AIMS

Arbeitspapiere des Instituts für Maschinelle Sprachverarbeitung
Lehrstuhl für Theoretische Computerlinguistik Universität Stuttgart

1998 • VOL. 4 • NO. 3

Inducing Lexicons with the EM Algorithm

Previous Issues of AIMS:

Vol.1 (1) 1994: *Wigner Distribution in Speech Research.* Master thesis by Wolfgang Wokurek (in German), papers by Grzegorz Dogil, Wolfgang Wokurek, and Krysztof Marasek (in English), and a bibliography.

Vol.2 (1) 1995: *Sprachentstörung.* Doctoral Dissertation. University of Vienna, 1994 by Wolgang Wokurek (in German with abstract in English). Full title: Sprachentstörung unter Verwendung eines Lautklassendetektors (Speech enhancement using a sound class detector).

Vol.2 (2) 1995: *Word Stress.* Master thesis by Stefan Rapp (in German) and papers mostly by Grzegorz Dogil, Michael Jessen, and Gabriele Scharf (in English).

Vol.2 (3) 1995: *Language and Speech Pathology.* Master theses by Gabriele Scharf and by Jörg Mayer (in German) and papers by Hermann Ackermann, Ingo Hertrich, Jürgen Konczak, and Jörg Mayer (mostly in German).

Vol.3 (1) 1997: *Tense versus Lax Obstruents in German.* Revised and expanded version of Ph.D. Dissertation, Cornell University, 1996 by Michael Jessen (in English). Full title: Phonetics and phonology of the tense and lax obstruents in German.

Vol.3 (2) 1997: *Electroglottographic Description of Voice Quality.* Habilitationsschrift, University of Stuttgart, 1997 by Krysztof Marasek (in English).

Vol.3 (3) 1997: *Aphasie und Kernbereiche der Grammatiktheorie (Aphasia and core domains in the theory of grammar).* Doctoral Dissertation, University of Stuttgart, 1997 by Annegret Bender (in German with abstract in English).

Vol.3 (4) 1997: *Intonation and Bedeutung (Intonation and meaning).* Doctoral Dissertation, University of Stuttgart, 1997 by Jörg Mayer (in German with abstract in English).

Vol.3 (5) 1997: *Koartikulation und glottale Transparenz (Coarticulation and glottal transparency).* Doctoral Dissertation, University of Bielefeld, 1997 by Kerstin Vollmer (in German with abstract in English).

Vol.3 (6) 1997: *Der TFS-Repräsentationsformalismus und seine Anwendung in der maschinellen Sprachverarbeitung (The TFS Representation Formalism and its Application to Natural Language Processing).* Doctoral Dissertation, University of Stuttgart, 1997 by Martin C. Emele (in German).

Vol.4 (1) 1998: *Automatisierte Erstellung von Korpora für die Prosodieforschung (Automated generation of corpora for prosody research).* Doctoral Dissertation, University of Stuttgart, 1998 by Stefan Rapp (in German with abstract in English).

Vol.4 (2) 1998: *Theoriebasierte Modellierung der deutschen Intonation für die Sprachsynthese (Theory-based modelling of German intonation for speech synthesis).* Doctoral Dissertation, University of Stuttgart, 1998 by Gregor Möhler (in German with abstract in English).

Inducings Lexicons with the EM Algorithm

Mats Rooth, Stefan Riezler, Detlef Prescher, Sabine Schulte im Walde, Glenn Carroll, and Franz Beil

Lehrstuhl für Theoretische Computerlinguistik
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart

CIP-Kurztitelaufnahme der Deutschen Bibliothek

Rooth, Mats:
Inducing Lexicons with the EM Algorithm / Mats Rooth; Stefan Riezler; Detlef Prescher; Sabine Schulte im Walde; Glenn Carroll; Franz Beil. - Stuttgart, 1998.
AIMS - Arbeitspapiere des Instituts für Maschinelle Sprachverarbeitung, Vol. 4, No. 3, 1998, Stuttgart, Germany.
ISSN 1434-0763

Copyright © by the authors

Lehrstuhl für Theoretische Computerlinguistik
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Azenbergstraße 12
70174 Stuttgart

www: `http://www.ims.uni-stuttgart.de/projekte/gramotron/`
email: `gramotron@ims.uni-stuttgart.de`

Contents

Inside-Outside Estimation of a Lexicalized PCFG for German

– Gold –

Franz Beil, Glenn Carroll, Detlef Prescher, Stefan Riezler, and Mats Rooth

Abstract

The paper describes an extensive experiment in inside-outside estimation of a lexicalized probabilistic context free grammar for German verb-final clauses. Grammar and formalism features which make the experiment feasible are described. Successive models are evaluated on precision and recall of phrase markup.

4.1 Introduction

Charniak (1995) and Carroll and Rooth (1998) present head-lexicalized probabilistic context free grammar formalisms, and show that they can effectively be applied in inside-outside estimation of syntactic language models for English, the parameterization of which encodes lexicalized rule probabilities and syntactically conditioned word-word bigram collocates. The present paper describes an experiment where a slightly modified version of Carroll and Rooth's model was applied in a systematic experiment on German, which is a language with rich inflectional morphology and free word order (or rather, compared to English, free-er phrase order). We emphasize techniques which made it practical to apply inside-outside estimation of a lexicalized context free grammar to such a language. These techniques relate to the treatment of argument cancellation and scrambled phrase order; to the treatment of case features in category labels; to the category vocabulary for nouns, articles, adjectives and their projections; to lexicalization based on uninflected lemmata rather than word forms; and to exploitation of a parameter-tying feature.

4.2 Corpus and morphology

The data for the experiment is a corpus of German subordinate clauses extracted by regular expression matching from a 200 million token newspaper corpus. The clause length ranges between four and 12 words. Apart from infinitival VPs as verbal arguments, there are no further clausal embeddings, and the clauses do not contain any punctuation except for a terminal period. The corpus contains 4128873 tokens and 450526 clauses which yields an average of 9.16456 tokens per clause. Tokens are automatically annotated with a list of part-of-speech (PoS) tags using a computational morphological analyser based on finite-state technology (Karttunen et al. (1994), Schiller and Stöckert (1995)).

A problem for practical inside-outside estimation of an inflectional language like German arises with the large number of terminal and low-level non-terminal categories in the grammar resulting from the morpho-syntactic features of words. Apart from major class (noun, adjective, and so forth) the analyser provides an ambiguous word with a list of possible combinations of inflectional features like gender, person, number (cf. the top part of Fig. 4.1 for an example ambiguous between nominal and adjectival PoS; the PoS is indicated following the '+' sign; the features enclosed between '^' and '+' indicate the derivation if available; the entry is headed by the lemma of the analysandum; `Pos` is the feature for an adjective's positive form as opposed to comparative or superlative; the '*'-sign marks uppercase forms.). In order to reduce the number of parameters to be estimated, and to reduce the size of the parse forest used in inside-outside estimation, we collapsed the inflectional readings of adjectives, adjective derived nouns, article words, and pronouns to a single morphological feature (see separated last line in Fig.

```
analyze> Deutsche
1. deutsch^ADJ.Pos+NN.Fem.Akk.Sg
2. deutsch^ADJ.Pos+NN.Fem.Nom.Sg
3. deutsch^ADJ.Pos+NN.Masc.Nom.Sg.Sw
4. deutsch^ADJ.Pos+NN.Neut.Akk.Sg.Sw
5. deutsch^ADJ.Pos+NN.Neut.Nom.Sg.Sw
6. deutsch^ADJ.Pos+NN.NoGend.Akk.Pl.St
7. deutsch^ADJ.Pos+NN.NoGend.Nom.Pl.St
8. *deutsch+ADJ.Pos.Fem.Akk.Sg
9. *deutsch+ADJ.Pos.Fem.Nom.Sg
10. *deutsch+ADJ.Pos.Masc.Nom.Sg.Sw
11. *deutsch+ADJ.Pos.Neut.Akk.Sg.Sw
12. *deutsch+ADJ.Pos.Neut.Nom.Sg.Sw
13. *deutsch+ADJ.Pos.NoGend.Akk.Pl.St
14. *deutsch+ADJ.Pos.NoGend.Nom.Pl.St

==>   Deutsche  { ADJ.E, NNADJ.E }
```

Figure 4.1: Collapsing Inflectional Features

```
während   { ADJ.Adv, ADJ.Pred, KOUS, APPR.Dat, APPR.Gen }
sich  { PRF.Z }
das   { DEMS.Z, ART.Def.Z }
Preisniveau   { NN.Neut.NotGen.Sg }
dem   { DEMS.M, ART.Def.M }
westdeutschen    { ADJ.N }
annähere  { VVFIN }
. { PER }
```

