Computer Assisted Mechanics and Engineering Sciences, 14: 737–743, 2007.

Associative learning of concepts

Viliam Rockai
Technical University of Košice, Department of Cybernetics and Artificial Intelligence
Letna 9/B, 042 00 Košice, Slovak Republic

Robert Kende
IMC, Strawinskylaan 377, 1088 XX, Amsterdam, The Netherlands

(Received in the final form September 18, 2007)

Humans find it extremely easy to say if two words are related or if one word is more related to a given word than another one. For example, if we come across two words — 'car' and 'bicycle', we know they are related since both are means of transport. Also, we easily observe that 'bicycle' is more related to 'car' than 'fork' is. In the paper we describe our approach on quantifying the semantic relatedness of concepts based on the theory of associative learning of concepts.

Keywords: semantic surrounding, associative learning, concept similarity, grammar, relatedness

1. Introduction

Ontology building is becoming a very popular area of research. Typically the existing methods and focus is on a semi-automatic ontology building based on natural language processing techniques. These methods typically try to fill an existing ontological structure with instances found in texts. They typically need a pre-tagged text corpora as input and are not trying to "learn" an ontological structure from scratch. We propose a method for inducing concept hierarchies from raw text corpora achieved by a statistical learning algorithm. Our experiments indicate, that the method is capable of effectively measuring semantical similarities between token pairs allowing to take a first step toward building a concept hierarchy (a simple ontological structure). Building a concept hierarchy in our first experiments means the clustering of word tokens into morphological or close semantical sets allowing to create word lexicons from texts in natural languages.

Our approach is based on the neuro-physiological model of thalamo-cortical information processing by R. Hecht-Nielsen [3] thus bringing a biological plausibility in the approach. The model assumes the existence of a fixed lexicon of "symbols" in the human brain thalamus, which is created during an early period of the growth of the individual. Learning occurs by establishing associations between the symbols mimicking neural connections between cortical regions. The dynamics of learning in the network is modelled by a Hebbian learning method [2]. The associative learning of concepts is an unsupervised learning processed over the stream of symbols. The goal of the process is to learn a knowledge representation structure resembling semantic networks or concept hierarchies. The method is based on the induction of associations between symbols observing their co-occurrences in a context window which the learning is processed through. The condition necessary to create an association between a particular token pair is given by their statistically "non-random" occurrences. After it has been determined that an association exists and therefore has to be created, only its weight is altered during the learning process [6].

2. ASSOCIATIVE LEARNING OF CONCEPTS

2.1. Representation

Based on the thalamo-cortical learning theory as described in [3], our implementation [6] works with tokens that are given by a set of excited neurons in some cortical region. This token represents an invariant sensory input and can be seen as an attribute of the perceived object. In the theory Symbols are defined as the representation of these attributes. A concept is defined as a symbol with its associated neighborhood. Thus, the same concept can be activated by different sets of symbols. Associations anchor the symbols to each other. The establishment of an association between two symbols depends on whether the co-occurrences are significant or not. The significance can be determined by utilizing the following formula,

$$S(i,j) = \frac{p(i,j)}{p(i)\cdot p(j)}, \tag{1}$$

where i and j are independent. $S(i,j)$ stands for mutual significance and is based on the information theoretic formula of the mutual information. In our case, using natural language corpus as a source, we can understand $p(i)$ and $p(j)$ as probability of seeing words i or j in the text. $p(i,j)$ then stands for probability of seeing these two words together. The mutual significance is defined as the ratio of mutual probability of symbols (discrete variables) i, j to their prior probabilities, where i and j are independent,

$$p(i,j) = p(i)\cdot p(j) \tag{2}$$

If two symbols coexist in joint context randomly, it will be represented by

$$S(i,j) = \frac{p(i,j)}{p(i)\cdot p(j)} < 1. \tag{3}$$

If $S(i,j) > 1$, the tokens will be considered as associated tokens. This will be the main parameter during the learning process. We call this parameter the threshold and it can be altered for filtering more non-random occurrences. The weight of an association of two symbols i and j is given by the formula

$$w(i,j) = \frac{p(i,j)}{p(j)}. \tag{4}$$

The weight of an association is the only knowledge we learn in the process. These weights are represented in fascicles — our knowledge bases $F_x = w(i,j)$ matrices, where x stands for the contextual distance. We learn several of these F_x knowledge bases each standing for different context distances (features) of the two symbols learned [6].

