Semantic Similarities and General-Specific Noun Relations from the Web

Gaël Dias*, Raycho Mukelov*, Guillaume Cleuziou** and Veska Noncheva***

*University of Beira Interior Covilhã - Portugal
ddg@di.ubi.pt , raicho@penhas.di.ubi.pt
**University of Orléans, Orléans - France
guillaume.cleuziou@univ-orleans.fr
***University of Plovdiv, Plodiv - Bulgaria
wesnon@pu.acad.bg

Résumé. Dans cet article nous proposons une nouvelle méthodologie utilisant les graphes orientés pondérés et l'algorithme TextRank proposé par Mihalcea et Tarau (2004) dans le but d'extraire automatiquement des relations de généralités entre noms à partir de collocations observées sur un corpus Web. Plusieurs mesures d'association (non-symétriques) ont été implémentées pour construire les graphes sur lesquels l'algorithme TextRank a été appliqué afin de produire une liste de noms ordonnés du plus général au plus spécifique. Les résultats ont été évalués quantitativement en utilisant la hiérarchie des noms de WordNet comme base de référence.

1 Introduction

Taxonomies are crucial for any knowledge-based system. They are in fact important because they allow to structure information, thus fostering their search and reuse. However, it is well known that any knowledge-based system suffers from the so-called knowledge acquisition bottleneck, *i.e.* the difficulty to actually model the domain in question. As stated by Caraballo (1999), WordNet has been an important lexical knowledge base, but it is insufficient for domain specific texts. So, many attempts have been made to automatically produce taxonomies (Grefenstette (1994)), but Caraballo (1999) is certainly the first work which proposes a complete overview of the problem by (1) automatically building a hierarchical structure of nouns based on bottom-up clustering methods and (2) labeling the internal nodes of the resulting tree with hypernyms from the nouns clustered underneath by using patterns like "B is a kind of A".

In this paper, we are interested in dealing with the second problem of the construction of an organized lexical resource *i.e.* discovering general-specific noun relations, so that correct nouns are chosen to label internal nodes of any hierarchical knowledge base, such as the one proposed by Dias et al. (2006). Most of the works proposed so far have (1) used predefined patterns or (2) automatically learned these patterns to identify hypernym/hyponym relations. From the first paradigm, Hearst (1992) first identifies a set of lexico-syntactic patterns that are easily recognizable *i.e.* occur frequently and across text genre boundaries. These can be called seed patterns. Based on these seeds, he proposes a bootstrapping algorithm to semi-automatically

acquire new more specific patterns. Similarly, Caraballo (1999) uses predefined patterns such as “X is a kind of Y” or “X, Y, and other Zs” to identify hypernym/hyponym relations. This approach to information extraction is based on a technique called selective concept extraction as defined by Riloff (1993).

A more challenging task is to automatically learn the relevant patterns for the hypernym/hyponym relations. In the context of pattern extraction, there exist many approaches as summarized by Stevenson et Greenwood (2006). The most well-known work in this area is certainly the one proposed by Snow et al. (2006) who use machine learning techniques to automatically replace hand-built knowledge.

Links between words that result from manual or semi-automatic acquisition of relevant predicative or discursive patterns (Hearst (1992); Caraballo (1999)) are fine and accurate, but such an acquisition is a tedious task that requires substantial manual work. On the other side, works done by Snow et al. (2006) have proposed methodologies to automatically acquire these patterns mostly based on supervised learning to leverage manual work. However, training sets still need to be built. Unlike other approaches, we propose an unsupervised methodology which aims at discovering general-specific noun relations which can be assimilated to hypernym/ hyponym relations detection The advantages of this approach are clear as it can be applied to any language or any domain without any previous knowledge, based on a simple assumption : specific words tend to attract general words with more strength than the opposite. As Michelbacher et al. (2007) state : *“there is a tendency for a strong forward association from a specific term like adenocarcinoma to the more general term cancer, whereas the association from cancer to adenocarcinoma is weak”*.