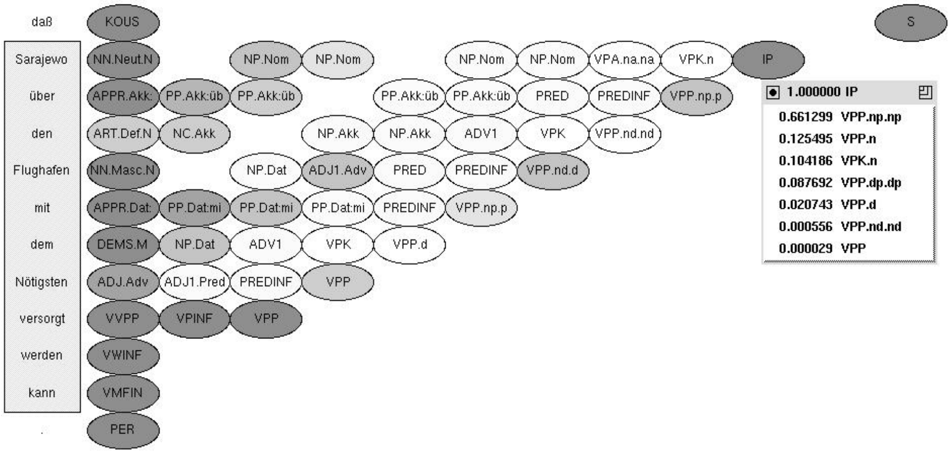
Figure 4.2: Corpus Clip

4.1 for an example). This reduced the number of low-level categories, as exemplified in Fig. 4.2: *das* has one reading as an article and one as a demonstrative; *westdeutschen* has one reading as an adjective, with its morphological feature N indicating the inflectional suffix.

We use the special tag `UNTAGGED` indicating that the analyser fails to provide a tag for the word. The vast majority of `UNTAGGED` words are proper names not recognized as such. These gaps in the morphology have little effect on our experiment.

4.3 Grammar

The grammar is a manually developed headed context-free phrase structure grammar for German subordinate clauses with 5508 rules and 562 categories, 209 of which are terminal categories. The formalism is that of Carroll and Rooth (1998), henceforth C+R:



```
● 1.000000 IP
  0.661299  VPP.np.np
  0.125495  VPP.n
  0.104186  VPK.n
  0.087692  VPP.dp.dp
  0.020743  VPP.d
  0.000556  VPP.nd.nd
  0.000029  VPP
```

Word-by-word gloss of the clause on the left:
'that Sarajevo over the airport with the essentials supplied will can'

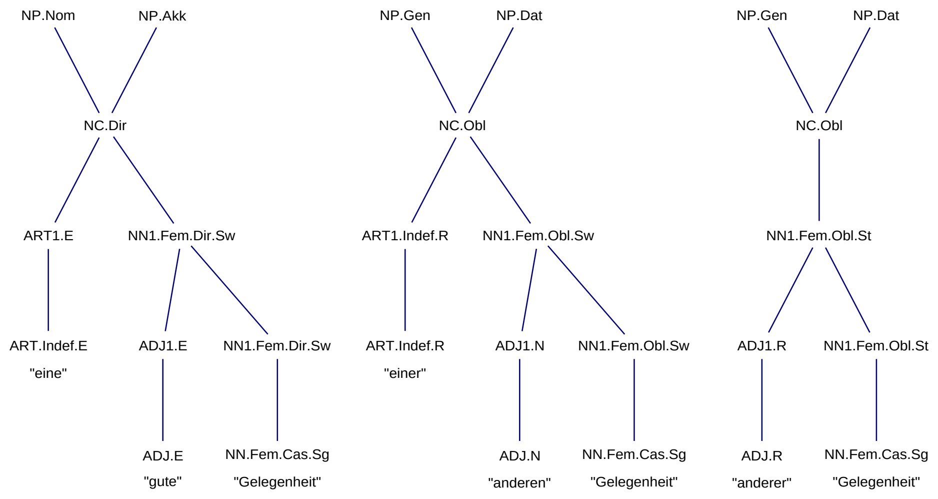
Figure 4.3: Chart browser

```
mother -> non-heads head' non-heads (freq)
```

The rules are head marked with a prime. The non-head sequences may be empty. `freq` is a rule frequency, which is initialized randomly and subsequently estimated by the inside outside-algorithm. To handle systematic patterns related to features, rules were generated by Lisp functions, rather than being written directly in the above form. With very few exceptions (rules for coordination, S-rule), the rules do not have more than two daughters.

Grammar development is facilitated by a chart browser that permits a quick and efficient discovery of grammar bugs (Carroll 1997a). Fig. 4.3 shows that the ambiguity in the chart is quite considerable even though grammar and corpus are restricted. For the entire corpus, we computed an average 9202 trees per clause. In the chart browser, the categories filling the cells indicate the most probable category for that span with their estimated frequencies. The pop-up window under `IP` presents the ranked list of all possible categories for the covered span. Rules (chart edges) with frequencies can be viewed with a further menu. In the chart browser, colors are used to display frequencies (between 0 and 1) estimated by the inside-outside algorithm. This allows properties shared across tree analyses to be checked at a glance; often grammar and estimation bugs can be detected without mouse operations.

The grammar covers 88.5% of the clauses and 87.9% of the tokens contained in the corpus. Parsing failures are mainly due to `UNTAGGED` words contained in 6.6% of the failed clauses, the pollution of the corpus by infinitival constructions (≈1.3%), and a number of coordinations not covered by the grammar (≈1.6%).



Glosses:*'a good opportunity'*, *'a different opportunity'*, *'different opportunity'*

Figure 4.4: Noun Projections

4.3.1 Case features and agreement

On nominal categories, in addition to the four cases `Nom`, `Gen`, `Dat`, and `Akk`, case features with a disjunctive interpretation (such as `Dir` for `Nom` or `Akk`) are used. The grammar is written in such a way that non-disjunctive features are introduced high up in the tree. Figure 4.4 illustrates the use of disjunctive features in noun projections: The terminal `NN` contains the four-way ambiguous `Cas` case feature; the N-bar (`NN1`) and noun chunk `NC` projections disambiguate to two-way ambiguous case features `Dir` and `Obl`; the weak/strong (`Sw/St`) feature of `NN1` facilitates or prevents combination with a determiner, respectively; only as soon as the `NP` projection, the case feature appears in disambiguated form. The use of disjunctive case features results in some reduction in the size of the parse forest, and some parameter pooling. Essentially the full range of agreement inside the noun phrase is enforced. Agreement between the nominative NP and the tensed verb (e.g. in number) is not enforced by the grammar, in order to control the number of parameters and rules.

For noun phrases we employ Abney's chunk grammar organization (Abney 1996). The noun chunk (`NC`) is an approximately non-recursive projection that excludes post-head complements and (adverbial) adjuncts introduced higher than pre-head modifiers and determiners but includes participial pre-modifiers with their complements. Since we perform complete context free parsing, parse forest construction, and inside-outside estimation, chunks are not motivated by deterministic parsing. Rather, they facilitate evaluation and graphical debugging, by tending to increase the span of constituents with high estimated frequency.

class	#	frame types
VPA	15	n, na, nad, nai, nap, nar, nd, ndi, ndp, ndr, ni, nir, np, npr, nr
VPP	13	d, di, dp, dr, i, ir, n, nd, ni, np, p, pr, r
VPI	10	a, ad, ap, ar, d, dp, dr, p, pr, r
VPK	2	i, n

Table 4.1: Number and types of verb frames

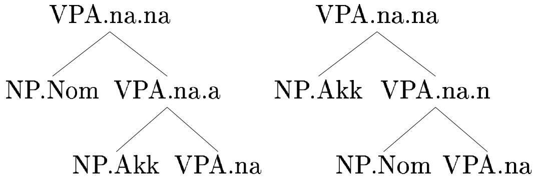


Figure 4.5: Coding of canonical and scrambled argument order

4.3.2 Subcategorisation frames of verbs

The grammar distinguishes four subcategorisation frame classes: active (VPA), passive (VPP), infinitival (VPI) frames, and copula constructions (VPK). A frame may have maximally three arguments. Possible arguments in the frames are nominative (n), dative (d) and accusative (a) NPs, reflexive pronouns (r), PPs (p), and infinitival VPs (i). The grammar does not distinguish plain infinitival VPs from *zu*-infinitival VPs. The grammar is designed to partially distinguish different PP frames relative to the prepositional head of the PP. A distinct category for the specific preposition becomes visible only when a subcategorized preposition is cancelled from the subcat list. This means that specific prepositions do not figure in the evaluation discussed below. The number and the types of frames in the different frame classes are given in Table 4.1.