In natural language corpus we understand the contextual distance as the count of words between two tokens. For example in the sentence "All work and no play makes Jack a dull boy" let's take the "play" token as reference token — j. The context distances for all other tokens in the sentence are then these (written in parenthesis):

All (-4) work (-3) and (-2) no (-1) play (0) makes (1) Jack (2) a (3) dull (4) boy (5).

In this paper the learning process — storing the weights was processed only on four straight (positive) and four inverse (negative) contextual distances. Thus we stored the weight information for fascicles $F_{-4}, F_{-3}, F_{-2}, F_{-1}$ and fascicles F_1, F_2, F_3, F_4.

The calculation of weights $w(i,j)$ is given by an approximation of the above given formulas,

$$S(i,j) = \frac{\frac{C(i,j)}{T}}{\frac{C(i)}{T}\cdot\frac{C(j)}{T}}, \tag{5}$$

$$w(i,j) = \frac{\frac{C(i,j)}{T}}{\frac{C(j)}{T}}. \tag{6}$$

where $C(i, j)$ is joint occurrence of tokens i and j and T is the total token count incremented during the learning process. Division of these variables defines the probabilities from Eqs. (3) and (4). In this paper we don't use the weight value to simplify the presentation. The weights here only indicate that the association was established because the significance was greater than the threshold value.

2.2. Semantic relatedness

It is observable [3] that humans find it extremely easy to say if two words are related, For example if we come across two words — "car" and "bicycle" we know they are related as both are means of transport. Also, we easily observe that one word is more related to a given word than another word — "bicycle" is more related to "car" than "fork" is. But is there some way to assign a quantitative value to this relatedness? By utilising the knowledge acquired by learning the fascicles, we constructed the following measure for R — semantic relatedness

$$R = \frac{|O(x) \bigcap O(y)|}{|O(x) \bigcup O(y)|} \tag{7}$$

where $O(x)$ stands for the semantic neighbourhood of the symbol x. $O(x)$ is such a set of symbols, where for every token t from the lexicon L exists an association between t and the symbol x. It is a set of all tokens associated to the question token x. As written above, the association is established only when the significance is greater than the threshold value *tresh*.

$$L \quad = \{t_1 , t_2 , t_3 , \dots , t_n\}, \tag{8}$$

$$O(x) = \{\forall t \in L : x \in L, S(x, t) > \mathit{tresh}\}. \tag{9}$$

Lexicon L is the set of all known tokens. The formula is the proportion of cardinalities of the sets of the commonly shared properties between symbols x and y to the set of all the properties of x and y. Since several fascicles can contribute with different features of symbols, we need to create an integral measure by a combination of all the contributing features,

$$R_{int} = \sum v_i \frac{|O_i(x) \bigcap O_i(y)|}{|O_i(x) \bigcup O_i(y)|} \tag{10}$$

R_{int} then represents weighted value of the relatedness between symbol x and y. v_i stands for the weight assigned to the relevant i-th fascicle.

3. Computing Similarities

Similarity (relatedness) computing is detailed described in Section 2.2. Since this is very important part of our experiments, little example showing computing of unweighted semantic relatedness will be helpful.

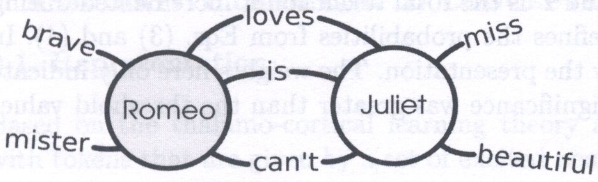
Example:

Assume three tokens "Romeo", "Juliet" and "Poison" whose associative environment is known. We want to compute the semantic similarity of token "Romeo" to tokens "Juliet" and "Poison". Lets assume that associations are defined in fascicles F_{-1} and F_1 like below.