Based on this assumption, we propose a methodology based on directed weighted graphs and the TextRank algorithm (Mihalcea et Tarau (2004)) to automatically induce general-specific noun relations from web corpora frequency counts. Indeed, asymmetry in Natural Language Processing can be seen as a possible reason for the degree of generality of terms (Michelbacher et al. (2007)). So, different asymmetric association measures are implemented to build the graphs upon which the TextRank algorithm is applied and produces an ordered list of nouns from the most general to the most specific. Experiments have been conducted based on the WordNet noun hierarchy and a quantitative evaluation proposed using the statistical language identification model (Beesley (1998)).

2 Asymmetric Association Measures

Michelbacher et al. (2007) clearly point at the importance of asymmetry in Natural Language Processing. In particular, we deeply believe that asymmetry is a key factor for discovering the degree of generality of terms. It is cognitively sensible to state that when someone hears about "mango", he may induce the properties of a *"fruit"*. But, when hearing *"fruit"*, more common fruits will be likely to come into mind such as *"apple"* or *"banana"*. In this case, there exists an oriented association between *"fruit"* and *"mango"* (mango $\rightarrow$ fruit) which indicates that *"mango"* attracts more *"fruit"* than *"fruit"* attracts *"mango"*. As a consequence, *"fruit"* is more likely to be a more general term than *"mango"*.

Based on this assumption, asymmetric association measures are necessary to induce these associations. Pecina et Schlesinger (2006) and Tan et al. (2004) propose exhaustive lists of association measures from which we present the asymmetric ones that will be used to measure

the degree of attractiveness between two nouns, x and y, where $f(.,.)$, $P(.)$ and $P(.,.)$ are respectively the frequency function, the marginal probability function and the joint probability function, and N the total of digrams.

$$Braun - Blanquet = \frac{f(x,y)}{\max\left(f(x,y) + f(x,\bar{y}), f(x,y) + f(\bar{x},y)\right)} \tag{1}$$

$$J\ measure = \max \left[\begin{array}{l} P(x,y)\log\frac{P(y|x)}{P(x)} + P(x,\bar{y})\log\frac{P(\bar{y}|x)}{P(\bar{y})}, \\ P(x,y)\log\frac{P(x|y)}{P(x)} + P(\bar{x},y)\log\frac{P(\bar{x}|y)}{P(\bar{x})} \end{array} \right]. \tag{2}$$

$$Confidence = \max[P(x|y), P(y|x)] \tag{3}$$

$$Laplace = \max\left[\frac{N.P(x,y)+1}{N.P(x)+2}, \frac{N.P(x,y)+1}{N.P(y)+2}\right]] \tag{4}$$

$$Conviction = \max\left[\frac{P(x).P(\bar{y})}{P(x,\bar{y})}, \frac{P(\bar{x}).P(y)}{P(\bar{x},y)}\right] \tag{5}$$

$$Certainty\ Factor = \max\left[\frac{P(y|x) - P(y)}{1 - P(y)}, \frac{P(x|y) - P(x)}{1 - P(x)}\right] \tag{6}$$

$$Added\ Value = \max[P(y|x) - P(y), P(x|y) - P(x)] \tag{7}$$

All seven equations show their asymmetry by evaluating the maximum value between two hypotheses *i.e.* by evaluating the attraction of x upon y but also the attraction of y upon x. As a consequence, the maximum value will decide the direction of the general-specific association *i.e.* $(x \rightarrow y)$ or $(y \rightarrow x)$.

3 TextRank Algorithm

Graph-based ranking algorithms are essentially a way of deciding the importance of a vertex within a graph, based on global information recursively drawn from the entire graph. The basic idea implemented by a graph-based ranking model is that of "voting" or "recommendation". When one vertex links to another one, it is basically casting a vote for that other vertex. The higher the number of votes that are cast for a vertex, the higher the importance of the vertex. Moreover, the importance of the vertex casting the vote determines how important the vote itself is, and this information is also taken into account by the ranking model. Hence, the score associated with a vertex is determined based on the votes that are cast for it, and the score of the vertices casting these votes.

Our intuition of using graph-based ranking algorithms is that more general words will be more likely to have incoming associations as they will be associated to many specific words. On the opposite, they will have few outgoing associations as they will not attract specific words. As a consequence, the voting paradigm of graph-based ranking algorithms should give more strength to general words than specific ones, thus resulting in an ordered list of words from general to specific.

