Chapter 13

A Goodness Measure for Phrase Learning via Compression with the MDL Principle

CHUNYU KIT

ABSTRACT. This paper reports our ongoing research on unsupervised language learning via compression within the MDL paradigm. It formulates an empirical information-theoretical measure, *description length gain*, for evaluating the goodness of guessing a sequence of words (or characters) as a phrase (or a word), which can be calculated easily following classic information theory. The paper also presents a best-first learning algorithm based on this measure. Experiments on phrase and lexical learning from POS tag and character sequence, respectively, show promising results.

1 Introduction

Grammar induction from a naturally-occurring text corpus is in the general domain of inferring a theory (or model) to account for observed data. Practical techniques for grammar induction have many important applications for a wide range of natural language (NL) and speech processing tasks. Researchers have devoted tremendous effort to the research in the past decades.

Inferring a probabilistic grammar from NL data, as revealed by most previous practice, involves in general two essential sub-tasks, one to infer a set of phrase structure rules (or productions), another to estimate a correspondent set of probabilistic parameters (i.e., production probabilities) that optimise the fit of the grammar to the data. The inference can be viewed as a search process for the best grammar allowable in a predefined *grammar* (or *hypothesis*) *space*. If a set of permissible rules or rule formats (e.g., *Chomsky normal form* (CNF)) are given, it is popular to apply the *Baum-Welch* (or *forward-backward*) *algorithm* and its extension, the *inside-outside algorithm* [Baker79, LY90], to estimate the probabilistic parameters for regular grammars (e.g., hidden Markov models (HMMs)) and SCFGs, respectively. There are also many other sophisticated algorithms to facilitate the searching, e.g., *genetic algorithm* [Hol75] and *stimulated annealing* algorithm [KGV83].

However, no matter how sophisticated is the search method in use, the goodness criterion to guide the searching remains a critical issue. It is this

Proceedings of the Third ESSLLI Student Session
Ivana Kruijff-Korbayová (editor).

criterion that tells a search algorithm which grammar is better. In this paper we report our ongoing research on this specific task to develop a suitable criterion for unsupervised phrase structure learning from a natural language. It is based on the classic and algorithmic information theory [Shan48, CT91, Solo64, Kol65, Chai66, LV93] and on the *Minimum Description Length* (MDL) principle [Ris78, Ris89].

The rest of the paper is organised as follows: section 2 reviews previous research on language learning, focusing on the goodness criteria; section 3 discusses language learning via compression; section 4 formulates description length calculation in bits, adhering to Rissanen's [Ris78, RR94] insight on the objectiveness of code length; section 5 develops an information-theoretical measure for phrase learning, the *description length gain* or *compression effect* as an indicator to how good a word sequence is as a phrase, and a best-first learning algorithm with this measure; section 6 reports phrase and lexical learning experiments, and the experimental results are given in Appendix A and B, respectively; the last section concludes the paper with a brief discussion on future work.

2 Criteria for Phrase Structure Learning

Given no prior knowledge but a corpus as a sequence of linguistic symbols (e.g., characters, words or POS tags) as data, unsupervised phrase learning needs to determine which segments of the sequence are phrases or other target linguistic units (or structures). This learning has to be carried out through guessing, guided by some reliable information-theoretical criterion on how good a fragment of the sequence is as a phrase. The guessing needs to consider all n-gram items in the corpus. Although n-grams of arbitrary lengths in a large-scale corpus are known to be huge in number, the *Virtual Corpus* (VC) [Kit95], based on *suffix array* data structure [MM90, NM94] and a bucket-radix sort, can be employed as a fairly efficient approach to handling them, including counting and, more importantly, storing and retrieval. It is reported that the VC system can prepare the entire PTB WSJ part-of-speech (POS) tag corpus, of 1.3M tags, for outputting all n-grams within about 1.5 minutes on a Sun SPARC station 4. It is an ideal operational basis for inducing phrases from n-grams.

