CORPUS-BASED LEARNING OF OT CONSTRAINT RANKINGS FOR LARGE-SCALE LFG GRAMMARS

Martin Forst
University of Stuttgart
Institute for NLP

Jonas Kuhn
Saarland University
Computational Linguistics

Christian Rohrer
University of Stuttgart
Institute for NLP

Proceedings of the LFG05 Conference

University of Bergen

Miriam Butt and Tracy Holloway King (Editors)

2005
CSLI Publications
http://csli-publications.stanford.edu/

Abstract

We discuss a two-stage disambiguation technique for linguistically precise broad-coverage grammars: the pre-filter of the first stage is triggered by linguistic configurations ("optimality marks") specified by the grammar writer; the second stage is a log-linear probability model trained on corpus data. This set-up is used in the Parallel Grammar (ParGram) project, developing Lexical Functional Grammars for various languages. The present paper is the first study exploring how the pre-filter can be empirically tuned by learning a relative ranking of the optimality marks from corpus data, identifying problematic marks and relaxing the filter in various ways.

1 Background

1.1 Linguistically precise grammars in NLP

In recent years, parsing based on large manually developed grammars that are directly informed by linguistic theory has made significant progress towards broad-coverage application. Efficient processing platforms for parsing and generation are available (in our context in particular the XLE system (Kaplan et al. 2002) for Lexical-Functional Grammar – LFG); advanced profiling techniques and tool support for grammar development are available (Oepen and Carroll 2000, King et al. 2004); effective fallback strategies have been established to achieve robustness while still taking advantage of the high depth of analysis (Riezler et al. 2002). The high initial cost of theory-driven manual grammar development pays off when grammars for new languages are added to a family of grammars, as syntactic theory often gives clear indications as to which parts of the rule system will carry over from an existing grammar and which parts have to be rewritten; the most effective methodology thus relies on multilingual grammar development based on clear cross-linguistic grammar writing conventions, as, e.g., practiced in the LFG-based Parallel Grammar (ParGram) project (Butt et al. 2002, 1997) and the Grammar Matrix approach (Bender et al. 2002) in the Head-driven Phrase Structure Grammar (HPSG) framework. The resulting grammars are particularly suited for application contexts requiring great depth of analysis (like language understanding systems with a reasoning component) and/or reversible grammars for parsing/generation (like high-quality machine translation or computer-aided language learning).

1.2 Disambiguation: a two-stage approach

Theory-driven grammar development typically leads to grammars that overgenerate only mildly, since lexical subcategorization information is taken into account and the grammatical constructions can be restricted by rich feature constraints. In other words, most of the parses that a grammar assigns to a string are linguistically justified. Nevertheless, due to the underspecified nature of natural language, ambiguity rates for non-trivial sentences are considerable: most disambiguation decisions cannot be made on strictly grammatical grounds, but involve some contextual or world knowledge. To ensure portability of the grammars across domains, hard-wiring such non-grammatical decisions in the grammar code is generally avoided. This justifies the need for sophisticated disambiguation techniques to complement the linguistic grammar in parsing.

Contrary to the situation in grammar writing, the most effective way of building a disambiguator is to exploit empirical, corpus-driven techniques. For constraint-based formalisms like LFG and HPSG, the use of log-linear probability models applied on fully or partially labeled training corpora has been established as a powerful, general machine learning technique (Johnson et al. 1999, Riezler et al. 2002, Kaplan et al. 2004, Toutanova et al. 2002). The log-linear models are typically trained using a large, schematically constructed set of learning features that check for structural and lexical configurations and co-occurrences in the linguistic representations. Trimming the features of the log-linear model is an engineering task separated quite clearly from the grammar writing task. Hence, there is a clear conceptual split between grammar writing, driven by linguistic theory, and disambiguator development work, involving advanced machine learning techniques. However, in the ParGram project, it has proven highly productive to assume an intermediate stage between the two components – a linguistically motivated pre-disambiguation filter. Motivation comes from both ends:

(1) *Linguistic motivation*: Beyond the strict grammatical rules and principles that are encoded in the grammar proper, there is a considerable number of soft principles that the grammar writer is well aware of when working on a linguistic construction. For instance, many "rare" constructions should only be appealed to in analysis when there is no "canonical" analysis of a string.[1]

Translating soft constraints into carefully conditioned hard grammatical constraints in order to keep up the two-way conceptual split often yields unintelligible, error-prone rules; it also goes against the idea of a theory-driven grammar development as it hides a clear intuitive explanation in a technically complicated rule. Leaving the soft constraints entirely out of the grammar writing picture, hoping that some constellation of the learning features will pick them up, is unsatisfactory, too. At the point when the linguist is working on a construction, s/he is best aware of the linguistically salient interactions, and it takes almost no extra work to encode the soft constraint explicitly.

