IMPROVED LANGUAGE MODELING BY UNSUPERVISED ACQUISITION OF STRUCTURE

Klaus Ries[†,§]
ries@informatik.uni-freiburg.de

Finn Dag Buø[†]
finndag@ira.uka.de

Ye-Yi Wang[‡]
yyw@cs.cmu.edu

Interactive Systems Laboratories[†,‡]

[†] University of Karlsruhe, Karlsruhe, Germany
[‡] Carnegie Mellon University, Pittsburgh, PA, USA
[§] University of Freiburg, Freiburg, Germany

ABSTRACT

The perplexity of corpora is typically reduced by more than 30% compared to advanced n-gram models by a new method for the unsupervised acquisition of structural text models. This method is based on new algorithms for the classification of words and phrases from context and on new sequence finding procedures. These procedures are designed to work fast and accurately on small and large corpora. They are iterated to build a structural model of a corpus.

The structural model can be applied to recalculate the scores of a speech recognizer and improves the word accuracy. Further applications such as preprocessing for neural networks and (hidden) markov models in language processing, which exploit the structure finding capabilities of this model, are proposed.

1. CLASSIFYING ENTITIES FROM CONTEXT VIA ITERATED REESTIMATION

The most widespread criterion for the classification of words and phrases in linguistics is the *replacement test*, which states, that two linguistic entities are the same, if they can be mutually substituted in sentences where they appear. [Fin93] extended this criterion and states, that linguistic entities, that have similar context should be assigned similar categories. Our definition still goes further and says that a *linguistic entity and its context are the same.* As the context of an entity itself consists of entities this definition is somewhat circular: If we know a categorization of linguistic entities, we can construct the context and hence a classification of these entities.

As an initial classification, we assign each of the most frequent n-1 entities a singleton class while all other entities are collected in a "rubbish" class. This initial assignment is motivated by the fact, that the most frequent entities usually cover most of the corpus, e.g. 89% of the English Scheduling Task is covered by just 200 words – this behavior is predicted by Zipf's law [Zip35][1]. A given classification may be improved by classifying the context-vectors of the entities. The context-vectors are calculated by counting the classes of the entities in fixed intervals around an entity.

Assume we want to calculate the context vector of the word town with the context-intervals $[1,1]$ resp. $[-3,-1]$. The context-vector consists of the counts of the classes found one word after resp. one to three words before the word town in the corpus. To get a context vector for the context specification $[1,1],[-3,-1]$ the vectors for these intervals are concatenated. Usual context specifications are $[-1,1],[1,1]$ and $[-3,-1],[-1,-1],[1,1],[1,3]$.

This classification procedure is iterated, and under certain restrictions to the classification procedure, an EM-algorithm [DLR77, Rie94] is obtained. The idea behind the EM formulation is that the classes used to calculate the context vectors and the classes found using the clustering algorithm are the same. Further restrictions on the context-vectors and the choice of the classification procedure lead to the classification criterion of class-based bigrams [BdP$^+$92, Rie94]. If we would not iterate the classification this procedure would be equivalent to [Fin93] – our algorithm is therefore a proper generalization of [BdP$^+$92]

The first author was supported by the Friedrich Naumann Foundation under a student grant during the completion of this work at the University of Karlsruhe, Germany, and is now doing graduate studies at the University of Freiburg, Germany. The second author was supported by the Research Council of Norway. This research was performed at the University of Karlsruhe and at the Carnegie Mellon University, Pittsburgh and was supported in part by the ARPA, NSF and BMFT. The authors gratefully acknowledge their chance to work under these conditions and the permanent support and inspiration of the staff at Interactive Systems Laboratories, especially Monika Woszczyna for providing the lattices and related software. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of any organization mentioned above.

[1] Zipf's law states $f_r \sim \frac{1}{r}$, where f_r is the frequency of the r-most probable entity. As the number of entities is very high this distribution of word frequencies is highly skewed and we will never get enough data to estimate parameters of entities, that occur seldom. This can be prevented only by clustering entities with similar properties together and hence limiting the number of entities involved.

0-7803-2431-5/95 $4.00 © 1995 IEEE

and [Fin93]. The iterative process improves the classification performance compared to [Fin93] especially on small corpora (<50.000 words), as the estimation of the context is better – it is based on classes, that have much higher probabilities than words. The speed of [Fin93] is maintained in our implementation, and we need approx. 10 min for 8 iterations of the classification process on a standard workstation for a corpus of 50.000 words and 100 classes. The algorithm is expected to scale up sublinear in practice as the speed of the classification procedure itself is not affected by the size of the corpus. [BdP+92] is improved by allowing wide context windows and the classification is sped up considerably.