However, a more crucial issue in inducing phrases is a sound criterion to guide the guessing. Many criteria have been explored in previous research. For example, Cook *et al.* [CRA76] explore a hill-climbing search for a grammar of a smaller weighted sum of grammar *complexity* and the *discrepancy* between grammar and corpus; Lari and Young [LY90], Carroll and Charniak [CC92a, CC92b] induce various types of probabilistic context-free grammar (PCFG) with *inside-outside* algorithm [Baker79] for probabilistic parameter re-estimation; Brill *et al.* [BMMS90] derive phrase structures from a tagged corpus with *generalised mutual information*; and Brill and Marcus [BM92] attempt to induce binary branching phrases with distribution analysis using the information-theoretical measure *divergence*, derived from *relative entropy*; among many others. They have achieved significant progress, and also encountered some problems that seem unable to be resolved in their approaches, e.g., Brill *et al.* have to predefine *distitutes* as brute force to eliminate some unlikely phrases, e.g., `[Noun Prep]` and `[Prep DT]`, that are favoured by their criteria. de Marcken [deMa95a] has an in-depth discussion on the kind of issues involved in pure distribution analysis and on the disadvantages of the inside-outside algorithm for grammar induction.

Recent work in language modelling such as Stolcke [Sto94] and Chen [Chen95, Chen96] are in the theoretical framework of Bayesian modelling. Basically, Stolcke's work follows Cook *et al.*'s [CRA76] paradigm of searching by hill-climbing, but guided by maximum likelihood. Chen follows Solomonoff's inductive learning [Solo59, Solo60, Solo64], and uses the universal prior probability $p(G) = 2^{-l(G)}$ for grammar induction. Their learning systems are reported to work well on small to medium size artificial corpora, in terms of the theoretical measures like entropy, perplexity or likelihood. However, more concrete evaluations are expected to be based on the performance of learning from a large-scale naturally-occurring corpus like the Brown [FK82] or PTB corpus [MSM93].

3 Language Learning via Compression

The idea of learning via compression has been practised by researchers for a long time. A notable early discussion on the dual relationship between learning (i.e., detecting regularities in data) and compression is in Solomonoff's pioneering work [Solo64]. Some earlier related thoughts may be traced back to, for instance, Chomsky's considerations of the simplicity of grammar [Chom55, Chom57] and Zipf's *Principle of Least Effort* [Zipf35, Zipf49]. Occam's Razor appears to be the philosophical thought behind the idea of learning via compression. One of its interpretations is that among several alternative theories (or models) that explain the data equally well, the simplest one is preferable.

Compression, or a procedure equivalent to compression, has been widely exploited for unsupervised language learning at different linguistic levels, e.g., as in [Wol80, Wol82, CB94a, CB94b, Sto94, Chen95, Chen96, deMa95b, deMa96], among many others. Recently, the MDL principle is extensively applied as a guidance to language learning, e.g., as in [Sto94, deMa95b, deMa96]. The central idea in MDL is that a model M (e.g., a grammar G) for a set of data X (e.g., a corpus C as a sequence of characters or words) with the shortest description length of both the model and the data (given the model), i.e., $|M| + |X_{\text{given } M}|$ in bits, is the best model.

The difference among various learning-via-compression approaches lies in how the compression is carried out. It is unnecessary to have a real compression procedure for learning-via-compression, but a calculation of the probability of a model, p(M), in terms of its structure (e.g., grammar rules) and parameters is needed. With $p(M)$ calculated, a learning procedure can search for the most likely model, i.e., the model that maximises the *posterior probability* $p(M|X)$, following the Bayes' rule as below

$$M_{max} = \arg\max_{M} \frac{p(X|M)\,p(M)}{p(X)} = \arg\max_{M} p(X|M)\,p(M) \tag{13.1}$$

where $p(X)$ is a constant. Stolcke and Chen both use the prior $p(M) = c^{-l(M)}$ to calculate $p(M)$, where $l(M)$ is the description length of M encoded by a coding scheme chosen, and $c = 2$ if $l(M)$ is in bits. Different *ad hoc* coding schemes are used by different researchers to estimate $l(M)$.