German, being a language with comparatively free phrase order, allows for scrambling of arguments. Scrambling is reflected in the particular sequence in which the arguments of the verb frame are saturated. Compare Figure 4.5 for an example of a canonical subject-object order in an active transitive frame and its scrambled object-subject order. The possibility of scrambling verb arguments yields a substantial increase in the number of rules in the grammar (e.g. 102 combinatorically possible argument rules for all in VPA frames). Adverbs and non-subcategorized PPs are introduced as adjuncts to VP categories which do not saturate positions in the subcat frame.

In earlier experiments, we employed a flat clausal structure, with rules for all permutations of complements. As the number of frames increased, this produced prohibitively many rules, particularly with the inclusion of adjuncts.

4.4 Parameters

The parameterization is as in C+R, with one significant modification. Parameters consist of (i) rule parameters, corresponding to right hand sides conditioned by parent category and parent head; (ii) lexical choice parameters for non-head children, corresponding to child lemma conditioned by child category, parent category, and parent head lemma. See C+R or Charniak (1995) for an explanation of how such parameters define a probabilistic weighting of trees. The change relative to C+R is that lexicalization is by uninflected lemma rather than word form. This reduces the number of lexical parameters, giving more acceptable model sizes and eliminating splitting of estimated frequencies among inflectional forms. Inflected forms are generated at the leaves of the tree, conditioned on terminal category and lemma. This results in a third family of parameters, though usually the choice of inflected form is deterministic.

A parameter pooling feature is used for argument filling where all parent categories of the form VP.x.y are mapped to a category VP.x in defining lexical choice parameters. The consequence is e.g. that an accusative daughter of a nominative-accusative verb uses the same lexical choice parameter, whether a default or scrambled word order is used. (This feature was used by C+R for their phrase trigram grammar, not in the linguistic part of their grammar.) Not all desirable parameter pooling can be expressed in this way, though; for instance rule parameters are not pooled, and so get split when the parent category bears an inflectional feature.

4.5 Estimation

The training of our probabilistic CFG proceeds in three steps: (i) unlexicalized training with the `supar` parser, (ii) bootstrapping a lexicalized model from the trained unlexicalized one with the `ultra` parser, and finally (iii) lexicalized training with the `hypar` parser (Carroll 1997b). Each of the three parsers uses the inside-outside algorithm. `supar` and `ultra` use an unlexicalized weighting of trees, while `hypar` uses a lexicalized weighting of trees. `ultra` and `hypar` both collect frequencies for lexicalized rule and lexical choice events, while `supar` collects only unlexicalized rule frequencies.

Our experiments have shown that training an unlexicalized model first is worth the effort. Despite our use of a manually developed grammar that does not have to be pruned of superfluous rules like an automatically generated grammar, the lexicalized model is notably better when preceded by unlexicalized training (see also Ersan and Charniak (1995) for related observations). A comparison of immediate lexicalized training (without prior training of an unlexicalized model) and our standard training regime that involves preliminary unlexicalized training speaks in favor of our strategy (cf. the different 'lex 0' and 'lex 2' curves in figures 4.7 and 4.8). However, the amount of unlexicalized training has to be controlled in some way.

A		B		C	
1:	52.0199	1:	53.7654	1:	49.8165
2:	25.3652	2:	26.3184	2:	23.1008
3:	24.5905	3:	25.5035	3:	22.4479
⋮	⋮	⋮	⋮	⋮	⋮
13:	24.2872	55:	25.0548	70:	22.1445
14:	24.2863	56:	25.0549	80:	22.1443
15:	24.2861	57:	25.0549	90:	22.1443
16:	24.2861	58:	25.0549	95:	22.1443
17:	24.2867	59:	25.055	96:	22.1444

Table 4.2: Overtraining (iteration: cross-entropy on heldout data)

A standard criterion to measure overtraining is to compare log-likelihood values on held-out data of subsequent iterations. While the log-likelihood value of the training data is theoretically guaranteed to converge through subsequent iterations, a decreasing log-likelihood value of the held-out data indicates overtraining. Instead of log-likelihood, we use the inversely proportional cross-entropy measure. Table 4.2 shows comparisons of different sizes of training and heldout data (training/heldout): (A) 50k/50k, (B) 500k/500k, (C) 4.1M/500k. The overtraining effect is indicated by the increase in cross-entropy from the penultimate to the ultimate iteration in the tables. Overtraining results for lexicalized models are not yet available.

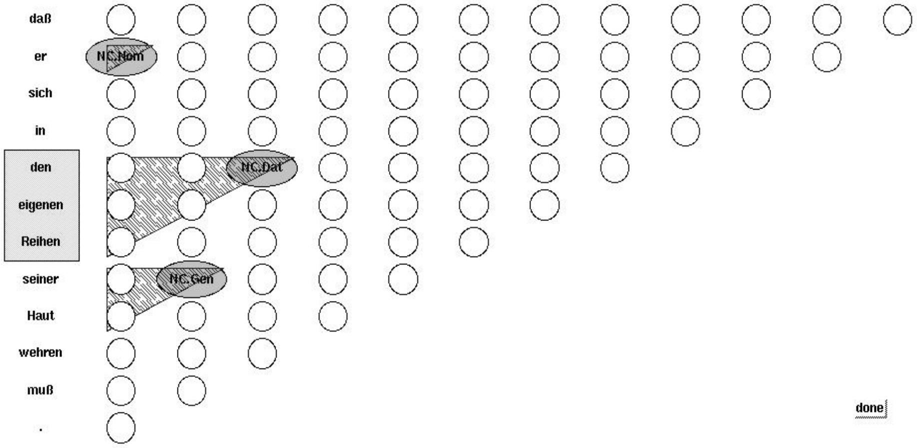
However, a comparison of precision/recall measures on categories of different complexity through iterative unlexicalized training shows that the mathematical criterion for overtraining may lead to bad results from a linguistic point of view. While we observed more or less converging precision/recall measures for lower level structures such as noun chunks, iterative unlexicalized training up to the overtraining threshold turned out to be disastrous for the evaluation of complex categories that depend on almost the entire span of the clause. The recognition of subcategorization frames through 60 iterations of unlexicalized training shows a massive decrease in precision/recall from the best to the last iteration, even dropping below the results with the randomly initialized grammar (see Figure 4.8).

4.5.1 Training regime

We compared lexicalized training with respect to different starting points: a random unlexicalized model, the trained unlexicalized model with the best precision/recall results, and an unlexicalized model that comes close to the cross-entropy overtraining threshold. (For the effect of differently randomized rule frequencies, we refer the reader to Appendix 2.) The details of the training steps are as follows:

(1) 0, 2 and 60 iterations of unlexicalized parsing with `supar`;

(2) lexicalization with `ultra` using the entire corpus;



Word-by-word gloss of the clause:
'that he himself in his own ranks hiw skin defend must'

Figure 4.6: Chart browser for manual NC labelling

(3) 23 iterations of lexicalized parsing with `hypar`.

The training was done simultaneously on four machines (two 167 MHz UltraSPARC with 188 MB and 312 MB main memory, and two 296 MHz SUNW UltraSPARC-II with 1.1 GB main memory). Using the grammar described here, one iteration of `supar` on the entire corpus takes about 2.5 hours, lexicalization and generating an initial lexicalized model takes more than six hours, and an iteration of lexicalized parsing can be done in 5.5 hours.

4.6 Evaluation

For the evaluation, a total of 600 randomly selected clauses were manually annotated by two labellers. Using a chart browser, the labellers filled the appropriate cells with category names `NC` and `PPART`, and those of maximal VP projections (cf. Figure 4.6 for an example of NC-labelling). We included prepositional phrases with preposition incorporated determiners (i.e. `PPART`, e.g. *'beim Zahnarzt'*, *'at the dentist'*) in the set of annotated noun chunks because the left boundary of the `NC` contained within the prepositional phrase is incorporated into the preposition word. Subsequent alignment of the labellers decisions resulted in a total of 1353 labelled NC categories: 627 `NC.Nom`, 319 `NC.Akk`, 253 `NC.Dat`, 75 `NC.Gen`, 73 `PPART.Dat`, and 6 `PPART.Akk`. The total of 584 labelled VP categories subdivides into 21 different verb frames with 340 different lemma heads. The dominant frames are active transitive (164 occurrences) and active intransitive (117 occurrences). They represent almost half of the annotated frames. Thirteen frames occur less than ten times, five of which just once (compare Table 4.3 for the numbers of individual frames).