In our implementation the classification is done by an extension of a hill-climbing procedure, which improves the variance criterion directly [SL77]. As any incremental clustering algorithm needs an initial class assignment, we may use the classification of the last step as the initial class assignment of the next step. This simple technical trick is crucical to improve the classification and also results in considerable speed up. As suggested by [Fin93], we preprocess the context vectors by calculating the ranks of their entries. [Rie94] also extends the calculation of context vectors to a more general class including fuzzy classification [Yan93].

2. SEQUENCE MEASURES AND SEQUENCE IDENTIFICATION

Marking sequences of entities, which are tied together, may be divided into two steps: First, we assign each sequence a score (the sequence measure), which determines their likelihood of being a sequence, and second, we use this measure to identify the sequences, that are present in some actual portion of the corpus by a sequence identification procedure.

The first type of sequence measures may be called *indirect sequence measures*, as they make use of a usual measure of coincidence $co(X,Y)$ between random variables X and Y. The *mutual information* $\mathcal{MI}(X,Y)$ has proven to be a good choice as a measure of coincidence in our studies. The indirect sequence measure of the sequence $\langle e_1, \ldots, e_n\rangle$ is

$$s(\langle e_1, \ldots, e_n\rangle) = min_i \quad co(\langle e_1, \ldots, e_i\rangle, \langle e_{i+1}, \ldots, e_n\rangle)$$

If we use $\mathcal{MI}(\cdot,\cdot)$ as a measure of coincidence the sequence measure of an entity is the minimal $\mathcal{MI}(\cdot,\cdot)$ between a prefix and the rest of that entity.

This formulation of a sequence measure is motivated by the idea that a sequence of words occurring in the corpus is an uninteresting sequence, if it is found as the suffix and the prefix of neighboring interesting sequences.

The second class of measures are direct measures and are not linked to measures of coincidence. The only direct measure found in the literature was [Suh73], but it did not proof as effective as the indirect measure based on mutual information. Suhotins measure is

$$\frac{1}{2n}\sum_{i=1}^{n-1} Pr(\langle e_1 \ldots, e_n\rangle | \langle e_1, \ldots e_j\rangle \vee \langle e_{j+1}, \ldots e_n\rangle)$$

Assume, though, that we already had a procedure that marks the sequences found in a corpus. We may then take the counts of the sequences found in the corpus as a direct sequence measure. The calculation of the sequence measure and the sequence identification procedure may be iterated and we call the resulting measure *iterated marking frequency*. This measure has proven to be the best one by manual inspection and the resulting entropy reduction. Note that we still need indirect measures or [Suh73] to have an initial sequence marking, and that the *iterated marking frequency* depends on it crucially.

	elements		elements
D	B D F G SCH	A	A I O U
CH	N CH NG	AEH	E AI AU EU AEH UEH
R	K P R	UE	AE OE UE OEH
T	L M S T	IE	AH EH ER IE OH UH
H	H J V Z	SIL	+K QK +EH +H# +GH +GN SIL

Figure 1: Phoneme Classification: This is an example of a classification of phonemes in the English Scheduling Task. The classification was made with 10 classes and the context was $[-1,-1],[1,1]$. The classes seem to reflect acoustic similarities, though the only information available to the classification algorithm was the phonetic transcription of dialogues.

The most effective sequence identification procedure we found, may be stated rather simple:

1. Rank all sequences found in the corpus by the assigned sequence measure.
2. Take the highest scoring sequence and replace any occurrence of it in the corpus by a new and unique symbol representing this sequence. The words, that make up this sequence, can be used in successive join steps, if all words in the sequence are used in the join.
3. Delete this sequence from the list of sequences and goto 2, unless no sequence is left.

Calculating the *indirect measures* takes approx. 1 min on a standard workstation for a 50.000 words corpus, using the *iterated marking frequency* adds another 2 min.

This sequence identification with the indirect measure based on $\mathcal{MI}(\cdot,\cdot)$ may be compared with [MM91]. Instead of using two word sequences plus their context to search for sequence boundaries we are looking for the sequences themselves. This has proven [Rie94] more effective than the corresponding approach to search for boundaries with sequences of length greater than 2. Context also plays a significant role in our approach: Assume we want to process the sequence `A B C` and use the sequences `A B` and `B C` to join `A B C`. If `A B` has a higher sequence measure than `B C`, the left context `A` of `B C` prohibits their join.