Romeo → brave (−1), mister(−1), loves(1), is(1), can't(1)

Juliet → beautiful(−1), miss(−1), loves(1), is(1), can't(1)

Poison → the(−1), is(1), caused(1), green(1), deadly(−1)



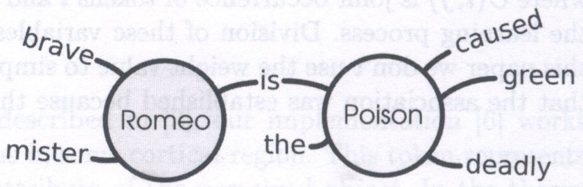


Fig. 1. Similarity for "Romeo & Juliet"

Fig. 2. Similarity for "Romeo & Poison"

We have chosen value 1 as the weights for fascicles F_1 and F_{-1}. The count of joint associations between "Romeo" and "Juliet" is 3, the count of all the associations of those tokens is 7. Thus this, using Eq. (10), the quantitative expression of their similarity is $3/7 = 0.42$.

The quantitative expression of similarity between "Romeo" and "Poison" is $1/7 = 0.14$.

The conclusion depending on the association learning theory will be, that the concept "Romeo" is more similar to concept "Juliet" than to concept "Poison".

Relatedness computing strongly depends on the source corpus. For example word "mouse" can be "an animal" or "computer device" as well. Relatedness between "mouse"-"device" and "mouse"-"animal" will then strongly depend on the corpus quality. If the corpus was computer related text, then "mouse" would be more related to "device" and it is possible, that there will be only minor relatedness (if any) with the word "animal".

4. Experiments

4.1. Artificial grammar corpus

To test our method we have generated 10 000 sentences using an artificial grammar able to form some fairly complex English sentences. We used The Simple Language Generator (SLG) program to generate the sentences. We have used sets of 2 articles, 7 adjectives, 10 nouns and 18 verb forms in our grammar input for SLG. An example of our generated training set can be seen here:

> *...John walks. Mangy dogs bite. A mangy cat walks. The nasty dog bites the girls who walk. The boy who eats walks. The dogs who Mary walks bark...*

We carried out learning based on our training set and performed clustering on the learned knowledge bases for all symbols existing in the Lexicon (terminals of our grammar). Then we counted the similarity matrix.

The similarity matrix **S**, where every element $S_{i,j}$ was the relatedness of tokens i and j was counted using Eq. (10). Weights were stored in 4 straight fascicles $F_1, \ldots, F_4$ and 4 inverse fascicles $F_{-1}, \ldots, F_{-4}$, where indexes define the contextual distances. The weights used in Eq. (10) were $v_1 = v_{-1} = 0.4$, $v_2 = v_{-2} = 0.3$, $v_3 = v_{-3} = 0.2$ and $v_4 = v_{-4} = 0.1$. These values are same for both experiments. The exact formula derived from Eq. (11) used in experiments will then look like this,

$$S_{i,j} = \sum_{\substack{k=-4 \\ k\neq 0}}^{4} v_k \cdot \frac{|O_k(i) \bigcap O_k(j)|}{|O_k(i) \bigcup O_k(j)|}. \tag{11}$$

The matrix was the input to the hierarchical clustering algorithm *hclust* in the *R* computing language. This function performs a hierarchical cluster analysis using a set of dissimilarities for the n objects being clustered. Initially, each object is assigned to its own cluster and then the algorithm proceeds iteratively, at each stage joining the two most similar clusters, continuing until there is just a single cluster. At each stage distances between clusters are recomputed by the Lance–Williams dissimilarity update formula [5] according to the particular clustering method being used, in our case, the complete method was used.