VPA.na	164	VPP	4
VPA.n	117	VPP.p	3
VPK.n	90	VPA.npr	3
VPP.n	64	VPA.ni	2
VPA.np	62	VPA.ndr	2
VPA.nr	19	VPP.d	1
VPA.nd	16	VPI.p	1
VPA.nap	12	VPI.a	1
VPP.np	9	VPA.nir	1
VPA.nad	8	VPA.ndp	1
VPP.nd	4		

Table 4.3: Frames in the test set

4.6.1 Methodology

To evaluate iterative training, we extracted maximum probability (Viterbi) trees for the 600 clause test set in each iteration of parsing. For extraction of a maximal probability parse in unlexicalized training, we used Schmid's `lopar` parser (Schmid 1999). Trees were mapped to a database of parser generated markup guesses, and we measured *precision* and *recall* against the manually annotated category names and spans. Precision gives the ratio of correct guesses over all guesses, and recall the ratio of correct guesses over the number of phrases identified by human annotators. Here, we render only the precision/recall results on pairs of category names and spans, neglecting less interesting measures on spans alone. For the figures of adjusted recall, the number of unparsed misses has been subtracted from the number of possibilities.

In the following, we focus on the combination of the best unlexicalized model and the lexicalized model that is grounded on the former. In addition, as mentioned in section 4.5.1, we compare the results for our best lexicalized model (i) to a trained lexicalized model resulting from lexicalization of a random grammar and (ii) to a trained lexicalized model derived from an excessively trained unlexicalized model. The precision and recall plots for those models are labelled `lex 00` and `lex 60`, respectively.

Furthermore, in Appendix 3, we compare the training results with respect to varying initial random states of the grammar.

4.6.2 NC Evaluation

Figure 4.7 plots precision/recall for the training runs described in section 4.5.1, with lexicalized parsing starting after 0, 2, or 60 unlexicalized iterations. The best results are achieved by starting with lexicalized training after two iterations of unlexicalized training. Of a total of 1353 annotated `NC`s with case, 1103 are correctly recognized in the best unlexicalized model and 1112 in the last lexicalized model. With a number of 1295 guesses in the unlexicalized and 1288 guesses in the final lexicalized model, we gain 1.2% in precision (85.1% vs. 86.3%) and 0.6% in recall (81.5% vs. 82.1%) through lexicalized training. Adjustment to parsed clauses yields 88% vs. 89.2% in recall. As shown in Figure 4.7, the gain is achieved already within the first iteration; it is equally distributed between corrections of category boundaries and labels.

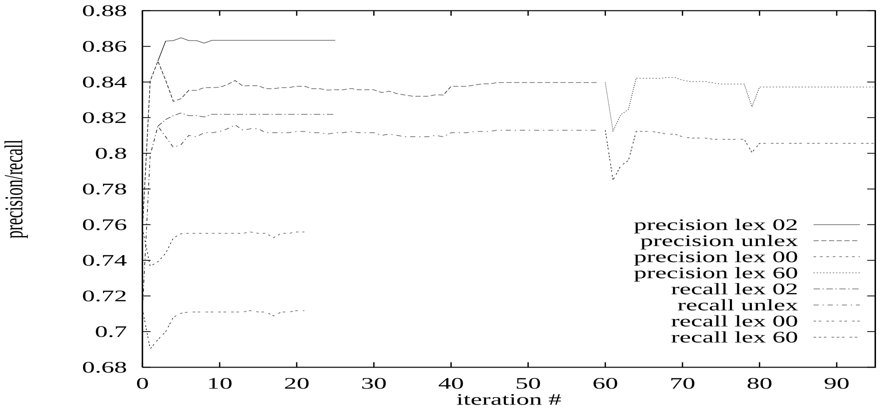


Figure 4.7: Precision/recall measures on NCs with case (random 0)

The comparatively small gain with lexicalized training could be viewed as evidence that the chunking task is too simple for lexical information to make a difference. However, we find about 7% revised guesses from the unlexicalized to the first lexicalized model. Currently, we do not have a clear picture of the newly introduced errors.

The plots labeled "00" are results for lexicalized training starting from a random initial grammar. The precision measure of the first lexicalized model falls below that of the unlexicalized random model (74%), only recovering through lexicalized training to equalize the precision measure of the random model (75.6%). This indicates that some degree of unlexicalized initialization is necessary, if a good lexicalized model is to be obtained.

Skut and Brants (1998) report 84.4% recall and 84.2% for NP and PP chunking without case labels. While these are numbers for a simpler problem and are slightly below ours, they are figures for an experiment on unrestricted sentences. A genuine comparison has to await extension of our model to free text.

4.6.3 Verb Frame Evaluation

Figure 4.8 gives results for verb frame recognition under the same training conditions. Again, we achieve best results by lexicalizing the second unlexicalized model. Of a total of 584 annotated verb frames, 384 are correctly recognized in the best unlexicalized model and 397 through subsequent lexicalized training. Precision for the best unlexicalized model is 68.4%. This is raised by 2% to 70.4% through lexicalized training; recall is 65.7%/68%; adjustment by 41 unparsed misses makes for 70.4%/72.8% in recall. The rather small improvements are in contrast to 88 differences in parser markup, i.e. 15.7%, between the unlexicalized and second lexicalized model. The main gain is observed within the first two iterations (cf. Figure 4.8; for readability, we dropped the recall curves when more or less parallel to the precision curves).

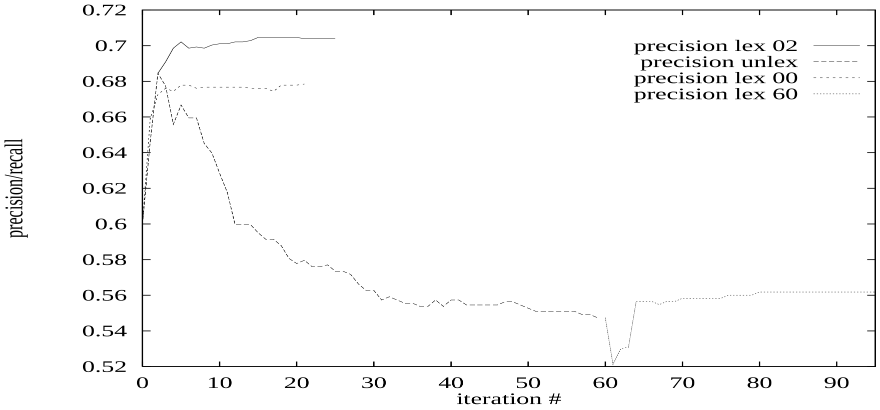


Figure 4.8: Precision measures on all verb frames (random 0)

Results for lexicalized training without prior unlexicalized training are better than in the NC evaluation, but fall short of our best results by more than 2%.

The most notable observation in verb frame evaluation is the decrease of precision of frame recognition in unlexicalized training from the second iteration onward. After several dozen iterations, results are 5% below a random model and 14% below the best model. The primary reason for the decrease is the mistaken revision of adjoined PPs to argument PPs. E.g. the required number of 164 transitive frames is missed by 76, while the parser guesses 64 `VPA.nap` frames in the final iteration against the annotator's baseline of 12. In contrast, lexicalized training generally stabilizes w.r.t. frame recognition results after only few iterations.

The plot labeled "lex 60" gives precision for a lexicalized training starting from the unlexicalized model obtained with 60 iterations, which measured by linguistic criteria is a very poor state. As far as we know, lexicalized EM estimation never recovers from this bad state.

4.6.4 Evaluation of non-PP Frames

Because examination of individual cases showed that PP attachments are responsible for many errors, we did a separate evaluation of non-PP frames. We filtered out all frames labelled with a PP argument from both the maximal probability parses and the manually annotated frames (91 filtered frames), measuring precision and recall against the remaining 493 labeller annotated non-PP frames.