Therefore the XLE grammar development platform employed in the ParGram project has been integrated with a soft constraint mechanism inspired by the strict constraint ranking system of Optimality Theory (OT) (Frank et al. 2001). The mechanism is conceptually quite simple: for particular structural configurations in the linguistic representation, an *optimality mark* or OT constraint can be introduced (e.g., for the occurrence of a topicalized object). Each optimality mark is assigned a polarity, i.e., defining it as preferred or dispreferred. Furthermore, all marks used in a grammar can be ordered in a relative ranking (where several marks can be given the same rank position). When the parser is applied, optimality mark instances are collected in a multiset and can then be used as a filter on the readings produced by the system. Following the ranking order, each mark will filter out readings that have fewer instances than the reading with the maximal instances (for preference marks) or more instances than the reading with the fewest instances (for dispreference marks). The readings of a sentence that pass all marks and are still left in the end are called "optimal", the readings that are filtered out are called "suboptimal".[2]

Experience showed that without a (potentially temporary) filtering mechanism for uncommon constructions, grammar writing would be considerably harder (King et al. 2004).

(2) *Technical motivation*: The log-linear models applied in empirical training of the disambiguator are a discriminative technique, involving the computation of the gold standard analysis as well as all alternative solutions (Johnson et al. 1999); furthermore, parameter estimation is an iterative process that passes the training data multiple times. Hence, to keep the process tractable on medium-size to large training corpora, the set of competing analyses has to be limited. Using only the analyses that pass a linguistically motivated prefilter is a very desirable set-up.

[1] Awareness of such linguistic interactions goes back to Panini's work, and in recent years, ways of including soft-constraint mechanisms in formal grammar formalisms have been explored, particularly in the framework of Optimality Theory (Prince and Smolensky 1993) or in probabilistic grammar models (Manning 2003).

[2] For example, the parser may assign four readings to a sentence. Reading one has the multiset $\{ C_1, C_1, C_2 \}$ of optimality marks, reading two has the multiset $\{ C_2, C_2, C_3 \}$, reading three $\{ C_1, C_1, C_2 \}$, and reading four $\{ C_1, C_1, C_3, C_3 \}$. Let us furthermore assume that the marks are ranked $C_1 \gg C_2 \gg C_3$, and that all have positive polarity, i.e., they are preference marks. In evaluation, C_1 is considered first. Readings one, three, and four have two marks of C_1 each, whereas reading two has none. Therefore reading two is filtered out at this step. For the remaining readings, C_2 is considered next. Reading four doesn't include any C_2 marks, so it is filtered out. For the final mark C_3, there is no difference between the remaining readings (one and three – both have no C_3 mark). Hence, there are two optimal readings: readings one and three.

2 Methodology

In this section, we discuss out experimental methodology at a conceptual level, a more detailed description and the results follow in section 3.

2.1 Trimming the linguistic pre-filter

In past work on the ParGram grammars, both the introduction of OT marks and the specification of their relative ranking was done manually. This is problematic since the various marks affect phenomena that were integrated into the grammar at different development stages, and often the appropriate relative ranking can only be determined empirically. Moreover, the question of whether or not a particular mark should be active in the two-stage filter architecture we described is also hard to answer in isolation. (But note that the structural specification and the polarity of the candidate marks *are* aspects about which the grammar writer can make an informed decision.) [3]

This paper is the first systematic study applying empirical methods in order (i) to determine the ranking of the OT marks, and (ii) to decide which marks should be left out of the first disambiguation stage. It has been part of this study to explore measures for the quality of a particular specification of the pre-filter. We present results for the German grammar from the family of ParGram grammars. Some of the results of our experiments are surprising and provide some interesting insights in the workings of the two-stage filter architecture.

