Automatic Extraction of Semantic Relations from Specialized Corpora

Aristomenis Thanopoulos, Nikos Fakotakis and George Kokkinakis

Wire Communications Laboratory
Electrical & Computer Engineering Dept., University of Patras
26500 Rion, Patras, Greece
aristom@wcl.ee.upatras.gr

Abstract

In this paper we address the problem of discovering word semantic similarities via statistical processing of text corpora. We propose a knowledge-poor method that exploits the sentencial context of words for extracting similarity relations between them as well as semantic in nature word clusters. The approach aims at full portability across domains and languages and therefore is based on minimal resources.

1 Motivation

Providing digital computers with the capability to acquire conceptual relations between lexical items by processing real-life text corpora is not only an exciting research activity but also a significant task in the framework of many NLP systems. Specifically:

1. State-of-the-art Language Modeling techniques (McMahon and Smith., 1996) require lexical information about word classes.
2. Thesauri creation in a (semi-) automatic manner in any domain and language with minimal dependence on specialized tools and resources is very important. Most thematic domains today in most of the languages lack semantic resources. Adopting a knowledge-poor corpus-based method not only much less labor is necessary in construction of conceptual structures but also domain-dependent semantic relations are obtained. New resources can be readily created in new domains or existing thesauri can be enlarged or refined by re-training on larger corpora as soon as they become available.
3. Many currently implemented, both spoken and written, NLP systems operate in a specific domain and usually utilize a constrained vocabulary related directly to their task domain. Therefore semantic domain-dependent knowledge can be acquired directly from relevant corpora.
4. Autonomous computational intelligence should rely mainly on processing of free flow electronic texts for acquiring new semantic and world knowledge.

The present approach aims at corpus-based automatic extraction of domain-dependent semantic similarity relations between lexical items and the formation of corresponding semantic clusters. For this purpose, the usage of readily available domain-specific text corpora is imperative. The guideline of our approach was the adaptation to the special characteristics of this type of corpora (specialization, restricted size) without imposing the need for other domain-dependent resources and obtaining portability across languages.

2 Related work

Three main approaches have been proposed for the automatic extraction of lexical semantics knowledge: *syntax*-based, *n-gram*-based and *window*-based. Syntax-based methods (referred also as knowledge-rich in contrast to the others - knowledge-poor methods) (Pereira and Thishby, 1992; Grefenstette, 1993; Li and Abe, 1997) represent the words under consideration as vectors containing statistic values of their syntactic properties in relation to a given set of words (e.g. statistics of *object* syntax relations referring to a set of verbs) and cluster the considered words according to similarity of the corresponding vectors. Methods that use bigrams (Brown et al., 1992) or trigrams (Martin et al., 1998) cluster words considering as a word's context the one or two immediately adjacent words and employ as clustering criteria the minimal loss of average

mutual information and the perplexity improvement respectively. Such methods are oriented to language modeling and aim primarily at rough but fast clustering of large vocabularies. Brown et al. (1992) also proposed a window method introducing the concept of "semantic stickiness" of two words as the relatively frequent close occurrence between them (less than 500 words distance). Although this is an efficient and entirely knowledge-poor method for extracting both semantic relations and clusters, the extracted relations are not restricted to semantic similarity but extend on thematic roles. Moreover its applicability to small and specialized corpora is uncertain.

3 A knowledge-poor approach

In order to achieve portability we approach the issue from a knowledge-poor perspective. Syntax-based methods employ partial parsers which require highly language-dependent resources (morphological/grammatical analysis), and/or properly tagged training corpus in order to detect syntactic relations between sentence constituents. On the other hand, n-gram methods operate on large corpora and, in order to reduce computational resources, consider as context words only the immediately adjacent ones. Medium-distance word context is not exploited. Since large corpora are available only for few domains we aimed at developing a method for processing small or medium sized corpora exploiting the most of contextual information, that is, the full sentential context of words. Our approach was driven by the observation that in domain-constrained corpora, unlike fiction or general journalese, the vocabulary is limited, the syntactic structures are not complex and that medium-distance lexical patterns are frequently used to express similar facts.

