (VVFIN) in ein finites Hilfsverb (VAFIN) falls ein Partizip innerhalb der folgenden drei Worte folgt.

Anwendungsphase des Taggers

1. Jedem Wort wird das nach dem Lexikon wahrscheinlichste Tag zugewiesen.
2. Unbekannte Wörter, die mit einem Grossbuchstaben beginnen, werden als Namen angesehen.
3. Unbekannte Wörter, die mit einem Kleinbuchstaben beginnen, werden aufgrund ihrer Endung klassifiziert, aufgrund der *lexikalischen Regeln* aus der Lernphase (z.B. *blablaous* als Adj). Zusätzlich wird für unbekannte Wörter eine Sammlung von aus dem Trainingskorpus extrahierten Bigrammen (wie beim statistischen Tagging) zu Hilfe gezogen.
4. Die in der Lernphase gelernten Kontextregeln werden angewendet. (siehe Beispiele oben)

Der Algorithmus für die englische Version hat eine Fehlerquote von 7,9%, wenn nur die Schritte 1-3 angewendet werden. Wenn auch die 71 automatisch ermittelten Kontextregeln eingesetzt werden, verbessert sich das System von 7,9% auf 5,1% Fehlerrate.

Typische Taggingfehler (im Deutschen)

Faule und übereifrige Kontextregeln

Zur obigen Regel `VVFIN VAFIN Next1or2or3tag VVPP` : Steht ein Partizip weiter weg als drei Tags vom provisorisch als Vollverb getaggten Kandidaten, so vermag sie das Vollverb nicht mehr in ein Hilfsverb zu verwandeln. Umgekehrt auch:

```
Der Brief ist lang.
```

richtig getaggt, aber die Regel verwandelt das *ist* im folgenden Satz zu unrecht in ein Hilfsverb:

```
Der Brief ist lang, erreicht hat er aber nichts.
```

Eine richtige Syntaxanalyse könnte hier Abhilfe schaffen, wäre aber rechnerisch wesentlich aufwendiger.

Ohne Syntaxregeln hat der Tagger auch grosse Schwierigkeiten, die Relativpronomen *der, die, den* etc. von Artikeln zu unterscheiden.

Eifrige lexikalische Regeln

Mit Worten, die nicht im Lexikon stehen, stellen lexikalische Regeln allerlei sinnvolles und sinnloses an. Während die weiter oben zitierte Regel

```
bar hassuf 3 ADJD 5
```

meist sinnvoll ist, wird durch sie das unbekannte Wort *Privatbar* auch zu einem Adjektiv gemacht. Da viele Adverbien in -ch enden, wird der unbekannte *Hirsch* durch eine ähnlich fleissige Regel vorläufig leider zum Adjektiv.

Ein Vergleich von statistischem und regelbasiertem Tagging für das Deutsche **von Martin Volk und Gerold Schneider (Publikation)**

==> **Brills These:** Regelbasiertes Tagging ist genauso gut wie probabilistisches Tagging. Mit Hilfe eines Lexikons kann es noch verbessert werden.

Untersuchte Tagger:

1. Statistischer Tagger: TreeTagger von der Universität Stuttgart (entwickelt von Helmut Schmid)
2. Regelbasierter Tagger: Brill-Tagger

Korpus

rund 70'000 Wörter aus der Frankfurter Rundschau (manuell getaggt)

- Trainingskorpus (7/8): 60'710 Tokens
- Testkorpus (1/8): 8'887 Tokens; mit einer durchschnittlichen Mehrdeutigkeit von 1,5 Tags/Token für alle Token, die im Lexikon enthalten sind. 1342 Tokens aus dem Testkorpus kommen im Trainingskorpus nicht vor und sind deshalb nicht im Lexikon enthalten.

Tagset

STTS (Stuttgart-Tübingen Tagset) mit 54 Tags einschl. 3 Tags für Satzzeichen

Training

- Beim TreeTagger: Dauer ungefähr 2 Minuten; Ausgabe-Datei rund 630 kByte. Das Lexikon muss vor dem Training berechnet werden und wird in die Ausgabe-Datei hineinkompiliert.
- Beim BrillTagger: Dauer rund 30 Stunden (!!); Ausgabe:
  1. Vollformen-Lexikon mit 14'147 Einträgen (212 kByte)
  2. 329 Kontext-Regeln (8 kByte)
  3. 378 Lexikalische Regeln (9 kByte)
  4. Bigram-Liste mit 42'270 Einträgen (609 kByte)