For that purpose, we first need to build a directed graph. Informally, if x attracts more y than y attracts x, we will draw an edge between x and y as follows ($x \rightarrow y$) as we want to give more credits to general words. Formally, we can define a directed graph $G = (V, E)$ with the set of vertices V (in our case, a set of words) and a set of edges E where E is a subset of $V \times V$ (in our case, defined by the asymmetric association measure value between two words). In Figure 1, we show the directed graph obtained by using the set of words $V = \{$*isometry, rate of growth, growth rate, rate*$\}$ randomly extracted from WordNet where *"rate of growth"* and *"growth rate"* are synonyms, *"isometry"* an hyponynym of the previous set and *"rate"* an hypernym of the same set. The weights associated to the edges have been evaluated by the confidence association measure (Equation 3) based on web search engine counts[1].

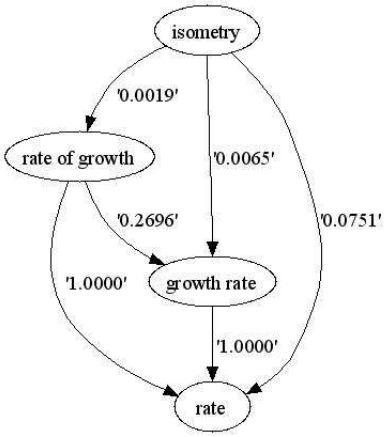

FIG. 1 – *Directed Graph Construction.*

Figure 1 clearly shows our assumption of generality of terms as the hypernym *"rate"* only has incoming edges whereas the hyponym *"isometry"* only has outgoing edges. As a consequence, by applying a graph-based ranking algorithm, we aim at producing an ordered list of words from the most general (with the highest value) to the most specific (with the lowest value). For that purpose, we present the TextRank algorithm proposed by Mihalcea et Tarau (2004) both for unweighted and weighted directed graphs.

Unweighted Directed Graph

For a given vertex V_i let $In(V_i)$ be the set of vertices that point to it (predecessors), and let $Out(V_i)$ be the set of vertices that vertex V_i points to (successors). The score of a vertex V_i is defined in Equation 8 where d (usually set to 0.85) is a damping factor that can be set between 0 and 1, which has the role of integrating into the model the probability of jumping from a given vertex to another random vertex in the graph.

$$S(V_i) = (1-d) + d \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} \times S(V_j) \quad (8)$$

[1]We used counts returned by http ://www.yahoo.com.

Weighted Directed Graph

In order to take into account the weights of the edges, a new formula is introduced in Equation 9.

$$WS(V_i) = (1-d) + d \sum_{j \in In(V_i)} \frac{w_{ji}}{\sum_{k \in Out(V_j)} w_{j,k}} \times WS(V_j) \quad (9)$$

After running the algorithm in both cases, a score is associated to each vertex, which represents the "importance" of the vertex within the graph. Notice that the final values obtained after TextRank runs to completion are not affected by the choice of the initial values randomly assigned to the vertices, only the number of iterations needed for convergence may be different. As a consequence, after running the TextRank algorithm, in both its configurations, the output is an ordered list of words from the most general one to the most specific one. In table 1, we show both the lists with the weighted and unweighted versions of the TextRank based on the directed graph shown in Figure 1.

Unweighted		Weighted		WordNet	
$S(V_i)$	Word	$WS(V_i)$	Word	Category	Word
0.50	*rate*	0.81	*rate*	Hypernym	*rate*
0.27	*growth rate*	0.44	*growth rate*	Synset	*growth rate*
0.19	*rate of growth*	0.26	*rate of growth*	Synset	*rate of growth*
0.15	*isometry*	0.15	*isometry*	Hyponym	*isometry*

TAB. 1 – *TextRank ordered lists.*

The results show that asymmetric measures combined with directed graphs and graph-based ranking algorithms such as the TextRank are likely to give a positive answer to our hypothesis about the degree of generality of terms. Moreover, we propose an unsupervised methodology for acquiring general-specific noun relations. However, it is clear that deep evaluation is needed.

4 Experiments and Results

Evaluation is classically a difficult task in Natural Language Processing. Human judgment or evaluation metrics are two possibilities. However, human evaluation is time-consuming and generally subjective even when strict guidelines are provided. As a consequence, in order to validate our assumptions, we propose an automatic evaluation scheme based on statistical language identification techniques (Beesley (1998)).