Specifically we have developed two different algorithms in respect to the context consideration they employ: *Word-based* and *Pattern-based*. The former acquires word-based contextual data (extended up to sentence boundaries), according to the distributional similarity of which, word similarity relations are extracted. The latter detects common patterns throughout the corpus that indicate possible word similarities. For example, consider the sentence fragments:

"...while the S&P index inched up 0.3%."

"The DAX index inched up 0.70 point to close..."

Although their syntactic structures are different, the common contextual pattern (appearing beyond immediately adjacent words) indicates a possible similarity between the tokens *'S&P'* and *'DAX'*. Word pairs that persistently appear such context similarity throughout the corpus (frequently observed in technical texts) are confidently indicated as semantically similar. Our method captures such context similarity and extracts a proportionate measure about semantic similarity between lexical items.

Most approaches (Brown et al., 1992; Li & Abe, 1997) inherently extract semantic knowledge in the abstracted form of semantic clusters. Our method produces semantic similarity relations as an intermediate (and information-richer) semantics representation formalism, from which cluster hierarchies can be generated. Of great importance is that soft clustering methods can also be applied to this set of relations and cluster polysemous words to more than one classes.

Stock market-financial news and Modern Greek, were used as domain and language test case respectively. However demonstrative examples taken from the WSJ corpus have been used throughout the paper as well.

4 Context Similarity Estimation

The main idea supporting context-based word clustering is that two words that can substitute one another in several different contexts always providing meaningful word sequences are probably semantically similar. Present n-gram based methods utilize this assumption considering as a context of a *focus* word only the one or two immediately adjacent *parameter* words.

In the present work, we consider as word context the whole sentence in which the examined word appears, excluding only the semantically empty (i.e. functional) words such as articles, conjunctions, particles, auxiliaries. Adopting this word context notion we proceed to the following analysis:

Let us consider a text corpus $\mathbf{T_C}$ with vocabulary V_C and $V_S \subseteq V_C$ the set of words that are of interest in extracting semantic similarity relations between them. V_S comprises the non-functional words of

V_C appearing in $\mathbf{T_C}$ with a frequency higher than a threshold (set to 20 in the presented experiments) in order to acquire sufficient data for every focus word. Let $V_P \subseteq V_C$ be the set of words that will be used as context parameters. Ideally, any word appearing at least twice in the corpus could be used as context parameter. However we specified this word frequency threshold to 10 in order to diminish computational time. Consider a sentence of $\mathbf{T_C}$:

$$S_m = w_1, w_2, ..., w_{j-1}, w_j, w_{j+1}, ..., w_k$$

We define as ***sentential context*** *of w_i in S_m* the set of the pairs of the sentence words which are members of V_P, accompanied by their corresponding distance from w_j:

$$C_{S_m}(w_j) = \{(i - j, w_i), i = 1..k, (i \neq j),\ \forall w_i \in V_P\}$$

Equation (1): Sentential context of w_i in S_m

More formally, $C_{S_m}(w_j)$ can be represented as a binary-valued matrix defined over the set $\mu = \delta \times \omega$ where $\boldsymbol{\delta}$={-1,1,-2,2,...,-L_m,L_m}, L_m being the maximum word distance we regard that carries useful contextual information (for full sentence distance L_m=L_{max}-1 where L_{max} the maximum sentence length in T_C), and $\boldsymbol{\omega}$ the ordered set V_S:

$$C_{S_m}(w_j) = \left\{ c_{j,m}(d,w) \right\}_{d \in \delta, w \in \omega}, \text{where:}$$

$$c_{j,m}(d,w) = \begin{cases} 1, & w = w_j, w_j \in \omega, d = i - j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Summing over all corpus sentences we obtain the contextual data matrices for every $w_j \in V_S$:

$$C_{T_C}(w_j) = \{c_j(d,w)\} = \sum_m C_{S_m}(w_j) \quad (3)$$

The word semantic similarity estimation has been reduced to matrices similarity estimation. The obtained contextual matrices are compared using a weighted Tanimoto measure (Charniak, 1993) and a word similarity measure $\mathbf{S_m(w_i,w_j)}$ is obtained:

$$S_m(w_i, w_j) = \frac{\sum_d \sum_w \left[h(d) \cdot c_i(d,w) \cdot c_j(d,w) \right]}{\sum_d \sum_w \max\left\{ c_i(d,w), c_j(d,w) \right\}} \quad (4)$$