For the best lexicalized model, we find somewhat but not excessively better results than those of the evaluation of the entire set of frames. Of 527 guessed frames in parser markup, 382 are correct, i.e. a precision of 72.5%. The recall figure of 77.5% is considerably better since

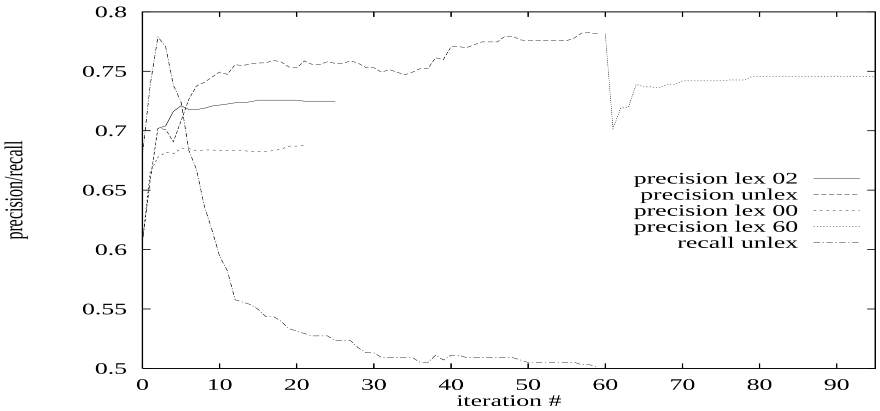


Figure 4.9: Precision measures on non-PP frames (random 0)

overgeneration of 34 guesses is neglected. The differences with respect to different starting points for lexicalization emulate those in the evaluation of all frames.

The rather spectacular looking precision and recall differences in unlexicalized training confirm what was observed for the full frame set. From the first trained unlexicalized model throughout unlexicalized training, we find a steady increase in precision (70% first trained model to 78% final model) against a sharp drop in recall (78% peek in the second model vs. 50% in the final). Considering our above remarks on the difficulties of frame recognition in unlexicalized training, the sharp drop in recall is to be expected: Since recall measures the correct parser guesses against the annotator's baseline, the tendency to favor PP-arguments over PP-adjuncts leads to a loss in guesses when PP-frames are abandoned. Similarly, the rise in precision is mainly explained by the decreasing number of guesses when cutting out non-PP frames. For further discussion of what happens with individual frames, see Appendix 1.

One systematic result in these plots is that performance of lexicalized training stabilizes after a few iterations. This is consistent with what happens with rule parameters for individual verbs, which are close to their final values within five iterations.

4.7 Conclusion

Our principal result is that scrambling-style free-er phrase order, case morphology and subcategorization, and NP-internal gender, number and case agreement can be dealt with in a head-lexicalized PCFG formalism by means of carefully designed categories and rules which limit the size of the packed parse forest and give desirable pooling of parameters. Hedging this, we point out that we made compromises in the grammar (notably, in not enforcing nominative-verb agreement) in order to control the number of categories, rules, and parameters.

A second result is that iterative lexicalized inside-outside estimation appears to be beneficial, although the precision/recall increments are small. We believe this is the first substantial investigation of the utility of iterative lexicalized inside-outside estimation of a lexicalized probabilistic grammar involving a carefully built grammar where parses can be evaluated by linguistic criteria.

A third result is that using too many unlexicalized iterations (more than two) is detrimental. A criterion using cross-entropy overtraining on held-out data dictates many more unlexicalized iterations, and this criterion is therefore inappropriate.

Finally, we have clear cases of lexicalized EM estimation being stuck in linguistically bad states. As far as we know, the model which gave the best results could also be stuck in a comparatively bad state. We plan to experiment with other lexicalized training regimes, such as ones which alternate between different training corpora.

The experiments are made possible by improvements in parser and hardware speeds, the carefully built grammar, and evaluation tools. In combination, these provide a unique environment for investigating training regimes for lexicalized PCFGs. Much work remains to be done in this area, and we feel that we are just beginning to develop understanding of the time course of parameter estimation, and of the general efficacy of EM estimation of lexicalized PCFGs as evaluated by linguistic criteria.

We believe our current grammar of German could be extended to a robust free-text chunk/phrase grammar in the style of the English grammar of Carroll and Rooth (1998) with about a month's work, and to a free-text grammar treating verb-second clauses and additional complementation structures (notably extraposed clausal complements) with about one year of additional grammar development and experiment. These increments in the grammar could easily double the number of rules. However this would probably not pose a problem for the parsing and estimation software.

Appendix 1: Evaluation of Individual Frames

	all guesses / correct guesses					
iter.	VPA.n	VPA.na	VPA.nd	VPA.nr	VPA.np	VPA.nap
possible	**117**	**164**	**16**	**19**	**62**	**12**
random	213/ 96	107/ 94	35/ 6	34/ 19	0/ 0	1/ 0
unlex 2	152/ 86	161/ 131	19/ 7	31/ 19	2/ 0	0/ 0
unlex 60	53 / 46	89/ 79	7/ 5	11/ 8	92/ 40	63/ 7
unlex 100	53 / 46	83/ 74	7/ 5	11/ 8	90/ 40	65/ 8
lex-00 1	196/ 99	129/ 117	28/ 7	34/ 19	0/ 0	2/ 0
lex-00 last	173/ 93	135/ 122	24/ 7	31/ 18	15/ 10	3/ 1
lex-02 2	139/ 88	147/ 129	19/ 7	27/ 17	13/ 9	0/ 0
lex-02 last	136/ 87	148/ 132	18/ 7	27/ 17	19/ 12	0/ 0
lex-60 1	54 / 50	68/ 66	25/ 7	11/ 9	85/ 38	51/ 7
lex-60 last	59 / 55	79/ 76	20/ 8	9/ 8	87/ 44	54/ 7

	all guesses / correct guesses			
iter.	copula	VPP.n	others	total
possible	**90**	**64**	**40**	**584**
random	69 / 60	67/ 51	34/ 10	560/ 336
unlex 2	96 / 77	68/ 52	32/ 12	561/ 384
unlex 60	102/ 78	47/ 8	116/ 23	580/ 294
unlex 100	102/ 78	26/ 21	124/ 25	560/ 305
lex-00 1	75 / 65	72/ 54	25/ 9	561/ 370
lex-00 last	79 / 67	72/ 54	31/ 10	563/ 382
lex-02 2	107/ 78	60/ 48	52/ 18	564/ 394
lex-02 last	106/ 78	60/ 47	50/ 17	564/ 397
lex-60 1	115/ 78	17/ 15	140/ 25	566/ 295
lex-60 last	114/ 78	20/ 18	124/ 24	566/ 318

Table 4.4: Absolute numbers of guesses and corrects for individual frames

In Table 4.4, we compare the absolute number of parser guesses and corrects of individual frames for our previously described training regime and the different resulting models. The table contains guesses and corrects of all frames that occur more than ten times in the annotator's markup. Combined measures for the set of frames that occur less than ten times in the markup are given in the column headed by *others*. The annotator's markup is indicated in the row labelled *possible*. The row labelled *random* indicates the figures for the unlexicalized model resulting from randomly initialized rule frequencies. We chose several key models of unlexicalized and lexicalized training that are indicated in the first column by iteration numbers. For unlexicalized training, we chose the the best model in iteration 2, the worst model in iteration 60, and the last model, which is already beyond the mathematical overtraining

