# Constructing semantic representations using the MDL principle


*Niels Fertig*
*Institut für Physik*
*LMU München*
*D 80797 München*
*Niels.Fertig@physik.uni-muenchen.de*

*Gabriele Scheler*
*Institut für Informatik*
*TU München*
*D-80290 München*
*scheler@informatik.tu-muenchen.de*



**Abstract**

Words receive a significant part of their meaning from use in communicative settings. The formal mechanisms of lexical acquisition, as they apply to rich situational settings, may also be studied in the limited case of corpora of written texts. This work constitutes an approach to deriving semantic representations for lexemes using techniques from statistical induction. In particular, a number of variations on the MDL principle were applied to selected sample sets and their influence on emerging theories of word meaning explored. We found that by changing the definition of description length for data and theory - which is equivalent to different encodings of data and theory - we may customize the emerging theory, augmenting and altering frequency effects. Also the influence of stochastic properties of the data on the size of the theory has been demonstrated. The results consist in a set of distributional properties of lexemes, which reflect cognitive distinctions in the meaning of words.


## 1 Introduction: The MDL principle

Minimum description length (MDL) is an inductive learning principle, that is explicitly based upon Solomonoffs ideas about formalizing induction ([6, 4]). The MDL principle states that the best theory to explain a set of data, is the one that minimizes the sum of a) the description length of the theory b) the description length of the data when encoded with the help of the theory.

This idea is intuitively appealing: to find a good theory, make things as simple as possible, while still modeling the data correctly. One can think of the description length of data + theory as a measurement of simplicity/complexity of the problem at hand, therefore the terms Kolmogorov Complexity or Algorithmic Complexity are commonly used in this context ([3]).

By building a theory MDL creates a reduced representation of the given data, where only the information of the absolute position of items in the data is lost, frequency and relative position with respect to other items is retained within the theory.

The original application of Solomonoff's idea has been to the extraction of grammatical rules (cf. also [1]). The aim of this work is to apply the MDL principle to derive semantic properties of individual words, given a set of example sentences. At the present time the approach is limited to artificially constructed data sets in order to explore the properties of different algorithms in a controlled way. However, the same approach may ultimately be applied to real text corpora, and more realistic semantic representations may be constructed.

## 2 Induction of semantic representations

### 2.1 Building distributive representations

A set of data,e.g. a text, is presented to the algorithm, which then builds abbreviations, so-called *signatures*, for re-occurring pieces of data. It is assumed that the distributive properties of lexemes may be described as a set of such signatures and that these signatures may be regarded as semantic features, i.e. the capture important aspects of the meaning of the lexemes (cf. [5] for a critical appraisal of this approach). However, the different contexts of a lexeme may be of varying importance. One approach consists in finding those features in a context which, when applied to the representation of the lexeme, leads to a maximally compressed version of the text.

Under this view, a lexeme is expanded into a set of signatures(=semes), which again expand into a set of contextual distributions, as the "theory" of the language. This allows to store a specific set of data efficiently if it complies with the theory. The task of the learning algorithm then is to construct such a theory.

### 2.2 Computing the description length

The decisive part of the learning algorithm is to compute the optimality criterion, i.e. the description length (DL). The overall DL is computed as a sum of the DL of the data ($DL_{data}$) and the DL of the theory ($DL_{theory}$), which establishes a basic trade-off of compressing the text and increasing the complexity of the theory. A very simple instruction for the DL would be:

$$DL_{data} = N$$

$$DL_{theory} = S \times x$$

$$DL = DL_{data} + DL_{theory}$$

where $S$ is the number of signatures stored, $N$ is the number of words in a text and $x$ is a factor to determine the cost of storing a signature.

Accordingly, during the process of learning, the DL of the text decreases, as word combinations are replaced by signatures, and the DL of the theory increases, as new signatures (of variable length) are added to it. This process is repeated as long as an overall decrease of DL can be achieved. The store of signatures designed makes up the theory of the text.

In subsequent experiments, we augmented the definition of the DL to achieve desired effects in the constructed theory. Speficically, we changed the data 'length' by giving different weights to different words, and we changed the theory length definition by making certain types of signatures longer than others. In this way, we can show that the general principle of minimizing description length is really a versatile tool in effecting statistical induction over texts.