Evaluation Metric

To identify the language of a text, a distance between its frequency-ordered list of N-grams and language baseline frequency ordered-lists can be computed. For each N-gram in the test document, there can be a corresponding one in the current language profile it is compared to.

N-grams having the same rank in both profiles receive a zero distance. If the respective ranks for an N-gram vary, they are assigned the number of ranks between the two as shown in Figure 2. Finally all individual N-gram rank distances are added up and evaluate the distance between the sample document and the current language profile.

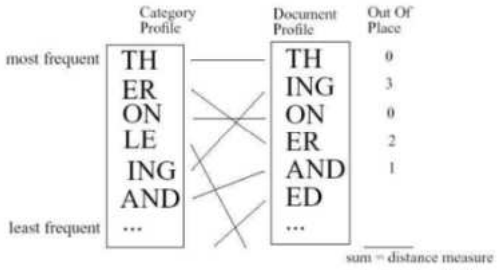

FIG. 2 – *Statistical Language Identification.*

For our purpose, we aim at calculating the distance between the lists of general-specific relations encountered by the TextRank algorithm and the original list given by WordNet. However, we face one problem. WordNet does not give an order of generality inside a synset. Then, we decided to order the words in each synset by their estimated frequency given by WordNet[2] and their frequency calculated in the web space, as our work is based on document hits. An example of both ordered lists is given in Table 2 showing different results.

WordNet Estimated Frequency		Web Estimated Frequency	
Category	Word	Category	Word
Hypernym	*statement*	Hypernym	*statement*
Synset	*answer*	Synset	*reply*
Synset	*reply*	Synset	*response*
Synset	*response*	Synset	*answer*
Hyponym	*rescript*	Hyponym	*feedback*
Hyponym	*feedback*	Hyponym	*rescript*

TAB. 2 – *Estimated Frequencies ordered lists.*

So, calculating the distance $d(.,.)$ on the lists proposed in Table 3 results in : *d(A,B)=5+1+0+2+1+1=10* and *d(A,C)=4+1+1+0+2+0=8*.

Weighted list (A)	WordNet Esti. List (B)	Web Esti. List (C)
feedback	*statement*	*statement*
statement	*answer*	*reply*
reply	*reply*	*response*
answer	*response*	*answer*
response	*rescript*	*feedback*
rescript	*feedback*	*rescript*

TAB. 3 – *Ordered lists to calculate* $d(.,.)$.

It is clear that this distance is a penalty factor which must be averaged by the length of the list. For that purpose, we propose the $matching - score(.,.)$ in Equation 10 (where $length(.)$

[2]We use WordNet 2.1.

is the number of words in a list and $n \in \mathbb{N}^+$) which aims at weighting positively the fact that two lists A and B are similar.

$$matching - score(A,B) = \begin{cases} 1 - \frac{d(A,B)}{2n^2} \; if \; length(A) = length(B) = 2n, \\ 1 - \frac{d(A,B)}{2n^2+2n} \; otherwise \end{cases} \quad (10)$$

Evaluation Scheme

In order to evaluate our methodology, we randomly extracted 115 seed synsets from which we retrieved their hypernym and hyponym synsets. For each seed synset, we then built the associated directed weighted and unweighted graphs based on the asymmetric association measures referred to in section 2[3] and ran the TextRank algorithm to produce a general-specific ordered lists of terms. For each produced list, we finally calculated their $matching - score(.,.)$ with both WordNet and Web Estimated Lists. In Table 4, we present the average results of the $matching - score(.,.)$ for the 115 synsets.

Equation	Type of Graph	Average *Matching-score* with Wordnet Estimated List	Average *Matching-score* with Web Estimated List
Braun-Blanquet	Unweighted	**51.94**	**52.83**
	Weighted	**51.94**	**52.83**
J Measure	Unweighted	47.41	48.74
	Weighted	46.76	48.93
Confidence	Unweighted	**51.94**	**52.83**
	Weighted	**51.94**	**52.83**
Laplace	Unweighted	**51.94**	**52.83**
	Weighted	**51.94**	**52.83**
Conviction	Unweighted	47.42	48.74
	Weighted	46.74	48.94
Certainly Factor	Unweighted	51.63	52.85
	Weighted	51.75	52.58
Added Value	Unweighted	51.63	52.85
	Weighted	51.77	52.58

TAB. 4 – *Average score in % for entire list comparison.*

In order to be more precise, we proposed another evaluation scheme by looking at the lists such as a sequence of three sub-lists.