3. COMBINING CLASSIFICATION AND SEQUENCE IDENTIFICATION

There are two alternatives, how classification and sequence identification may be combined. The first one would

```
specif specif no_i'm +ls+_+ you_li +h#+ +h#+
                                   +uh+ +uh+
                                   okay okay
                             i_gues i i
                                   guess guess
                    seem i i
                         need need
                         to to
                         meet meet
                         with with
                         you you
          +ls+_m +ls+_m for for
                             aftern about about
                                   two two
                                   hours hours
                     though during during
                            the the
                            week week
     it_is +um+_m or_wel here's +um+ +um+
                                   +muell +muell+
```

Figure 2: Iterated sequence identification: The classification/identification procedure was iterated eight times, the entity context vectors chosen according to the intervals $[-1,-1][1,1]$, the classification resulted in 400 classes and 4000 sequences were used in each run. An identified sequence is indented, and each sequence is represented by a truncated identifier. The original words are printed in boldface.

be to redefine context in terms of sequences and may be seen as an idea to include more global information into the entity classification process. The second one would be to identify the sequences based on word classes. In our current studies we only pursued the second possibility since the performance of the word classification procedures are satisfactory without the use of long distance knowledge.

The process of entity classification and sequence identification may be iterated to build a structural model of the corpus by the following procedure:

1. Classify the entities.
2. Identify sequences of entities by identifying sequences of their respective classes found in step 1.
3. Replace each identified sequence of entities by a new unique symbol representing the classes in the sequence.
4. Goto 1 or Stop.

4. CALCULATING THE PERPLEXITY

The entity classification and sequence measure, which is obtained by one iteration of the classification/identification procedure, may be used to transmit a corpus by the following procedure:

1. Run the sequence finding algorithm based on the predetermined sequences and classes.
2. Transmit the corpus obtained by replacing each identified sequence of classes by a unique symbol.

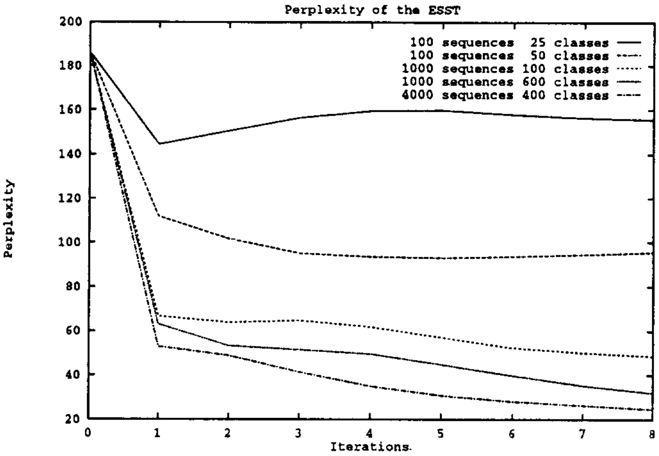

Figure 3: Perplexity of the ESST while iterating the model: Most of the reduction is done in the first iteration. As opposed to bigrams, where no further iterations are sensible, this model can be be used iteratively. The use of 4000 sequences and 400 classes in each step is the best choice possible on the ESST.

3. For each class: Transmit the corpus, that consists of all words, that are found in marked sequences and belong to this class.

The transmission of the testset is done similar to a monogram model: Each word w on the testset is transmitted with a code of length $\log_2(p_w)$, where p_w is the probabiliy of w on the trainingset. To transmit the corpus with the iterated model the corpus in step 2 may again be transmitted with its own structural model. This process must of course terminate and the last corpus must be transmitted with a monogram model. The perplexity PP is calculated straightforward from the number of bits $\log_2(PP)$ to transmit a word using the structural model on the testset. This corresponds to the usual definition of PP for n-gram models.

The unknown word is modeled by linear discounting. We do not transmit sequences as new symbols, that are not in the training set – instead, we split the sequence into its words again. This decision is sensible since an unknown sequence will never be used to further joins.

5. EXPERIMENTS

All experiments were run under standard settings on the English, German and Spanish Scheduling Task (ESST, GSST and SSST) that consists of spontaneous speech dialogs concerning time scheduling. The results of figure 3 indicate, that using 4000 sequences and 400 classes is a sensible setting[2]. These results compare to n-gram models [SW94] as follows:

[2] The application of a complexity measure based on the description length [Ris89] for determining the optimum setting indicates the choice of 1000 sequences and 400 classes [Rie94].