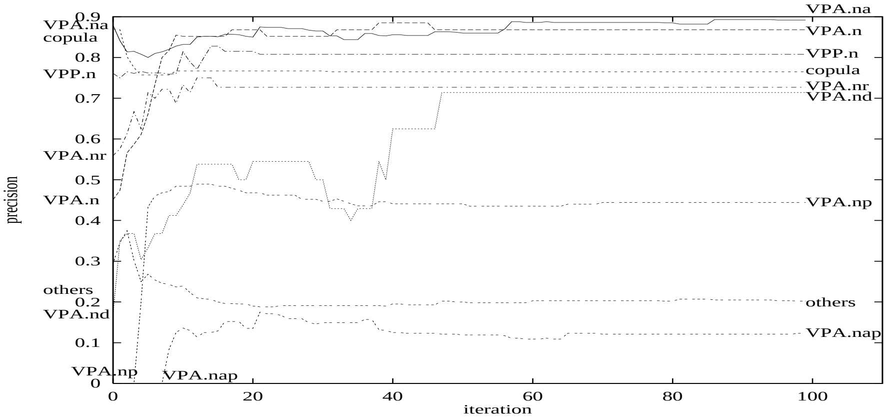


Figure 4.10: Precision of individual frames in unlexicalized training

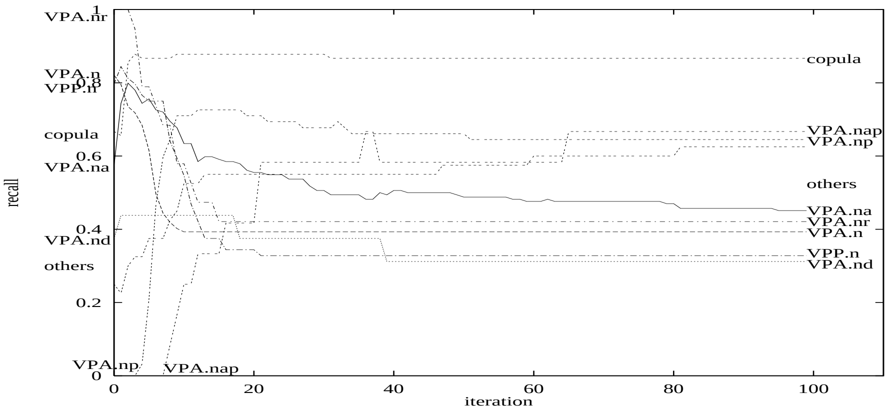


Figure 4.11: Recall of individual frames in unlexicalized training

threshold. For lexicalized training from a grammar with random frequency rules, we show the figures for the worst model immediately after lexicalization (*lex-00*), and the last model. In the best lexicalized training, we focus on the second and the last model (here, the start before lexicalization is *unlex 2*). The figures for the lexicalized training stuck in a bad state (*lex-60*) are again from immediately after lexicalization and from the last iteration.

In addition to the absolute figures of correct and incorrect guesses, we plot overall precision and recall results for individual frames in the course of unlexicalized training (Figures 4.10 and 4.11) and for those in lexicalized training starting from the best unlexicalized model (Figures 4.12 and 4.13).

unlex. The severe loss in precision following the second iteration during lexicalized training described in section 4.6.3 is due to a loss in decision mass for intransitive and transitive frames. Of 131 correctly recognized transitive frames in the second iteration more than 40 are lost in further training. Of 86 correct guesses of intransitive frames 40 are lost.

The loss with respect to intransitive frames is surprising given that the initial grammar is biased towards intransitive frames as witnessed by the figures for the random model (more than 38% of the total guesses are guesses of intransitive frames). Although the revision of intransitive guesses is desirable, it oversteps the mark of 117 in the annotator's handicap by far (213 guesses in random, 152 guesses in *unlex 2*, and 53 guesses in *unlex 60* and *100*). In the first iterations, intransitive guesses are changed to transitive and copula guesses. Concerning revisions to copula, we found a systematic error. Past participle forms of verbs are always associated with a tag list that in addition to the past participle tag `VVPP` also contains the tags `ADJ.Pred` for predicative and `ADJ.Adv` for adverbial adjectives. Given the ambiguous tag list, verbs forming their perfect tense forms with the auxiliary *'sein'* (*'to be'*) allow for an analysis of *'sein'* as a perfect tense auxiliary and also as a copula verb requiring a predicative. Similarly, the revision of passive frames to copula constructions in later iterations is to be explained by reanalysis of the passive auxiliary *'werden'* as a copula verb (cf. the decrease in guesses of `VPP.n` in *unlex 60* and *100*).

Apart from the shift to copula constructions, the dislike for intransitive and transitive is balanced by the tendency to choose frames that contain a PP-argument. The almost entire initial lack of PP-frame guesses in unlexicalized models is revised to more than 90 choices of `VPA.np`, more than 60 choices of `VPA.nap` and several low frequency PP-frames in *others*. The considerable increase in PP-frame choices contrasts with rather poor precision and recall results, less than 45% for `VPA.np`, less than 15% for `VPA.nap`, and about 20% for the PP-frames contained in *other*.

Let us add a remark on the interpretation of the precision and recall plots in figures 4.10 and 4.11. Although the majority of plots for individual frames shows a gain in precision during unlexicalized training, we noted the dramatic loss in overall precision illustrated back in Figure 4.8. As already mentioned in the above discussion of absolute numbers, this is due shift of the parser's decision mass as illustrated by the decrease in recall of intransitive, transitive, and passive frames in Figure 4.11.

lex-00/02/60. Apart from very few (mostly insignificant) exceptions, our observation concerning iterative inside-outside estimation succeeding lexicalization is confirmed by the evaluation of individual frames. The overall frame evaluation showed small but reliable precision and recall gains for frame recognition. Irrespective of different starting points, iterative lexicalized estimation similarly leads to either slightly increased or steady results for each of the individual frames with respect to absolute figures of correct guesses. The only notable negative exception is the loss of six correct choices of intransitive frames in *lex-00* accompanied by a correction from 97 to 80 wrong guesses of intransitives. The most significant gain in early lexicalization (*lex-00* and *lex-02*) is achieved for `VPA.np` frames. In late lexicalization, i.e. *lex-60*, lexicalized training mainly rectifies the loss of correct guesses of intransitive frames and of passive frames with a single nominative argument.

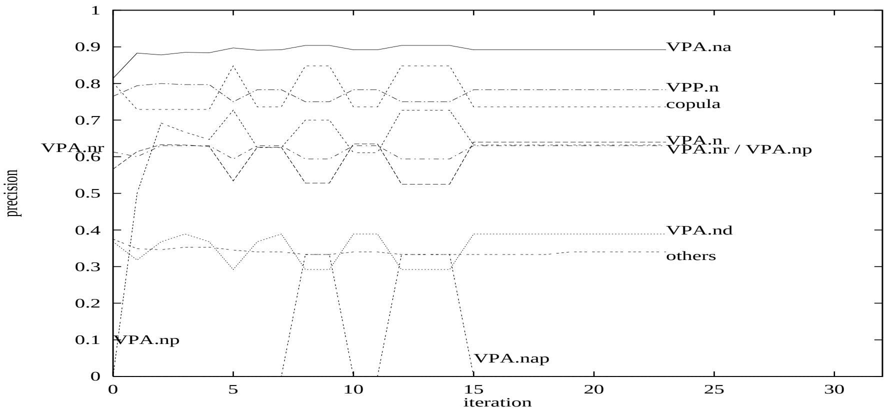


Figure 4.12: Precision of individual frames in training the best lexicalized model

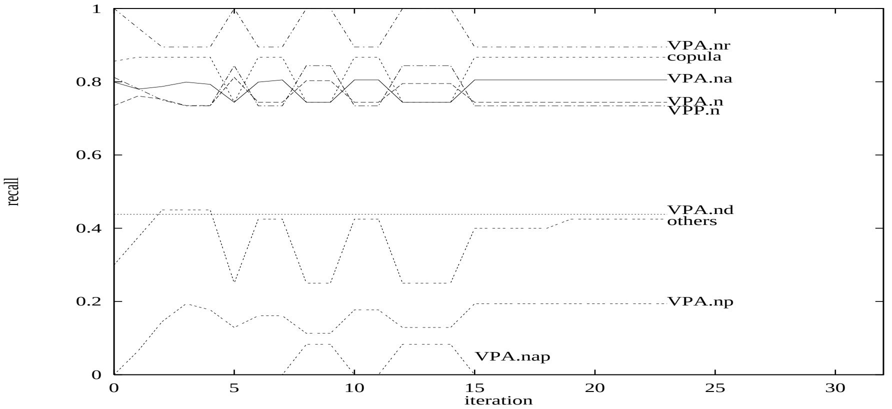


Figure 4.13: Recall of individual frames in training the best lexicalized model

The plots for precision in Figure 4.12 and recall in Figure 4.13 of individual frames in lexicalized training *lex-02* show main gains within the first four iterations. Later shifts in precision and recall up to iteration 15 are equalled out. In the above discussion of unlexicalized training, we mentioned `VPP.n` and the copula construction as an instance of a mutually dependent pair of frames. In the plots, this can be reidentified by precision/recall gains in the former and respective losses in the latter.

Appendix 2: Random Models

In order to give an impression of possible gains through grammar training, we did an evaluation of 50 different grammars with randomly initialized rule frequencies.

The following plots present precision and recall measures on noun chunk evaluation (Figure 4.14), frame evaluation (Figure 4.15). The precision of noun chunk recognition is in the range of 58% and 77%, frame recognition varies between 48% and 62%. We abstained from including a plot for the variability of non-PP frame recognition in different random models. It ranges from 52% to 66%.