Theoretically, searching for the most likely model for a given data set in a Bayesian modelling framework is equivalent to searching for a model with minimum description length, since using the negative logarithm, (13.1) above becomes

$$M_{max} = \arg\min_{M} -\log p(X|M) - \log p(M) = \arg\min_{M} |X_{\text{given } M}| + |M| \tag{13.2}$$

It straightforwardly follows from information theory [Shan48, CT91]. A Bayesian interpretation of MDL principle can also be found in [QR89, LV93].

There is another option that strictly follows the algorithmic information theory and the MDL principle to calculate bits, instead of probabilities: the model M and the data X (generated by M) can be encoded as one, instead of as two separate parts, using a universal coding. The universal coding is assumed, in theory, to be a coding scheme that is as good as any other scheme in coding a random sequence. For the purpose of language learning, a coding scheme such as Shannon-Fano (or Huffman) code [Shan48, Huf52, CT91] or arithmetic coding [Ris78, Ris89] would suffice for the purpose of description length calculation. Consequently, a general criterion for selecting phrase candidates among n-grams within a given corpus C is established as this: an n-gram is a better candidate if putting it as a phrase into G can lead to a shorter $|G + C_{\text{given } G}|$. That is, only n-grams with a good compression effect are candidates. Accordingly, phrase learning is to search for a subset G of this candidate set such that $|G + C_{\text{given } G}|$ is minimal.

4 Calculating Description Length

The application of MDL is independent of encoding scheme [Ris89, QR89]. To resolve the critical issue of calculating the description length $|G| + |C_{\text{given } G}|$, what we need is an ideal encoding scheme for calculation, rather than a real compression program.

Once an ideal scheme that can reach the best compression on a random sequence is assumed, the description length $DL(X)$ on a given finite data set

$X = x_1 x_2 \cdots x_n$, with a vocabulary V, can be calculated as below in terms of the *empirical entropy* $\hat{H}(X)$, following information theory [Shan48, CT91].

$$DL(X) = n\hat{H}(X) = -n \sum_{x \in V} \hat{p}(x) \log \hat{p}(x) = -\sum_{x \in V} c(x) \log[c(x)/n] \tag{13.3}$$

where $c(x)$ is x's count in X, and $\hat{p}(x) = c(x)/n$. $\hat{H}(X)$ is the minimum expected average codeword length per symbol in X. The relation between this description length and the perplexity $PP(X) = \hat{p}(x_1 x_2 \cdots x_n)^{-1/n}$ under a uni-gram model is straightforward as below.

$$\begin{aligned} \log PP(X) &= -\tfrac{1}{n} \log \hat{p}(x_1 x_2 \cdots x_n) = -\tfrac{1}{n} \log[\hat{p}(x_1)\hat{p}(x_2) \cdots \hat{p}(x_n)] \\ &= -\tfrac{1}{n} \sum_{i=1}^{n} \log \hat{p}(x_i) = -\tfrac{1}{n} \sum_{x \in V} c(x) \log \hat{p}(x) = \tfrac{1}{n} DL(X) \end{aligned} \tag{13.4}$$

Perplexity is an indication of the quality of a language model: a lower perplexity indicates a better model [Jel85]. The description length in (13.3) and its average (i.e., the empirical entropy) in (13.4) play the same role.

5 Learning Algorithm

Following the DL calculation above, the *description length gain* of selecting each instance of the substring $x_i x_{i+1} ... x_j$ (denoted as $x_{i..j}$ hereafter for simplicity) $(i < j)$ in a given corpus X as a phrase is defined as

$$DLG(x_{i..j} \epsilon X) = DL(X) - DL(X[r \rightarrow x_{i..j}] \oplus x_{i..j}) \tag{13.5}$$

where $X[r \rightarrow x_{i..j}]$ represents a resultant corpus from the operation of replacing all instances of $x_{i..j}$ with r throughout X, and $\oplus$ denotes a string concatenation operation with a delimiter[1] inserted in between its two operands, i.e., the current corpus X and the newly learned phrase $x_{i..j}$.

Accordingly, a best-first learning algorithm using this goodness measure is put forward as below. It reaches a lower DL when more phrases are acquired.