While the technical results we report in this paper make reference to project-specific details of our system architecture, we believe that many of the higher-level observations will carry over very well to other projects involving a linguistically motivated core module that is applied in a broader context of empirically tuned system components.

2.2 Measuring the quality of the pre-filter

The fact that the component we are interested in here is a pre-filter in the context of a two-stage system has special consequences for quality assessment. It is not necessary that the pre-filter remove *all* incorrect readings – since it is followed up by a sophisticated second disambiguator. On the other hand, it is

[3] Let us consider the following sentences as examples that illustrate the way dispreference and preference marks work, but that also show that they sometimes prove problematic:

(1) Der Journalist stellt ihn ihr gegenüber.
The journalist put him her opposite.
'The journalist confronted him with her.'

(2) Weil er ihr gegenüber arrogant war, verliert er Sympathie.
Because he her opposite arrogant was, loses he sympathy.
'Since he was arrogant towards her, he is losing sympathy.'

(3) Weil er Frau Merkel gegenüber Schwachsinn erzählte, verliert er Sympathie.
Because he Ms Merkel opposite nonsense told, loses he sympathy.
'Since he told nonsense to Ms Merkel, he is losing sympathy.'

In examples 1 and 3, ambiguity arises due to the fact that *gegenüber* can be a separable verb particle, a preposition, a postposition or an adverb. This ambiguity can be (at least partly) resolved by the OT marks *VerbParticle*, which is a preference mark, and *Ppost*, which is a dispreference mark. In example 1, *VerbParticle* correctly filters out the readings where *gegenüber* is analyzed as an adverb or postposition, keeping only the parse where it is analyzed as a separable verb particle as optimal. In example 2, *gegenüber* is analyzed as a postposition. This analysis survives the filtering by *VerbParticle* and *Ppost* because no alternative analysis of *gegenüber* is available. In example 3, *gegenüber* is wrongly analyzed as a preposition, because *Ppost* makes its intended analysis as a postposition suboptimal. Realistically, the fine-tuning between instances parallel to 2 vs. 3 can only be done taking corpus frequencies into account.

highly undesirable if the pre-filter accidentally removes the correct reading, since this would make a data point unusable for the second stage. One might describe this as a task in which recall is of greatest importance, and precision should be traded off for recall; but in order to do justice to the special setting, we will call the relevant measures “filter fidelity” and “filter efficiency”. “Filter fidelity” is defined as the proportion of sentences for which the OT mark ranking under consideration keeps the correct reading among the optimal reading(s). The intuition behind “filter efficiency”, on the other hand, is to measure the proportion of readings among all incorrect readings of a sentence which are discarded by the OT mark ranking as suboptimal. Concretely, we calculate it as the quotient of the number of readings discarded by the filter divided by the total number of readings minus one.[4] Filter fidelity is our main criterion and it should be as close as possible to 100%, but filter efficiency does have a certain importance as well, of course, since filtering a maximum of bad readings while losing a minimum of good readings is the goal of this whole enterprise. As a combined quality measure, we therefore provide a weighted F-score where filter fidelity is weighted more strongly than filter efficiency. [5]

2.3 Corpus-based learning of a ranking

Our experiments start out with the manually specified OT mark ranking in the German grammar. An obvious technique to try out is to learn a ranking automatically from corpus data for which the correct reading has been labeled. The filter quality with the learned ranking can then be compared against the manual ranking and a uniform ranking (giving all marks the same rank).

For corpus-based learning of the OT ranking, one could in theory apply the classical Constraint Demotion Algorithm from the OT literature (Tesar and Smolensky 1998); however, due to the variation in the data the algorithm might not converge. Therefore we transform the classical discrete constraint ranking into a continuous numerical ranking for the purpose of learning. This allows us to apply robust learning algorithms like the Gradual Learning Algorithm (GLA) proposed by Boersma (1998), which is related to the perceptron algorithm. In learning, the system’s current numerical ranking (with some noise added to determine each constraint’s particular rank) is used in order to disambiguate a sentence from the training data. When the predicted solution does not match the gold standard analysis, all constraints ranked too low are promoted by a small increment (controlled by the so-called plasticity parameter); all constraints ranked too high are demoted. The noise added in application has the effect that constraints with a similar ranking can “swap” their relative rank, which leads to variation in the data, as it is often observed. This variant of OT is thus often called Stochastic OT.