	ESST	GSST	SSST
monogram	179	223	164
bigram	38	82	74
trigram	36	75	67
class-based bigram	39	84	73
interpolated standard and class-based bigram	35	73	66
our structural model	25	41	19

Interesting enough the perplexity reduction on the Spanish Scheduling Task is the most impressive one and our structural model tells us, that the perplexity of our Spanish corpus is lower than our English corpus. We have found in experiments, that the use of bigram models instead of monograms for the transmissions of an intermediate corpora used in our iterated model usually rises the perplexity of the model. This result is explained by the fact, that we have already extracted all bigram information available – further attempts lead to performance loss.

We have also started to evaluate the impact of this dramatic perplexity reduction on the recognition performance and ran a small test on the German Scheduling Task. We have mixed the score of the original language model (a bigram model), the original acoustic scores and the scores of our structural model in a weighted sum. We have then calculated a new score for the 150 best hypothesis according the recognizer and extracted the best hypothesis according to the new score. As a result the word accuracy rises from 59.7% to 62.2% on our testset. Naturally, this performance is limited by the number of correct words in the 150 best hypothesis from the decoder. If we rescore only N-best lists, that do contain the correct hypothesis, rescoring with our model yields an improvement from 81.6% to 89.5%.

6. CONCLUSION AND FURTHER APPLICATIONS

The structural model has been proven to obtain a much lower perplexity on the ESST, GSST and SSST data than traditional n-gram methods and their extensions. This was achieved by classifying entities and joining them into larger sequences. At the core of the reduction is the idea, that this template of classes is tied to together and uttered as a whole. The only variation, that is possible, is the actual instance, the surface form of this template, that results from instantiating the words into the template. The transmission policy of the data is thereby rather simple – we just transmit monograms. Our model cannot replace an n-gram model in the recognizer currently, as these must work incrementally. The actual influence of this model on the recognizer performance is still an open question, as the mechanisms applied are rather different. The first experiments show the feasibility of this approach. Current work on the structural model includes tight coupling with n-gram models and the improved handling of unknown words. We will also try to investigate the notion of government and binding in our model, that is used in modern linguistic theory. It might be reformulated as a special connection between instances of a template. Simple examples as the genus agreement between article and noun in German may already be captured by introducing bigram information in our model.

The authors believe, that the structure, which performs the reduction, may also be used to enhance the power of other non-symbolic methods. [Rie94] has shown how to produce 1 out of n and distributed representations of words for the usage in connectionists parsing. The same holds for sequences, that may introduce long distance information by providing a parse stack visible to the network. [WW94] used (hidden) markov models for speech-act modeling and reported, that the word classification using our techniques is almost as powerful as handmade word classes and outperforms the approach without classes considerably. Another application of the word classification procedures is the generalization to bilingual word classification [Rie94].

7. REFERENCES

[BdP+92] Peter F. Brown, Peter V. deSouza, Vincent J. Della Pietra, Robert L. Mercer, and Jennifer C. Lai. Class based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992.

[DLR77] A. P. Dempster, N.M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38, 1977. with discussion.

[Fin93] Steven Finch. *Finding Structure in Language.* PhD thesis, University of Edinburgh, 1993.

[MM91] David M. Magerman and Mitchell P. Marcus. Distituent parsing and grammar induction. volume D-91-09, pages 122a–122e. DFKI, March 1991.

[Rie94] Klaus Ries. Korpusbasierte Techniken zum Lernen von übersetzung spontan gesprochener Sprache (Corpus-Based Techniques for Learning the Translation of Spontanous Speech, in German). Diploma thesis, Universität Karlsruhe, 1994.

[Ris89] J. Rissanen. *Stochastic Complexity in Statistical Inquiry.* World Scientific Publishing, 1989.

[SL77] Detlef Steinhausen and Klaus Langer. *Clusteranalyse.* de Gruyter, 1977.

[Suh73] B. V. Suhotin. Methode de dechiffrage, outil de recherche en linguistique. *TA Informationes*, 2:3–43, 1973.

[SW94] B. Suhm and A. Waibel. Towards better language models for spontaneous speech. In *ICLSP*, volume II, pages 831–834, Yokohama, Japan, 1994.

[WW94] Monika Woszczyna and Alex Waibel. Inferring linguistic structure in spoken language. In *Proceedings of the International Conference on Spoken Language Processing.* ASJ, 1994.

[Yan93] Miin-Shen Yang. On a class of fuzzy classification maximum likelihood procedures. *Fuzzy Sets and Systems*, 57:365–375, 1993.

[Zip35] G. Zipf. *The Psycho-Biology of Language.* Houghton Millin, 1935.