### 2.3 Optimization

The choice of an optimization or learning algorithm becomes crucial, when we have to make local decisions without being able to compute the global gain immediately. In our case this means a choice as to which signature is to be built from a competing set of alternatives. There are a number of solutions for this problem which has become known as the problem of local minima: Adding noise to a greedy algorithm (which always chooses the local best alternative), keeping an exact (or inexact, compressed) log on decisions taken to be able to "backtrack" to an earlier point when performance does not improve or using a look-ahead algorithm which would compute the gain in DL more than one signature ahead. In the following experiments, we have used the simple greedy learning approach, i.e. always selecting the local best choice.

## 3 A first experiment

### 3.1 The data and the algorithm

A first experiment on computing description lengths and building a store of signatures was performed to implement the basic principles and to explore the stochastic properties of the artificially constructed example set.

The example sets were constructed in the following way: A set of 7 prepositions and 10 nouns was used to build sequences of three words of the form noun-prep-noun, such as *tree at house, bird in cage* etc. The rationale for this specific type of example is to explore the semantic distinctions imposed on lexemes by prepositional use and by other lexemes in close proximity. Data sets of various size (150/600/4500 words) were created. Also, the distributive properties of the lexemes have been varied to explore the influences of the algorithm. Specifically, we used data sets with an equal statistical distribution of the prepositions and data sets where a single preposition accounted for 50 % of the examples in a data set.

In our first approach the number of words in the text was used as $DL_{data}$, and the $DL_{theory}$ was the number of signatures built * $x$. Accordingly, when a two-word combination occurs $n$ times in the text, the abbreviation by a signature decreases the DL of the data by $n$. The DL of the theory is increased by $x$ for storing the signature. Therefore a signature needs to occur at least $x+1$ times in the text to effect a decrease of the overall DL.

The DL of data + theory was computed after each step with or without a new signature and if the DL was less with the new signature, the abbreviation was performed and stored.

When the text is structured into noun-prep-noun patterns and combinations of two words are to be abbreviated, one can build noun-prep or prep-noun combinations. Since these two options compete in constructing a signature, in principle it is possible to use a bias term favoring one type of signature over the other. This use of bias terms in manipulating the "length" of signatures is exemplified in the second set of experiments.

### 3.2 Results

Two different ways to select word combinations were applied. In the first case, word combinations were picked randomly, and any combinations that effected a decrease of overall DL were used. (For x=2, these are all the combinations which occur at least three times in the text.) In the second case, word combinations with a high decrease of DL were picked first. In the current implementation, a high decrease of DL means a high mutual information value of a word combination. Accordingly, the mutual information value of all the word combinations was computed using the most frequent word to build combinations and then the most frequent of those combinations was picked. As long as the DL criterion was met with a word combination, the signature was built and the words in the remaining abbreviated text were counted again. The results are shown in Fig. 1 and Fig. 2. We can see that the reduction is faster and more effective in the case of frequency-based selection.

The algorithm was also applied to datasets with different stochastic properties. In particular, the

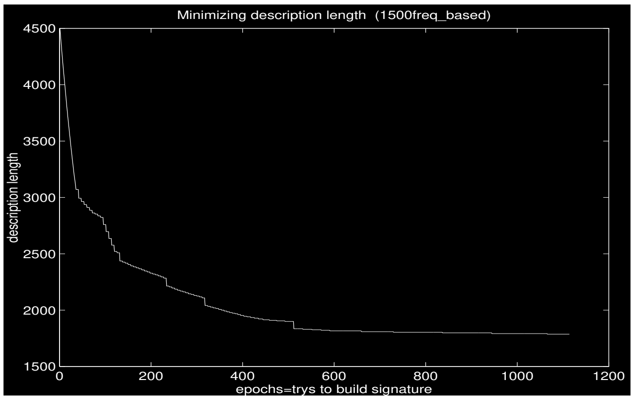


Figure 1: Reduction of $DL_{data}$ with frequency-based selection of word combinations

frequency of a single preposition was varied, while all other lexemes occurred with approximately equal probability. The results are shown in Fig. 3.

With a high frequency of a lexeme, the overall DL is decreased, which is due to the decrease of the DL of the data. For the long text (4500 words), the algorithm is able to reduce the DL down to less than 40% of the original DL. This relates to the increase of mutual information in the text, when certain words are used with high frequency. So the decrease of DL during the learning process is coupled to stochastic properties of the text.