2.4 Augmenting the set of OT marks

We also performed an additional experiment besides learning a ranking for just the OT marks specified by the grammar writers: we explored how pre-filter quality is affected if we systematically augment the existing set of OT marks to ensure that for common disambiguation decisions, sufficiently fine-grained distinctions in the OT marks are available. It is conceivable that for certain decisions, the OT mark set is too “sparse” to produce a reliable result, whereas a richer OT mark set might behave in a more balanced way. This is because in stochastic OT, competing marks may form clusters in the numerical ranking, and the addition of new constraints may have the effect of making such a cluster more stable.[6]

[4] At the LFG Conference in Bergen we presented figures that were based on a slightly different definition of filter efficiency, namely the quotient of the number of readings discarded by the filter divided by the total number of readings. Since this initial definition prevents filter efficiency from taking 1.0 as a value and it is highly dependent on the total number of readings for a given sentence, our new definition is more appropriate.

[5] The exact definition is $F_\beta = (1+\beta^2)\frac{FE \times FF}{FE+\beta^2 FF}$, β being set to 0.5.

[6] To anticipate the experimental results however, we could not observe the effect of getting a more relaxed filter by providing a larger set of interacting OT marks.

In a pilot study, we thus established OT tableaux containing the OT marks employed in the German ParGram LFG and ran the GLA on these. This allowed us to identify OT marks which were reranked particularly often and/or which were regularly both demoted and promoted. Two such marks were *ObjInVorfeld*[7] and *LabelP*.[8] After inspection of a certain number of sentences where these OT marks made the correct reading(s) suboptimal, we introduced new, more fine-grained OT marks such as *ObjPersPronoun* (which disprefers the interpretation of personal pronouns as objects) and *SubjIndef* (which disprefers the interpretation of indefinite noun phrases as subjects), hoping these would allow to make the correct reading(s) optimal for more sentences.

In order to be able to control whether this is effectively the case, we established two sets of tableaux: the first one, henceforth the "all marks" set, contains both the 59 original and the 54 additional, newly introduced OT marks; the second one, henceforth the "original marks" set, contains only the original marks. Both sets were in turn split up into a training and a test set, so that we can examine how well rankings learned from the training sets generalize to unseen data.

We then ran the GLA on the training portions of both the "all marks" set and the "original marks" set. For training, we used a "traditional" GLA setting, i.e. a setting where the effective numerical rank of an OT mark diverts from its grammatical rank within a normal distribution due to added noise and where marks making wrong predictions are demoted or promoted on the numerical scale by a constant called plasticity (cf. Boersma (1998)).

In order to evaluate the resulting rankings, we used a variant of the GLA without any noise intervening at evaluation time. This allowed us to evaluate the resulting numerical rankings as if they were strict relative rankings, which is the type of ranking used in XLE. Moreover, the application of this variant of the GLA to the data allows us to identify marks that, even with an "optimal" OT mark ranking, cause correct readings to be evaluated as suboptimal. In this sense, it is not only a tool for the evaluation of OT mark rankings as a whole, but it can also be used to evaluate how reliable single OT marks are.

2.5 Relaxing the filter

An important additional step in our experiments (based both on the "original marks" and based on "all marks") was the attempt to modify an existing set of marks and ranking in order to increase filter fidelity – without decreasing filter effectiveness too much. Besides learning a more adequate ranking, this could be achieved in the following ways: (1) deactivating certain OT marks, such that their filtering effect is removed, and (2) grouping together constraints with a very similar rank. For step (1), it is important to identify appropriate marks for deactivation. In the pre-filter scenario, marks that are typically involved in "highly contingent disambiguation decisions" (i.e., decisions that may turn out one way or the other) should be excluded from the set, since they will eliminate the correct solution in relatively many cases. To identify marks for deactivation we explored two strategies: (a) inspecting the results obtained with the GLA variant without noise given a ranking obtained through a training run and to deactivate the marks that caused wrong predictions and (b) automatically deactivating a certain proportion of marks being associated with ranks at the lower side of the numerical scale.

For step (2) – grouping of similarly ranked constraints – we used various threshold values representing the minimal distance that two marks have to be away from each other in order to be attributed to distinct groups with distinct ranks.[9]

[7] *ObjInVorfeld* disprefers the interpretation of case-ambiguous noun phrases in the vorfeld, i.e. the position in front of the finite verb in verb-second clauses, as objects in sentences such as [$_{\text{NP}-\text{SUBJ}}$ *Hans*] *sieht Maria* 'John sees Mary' vs. [$_{\text{NP}-\text{OBJ}}$ *Hans*] *sieht Maria* 'John, Mary sees'.