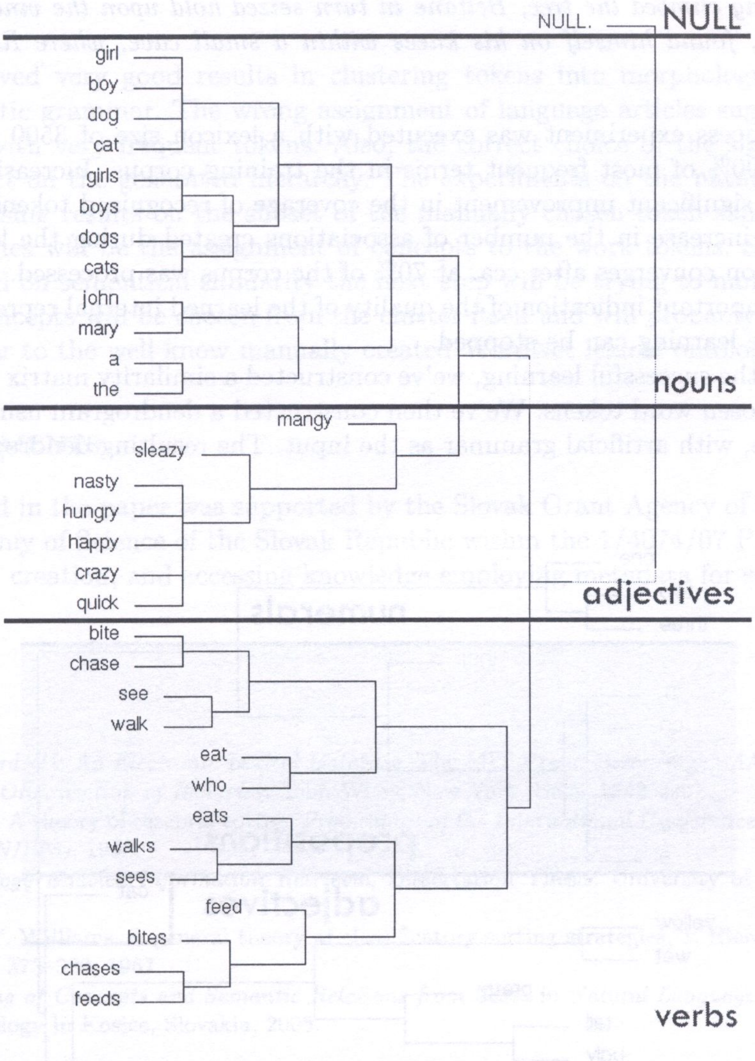


Fig. 3. Dendrogram computed upon artificial grammar corpus

The resulting dendrogram is depicted on Fig. 3.

From the figure one can see that the algorithm clustered tokens to morphological language classes with nearly 99% success rate. The only difference are "a" and "the" tokens, which are very frequent in the text and thus they have many associations. This suggests that our formula for computing mutual significance should be adjusted for very frequent symbols.

4.2. Natural language corpus

For learning the probabilities in our algorithm we used a corpus containing texts in natural language acquired from the Gutenberg library (www.gutenberg.com) consisting from randomly chosen publications in texts in English. The corpus was 200 MB in size and the count of associations was measured every 100 000 tokens.

Example from the corpus text:

> *...But at the foot of this cliff grew a tree, gnarled and stunted, the which, as Beltane watched, Black Roger began to climb, until, being some ten feet from the ground, he, reaching out and seizing a thick vine that grew upon the rock, stepped from the tree and vanished into the face of the cliff. But in a moment the leaves were parted and Roger looked forth, beckoning Beltane to*

follow. So, having climbed the tree, Beltane in turn seized hold upon the vine, and stumbling amid the leaves, found himself on his knees within a small cave, where Roger's hand met his...

The learning process experiment was executed with a lexicon size of 3500 tokens. The size of the lexicon covers 90% of most frequent terms in the training corpus. Increasing the lexicon size does not bring any significant improvement in the coverage of recognized tokens of the corpus. We have measured the increase in the number of associations created during the learning phase. The number of association converges after cca. at 70% of the corpus was processed. The convergence of associations is an important indication of the quality of the learned internal representation structure and signals that the learning can be stopped.

To demonstrate the successful learning, we've constructed a similarity matrix from a small subset (30) of manually chosen word tokens. We've then constructed a dendrogram using the same process as in the case above, with artificial grammar as the input. The resulting dendrogram is depicted on Fig. 4.