The weight function **h(d)** defines the desired influence that the distance between words should have to context similarity estimation. In this experiment we set: **h(d)=1/|d|**. In order to reduce computational time the denominator was set to $\sum_d \sum_w c_i(d,w) + \sum_d \sum_w c_j(d,w)$, a modification that has minimal effect on the final result. Experimental results of this method (Word-based Context Similarity Estimation – **WCSE**), are shown in Table 1. Note that, since the $C_{T_C}(w_j)$ matrix is sparse, (1) was used as data storing formula instead of (2), in order to diminish computational cost.

The previously described algorithm is handling all contextual data in a uniform way. However, study of the results showed that preference should be given to hits derived from many different similar contexts instead of few ones appearing many times. This would clearly give better results since the latter case may be due to often-used stereotyped expressions or repeated facts. In order to achieve this we modified (4) to:

$$S_m(w_i, w_j) = \frac{\sum_d \sum_w \left[h(d) \cdot \log_2 \left[c_i(d,w) \cdot c_j(d,w) \right] \right]}{\sum_d \sum_w c_i(d,w) + \sum_d \sum_w c_j(d,w)} \quad (5)$$

Indeed the experimental results of this variation (Variant WCSE – **VWCSE**) show a significant improvement (see Table 1).

5 Dynamic pattern detection for context similarity estimation

In the previously described method the notion of word context is based on independent intra-sentential word co-occurrences. However similarity of contextual patterns is much more reliable word similarity criterion than word-based context similarity. That is, if the sentential contexts $C_{S_m}(w_i)$ and $C_{S_n}(w_j)$ have at least two common elements, we count this as a much more confident hit regarding the w_i and w_j similarity. A measure expressing the weight of the common pattern is obtained. Since the patterns under detection vary across languages and domains we need a method that extracts them dynamically, regardless of the text genre, domain or language.

For this purpose we propose an algorithm that performs a sentence-by-sentence comparison along the corpus. This comparison is based on the

cross-correlation concept as it is used in digital signal processing. A sentence can be considered as a digital signal where every semantic token corresponds to a signal sample. In order to detect words with common contexts every sentence is checked on matching every other one partially (i.e. matching the semantic category of one or more tokens) on every possible relative position between the two sentences. Wherever common patterns of semantic tokens are found the neighboring respective tokens on the two sentences are stored as candidate semantic relatives.

During this process contextual data are not maintained in memory; instead the detection of a common pattern in both sentences results to the storage of several hits (i.e. candidate similar word pairs) or to the increase of their corresponding similarity measure according to the pattern similarity of their contexts.

Let S_m and S_n be two sentences that undergo the cross-correlation procedure. If $\delta_x=\{d_x,\ x=1..x_1,\ x_1>1\}$, is the set of word distances that satisfy the equality: $c_{i,m}(d_x,w_y)=c_{j,n}(d_x,w_y)=1$, then the pair (w_i,w_j) is stored as a hit accompanied by the following context similarity measure:

$$F_{m,n}(w_i,w_j)=\sum_{p}^{X}\frac{1}{|d_p|}+\sum_{\substack{q,p\\ q<p}}^{X}\frac{1}{\sqrt{|d_p\cdot d_q|}} \qquad (6)$$

Keeping only the first term we obtain the same result as in the WCSE method with weight function h(d)=1/|d|. The second term augments the score in proportion to the cohesion and the size of the detected pattern depending on the position of w_i (or, equivalently, w_j). Dividing (6) by the total length of S_m and S_n (i.e. $L_{mn}=L_m+L_n$) we obtain a normalized measure of the cross-correlation of the two sentences:

$$F^{N}{}_{m,n}(w_i,w_j)=\frac{F_{m,n}(w_i,w_j)}{L_{mn}} \qquad (7)$$

The total similarity measure is obtained from:

$$F(w_i,w_j)=\sum_{m}\sum_{\substack{n\\ n>m}}F_{mn}^{(N)}(w_i,w_j) \qquad (8)$$

applied throughout the corpus.