[8] *LabelP* disprefers the interpretation of a noun phrase as a close apposition to another noun phrase in sentences such as *Hans stellt* [$_{\text{NP}-\text{OBJ}}$ *das Auktionshaus Ebay*] *vor* 'Hans presents the auction house Ebay' vs. *Hans stellt* [$_{\text{NP}-\text{OBJ}}$ *das Auktionshaus*] [$_{\text{NP}-\text{OBJ}\theta}$ *Ebay*] *vor* 'Hans presents the auction house to Ebay'.

[9] Such a grouping can "relax" the pre-filter in the following way: assume two dispreference marks C_1, C_2 end up with

3 Experiments

3.1 Data

For our experiments, we parsed the 40,020 sentences of the Release 1 of the TiGer Corpus[10] with a variant of the German ParGram LFG (Dipper 2003) into which we had integrated the new, more fine-grained OT marks and in which the evaluation of almost all OT marks had been deactivated.[11] Out of 40,020 TIGER sentences, 23,962 received a full parse.[12] The resulting f-structure charts (packed f-structure representations) were matched against the f-structure charts previously derived from the TiGer graph annotation (Forst 2003a,b); OT mark profiles corresponding to the f-structure charts produced by the grammar were established, the TiGer-compatible readings being marked as target winners in them. Since the matching of the grammar output against the TiGer-derived representations is very time-intensive when the number of different analyses contained in the two f-structure charts involved (or at least in one of them) is very high, we had to limit the maximum number of matches performed between individual analyses to 10,000.[13] By doing this, we obtained 6,418 OT mark profiles, associated with the sentences for which a proper subset of all parses is compatible to the TiGer-derived f-structure charts. (The granularity of the TiGer annotation is not sufficient to always determine one single parse as the correct one.) Sentences for which all analyses were compatible with the TiGer-derived representations had to be discarded, since there would be nothing to be learned from OT tableaux associated with sentences of this kind.

An example of an OT mark profile obtained this way is the one associated with the following sentence:

(4)

Anlaß	für	all	das	gab	aus	Schweizer	Sicht	das	neue	österreichische	Abfallwirtschaftsgesetz.
rise	for	all	that	gave	from	Swiss	view	the	new	Austrian	waste management law

‘From the Swiss point of view, it was the new Austrian waste management law that gave rise to all that.’

reading	AdvAttach	Obj	ObjCommon	ObjDef	ObjNoSpec	...
A1	0	1	2	1	1	...
A2	0	1	2	0	2	...
A3	0	1	2	0	2	...
A4	0	1	2	1	1	...
A5-B1	1	1	1	0	1	...
☞A5-B2	0	1	1	0	1	...
A6-B1	1	1	1	1	0	...
A6-B2	0	1	1	1	0	...

Table 1: Sample OT mark profile for sentence (4)

similar, but distinct, decreasing ranks. Without grouping, all readings with a C_1 mark would be filtered out, independent of their C_2 marking, whereas after grouping a reading with a C_1 and no C_2 mark is treated like one with a C_2 and no C_1 mark.

[10] http://www.ims.uni-stuttgart.de/projekte /TIGER/TIGERCorpus/

[11] The OT mark *GuessedMassNoun* was kept active, as its deactivation would have led to such an enormous increase in the numbers of readings produced by the grammar that the matching mentioned below would not have been feasible for most sentences.

[12] The grammar version employed was not chosen for its coverage but for its adherence to ParGram f-structure decisions which are reflected in the TiGer-derived representations. Moreover, the newly introduced OT marks caused a slight slow-down of the grammar, which caused additional timeouts wrt. other grammar versions.

[13] This means that sentences for which the product of the number of analyses in the grammar output and the number of analyses in the TiGer-derived representations was greater than 10,000 were discarded.

The directly resulting 6,418 OT mark profiles, which correspond to the "all marks" set, were randomly split up into a trainings set of 5,755 and a test set of 663. Then we created the "original marks" set, by replacing all values in the columns of the newly introduced OT marks by zero, and split it up into a trainings set of 5,755 profiles and a test set of 663 along the same lines as the "all marks" set.