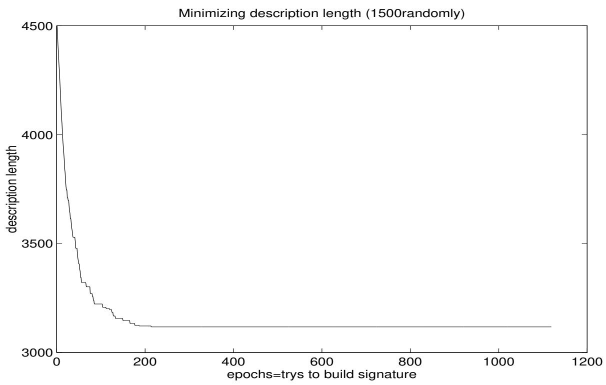


Figure 2: Reduction of $DL_{data}$ with random selection of word combinations

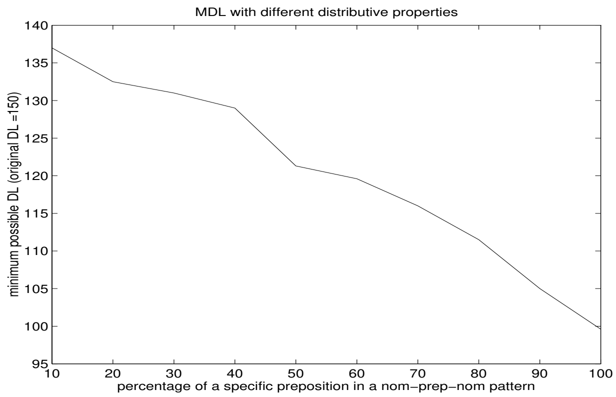


Figure 3: Reduction of DL of the data with respect to its stochastic properties. Nouns and 6 prepositions occurred with equal probability and the frequency of a single preposition was varied. (Equal probablity would be 14%).

# 4 Customizing the description length

For the second set of experiments, we selected a different, more complex data set, extracted from a text currently used in our group[1], and we changed the definition of description length in various ways to explore the influence on the emerging theory. We used a small set of examples (134) which followed the pattern [subj verb obj prep pnoun]. It must be borne in mind that these examples were only constructed for illustration of the properties and the possibilities of an MDL algorithm, no linguistically useful results are expected from them. Still it will be shown that the MDL principle is well adapted to extract the desired information by adapting the optimality criterion (mimimal length) to the problem at hand.

## 4.1 Data description

In the first set of experiments the definition of theory length was left unaltered, i.e. we built signatures on the basis of frequency of co-occurrence. However, the basic definition of $DL_{data}$ allows just one feature for each word - whereas we want to find a number of semantic descriptions based on co-occurence data for every word.

[1] "The Ghostway", a novel by T. Hillerman

| | | |
|---|---|---|
| sprinkle | he sprinkle | subject-verb |
| sprinkle | sprinkle on | verb-prep |
| take | he take | subject-verb |
| take | take from | verb-prep |
| take | take pocket | verb-pnoun |
| put | he put | subject-verb |
| put | put in | verb-prep |
| put | put on | verb-prep |
| put | put pocket | verb-pnoun |

Table 1: Contextual features for selected verbs based on frequency

We may overcome this limitation by changing $DL_{data}$ in the following way:

$$DL_{data} = \sum_{i=1}^{k} N_i \times \theta_i$$

where there are $k$ different types of words recognized, $N_i$ denotes the number of words of type $i$ found in the text and $\theta_i$ is a weight factor for different word types.

In this way, rather than simply counting the words in the text, each word receives a different numeric value ("weight") determining how many features it may be assigned. The weight is decreased by a constant for each signature formed and the weights of the words are counted for the description length. This may be seen as a primitive encoding step performed on the data - in this case determining the importance or relevance of a word with respect to the semantic representation. A useful assignment of word types would be to use part of speech classes, i.e. a different assignment for nouns, verbs, adjectives and prepositions, and leaving certain types of function words without any semantic representation (determiners, conjunctions etc.). In the current implementation, we used slots rather than word classes to determine the weight of each word.