In order to reduce search time and required memory during the whole process a pruning mechanism is applied at regular time intervals to eliminate word pairs with a relatively very low semantic similarity score.

Dividing (8) by the product of the word probabilities $P(w_i)\cdot P(w_j)$ we obtain the normalized similarity measure $\mathbf{F_N(w_i,w_j)}$.

In order to constrain the degradation of our results due to sparse data regarding less frequent words, we multiply (8) by P_C, a data sufficiency measure function of $P(w_i)$ and $P(w_j)$, obtaining F_U, a more reliable measure. Here we employed:

$$P_C(P_i,P_j)=\begin{cases}\dfrac{P_i\cdot P_j}{P_{Th}^2}, & P_i\cdot P_j<P_{Th}^2\\ 1, & \text{otherwise}\end{cases} \qquad (9)$$

where we used $P_{Th}=30/|T_C|$, $|T_C|$ being the size of the corpus.

Finally, sorting the resulting pairs by F_U and keeping the N-best scoring pairs, we obtain the preponderant semantically related candidates.

6 Preprocessing

In order to apply the above described algorithms some preprocessing is necessary:

1. A trainable sentence splitter and a rule-based chunker are applied. Sentence boundaries confine the scope of context while phrase boundaries determine the maximum extent of semantic tokens (see below).

2. The next step of the preprocessing is what we call *"semantic tokenization"*. We try to reduce context parameters and simultaneously to increase the volume of contextual data either by reducing the volume of both the focus and parameter word set or by discaring or merging lexical items resulting in reduction of the distance between semantic tokens. Words or word sequences are thus classified in common semantic categories employing syntactical, morphological and collocational information:

a. Functionals (auxiliaries, determiners) are discarded since they do not modify semantically their head words. Words of indeterminable semantic content (pronouns, low frequency words) are treated as empty tokens.
b. Known domain-independent lexical patterns incorporating arithmetic and temporal

expressions (e.g. dates, numbers, amounts, etc.) are regarded as a single semantic token and tagged accordingly. Their information content is indifferent to semantic knowledge acquisition; therefore we preserve only class information.

c. Frequently appearing lexical patterns which represent single semantic entities in the specific domain are treated as a single (albeit composite) "semantic token". Their detection is based on the following algorithm (cf. Smadja, 1993):

   1. Extract "significant bigrams" confined inside noun phrases i.e. immediately adjacent words that contain a relatively high amount of mutual information:

$$I_{mutual}(w_1, w_2) = \log_2 \frac{P(w_1 w_2)}{P(w_1)P(w_2)} \qquad (10)$$

   2. Combine *significant bigrams* together to obtain "significant n-grams" found in the corpus and confined inside noun phrases as well. Discard subsumed m-grams (m<n) only if they do not occur indepentently in the corpus.

   3. Tag throughout the corpus the significant n-grams as single semantic tokens, starting from the higher-order ones.

   Semantic entities that are lexically represented as sticky word chains may be either standard – in the framework of the information extraction task – named entities, such as "Latin America" (location), "Russian president Boris Yeltsin" (person), "Τράπεζα Μακεδονίας-Θράκης" ("*Bank of Macedonia and Thrace*"; organization) or representations of domain-specific typical events ("αύξηση μετοχικού κεφαλαίου" = rise of equity capital), abstract concepts ("Dow Jones industrials"), etc. To ensure that the detected "sticky" phrases actually represent semantic entities, human inspection is necessary for discarding the spurious ones, since repeated word sequences that do not constitute always single semantic entities often appear in specialized texts.

From the above it is apparent that we use the term "***semantic token***" to refer to a recognized semantic pattern (e.g. <*date*>), a rigid word chain (e.g. *"Dow_Jones_industrials"*) or a single content word. The context similarity estimation algorithms were run using vocabularies of focus and parameter words derived from the extracted set of semantic tokens.