3.2 Training and first results

The 5,755 OT mark profiles of both the "all marks" trainings set and the "original marks" training set were input to an implentation of the GLA that allows for multiple target winners. The learning was performed with a plasticity of 0.2 and in 10 iterations over the whole training set, each datum being considered 5 times. The result of these training runs were two different numerical OT mark rankings, one for the "all marks" set and another one for the "original marks" set.

The results of these two training runs are summarized in table 2.

	original marks			all marks		
ranking employed	filter fidelity	filter efficiency	weighted F-score	filter fidelity	filter efficiency	weighted F-score
after 10 iterations GLA	81.9	84.9	82.5	78.3	87.2	79.9
uniform ranking	82.7	83.3	82.8	77.2	84.1	78.5
original manual ranking	80.5	84.8	81.3			

Table 2: Results of GLA learning on test sets

As can be seen from the figures, the ranking of the OT marks does not play a major role. Although the automatically learned ranking performs better than the manually determined ranking originally used in the German ParGram LFG, both in terms of filter fidelity (81.9% vs. 80.5%) and filter efficiency (84.9% vs. 84.8%), the improvement from the latter to the former is very slight. Also, we have to state that, for the "original marks" set, the automatically learned ranking performs worse than a uniform ranking, i.e. a ranking where all marks are equally strong, in terms of filter fidelity (81.9% vs. 82.7%), even if filter efficiency is better (84.9% vs. 83.3%). The weighted F-score we employ confirms this picture (82.8% vs. 82.5%).

Comparing the results for the "original marks" set and the "all marks" set, the observation is that although the additional marks allow for a better filter efficiency, they have a negative effect on filter fidelity. This result is a bit disappointing, because initially, we had hoped to improve both filter efficiency and filter fidelity by providing the new marks. At the same time, it is not all that surprising, since the more OT marks are used for disambiguation, the more difficult it is, of course, to maximise filter fidelity.

As to the rankings' ability to generalize from the training data to the unseen test data, we can see in table 3 that both the figures themselves and the patterns observed above are comparable between the training sets and the test sets.

3.3 Relaxing the filter

Given that the filter fidelity we achieved with the learned ranking hardly exceeded 80%, we thought of ways of relaxing the OT filter in order to increase this value. At the same time, filter efficiency was not supposed to be affected too badly.

Inspecting (and deactivating) "problematic" OT marks: The first approach we took was to inspect the OT marks that, even with the automatically learned ranking, caused correct readings to be evaluated

	original marks			all marks		
ranking employed	filter fidelity	filter efficiency	weighted F-score	filter fidelity	filter efficiency	weighted F-score
after 10 iterations GLA	80.2	85.4	81.2	78.0	87.3	79.7
uniform ranking	81.8	82.7	82.0	76.7	83.7	78.0
original manual ranking	79.6	85.2	80.7			

Table 3: Results of GLA learning on training sets

as suboptimal. Examples of these were, as in the pilot study mentioned in 2.4, *ObjInVorfeld* and *LabelP*. Apparently, even the newly introduced OT marks did not allow us to counterbalance them in cases where they caused wrong predictions, which leads us to the opinion that these OT marks, instead of being evaluated in the pre-filter step, should be integrated into the log-linear model as properties. As such, they can contribute to choosing the correct reading in the final disambiguation step, where, moreover, they can interact with other properties, such as the ones that weigh subcategorization frames.

Another category of OT marks that still made wrong predictions were robustness OT marks such as *AdvAttach* and *MassInPl*. The purpose of these OT marks is mainly to disprefer fall back rules that are implemented for cases where lexical information is lacking and, as a consequence, they interact tightly with this kind of information. Due to missing or erroneous information in the lexicons, it can happen that they make wrong predictions, although they are fairly reliable in all other cases. We deactivated most of these OT marks, i.e. those which caused relatively many wrong predicions, but they can potentially be reactivated once the lexicons they interact with have been improved.

As a reaction to the inspection of the "problematic" OT marks, a variant of the data was created where these marks are deactivated. We henceforth call this set of data the "unproblematic marks" set.