1. Input: X_0 (with a vocabulary V_0); set $k = 0$.
2. Examine all substrings $x_{i..j}$ $(i < j)$ in X_k,
   (a) If no more $x_{i..j}$ that $DL(X_k[r \rightarrow x_{i..j}] \oplus x_{i..j}) < DL(X_k)$, output phrases and exit;
   (b) Else $x_{i..j} = \operatorname*{arg\,max}_{x_{i..j} \epsilon X_k} DLG(x_{i..j})$.
3. $X_{k+1} = X_k[r^k \rightarrow x_{i..j}] \oplus x_{i..j}$, $V_{k+1} = V_k \cup \{r^k\}$, $k = k + 1$, goto 2.

[1]For the sake of simplicity, the sign $\oplus$ is also used to denote a delimiter.

It is worth noting that only a learned phrase, but not its index r^k, is concatenated to the updated corpus $X_k[r^k \to x_{i..j}]$. This eliminates redundant r^k's in the grammar, since the phrase that r^k represents immediately follows the k_th $\oplus$ in the corpus.

One may see that this learning algorithm may not reach the shortest description length, since it is a best-first strategy that stops at a local minima. To alleviate this problem, an optimisation process is necessary. We are developing a learning algorithm to derive an optimal bracketing for a sentence with regard to DLG criterion[2]. Beyond this, it is highly possible that the criterion can be integrated into a GA or other sophisticated searching algorithms. Our goal here is to test the effectiveness of the DLG criterion for phrase learning. Also, the above learning need to observe some natural constraints (but not brute-force ones like *distitutes* [BMMS90, BM92]) on phrases, e.g., a phrase does not cross a sentence boundary.

6 Experiments

A number of preliminary experiments on unsupervised phrase and lexical learning have been conducted on parts of the PTB corpus. These experiments show that the learning approach with the DLG measure gives promising results. It has illustrated a very good capacity to capture the regularities in the data. Since it takes a learning-via-compression approach, the learning result is a set of deterministic CFG rules.

From Appendix A, we can see that most phrases learned form a PTB-II Brown sub-corpus of POS tags are good ones. Although `[IN DT]` is also guessed to be a phrase, this notorious problem in phrase structure learning that has been handled by brute-force distitutes in previous research appears to have been alleviated, in the sense that so many instances of `IN` and `DT` appear as a proper part in other phrases. When an optimal hierarchical chunking algorithm is available and runs for some rounds, bad phrases are expected to be further filtered out, if there is another better choice of bracketing leading to a shorter description length on the sequence of words (or tags) in question. For example, `[[IN DT] CD NN]` may be changed to `[IN [DT CD NN]]`, if the latter has a shorter description length.

Lexical learning result presented in Appendix B appears more impressive. Not only words and multi-word linguistic units (e.g., place and organisation names) are properly learned, but many morphological regularities are also detected.

We have not applied the phrase learner to a corpus as large as the entire Brown corpus yet, but a lexicon learning experiment on the PTB ATIS corpus shows that it can achieve a compression rate of 1.92 bits/character, over

[2] We wish to report our ongoing work in this direction with an *optimal hierarchical chunking* algorithm in a coming paper soon.

the data *and* the grammar (i.e., the lexical items learned). This shows that our algorithm compares favourably with other researchers' work. The best result reported before in this direction is de Marcken's MDL lexicon learner [deMa95b] with a Baum-Welch algorithm to estimate probabilities. The entropy rates it achieves in lexical learning from Brown corpus are reported to be 1.92 bits/characters on training data and 2.04 bits/characters on test data, omitting the grammar (i.e., the lexical items learned) and some other overheads. Ristad and Thomas' [RT95] non-monotonic context modelling with MDL principle achieves a massage entropy of 1.97 bits/characters on Brown corpus, leaving the model apart. We expect the ongoing optimal chunking algorithm to achieve a better compression effect.

7 Concluding remarks

We have developed an information-theoretical criterion, the *description length gain*, for unsupervised learning of phrase structures from natural language corpus within the MDL learning paradigm. The preliminary learning experiments show promising results.