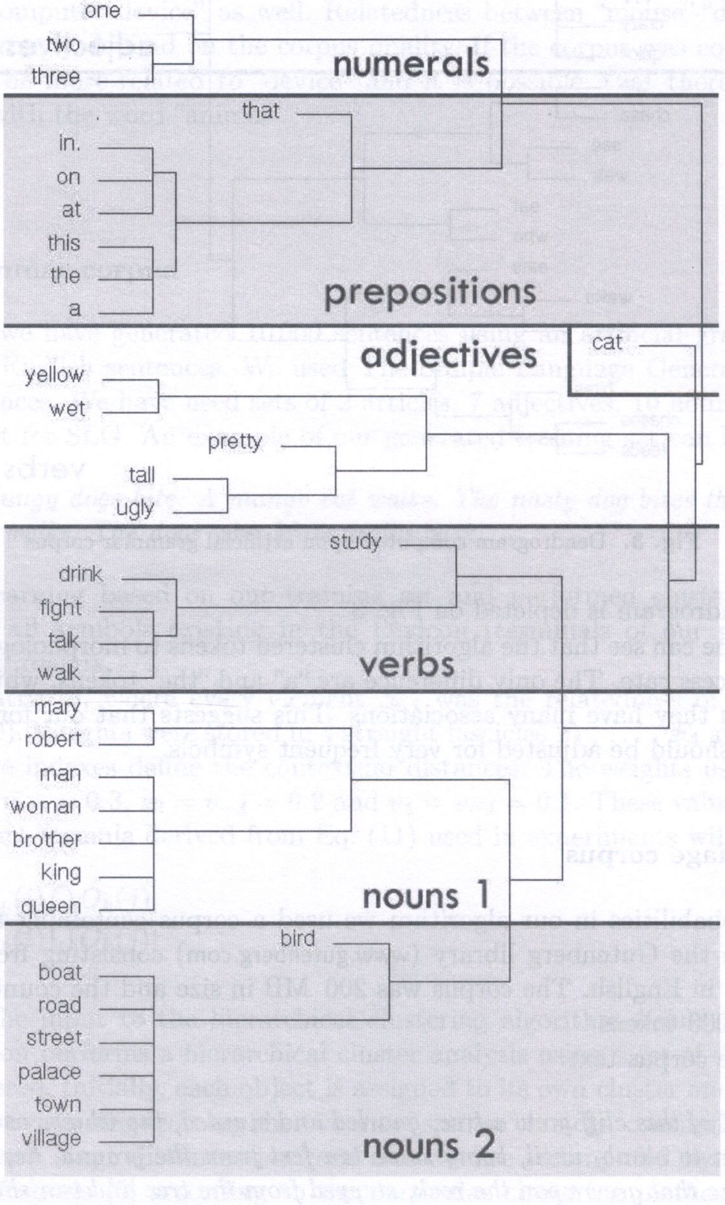


Fig. 4. Dendrogram computed upon natural language corpus

5. Conclusion

Our method achieved very good results in clustering tokens into morphological subsets on the artificial probabilistic grammar. The wrong assignment of language articles suggests adjusting our formulas to cope with very frequent tokens. Also, the correct choice of the significance threshold has a strong impact on the generated hierarchy. The experiments on the natural language corpus showed very promising results on the subset of the manually chosen token sample. Our next goal in building ontologies will be the assignment of concepts to the work tokens. Since we are able to build clusters based on semantical similarity the next step will be trying to name the clusters with concepts. These concepts will be chosen from the cluster itself and will probably represented as sets of synonyms similar to the well know manually created WordNet lexical ontology [1].

Acknowledgements

The work presented in the paper was supported by the Slovak Grant Agency of the Ministry of Education and Academy of Science of the Slovak Republic within the 1/4074/07 Project "Methods for annotation, search, creation, and accessing knowledge employing metadata for semantic description of knowledge".

References

[1] C. Fellbaum. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, MA, 1998.

[2] D.O. Hebb. *The Organization of Behavior*. John Wiley, New York, USA, 1949.

[3] R. Hecht-Nielsen. A theory of cerebral cortex. *Proceedings of the International Conference on Neural Information Processing (ICONIP98)*, 1998.

[4] R. Kende. *Ontology Enabled Information Retrieval*, Dissertation Thesis. University of Technology in Kosice, Slovakia, 2006.

[5] G.N. Lance, W.T. Williams. A general theory of classificatory sorting strategies, 1. Hierarchical systems. *Computer Journal*, **9**: 373–380, 1967.

[6] V. Rockai. *Mining of Concepts and Semantic Relations from Texts in Natural Language*, Diploma Thesis. University of Technology in Kosice, Slovakia, 2005.