$$DL_{data} = N_{subj,obj,verb,pnoun} \times 5 + N_{prep} \times 3$$

For each lexeme, we get distributive representations consisting of the type of syntactic relation (e.g. verb-obj) and the lexical item that fills that relation. For example, for verbs such as *sprinkle, take, put*, we get contextual feature representations as in Table 1. We may use this mechanism to look for similar profiles of words. In this case, if we look at the first 5 semes for *pocket* and *pouch*, we find a correspondence there:

| | | | | | |
|---|---|---|---|---|---|
| pocket: | in | prep-pnoun | pouch: | in | prep-pnoun |
| | he | subj-pnoun | | they | subj-pnoun |
| | put | verb-pnoun | | he | subj-pnoun |
| | they | subj-pnoun | | put | verb-pnoun |
| | stone | object-pnoun | | knife | object-pnoun |

We may rate the similarity of lexical items on the basis of these data.

Assuming that the application of this mechanism to a large data set would yield useful lexical groups such as *twig, stick, tube, sac, knife, blade, sheet, stone* (manipulable physical objects) or *place, replace, stick, pull, slip, put* (verbs of placement) we may replace these lexemes by a generic item to be used in further frequency counts. I.e. initially we collect data on individual word-word co-occurrences, but in subsequent applications of the MDL principle we may use the established signatures (the 'theory') to collect further co-occurrence frequencies. This is very much in line with other work using the MDL principle ([1, 2]), where the repeated application of the algorithm leads to the hierarchical feature sets used in syntactic theory or thesaurus building.

| | | |
|---|---|---|
| generic-put | put on<br>put in<br>put stick<br>put sheet<br>put pouch | verb-prep<br>verb-prep<br>verb-obj<br>verb-obj<br>verb-pnoun |
| generic-stone | put stone<br>stone in<br>stone from<br>stone with<br>stone pocket | verb-obj<br>obj-prep<br>obj-prep<br>obj-prep<br>obj-pnoun |

Table 2: Contextual features for word groups: similar words are replaced by a generic term

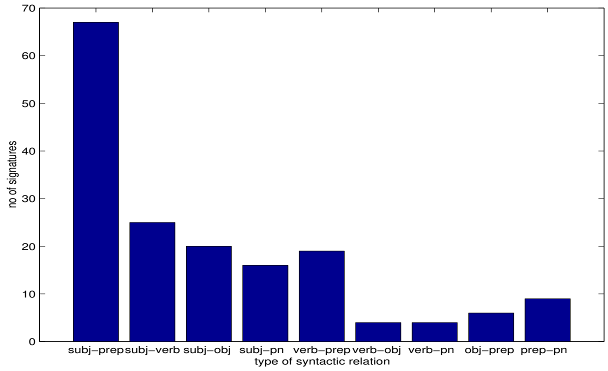


Figure 4: Distribution of syntactic relations relative to number of signatures formed

Applied to this example set we would get results as in Table 2. We can see that common properties of replacement verbs (i.e. use with the prepositions *in* and *on*, with names for manipulable objects as direct objects and with containers in prepositional phrases ) and manipulable objects (use with replacement verbs, with a *in, from* and *with* and with containers in prepositional phrases) are captured even with this small (but uniform) dataset.

## 4.2 Manipulating the length of the theory

So far, we have operated with the basic definition of the DL for the theory as the number of signatures stored. We already pointed to the possibility of using "shorter" and "longer" signatures for pieces of context by using shorter signatures for certain low frequency data and longer signatures for high frequency data. In this way we can balance the effect of frequency data in a text, by biasing for certain types of signatures. For instance, in our text we find that some syntactic relations are much more common in the constructed signatures than others (cf. Fig. 4). In particular, the subject terms are used most frequently in defining semantic representations: this is due to a high proportion of re-occurring words in this slot (pronouns). We may introduce a bias term for subject involvement and thus "penalize" signatures which contain a subject term. Although this may seem like an ad-hoc method here, such a bias term for the type of syntactic relation is actually quite useful. For instance, we may want to concentrate on verb-object relations at the expense of subject-verb relations, or adjective-noun relations at the expense of verb-noun relations.