This learning approach has two distinct features. First, it counts the description length in bits, instead of calculating probabilities then estimating compression effect in terms of the probability of the data given the model. Counting bits and calculating probability are theoretically equivalent, but operationally different in practice. This leads to another distinct feature of our approach: the corpus as input and the grammar as learning result are encoded together as one by the same coding scheme, instead of as two separate parts encoded by two different schemes. Thus, an ideal universal compression is assumed for calculating description length. In practice, the Shannon-Fano coding can suffice for phrase learning and text compression. In this sense, our learning approach is theoretically and operationally elegant, in addition to its effectiveness.

Although experiments show encouraging results, in order to have a clearer idea on how good the DLG criterion is, we still need to have more thorough evaluations on it with more sophisticated learning algorithms, e.g., the optimal hierarchical chunking under development. The evaluations and advanced learning algorithms are the two main tasks in our future work.

Acknowledgement

Thanks to Yorick Wilks for his supervision and support, and to Ted Dunning, Rob Gaizauskas, Mark Hepple, Ming Li, Paul McKevitt, Steve Renals and Paul Vitányi for valuable discussions and helps. Thanks also go to two anonymous reviewers for their beneficial comments. It is gratefully acknowledged that the author was supported by a University of Sheffield Research Scholarship to the Institute of Language, Speech and Hearing (ILASH) for this work as part of his

Ph.D. research. This paper is based on the Research Memorandum CS-96-08, Department of Computer Science, University of Sheffield.

Bibliography

[Baker79] J. Baker. Trainable grammars for speech recognition. In J. J. Wolf and D. H. Klatt (eds.), *Speech Communication Papers Presented at the 97th Meeting of the Acoustical Society of America*, MIT Press, Cambridge, MA, 1979

[BMMS90] E. Brill, D. Magerman, M. Marcus, and B. Santorini. Deducing linguistic structure from the statistics of large corpora. In *Proceedings of the DARPA Speech and Natural Language Workshop*, 275-282, 1990

[BM92] E. Brill, and M. Marcus. Automatically acquiring phrase structure using distributional analysis. In *Proceedings of 1992 DARPA Speech and Language Workshop*, Harriman, N.Y., 1992

[CC92a] G. Carroll, and E. Charniak. Learning probabilistic dependency grammars from labelled text. In *AAAI-92*, 1992

[CC92b] G. Carroll, and E. Charniak. *Two Experiments on Learning Probabilistic Dependency Grammars from Corpora*. Technical Report CS-92-16, Department of Computer Science, Brown University, 1992

[CB94a] T. A. Cartwright, and M. R. Brent. Segmenting speech without a lexicon: Evidence for a bootstrapping model of lexical acquisition. *Proceedings of the 16th Annual Conference of the Cognitive Science Society*, 148-152, Erlbaum, Hillsdale, N.J., 1994

[CB94b] T. A. Cartwright, and M. R. Brent. Segmenting speech without a lexicon: The roles of phonotactics and speech source. In *First Meeting of the ACL Special Interest Group in Computational Phonology*, 83-90, Association for Computational Linguistics, 1994

[Chai66] Gregory Chaitin. On the length of programs for computing finite binary sequences. *JACM*, **13**:547-569, 1966

[Chen95] S. F. Chen. Bayesian grammar induction for language modelling. In *ACL-95*, 228-235, Cambridge, MA, June, 1995

[Chen96] S. F. Chen. *Building Probabilistic Models for Natural Language*. PhD thesis, Harvard University, Cambridge, MA, 1996

[Chom55] N. A. Chomsky. *The Logical Structure of Linguistic Theory*. Plenum Press, New York, 1955

[Chom57] N. A. Chomsky. *Syntactic Structure*. Mouton, Hague,1957

[CRA76] C. M. Cook, A. Rosenfeld, and A. R. Aronson. Grammatical inference by hill climbing. *Information Science*, **10**:59-80, 1976

[CT91] T. M. Cover, and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., New York, 1991