7 Incorporating heuristics

From the study of the erroneously extracted semantic relations certain systematic errors were detected. For example, adjectives, adverbs or adjunctive nouns that occur interpolating in otherwise similar contexts lead to the extraction of spurious pairs. Consider for example the phrases: "η αύξηση της τιμής της βενζίνης" (= *the increase of the benzine price)* and "η αύξηση της τιμής πώλησης της βενζίνης" (=*the increase of the disposal benzine price)*. Every algorithm based on word adjacency data outputs as erroneous hits the pairs {*benzine-disposal*} and { *increase-disposal*}. A rule that was applied to deal with this problem is:

> If $w_i \in S_m$ and $w_j \in S_n$ have similar contexts, count the pair (w_i, w_j) as a hit only if $w_i \neq w_{j\pm1}$ and $w_j \neq w_{i\pm1}$.

Such contextual rules can be applied only using the cross-correlation method for the context similarity estimation (either pattern-based or word-based).

8 Word Clustering

Although the obtained semantically related N-best pair list constitutes already a thesaurus-like and information-rich form of semantic knowledge representation, many NLP applications (e.g. language modeling) require word clusters instead of word relations. However, since a word similarity measure has been extracted, the formation of clusters is a rather trivial problem, although more complex for "soft clustering" (i.e. a word can be classified in more than one classes).

In order to construct word classes we applied the unsupervised agglomerative hard clustering algorithm shown in *Figure 1* over the set of semantic relations. Each distinct lexical item is initially assigned to a cluster and then clusters are merged into larger ones according to the *average linkage* measure. Merging of clusters stops when the distance between the more proximate clusters exceeds a threshold proportional to the average distance between words. Tracking the successive merges we obtain sub-cluster hierarchies, such as the one shown in *Figure 2*.

Repeat until $\min(\text{AvgDistance}(C_J, C_I)) > k \cdot \frac{1}{|V_S|^2} \cdot \sum_{\forall w_i, w_j \in V_S} \text{distance}(w_i, w_j)$

  for every cluster C_I
    for every cluster $C_J \neq C_I$

      calculate $\text{AvgDistance}(C_J, C_I) = \frac{1}{|C_I| \cdot |C_J|} \cdot \sum_{\forall w_i \in C_I, \forall w_j \in C_J} \text{distance}(w_i, w_j)$

  merge $C_I, \underset{C_J}{Arg\min}\,(AvgDistance(C_j, C_I))$

Figure 1: Unsupervised Hard Clustering Algorithm

9 Experimental Results

The reported experiments have been carried out on a 220.000 words corpus, comprised of financial news of 1998, which was constructed in the framework of a currently carried out R&D project for Information Extraction from raw text[1].

The methods and their variations described in sections 4 and 5 for obtaining lexical semantic relations were tested and their accuracy per number of best hits was measured by human inspection. The VWCSE method was tested using only the previous and next word as context parameters (N&P method), to sketch a method baseline for the particular corpus. Using a Morphological Analyzer & Part-of-Speech tagger to restrict semantic relations only between words of the same Part-of-Speech (**PoS**) we obtain apparently higher accuracy, though we loose some interesting verb - noun pairs referring to the same action or condition, e.g. *αυξήθηκε* (=increased) and *άνοδος* (=increment). The results indicate that the normalization factors indeed improve the accuracy of the methods and that context similarity detection based on dynamic pattern-matching yields significantly more reliable results than the word-based method. This demonstrates the importance of the cross-correlation algorithm, which is the only suitable for pattern-based context similarity detection.

Regarding the clustering procedure, a set of 1300 words was clustered to 84 hierarchically structured clusters. Considering an interested cluster formed (Figure 2) we note that from the 18 lexical entities (words or rigid phrases) that constitute the cluster all but two refer to money investment or profit. From the vocabulary subject to clustering 4 words belonging to the same class were not detected; therefore accuracy and recall for the specific cluster were found at 88.9% and 80% respectively.

Although comparision with other knowledge-poor methods would be very useful it was not realized, mainly because our method produces semantic relations while other methods produce semantic clusters and our clustering process is not yet elaborated enough to yield quality results.