"Translating" the numerical rankings into strict rankings: For use in XLE, the numerical rankings obtained from GLA learning have to be "translated" into strict rankings in which marks may be grouped as equally strong. The easiest way of doing this is, of course, to have one group for each distinct numerical ranking. However, this may not be the most appropriate method of "translating" a numerical ranking into a strict ranking, because it completely ignores the information contained in the distance between two rankings. A possible alternative is to group all OT marks whose ranking have a distance smaller than a given threshold. This way, the number of groups of equally ranked OT marks is reduced, which should allow for better generalisation, and, more importantly, some of the information contained in the distance between rankings is taken into account. We experimented with groupings of this kind with thresholds 2.0 and 5.0.

The resulting rankings were then applied to both the "original marks" data set and the "unproblematic marks" set. The results are shown in table 4.

Just as in our first results (cf. subsection 3.2), we observe that the ranking has only a little influence on the results. Nevertheless, filter fidelity can be improved slightly by grouping marks whose ranks are not very distant, whithout filter efficiency being affected considerably. Taking the figures from the training data into account (which are not displayed here), the conclusion could be that a grouping with a threshold value of 5.0 performs best, since it basically achieves the same filter fidelity as the uniform ranking, while allowing for a slightly higher filter efficiency.

More importantly, table 4 shows that the deactivation of "problematic" marks can increase the filter fidelity considerably. We achieve a filter fidelity of about 96%, while still discarding more than 62% of the readings as suboptimal.[14] This set-up also yields the highest weighted F-score of all our experiments:

[14] This can arguably be considered an underestimation, because the effect of the OT mark GuessedMassNoun, mentioned in

ranking employed	original marks filter fidelity	 filter efficiency	 weighted F-score	unproblematic marks filter fidelity	 filter efficiency	 weighted F-score
after 10 iterations GLA	81.9	84.9	82.5	95.9	62.2	86.5
grouped with threshold 2.0	82.4	84.8	82.9	96.2	62.1	**86.7**
grouped with threshold 5.0	82.5	84.7	82.9	96.2	62.1	**86.7**
uniform ranking	82.7	83.3	82.8	96.1	60.3	85.9

Table 4: Results of disambiguation with “original marks” and “unproblematic marks”, marks being grouped according to different methods

86.7%.

Deactivating portions of the OT marks according to their ranks: An alternative approach to disactivating “unreliable” OT marks we experimented with was to discard a certain proportion of the marks corresponding to the ranks at the lower end of the numerical scale. We ran this experiment for both the “original marks” set and the “all marks” set, deactivating the lower 50% of the OT marks. The resulting variants of the data are henceforth called “upper 50% original” and “upper 50% all” respectively.

Tables 5 and 6 show the effect of discarding the lower 50% of the two OT mark sets. (The ranking used is the one obtained after 10 iterations of the GLA.)

original marks			upper 50% original		
filter fidelity	filter efficiency	weighted F-score	filter fidelity	filter efficiency	weighted F-score
81.9	84.9	82.5	99.5	41.3	77.6

Table 5: Results of disambiguation with “original marks” and “upper 50% original”

all marks			upper 50% all		
filter fidelity	filter efficiency	weighted F-score	filter fidelity	filter efficiency	weighted F-score
78.3	87.2	79.9	97.0	50.5	81.9

Table 6: Results of disambiguation with “all marks” and “upper 50% all”

Filter fidelity is greatly improved by this strategy, but unfortunately, there is a high price to be paid in terms of filter efficiency. In both settings, it drops to 50% or even less. Given that the filter fidelity for the “upper 50% all” set is comparable to the filter fidelity for the “unproblematic marks” set, but that the filter efficiency for it is considerably lower than for the “unproblematic marks” set, we conclude that it is a better strategy to identify problematic marks and then deactivate them than just to deactivate a certain proportion of the lower ranked marks.

subsection 3.1 as well, is not taken into account here, although it cuts down the number of readings considerably.

4 Discussion and conclusions

We presented a sequence of experiments exploring ways of empirical tuning for the first stage of a disambiguation architecture for linguistic grammars. This pre-filter is triggered by configurations that the grammar writer specifies as OT marks and uses a relative ranking among the marks. A somewhat surprising result is that training the constraint ranking on corpus data does not lead to a noticeable improvement over the use of a uniform ranking. However, it is very effective for identifying and deactivating marks that tend to exclude the correct readings in some cases. Both results are of course directly related to the somewhat unusual application context of the disambiguation routine as a pre-filter: if it were used as the only filter, one should certainly rely on a learned ranking to maximize filter effectiveness, and the OT marks that are problematic in the pre-filter scenario might well play an important role. For the given two-stage scenario however, our systematic empirical exploration showed that filter fidelity can be maximized most effectively by removing unreliable marks.