[deMa95a] C. de Marken. Lexical heads, phrase structure and the induction of grammar. In D. Yarowsky and K. Church (eds.), *Third Workshop on Very Large Corpora*, 14-26, Cambridge, MA, June, 1995 [deMa95b] C. de Marken. *The Unsupervised Acquisition of a Lexicon from Continuous Speech*. Technical Report A.I. Memo No. 1558, AI Lab., MIT, Cambridge, MA, November, 1995

[deMa96] C. de Marken. Linguistic structure as composition and perturbation. In *ACL-96*, 335-341, Santa Cruz, CA, 1996

[FK82] W. N. Francis, and H. Kucera. *Frequency Analysis of English Usage: Lexical and Grammar*. Houghton-Mifflin, Boston, 1982

[Hol75] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975

[Huf52] D. A. Huffman. A method for construction of minimum-redundancy codes. In *Proceedings IRE*, **40**:1098-1101, 1952

[Jel85] F. Jelinek. Self-organized language modeling for speech recognition. IBM T.J. Watson Research Center, Continuous Speech Recognition Group, Yorktown Heights, NY, 1985. Also in A. Waibel and K. F. Lee (eds.), *Readings in Speech Recognition*, 450-506, Morgan Kaufmann, San Mateo, CA, 1990

[KGV83] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimisation by stimulated annealing. *Science*, **200**:671-680, 1983

[Kit95] C. Kit. *Speeding up the Virtual Corpus approach to deriving and retrieving n-grams for any n from large-scale corpora*. Manuscript, Department of Computer Science, University of Sheffield, August, 1995

[Kit96] C. Kit. *Current Progress in Learning Phrase Structure with the Minimum Description Length Principle*. Technical Report CS-96-08, Department of Computer Science, University of Sheffield, August, 1996

[Kol65] A. N. Kolmogorov. Three approaches for defining the concept of 'information quantity'. *Problem of Information Transmission*, **1**:4-7, 1965

[LY90] K. Lari, and S. J. Young. The estimation of stochastic context-free grammars using the Inside-Outside algorithm. *Computer Speech and Language*, **4**:35-56, 1990

[LV93] M. Li, and P. M. B. Vitányi. *Introduction to Kolmogorov Complexity and its Applications*. Springer-Verlag, New York, 1993

[MM90] U. Manber, and E. Myers. Suffix array: a new method for on-line string searches. *First ASM-SIAM Symposium on Discrete Algorithms*, 319-327, American Mathematical Society, Providence, 1990

[MSM93] M. Marcus, B. Santorini, and M. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, **19**(2):313-330, 1993

[NM94] M. Nagao, and S. Mori. A new method of n-gram statistics for large number of n and automatic extraction of words and phrases from large text data of Japanese. *COLING-94*, 611-615, 1994

[QR89] J. R. Quinlan, and R. L. Riverst. Inferring decision tree using the Minimum Description Length principle. *Information and Control*, **80**:227-248, 1989

[Ris78] J. Rissanen. Modeling by shortest data description. *Automatica*, **14**:465-471, 1978

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

[RR94] J. Rissanen, and E. S. Ristad. Language acquisition in the MDL framework. In E. Ristad (ed.), *Language Computations*, American Mathematical Society, Philadelphia, PA, 1994

[RT95] E. S. Ristad, and R. G. Thomas. New techniques for context modelling. In *ACL-95*, Cambridge, Massachusetts, 1995

[Shan48] C. Shannon. *A mathematical theory of communication. Bell System Technical Journal*, **27**:379-423, 623-656, 1948

[Solo59] R. J. Solomonoff. A new method for discovering the grammars of phrase structure languages. In *Information Processing*, 258-290, 1959

[Solo60] R. J. Solomonoff. The mechanization of linguistic learning. In *Proceedings of the 2nd International Conference on Cybernetics*, 180-193, 1960,

[Solo64] R. J. Solomonoff. A formal theory of inductive inference, part 1 and 2. *Information Control*, **7**:1-22, 224-256, 1964

[Sto94] A. Stolcke. *Bayesian Learning of Probabilistic Language Models*. PhD thesis, University of California at Berkeley, Berkeley, CA, 1994