Lexical Item	*English Transl.*
κονδυλίων	outlays/g
κεφάλαια	capitals
κεφαλαίων	capitals /g
ρευστότητας	fluidity /g
επενδύσεις	investments
επένδυση	investment
*προγράμματος	program/g
τίτλων_δημοσίου	state stocks/g
ομολόγων	income bonds/g
εντόκων_γραμματίων	time notes /g
ζημιές	losses
κέρδη	profits
καταθέσεις	deposits
ζημιών	losses /g
πωλήσεις	purchases
εσόδων	incomes/g
έσοδα	incomes
συναλλαγές	dealings
*σύμβαση	contract

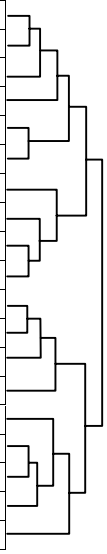

Figure 2: A derived sample hierarchial cluster of lexical entities ('/g' denotes genitive case)

[1] Project "MITOS" of the Greek General Secretariat for Reseach and Technology

METHOD		Precision (%) per number of best hits			
		100	200	300	400
N&P		64	61	57.7	54.75
WCSE		72	65.5	61.7	57
VWCSE		81	70	66	62.5
CCPM		74	59	54	50.5
CCPM-N		89	81.5	70.3	63
CCPM-N-F		90	80.5	72.7	67.25
CCPM-N-F-P_C		93	82	71.3	66.75
PoS &	VWCSE	86	80	75	67.5
	CCPM	86	77.5	68	59.5
	CCPM-N	93	88	79	74
	CCPM-N-F	95	88.5	83	77
	CCPM-N-F-P_C	97	89	82.7	77

N&P: Context = next and previous word

WCSE: every word into the sentence is taken as context parameter evenly - Eq.(4)

VWCSE: contextual similarity variance is favored - Eq.(5)

CCPM: Dynamic Pattern-Matching based on Cross-Correlation - Eq. (6)&(8)
CCPM-N: normalized by L_{mn} (7)&(8)

CCPM-N-F: normalized by $P(w_i)\cdot P(w_j)$

CCPM-N-F-P_C: normalized by P_C, Eq.(9)

Table 1: Comparative Results and Explanation Memo

10 Conclusion

Initiating from the conception of word similarity estimation in terms of context similarity we have proposed an approach with several variations for extracting semantic similarity relations betweenlexical entities by processing word adjacency data obtained from small or medium sized corpora. The described cross-correlation procedure, offers the possibility to dynamically detect pattern context similarities offering strong evidence for semantic similarity. The presented algorithm featureslanguage and domain portability and the ability to classify keywords irrespective of their grammatical characteristics.

The implementation of the soft clustering algorithm, the test of the method to a different domain and language and the quantified comparison with other knowledge-poor methods are quite interesting matters belonging to future work.

References

Brown P.F., DellaPietra V.J., DeSouza P.V., Lai J.C., Mercer R.L.: *Class-Based n-gram Models of Natural Language*. Computational Linguistics, 18(4): pp. 467-479, 1992.

Charniak E.: Statistical Language Learning. The MIT Press, 1993.

Grefenstette, G.: *SEXTANT: Extracting Semantics fromRaw Text: Implementation Details*. The Journal of Knowledge Engineering, 1993.

Li H., Abe N.: *Clustering Words with the MDL Principle*. Journal of Natural Language Processing v.4, n.2, 1997.

Martin S., Liermann J., Ney H.: *Algorithms for bigram and trigram word clustering*. Speech Communication 24, pp.19-37, 1998.

McMahon J.G., Smith F.J.: *Improving Statistical Language Model Performance with Automatically Generated Word Hierarchies*. Computational Linguistics, 22(2) 1996.

Pereira F., Tishby N.: *Distributional Similarity, Phrase Transitions and Hierarchical Clustering.* In Working Notes, Fall Symposium Series. AAAI pp.108-112, 1992.

Smadja F.: *Retrieving Collocations from text: Xtract*. Computational Linguistics, 19(1): pp. 143-177, 1993.