Tagging des Testkorpus mit dem TreeTagger

ambiguity	tokens	in %	correct	in %	lexical errors	in %	disambig. errors	in %
0	1342	15.10	1128	84.05	214	15.95	0	0.00
1	5401	60.77	5330	98.69	71	1.31	0	0.00
2	993	11.17	929	93.55	3	0.30	61	6.14
3	795	8.95	757	95.22	0	0.00	38	4.78
4	260	2.93	240	92.31	0	0.00	20	7.69
5	96	1.08	83	86.46	0	0.00	13	13.54
total	8887	100.00	8467	95.27	288	3.24	132	1.49

Fehlertypen

- Lexical errors: Das korrekte Tag ist nicht im Lexikon, und der Tagger rät ein inkorrektes Tag.
- Disambiguation errors: Das korrekte Tag ist im Lexikon, aber der Tagger wählt ein falsches Tag.

Tagging des Testkorpus mit dem Brill-Tagger

ambiguity	tokens	in %	correct	in %	lexical errors	in %	disambig. errors	in %
0	1342	15.10	1094	81.52	248	18.48	0	0.00
1	5401	60.77	5330	98.69	71	1.31	0	0.00
2	993	11.17	906	91.24	3	0.30	84	8.46
3	795	8.95	758	95.35	0	0.00	37	4.65
4	260	2.93	245	94.23	0	0.00	15	5.77
5	96	1.08	87	90.62	0	0.00	9	9.38
total	8887	100.00	8420	94.75	322	3.62	145	1.63

Vergleich der Ergebnisse

- TreeTagger: Besser bei unbekannten Wörtern
- BrillTagger: Besser bei mehrfach (4-fach oder 5-fach) ambigen Wörtern

TreeTagger Fehler			**Brill-Tagger Fehler**		
Anzahl	korrektes Tag	Tagger-Tag	Anzahl	korrektes Tag	Tagger-Tag
48	NE	NN	54	NE	NN
21	VVINF	VVFIN	31	NN	NE
20	NN	NE	19	VVFIN	VVINF
17	VVFIN	VVINF	19	VVFIN	ADJA
10	VVPP	VVFIN	17	VVINF	VVFIN
10	VVFIN	VVPP	15	VVPP	VVFIN
8	CARDNUM	VMPP	11	VVPP	ADJD
7	ADJD	VVFIN	11	ADJD	VVFIN
7	ADJD	ADV	8	VVINF	ADJA

Tagging des Testkorpus mit einer Kombination von Gertwol und Tagger

Um das Problem der Erkennung unbekannter Wörter einzudämmen, kann man ein `externes' Lexikon zuschalten. Z.B. kann man alle unbekannten Wörter zunächst von Gertwol analysieren lassen, die Gertwol-Ausgabe auf die möglichen Tags abbilden und dann dem Tagger-Lexikon hinzufügen. Dadurch kann man die Tagger-Genauigkeit weiter verbessern. Die besten Ergebnisse erzielten wir mit der Kombination von Gertwol und dem TreeTagger.

ambiguity	tokens	in %	correct	in %	lexical errors	in %	disambig. errors	in %
0	109	1.23	72	66.06	37	33.94	0	0.00
1	6307	70.97	6209	98.45	98	1.55	0	0.00
2	1224	13.77	1119	91.42	10	0.82	95	7.76
3	852	9.59	805	94.48	2	0.23	45	5.28
4	296	3.33	266	89.86	0	0.00	30	10.14
5	99	1.11	86	86.87	0	0.00	13	13.13
total	8887	100.00	8557	96.29	147	1.65	183	2.06

Ein Vergleich von statistischem und regelbasiertem Tagging für das Französische

Lit.: [Chanod und Tapanainen 95]: "Tagging French - comparing a statistical and a constraint-based method"; Online-Version (Postscript 130 KByte).

Die untersuchte **statistische Methode** entspricht der von Cutting et al. entwickelten und führte auch für Französisch zu 96,8% korrektem Tagging.

Das Ändern der `Parser-Tendenz' (engl. *bias*) ist manchmal sehr kompliziert. Die Sequenz

`Det N N/V Präp (Wie in *Le train part à cinq heures* .)`

wird oft falsch disambiguiert. Der Tagger bevorzugt die N-Lesart für das Verb. Zwei Tendenzen wurden hinzugefügt:

```
Auf ein Singular-Nomen folgt meist kein Nomen.
Auf ein Singular-Nomen folgt oft ein Singular-3.Pers.-Verb.
```

Danach wurde der obige Satz richtig disambiguiert, aber die Fehlerrate insgesamt stieg um 50%.