[Wol80] Gerard J. Wolff, Language acquisition and the discovery of phrase structure. *Language and Speech*, **23**:255-269, 1980

[Wol82] G. J. Wolff. Language acquisition, data compression and generalisation. *Language and Communication*, **2**(1):57-89, 1982

[Zipf35] G. K. Zipf. *The Psycho-Biology of Language*. Houghton Mifflin, 1935

[Zipf49] G. K. Zipf. *Human Behaviour and the Principle of Least Effort*. Hafner, New York, 1949

Appendix A

The first 50 grammar rules learned from a tagged PTB-II Brown sub-corpus of 101,723 tags are listed below. Phrases marked with `y`, `x` and `?` are correct, false and questionable ones, respectively. All noun sub-classes in PTB are merged into NN. Other PTB POS tags involved are listed below. The number preceded by `curDL` in each line is the current combined description length of data and model before the phrase is acquired. It is observed that when all phrases with positive DLG are learned, the algorithm shortens the combined description length by about 13% beyond Shannon-Fano coding.

```
curDL: 386245, R0: y[ IN DT NN ]
curDL: 379371, R1: y[ DT JJ NN ]
curDL: 374673, R2: y[ TO VB ]
curDL: 370096, R3: y[ MD VB ]
curDL: 364915, R4: y[ DT NN ]
curDL: 362917, R5: y[ MD RB VB ]
curDL: 361290, R6: ?[ PRP VBD ]
```

```
 curDL: 359755, R7: y[ $ CD ]
 curDL: 358687, R8: ?[ PRP VBP ]
 curDL: 357785, R9: y[ DT JJ JJ NN ]
 curDL: 356937, R10: y[ IN R1[DT JJ NN ] ]
 curDL: 356103, R11: y[ JJ NN ]
 curDL: 355341, R12: y[ DT JJ ]
 curDL: 354628, R13: y[ IN PRP$ NN ]
 curDL: 353926, R14: ?[ PRP VBZ ]
 curDL: 353330, R15: ?[ PRP R3[MD VB ] ]
 curDL: 352674, R16: y[ IN CD ]
 curDL: 352190, R17: y[ IN NN ]
 curDL: 351425, R18: y[ IN R11[JJ NN ] ]
 curDL: 350602, R19: x[ IN DT ]
 curDL: 349674, R20: y[ IN PRP$ R11[JJ NN ] ]
 curDL: 349044, R21: ?[ IN VBG ]
 curDL: 348405, R22: y[ IN R12[DT JJ ] ]
 curDL: 347959, R23: ?[ POS NN ]
 curDL: 347556, R24: ?[ VBZ VBN ]
 curDL: 347156, R25: ?[ VBD VBN ]
 curDL: 346691, R26: ?[ R3[MD VB ] VBN ]
 curDL: 346292, R27: ?[ NN CC NN NN ]
 curDL: 345922, R28: y[ TO R4[DT NN ] ]
 curDL: 345562, R29: y[ PRP$ NN ]
 curDL: 345209, R30: y[ PRP$ R11[JJ NN ] ]
 curDL: 344855, R31: ?[ POS R11[JJ NN ] ]
 curDL: 344498, R32: x[ R19[IN DT ] CD NN ]
 curDL: 344163, R33: ?[ VBP VBN ]
 curDL: 343893, R34: y[ IN R9[DT JJ JJ NN ] ]
 curDL: 343650, R35: ?[ PRP R5[MD RB VB ] ]
 curDL: 343426, R36: ?[ PRP RB VBD ]
 curDL: 343212, R37: y[ IN R7[$ CD ] ]
 curDL: 343001, R38: x[ IN PRP$ ]
 curDL: 342788, R39: y[ IN PRP ]
 curDL: 342574, R40: y[ IN R6[PRP VBD ] ]
 curDL: 342343, R41: x[ IN JJ ]
 curDL: 342134, R42: ?[ WP VBD ]
 curDL: 341940, R43: y[ DT CD NN ]
 curDL: 341758, R44: y[ RB VBN ]
 curDL: 341579, R45: y[ DT JJS ]
 curDL: 341403, R46: y[ RB JJ ]
 curDL: 341232, R47: y[ RB VB ]
 curDL: 341051, R48: ?[ FW FW ]
 curDL: 340888, R49: y[ JJ R11[JJ NN ] ]
 curDL: 340740, R50: x[ CC R11[JJ NN ] ]
```

```
CC  - Coordinating conjunction           CD  - Cardinal number
DT  - Determiner                         IN  - Preposition or subordinating conjunction
JJ  - Adjective                          JJS - Adjective, superlative
MD  - Modal                              POS - Possessive ending
PRP - Personal pronoun                   PRP$ - possesive personal pronoun
RB  - Adverb                             RBR - Adverb, comparative
TO  - 'to'                               VB  - Verb, base form
VBD - Verb, past tense                   VBG - Verb, gerund or present participle
VBN - Verb, past participle              VBP - Verb, non-3rd person singular present
```

```
VBZ - Verb, 3rd person singular present   WDT - Wh-determiner
WP  - Wh-pronoun                          $   - Dollar sign
```