In fact, we calculated the average $matching - score(.,.)$ for the three sub-lists that are contained in any general-specific list. Indeed, we can look at a list as the combination of the hypernym list, the synset list and the hyponym list. The idea is to identify differences of results in different parts of the lists (*e.g.* if hypernyms are more easily captured than hyponyms). In Table 5, we illustrate the results by representing a list of words as three sub-lists just in the case of weighted graphs as results between weighted and unweighted are negligible.

[3]The probability functions are estimated by the Maximum Likelihood Estimation (MLE).

Equation	Sub-List	Average *Matching-score* with Wordnet Estimated List	Average *Matching-score* with Web Estimated List
Braun-Blanquet	Hypernym	**68.34**	65.84
	Synset	**55.95**	54.17
	Hyponym	**56.19**	54.54
J Measure	Hypernym	61.98	60.83
	Synset	52.47	51.12
	Hyponym	52.91	54.62
Confidence	Hypernym	**68.34**	65.84
	Synset	**55.95**	54.17
	Hyponym	**56.19**	54.54
Laplace	Hypernym	**68.34**	65.84
	Synset	**55.95**	54.17
	Hyponym	**56.19**	54.54
Conviction	Hypernym	62.14	60.89
	Synset	51.75	50.62
	Hyponym	53.87	55.68
Certainly Factor	Hypernym	67.96	65.34
	Synset	56.03	54.32
	Hyponym	56.07	54.25
Added Value	Hypernym	67.32	64.70
	Synset	55.29	53.70
	Hyponym	56.55	54.52

TAB. 5 – *Average score in % for sub-list comparison.*

Discussion

Based on Table 4, the first conclusion to be drawn from our experiments is that unweighted graphs and weighted graphs perform the same way *i.e.* the importance of the graph is its topology and not its weights. In fact, the number of incoming compared to the number of outgoing edges makes the difference in the results.

The second conclusion is the fact that using any of the asymmetric measures does not drastically influence the results. This is a clear consequence of our first conclusion, as the topology is more important than the values given to the edges and most of the asymmetric association measures are able to catch the correct directions of the edges. In fact, the simplest measure, the Confidence, performs best with a $matching - score(.,.)$ of 52.83% which means that the list obtained with our methodology overlaps more than a half the Web Estimated List.

An important remark needs to be made at this point of our discussion. There is a large ambiguity introduced in the methodology by just looking at web counts. Indeed, when counting the occurrences of a word like *"answer"*, we count all its occurrences for all its meanings and forms. For example, based on WordNet, the word *"answer"* can be a verb with ten meanings and a noun with five meanings. Moreover, words are more frequent than others although they are not so general, unconfirming our original hypothesis. Looking at Table 3, *"feedback"* is a clear example of this statement. As we are not dealing with a single domain within which

one can expect to see the "one sense per discourse" paradigm, it is clear that the $matching - score(.,.)$ would not be as good as expected as it is clearly biased by "incorrect" counts. For that reason, we proposed to use Web Estimated Lists to evaluate the $matching - score(.,.)$. As expected, the results show improvements although negligible for most measures. Lately, with (Kilgarriff (2007)), there has been great discussion whether one should use web counts instead of corpus counts to estimate word frequencies. In our study, we clearly see that web counts show evident problems, like the ones mentioned by Kilgarriff (2007). However, they cannot be discarded so easily. In particular, we aim at looking at web counts in web directories that would act as specific domains and would reduce the space for ambiguity. Of course, experiments with well-known corpora will also have to be made to understand better this phenomenon.

Finally, Table 5 shows very interesting results. On average, the $matching - score(.,.)$ works better to discover hypernyms (68.34%) and hyponyms (56.19%). The worst results are shown for the words in the seed synsets (55.95%). These results are encouraging as defining an order in the seed synset is a difficult task or even impossible. Indeed, it would mean that one is capable of giving a fine-grained level of generalization-specification between synonyms. For example, is it possible to clearly define a level of generalization between the *"answer"* and *"response"* ? It does not seem so. However, with our algorithm, each word has a specific order, even within the seed synset. Based on these results, we clearly believe that future research will lead to improved results.