In future work, we plan to explore in more detail the technique of deactivating certain marks from the ranking automatically, among other things by combining this technique with the approach of grouping similarly ranked constraints together. Moreover, we will evaluate the effect of pre-filter variants on the training of the log-linear model used as the second disambiguation stage.

References

Bender, Emily M., Dan Flickinger, and Stephan Oepen. 2002. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe (eds.), *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pp. 8–14, Taipei, Taiwan.

Boersma, Paul. 1998. *Functional Phonology. Formalizing the interactions between articulatory and perceptual drives*. PhD thesis, University of Amsterdam.

Butt, Miriam, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*, Taipei, Taiwan.

Butt, Miriam, Tracy Holloway King, Maria-Eugenia Niño, and Frederique Segond. 1997. *A Grammar-Writer's Cookbook*. CSLI Publications.

Dipper, Stefanie. 2003. *Implementing and Documenting Large-scale Grammars – German LFG*. PhD thesis, IMS, University of Stuttgart. Arbeitspapiere des Instituts für Maschinelle Sprachverarbeitung (AIMS), Volume 9, Number 1.

Forst, Martin. 2003a. Treebank Conversion – Creating a German f-structure bank from the TIGER Corpus. In *Proceedings of the LFG03 Conference*, Saratoga Springs. CSLI Publications.

Forst, Martin. 2003b. Treebank Conversion – Establishing a testsuite for a broad-coverage LFG from the the TIGER Treebank. In *Proceedings of the EACL Workshop on Linguistically Interpreted Corpora (LINC '03)*, Budapest.

Frank, Anette, Tracy H. King, Jonas Kuhn, and John Maxwell. 2001. Optimality Theory style constraint ranking in large-scale LFG grammars. In Peter Sells (ed.), *Formal and Empirical Issues in Optimality-theoretic Syntax*, pp. 367–397. Stanford: CSLI Publications.

Johnson, Mark, Stuart Geman, Stephen Canon, Zhiyi Chi, and Stefan Riezler. 1999. Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99), College Park, MD*, pp. 535–541.

Kaplan, Ronald M., Tracy H. King, and John T. Maxwell. 2002. Adapting existing grammars. The XLE approach. In *Proceedings of the COLING-2002 Workshop on Grammar Engineering and Evaluation*, Taipei, Taiwan.

Kaplan, Ronald M., Stefan Riezler, Tracy King, John T. Maxwell III, Alexander Vasserman, and Richard Crouch. 2004. Speed and Accuracy in Shallow and Deep Stochastic Parsing. In *Proceedings of the Human Language Technology Conference and the 4th Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'04)*, Boston.

King, Tracy Holloway, Stefanie Dipper, Annette Frank, Jonas Kuhn, and John Maxwell. 2004. Ambiguity management in grammar writing. *Research on Language and Computation* 2:259–280.

Manning, Christopher D. 2003. Probabilistic syntax. In Rens Bod, Jennifer Hay, and Stefanie Jannedy (eds.), *Probabilistic Linguistics*, pp. 289–341. Cambridge, MA: MIT Press.

Oepen, Stephan, and John Carroll. 2000. Parser engineering and performance profiling. *Natural Language Engineering* 6:81–97.

Prince, Alan, and Paul Smolensky. 1993. Optimality theory: Constraint interaction in generative grammar. Technical Report Technical Report 2, Rutgers University Center for Cognitive Science.

Riezler, Stefan, Dick Crouch, Ron Kaplan, Tracy King, John Maxwell, and Mark Johnson. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02), Pennsylvania, Philadelphia*.

Tesar, Bruce B., and Paul Smolensky. 1998. Learnability in Optimality Theory. *Linguistic Inquiry* 29: 229–268.

Toutanova, Kristian, Christopher D. Manning, Stuart M. Shieber, Dan Flickinger, and Stephan Oepen. 2002. Parse disambiguation for a rich HPSG grammar. In *First Workshop on Treebanks and Linguistic Theories (TLT2002)*, pp. 253–263.