Appendix B

Experimental results of lexicon learning from a 2K-character fragment of the PTB WSJ corpus. The spaces are replaced by underscore for visualisation. It is observed the algorithm further shortens the combined description length by 15% beyond Shannon-Fano coding.

```
curDL: 8252, R0: [ _ c o a l i t i o n _ g o v e r n m e n t ]
curDL: 8093, R1: [ _ t h e _ k h m e r _ r o u g e _ ]
curDL: 8010, R2: [ _ n o n - c o m m u n i s t ]
curDL: 7958, R3: [ _ t h e _ s t a t e _ d e p a r t m e n t _ ]
curDL: 7807, R4: [ _ c o m m u n i s t ]
curDL: 7670, R5: [ _ c a m b o d i a ]
curDL: 7623, R6: [ _ c o a l i t i o n _ ]
curDL: 7575, R7: [ _ i s _ t h a t _ t h e ]
curDL: 7549, R8: [ _ k h m e r _ r o u g e _ ]
curDL: 7528, R9: [ _ s a n d i n i s t a s _ ]
curDL: 7498, R10: [ _ v i e t n a m ]
curDL: 7478, R11: [ e _ g u e r r i l l a _ ]
curDL: 7462, R12: [ _ n i c a r a g u a _ ]
curDL: 7350, R13: [ _ t h e _ ]
curDL: 7314, R14: [ o u l d _ ]
curDL: 7301, R15: [ c o m m u n i s t ]
curDL: 7255, R16: [ _ t h a t ]
curDL: 7243, R17: [ m o d e r a t e s _ ]
curDL: 7188, R18: [ _ t h e ]
curDL: 7167, R19: [ _ w i t h ]
curDL: 7159, R20: [ _ p a r t n e r s ]
curDL: 7150, R21: [ n _ a m e r i c a ]
curDL: 7143, R22: [ _ i n t e r i m _ ]
curDL: 7124, R23: [ _ a n d ]
curDL: 7097, R24: [ _ o f ]
curDL: 7085, R25: [ t i o n _ ]
curDL: 7060, R26: [ _ i n _ ]
curDL: 7052, R27: [ _ f o r _ ]
curDL: 7031, R28: [ i n g ]
curDL: 7026, R29: [ g r o u p s _ ]
curDL: 7021, R30: [ _ r e g i m e _ ]
curDL: 7012, R31: [ p a r t ]
curDL: 7008, R32: [ c u r r e n t ]
curDL: 7005, R33: [ i n c l u d ]
curDL: 7002, R34: [ g o v e r n ]
curDL: 6999, R35: [ w e r e _ ]
curDL: 6994, R36: [ a l l ]
curDL: 6991, R37: [ _ i n  R5 ]
curDL: 6987, R38: [ _ t o _ ]
curDL: 6986, R39: [ g e s t ]
curDL: 6982, R40: [ r e ]
curDL: 6976, R41: [ e r ]
curDL: 6974, R42: [ o u ]
```