5 Conclusions and Future Work

In this paper, we proposed a new methodology based on directed weighted/unweighted graphs and the TextRank algorithm to automatically induce general-specific noun relations from web corpora frequency counts. To our knowledge, such an unsupervised experiment has never been attempted so far. In order to evaluate our results, we proposed a new evaluation metric, the $matching - score(.,.)$, based on an adaptation of the statistical language identification model. The results obtained by using seven asymmetric association measures based on web frequency counts showed promising results reaching levels of $matching - score(.,.)$ of 68.34% for hypernyms detection.

Nevertheless, future work is needed. First, based on the statements of Kilgarriff (2007), we aim at reproducing our experiments based on web directories and reference corpora such as the Reuters to avoid large scale ambiguity from web counts. Second, the $matching - score(.,.)$ generally penalizes the overall results as hypernyms and hyponyms are not so much represented in terms of words than the seed synset. As a consequence, we aim at gathering more hypernyms and hyponyms of the seed synset to provide a more representative test set. Third, we want to propose another way of evaluating the results. Instead of applying the $matching - score(.,.)$ function, we could run clustering algorithms to reproduce the three original sub-lists of words. So far, our experiments with the K-means and the PAM algorithm have not been fruitful but we aim at using more sophisticated algorithms such as the PoBOC or the QT-Clustering to perform this task. Finally, we want to study the topologies of the built graphs to understand if simplifications can be made based on their topologies as it is done in (Patil et Brazdil (2007)).

Références

Beesley, K. (1998). Language identifier : A computer program for automatic natural-language identification on on-line text.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In *Conference of the Association for Computational Linguistics (ACL 1999)*.

Dias, G., C. Santos, et G. Cleuziou (2006). Automatic knowledge representation using a graph-based algorithm for language-independent lexical chaining. In *Proceedings of the Workshop on Information Extraction Beyond The Document (COLING/ACL 2006)*, Sydney, Australia, pp. 36–47.

Grefenstette, G. (1994). *Explorations in automatic thesaurus discovery*. Kluwer Academic Pub.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics*, Morristown, NJ, USA, pp. 539–545. Association for Computational Linguistics.

Kilgarriff, A. (2007). Googleology is bad science. *Comput. Linguist. 33*(1), 147–151.

Michelbacher, L., S. Evert, et H. Schütze (2007). Asymmetric association measures. In *Recent Advances in Natural Language Processing (RANLP 2007)*.

Mihalcea, R. et P. Tarau (2004). TextRank : Bringing order into texts. In *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*.

Patil, K. et P. Brazdil (2007). Sumgraph : Text summarization using centrality in the pathfinder network. *International Journal on Computer Science and Information Systems 2*(1), 18–32.

Pecina, P. et P. Schlesinger (2006). Combining association measures for collocation extraction. In *Proceedings of the COLING/ACL on Main conference poster sessions*, Morristown, NJ, USA, pp. 651–658. Association for Computational Linguistics.

Riloff, E. (1993). Automatically constructing a dictionary for information extraction tasks. In *National Conference on Artificial Intelligence*, pp. 811–816.

Snow, R., D. Jurafsky, et A. Y. Ng (2006). Semantic taxonomy induction from heterogenous evidence. In *ACL '06 : Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, Morristown, NJ, USA, pp. 801–808. Association for Computational Linguistics.

Stevenson, M. et M. A. Greenwood (2006). Comparing information extraction pattern models. In *Proceedings of the Workshop on Information Extraction Beyond The Document (COLING/ACL 2006)*, Sydney, Australia, pp. 29–35.

Tan, P.-N., V. Kumar, et J. Srivastava (2004). Selecting the right objective measure for association analysis. *Information Systems 29*(4), 293–313.

Summary

In this paper, we propose a new methodology based on directed weighted graphs and the TextRank algorithm (Mihalcea et Tarau (2004)) to automatically induce general-specific noun relations from web corpora frequency counts. Different asymmetric association measures are implemented to build the graphs upon which the TextRank algorithm is applied and produces an ordered list of nouns from the most general to the most specific. Experiments are conducted based on the WordNet noun hierarchy with a quantitative evaluation.