We may also want to combine different kinds of bias terms for signatures.

| | | | | | |
|---|---|---|---|---|---|
| put | put on | verb-prep | put | put on | verb-prep |
| he | he on | subj-prep | cut | cut with | verb-prep |
| he | he with | subj-prep | put | put blade | verb-obj |
| cut | cut with | verb-prep | stick | stick in | obj-prep |
| he | he put | subj-verb | he | he put | subj-verb |
| he | he in | subj-prep | put | put stick | verb-obj |
| they | they be | subj-verb | be | be face | verb-obj |
| it | it put | subj-verb | face | face with | obj-prep |
| put | put blade | verb-obj | blade | blade on | obj-prep |
| stick | stick in | obj-prep | tube | tube with | obj-prep |
| be | be by | verb-prep | rabbit | rabbit with | obj-prep |
| be | be with | verb-prep | of | of valley | prep-pnoun |
| they | they with | subj-prep | in | in pocket | prep-pnoun |
| it | it with | subj-prep | be | be by | verb-prep |
| he | he cut | subj-verb | be | be with | verb-prep |

Table 3: 15 word combinations with highest $\Delta$ DL for $\delta = 0$, $\sigma = 0$ (left) and $\delta = 1 \ldots 4, \sigma = 2$ (right)

In this way we can explore the combined influence of several factors on the emerging theory by changing the values of each of them independent of the other. Here we have introduced a term for the distance of words in the pattern, i.e. favoring adjacent word patterns over long-distance patterns.

The resulting definition looks like this:

$$DL = DL_{data} + \sum_{i}^{k} \sigma_i \times S_i + \sum_{j}^{m} \delta_j \times S_j$$

where $\delta$ is the term for distance and $\sigma$ for subject involvement, and $S$ the cost of storing a signature.

Thus we can untie the likelihood of forming a signature (entering into the semantic description) from the frequency of a word combination. For instance, the list of the 15 signatures with the highest gain in DL for the original version (i.e. based only on frequency) is significantly altered if we set $\delta$ to 1-4, depending on distance, and $\sigma$ to 2 if a subject is present in the relation, as shown in Table 3.

We see that with the change of parameter settings, certain word combinations are ranked higher that they would be on the basis of freqeuncy, other word combinations are raked lower or disappear to make room for further novel word combinations. For instance, with penalizing the subject relation, new verb-object relations appear, and with penalizing distance, subject-preposition relations (which are among the most frequent) disappear. The combination of different parameters allows for a subtle balancing of the emergent results.

We can expect with a larger dataset to be able to extract a set of distributional relations between words and use them to cluster words into higher-level semantic classes. However, with that goal in mind, a specific local learning method needs to be implemented, rather than the exhaustive search that was used in these experiments.

## 5 Conclusion

In the preceding sections, we have shown how we can apply the principle of a compression algorithm which optimizes for length of theory abbreviating the data to the task of building semantic representations from texts. We have concentrated on the definition of the optimality criterion itself, namely the description length of data and theory, and shown various possibilities of customizing this definition to fit the needs of a particular problem. We have pointed to the possibility of repeated application of the optimality criterion for the creation of feature hierarchies without using this option in the reported experiments. In order to apply such methods to a large text ( $>$ 10000 words), we need to pay attention to the learning method, i.e. a method of selecting an action given the local gain for that action. We have shortly referred to various methods of accomplishing that aim. Although all the reported experiments were conducted on artificial data, or small data sets, a number of interesting observations could be made. In particular, the influence of the stochastic distribution of words in the data on compression rates and on the constructed theory could be shown. Furthermore, it was demonstrated how simple changes in the definition of the description length have a significant impact on the emerging theory, and how in-built biases can override frequency effects in the data.

**Acknowledgments:** The project was funded by the DFG under contract no. Br 609/7-1.

## References

[1] P. Gruenwald. Automatic grammar induction using the MDL principle. Master's thesis, University of Amsterdam, 1994.

[2] Hang Li and Naoki Abe. Clustering words with the MDL principle. In *Proceedings of COLING-96*, 1996. 1C.

[3] Ming Li and Paul Vitanyi. *An Introduction to Kolmogorov Complexity and its Applications.* Graduate Texts in Computer Science. Springer, 2nd edition, 1997.

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

[5] Gabriele Scheler. LexematischeÄquivalenz in der maschinellen Übersetzung. Technical Report FKI-219-96, Institut für Informatik, TU München, 1996.

[6] Ray Solomonoff. A formal theory of inductive inference. part I and II. *Information and Control*, 7:1–22, 224–254, 1964.