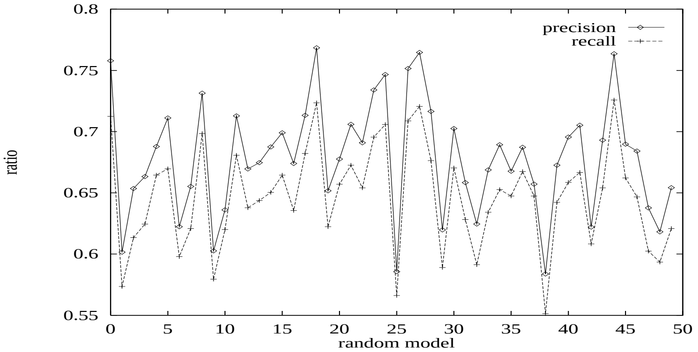


Figure 4.14: Precision/recall measures on noun chunks of 50 different unlexicalized random models

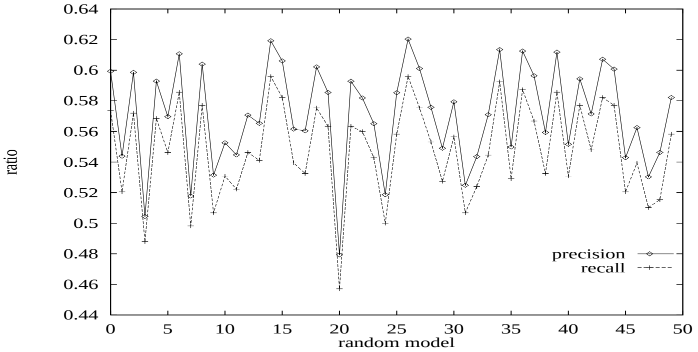


Figure 4.15: Precision/recall measures on VP frames of 50 different unlexicalized random models

Appendix 3: Different Starting Points

In order to validate our results, we chose two grammars from the 50 different rule frequency randomizations as starting points for unlexicalized and lexicalized training. Random model 20 delivers the worst precision and recall results for frames among all random states. Random model 25 yields almost identical precision and recall ratios for both noun chunks and subcategorization frames, i.e., in the context of all random models, intermediate results for noun chunks and fair results for frames. The initial grammar used in our main experiment (random grammar 0) is well above average in precision and recall for both evaluations.

Noun chunks.

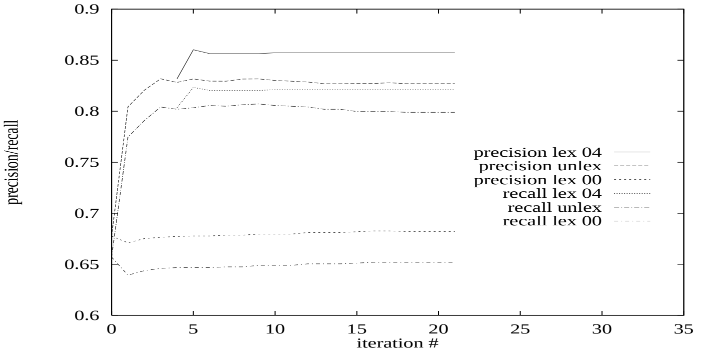


Figure 4.16: Precision/recall measures for `NC` with case (random 20)

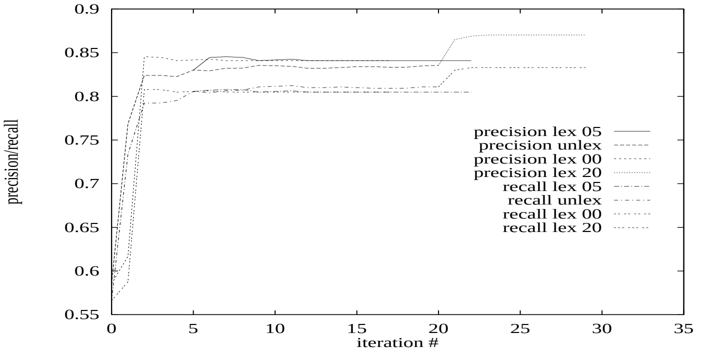


Figure 4.17: Precision/recall measures for `NC` with case (random 25)

Starting from different random grammars, precision and recall measures in noun chunk evaluation for unlexicalized training lead to similar results. Irrespective of initialization, unlexicalized training yields between 83% and 84% in precision and between 80% and 81% in recall of noun chunk recognition. A minor difference is found in the number of iterations within which the best results will be reached. Starting from a random grammar with poor precision and recall requires few more iterations of unlexicalized training in order to reach maximally possible results (cf. Figure 4.16).

For lexicalized training with different bases for lexicalization, the comparison of different starting points shows a much more diverse picture (in addition to Figure 4.18 and 4.19 see also Figure 4.8 in section 4.6.3). In contrast to the models *random 0* and *random 20*, immediate lexicalization of the random model does not produce poor results for noun chunk recognition. Furthermore, lexicalized training based on an early model of unlexicalized training may have hardly any positive effect at all as witnessed by the plots for *lex 05* in Figure 4.19. Rather, the precision results for *lex 20* in Figure 4.19 suggest the need for a more careful selection of the base for lexicalization and further lexicalized inside-outside estimation. However, since our prime interest is in good results for frame recognition, i.e. good overall results for the entire model, we have to compromise here.

Frames.

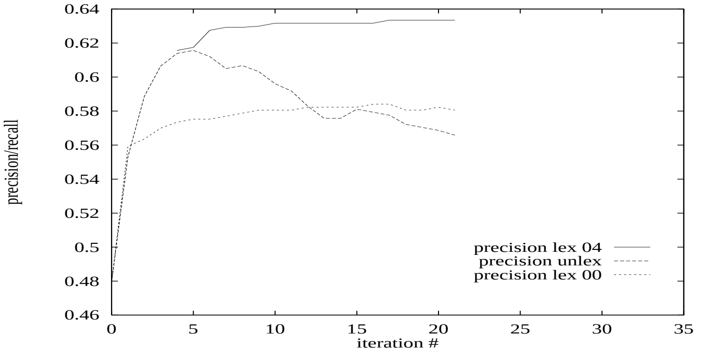


Figure 4.18: Precision measures for all verb frames (random 20)

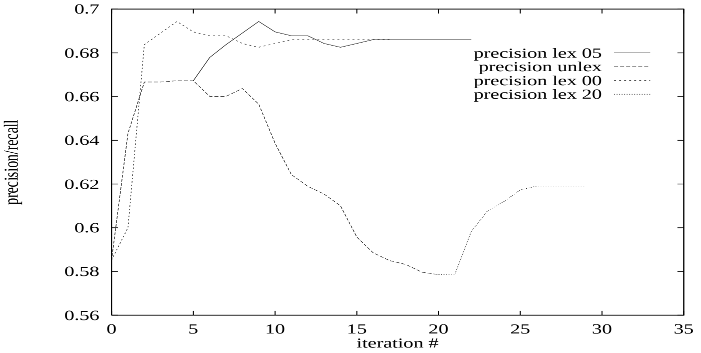


Figure 4.19: Precision measures for all verb frames (random 25)

Again, the results concerning frame recognition in iterative unlexicalized reestimation from different starting points corroborate our observations presented in section 4.6. Precision gains in initial iterations are followed by losses more or less severe depending on both precision in the initial random state and precision in the best unlexicalized model. In contrast to the observations for precision results in NC evaluation, the precision of frame recognition found for random models correlates with precision maxima. Low precision of 48% in *random 20* restricts the maximum to less than 62% within five iterations of unlexicalized training, high precision in *random 0* allows for a maximum of more than 68% within only two iterations. Irrespective of the particular starting point, 20 iterations of unlexicalized training show a uniform decrease to about 57% precision after 20 iterations.

The plots for lexicalized training with lexicalization of different base models also support our findings in section 4.6. In general, it is a good strategy to start with lexicalized training from a well-trained unlexicalized model in order to obtain best precision results. A notable exception is *lex 00* in *random 25*, where lexicalization of the random model yields similar precision for frame recognition as lexicalized training with the unlexicalized model from iteration 5 as its base.

Bibliography

Abney, S. (1991). Parsing by chunks. In D. Bouchard and K. Leffel (Eds.), *Views on Phrase Structure*. Dortrecht: Kluwer Academic Publishers.

Abney, S. (1995). Chunks and dependencies: Bringing processing evidence to bear on syntax. In *Computational Linguistics and the Foundations of Linguistic Theory.* Stanford, CA: CSLI.

Abney, S. (1996). Chunk stylebook. Technical report, SfS, Universität Tübingen.

Baker, J. (1979). Trainable grammars for speech recognition. In D. Klatt and J. Wolf (Eds.), *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pp. 547–550.