Die **constraint-basierte Methode** (nach Chanod, Tapanainen)

Motivation:

1. In einem Zeitungscorpus mit 1 Mio laufenden Wörtern machen die 16 häufigsten ambigen Wortformen 50% aller Ambiguitäten aus. (Zwei Drittel aller Ambiguitäten gehen auf die 97 häufigsten Wortformen zurück.)

2. Die häufigsten ambigen Wortformen sind corpus-unabhängig.

Methode:

1. Für die häufigsten ambigen Wortformen werden Regeln aufgestellt, die kontextuelle Bedingungen angeben. (Dadurch wird z.B. die Mehrdeutigkeit zwischen Clitic und Determiner für *le* oder *la* geklärt.)

   ```
   Je le veux.                   (Ich will es.)
   Je travaille dans le jardin.  (Ich arbeite im Garten.)
   ```

   Einige dieser häufigsten ambigen Wortformen haben sehr seltene Lesarten: Die Hilfsverben *a* und *est* können auch Nomen sein. Für diese Fälle wird genau festgelegt, wie der Kontext aussehen muss, damit diese Wortformen die seltene Lesart haben können. In allen anderen Fällen wird die wahrscheinlichere Lesart angenommen.

2. Für schwierigere Fälle werden kontextuelle Heuristiken aufgestellt.

   Bsp. Unterscheidung zwischen *des* als Determiner bzw. kontrahierter Präposition-Determiner

   ```
   Jean mange des pommes.
   Jean aime le bruit des vagues.
   ```

   Eindeutige Regelung nur über Verbsubkategorisierung möglich. Hier Heuristiken:

   - Det-Lesart, wenn *des* am Satzanfang
   - Prep-Det-Lesart, wenn *des* hiner einem Substantiv

3. Für weitere Probleme werden nicht-kontextuelle Heuristiken aufgestellt. Sie entsprechen lexikalischen Wahrscheinlichkeiten. (Die Autoren raten, welche Lesart wahrscheinlicher ist.) Bsp.:

   ```
   Präposition vor Adjektiv
   Pronomen vor Partizip Perfekt
   ```

   Werden nur auf die Fälle angewendet, die durch die vorherigen Schritte nicht disambiguiert werden konnten.

System:

Regeln und Heuristiken sind als Transducer implementiert.

```
39 Regeln
25 kontextuelle Heuristiken
11 nicht-kontextuelle Heuristiken
```

Leistung:

Test A: Corpus mit 255 Sätzen (5752 Wörter)

=> 54% Wörter sind mehrdeutig. Nach Anwendung aller Regeln: 1,3% Fehlerrate (s. Tabelle 1)

Test B: Zeitungscorpus mit 12.000 Wörtern (mit Schreibfehlern und vielen Eigennamen)

Nach Anwendung aller Regeln: 2,5% Fehlerrate (s. Tabelle 2)

Kombination von statistischem und constraint-basiertem Tagging

Versuchsanordnung:

1. Einsatz des constraint-basierten Taggers ohne die nicht-kontextuellen Heuristiken (Aus Zeitungskorpus mit 12.000 Wörtern bleiben 1400 mehrdeutig.)
2. Einsatz des statistischen Taggers unabhängig vom vorherigen Lauf des constraint-basierten Taggers. Für die Fälle, wo der constraint-basierte Tagger keine Eindeutigkeit herstellt, wird das vom statistischen Tagger ermittelte Tag genommen. (Erzeugt 220 Fehler auf die 1400 Mehrdeutigkeiten.)
3. Die verbleibenden Mehrdeutigkeiten (0,5%) werden durch die nicht-kontextuellen Heuristiken behandelt. (Erzeugt nur 150 Fehler auf die 1400 Mehrdeutigkeiten.)

Fehleranalyse

1. Fehler durch Mehr-Wort-Ausdrücke (15 Fehler).

   Lösung: Lexikalisierung der Ausdrücke

2. korrigierbare Fehler (41 Fehler)

   Lösung: Korrektur und Ergänzung der Regeln (das vorliegende Ergebnis wurde unter Zeitbegrenzung erzielt.)

   Bsp.: "Prep + Clitic + Fin-Verb" war nicht verboten und wurde anwendet auf

   à l'est

3. problematische (schwer zu korrigierende) Fehler (28 Fehler)