Baum, L. E., T. Petrie, G. Soules, and N. Weiss (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics 41*(1), 164–171.

Baum, L. E. and G. A. Sell (1968). Growth transformations for functions on manifolds. *Pacific Journal of Mathematics 27*(2), 211–227.

Beckwith, R., C. Fellbaum, D. Gross, and G. A. Miller (1991). Wordnet: A lexical database organized on psycholinguistic principles. In *Lexical Acquisition – Exploiting On-Line Resources to Build a Lexicon*, Chapter 9, pp. 211–232.

Beil, F., G. Carroll, D. Prescher, S. Riezler, and M. Rooth (1998). Inside-outside estimation of a lexicalized PCFG for German. –Gold–. In *Inducing Lexicons with the EM Algorithm*, AIMS Report 4(3). IMS, Universität Stuttgart.

Booth, T. L. and R. A. Thompson (1973). Applying probability measures to abstract languages. *IEEE Transactions on Computers C-22*(5), 442–450.

Brown, P., P. deSouza, R. Mercer, V. D. Pietra, and J. Lai (1992). Class-based n-gram models of natural language. *Computational Linguistics 18*(4), 467–479.

Buschbeck-Wolf, B. (1997). Resolution on demand. Technical Report 196, Verbmobil.

Carroll, G. (1997a). *Manual pages for* `charge`*,* `hyparCharge`*, and* `tau`. IMS, Universität Stuttgart.

Carroll, G. (1997b). *Manual pages for* `supar`*,* `ultra`*,* `hypar`*, and* `genDists`. IMS, Universität Stuttgart.

Carroll, G. and M. Rooth (1998). Valence induction with a head-lexicalized PCFG. In *Proceedings of EMNLP-3*, Granada.

Charniak, E. (1995). Parsing with context-free grammars and word statistics. Technical Report CS-95-28, Department of Computer Science, Brown University.

Chomsky, N. (1965). *Aspects of the Theory of Syntax.* Cambridge, MA: M.I.T. Press.

Church, K. W., W. A. Gale, P. Hanks, and D. Hindle (1991). Using statistics in lexical analysis. In U. Zernik (Ed.), *Lexical acquisition: exploiting on-line resources to build a lexicon.*

Dagan, I. and A. Itai (1994). Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics 20*, 563–596.

Dagan, I., L. Lee, and F. Pereira (1998). Similarity-based models of word cooccurrence probabilities. To appear in *Machine Learning.*

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the *EM* algorithm. *Journal of the Royal Statistical Society 39*(B), 1–38.

Duda, R. O. and P. E. Hart (1973). *Pattern classification and scene analysis.* New York: Wiley.

Ersan, M. and E. Charniak (1995). A statistical syntactic disambiguation program and what it learns. Technical Report CS-95-29, Department of Computer Science, Brown University.

Fodor, J. A. (1998). *Concepts: Where Cognitive Science Went Wrong.* Oxford: Oxford Cognitive Science Series.

Gazdar, G., E. Klein, G. K. Pullum, and I. A. Sag (1985). *Generalized Phrase Structure Grammar.* Cambridge, MA: Harvard University Press.

Grimshaw, J. (1990). *Argument Structure.* Cambridge, MA: MIT Press.

Hale, K. and S. Keyser (1993). Argument structure and the lexical expression of syntactic relations. In K. Hale and S. Keyser (Eds.), *The View from Building 20.* Cambridge, MA: MIT Press.

Hindle, D. (1983). User manual for fidditch, a deterministic parser. Technical Memorandum 7590-142, Naval Research Laboratory.

Hindle, D. (1994). A parser for text corpora. In B. Atkins and A. Zampolli (Eds.), *Computational Approaches to the Lexicon.* Oxford: Oxford University Press.

Hindle, D. and M. Rooth (1993). Structural ambiguity and lexical relations. *Computational Linguistics 19*, 103–120.

Hofmann, T. and J. Puzicha (1998). Unsupervised learning from dyadic data. Technical Report TR-98-042, International Computer Science Insitute, Berkeley, CA.

Hughes, J. (1994). *Automatically Acquiring Classification of Words.* Ph. D. thesis, University of Leeds, School of Computer Studies.

Jackendoff, R. (1977). *$\bar{X}$ Syntax: A Study in Phrase Structure.* Cambridge, MA: MIT Press.

Karttunen, L., T. Yampol, and G. Grefenstette (1994). *INFL Morphological Analyzer/Generator 3.2.9 (3.6.4).* Xerox Corporation.

Katz, S. M. (1980). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing 35.*

Klavans, J. L. and M.-Y. Kan (1998, August). The role of verbs in document analysis. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING-ACL '98)*, Montreal, Canada.

Kullback, S. and R. A. Leibler (1951). On Information and Sufficiency. *Annals of Mathematical Statistics 22*, 79–86.

Kural, M. (1996). *Verb Incorporation and Elementary Predicates.* Ph. D. thesis, University of California, Los Angeles.

Lang, B. (1989). A uniform formal framework for parsing. In *Proceedings of the 1st International Workshop on Parsing Technologies (IWPT 1)*, Pittsburgh, PA, pp. 28–42.

Lang, S. (1993). *Algebra* (Third edition ed.). Reading, MA: Addison-Wesley.

Levin, B. (1993). *English Verb Classes and Alternations. A Preliminary Investigation.* Chicago/London: The University of Chicago Press.

Liberman, M. (1992). *ACL-DCI CD ROM 1.* Association for Computational Linguistics.

McLachlan, G. J. and T. Krishnan (1997). *The EM Algorithm and Extensions.* New York: Wiley.

Melamed, I. D. (1997). Word-to-word models of translational equivalence. Technical Report IRCS, Nr. 98-08, University of Pennsylvania.

Ney, H., U. Essen, and R. Kneser (1994). On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language 8*, 1–38.

Pereira, F., N. Tishby, and L. Lee (1993). Distributional clustering of English words. In *Proceedings of the 31th Annual Meeting of the ACL*, Columbus, Ohio.

Pollard, C. and I. A. Sag (1987). *Information-Based Syntax and Semantics, Vol. 1: Fundamentals.* Stanford, CA: CSLI.

Resnik, P. (1993). *Selection and information: A class-based approach to lexical relationships.* Ph. D. thesis, University of Pennsylvania, CIS Department.

Resnik, P. (1997). Selectional preference and sense disambiguation. Paper presented at: Tagging Text with Lexical Semantics: Why, What, and How?, Workshop in connection with ANLP 97, Washington, D.C.

Ribas, F. (1994). An experiment on learning appropriate selectional restrictions from a parsed corpus. In *Proceedings of COLING-94*, Kyoto, Japan.

Ribas, F. (1995). On learning more appropriate selectional restrictions. In *Proceedings of the 7th Conference of the EACL*, Dublin, Ireland.

Rooth, M. (1998). Two-dimensional clusters in grammatical relations. In *Inducing Lexicons with the EM Algorithm*, AIMS Report 4(3). Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Rooth, M. (Ms.). Two-dimensional clusters in grammatical relations. In *Symposium on Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity*, AAAI 1995 Spring Symposium Series. Stanford University.

Rooth, M., S. Riezler, D. Prescher, G. Carroll, and F. Beil (1998). EM-based clustering for NLP applications. In *Inducing Lexicons with the EM Algorithm*, AIMS Report 4(3). Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Saul, L. K. and F. Pereira (1997). Aggregate and mixed-order Markov models for statistical language processing. In *Proceedings of EMNLP-2*.

Schiller, A. and C. Stöckert (1995). *DMOR*. IMS, Universität Stuttgart.

Schmid, H. (1999). *Manual page for* `lopar`. IMS, Universität Stuttgart.

Schuhmacher, H. (1986). *Verben in Feldern. Valenzwörterbuch zur Syntax und Semantik deutscher Verben*. Berlin: de Gruyter.

Schulte im Walde, S. (1998). Automatic semantic classification of verbs according to their alternation behaviour. Master's thesis, Institut für maschinelle Sprachverarbeitung, Universität Stuttgart.

Skut, W. and T. Brants (1998). A maximum-entropy partial parser for unrestricted text. In *Proceedings of the Sixth Workshop on Very Large Corpora*, Montréal, Québec.

Stowell, T. (1981). *Origins of Phrase Structure*. Ph. D. thesis, MIT.

Wahba, G. and S. Wold (1975). A completely automatic french curve: Fitting spline functions by cross validation. *Commun. Statist. 4*, 1–17.