Evaluating co-occurrence order for automatic thesaurus construction

Roger L. Granada, Renata Vieira, Vera Lúcia Strube de Lima
Pontifical Catholic University of Rio Grande do Sul (PUCRS)
Ipiranga Av., 6681. FACIN. CEP 90169-900. Porto Alegre, Brazil.
roger.granada@acad.pucrs.br, {renata.vieira, vera.strube}@pucrs.br

Abstract

Identifying the best semantically similar terms in an automatic thesaurus construction task is still an open problem in natural language processing. Many methods have been proposed to solve this problem. In this work we present a comparison between three corpus based methods for automatically build thesaurus. These methods look for related terms using the relations between terms and they differ among themselves in the co-occurrences order of these relations. The evaluation process was carried out by domain specialists who evaluated the related terms generated by each method in an experiment applied to the data privacy domain.

Keywords: Syntactic dependency, Vector Space Model, Semantic similarity, Automatic Thesaurus construction

1. Introduction

Thesauri have supported the discovery of domain knowledge. Besides, the association of a thesaurus to a domain ontology serves as the double purpose of presenting a better understanding of the concepts in the domain, as well as suggesting alternative words and phrases that may be used to describe each concept. These words and phrases must be semantically similar to the concept of the ontology to better describe this concept.

To better understand what semantic similarity is, Lemaire and Denhière in [23] point out that it could be viewed as an association of two terms, that is, the mental activation of one term when another term is presented. This type of semantic association is the same that a thesaurus uses to relate terms. According to the definition of Kilgarriff and Yallop [21], a thesaurus can be seen as a resource in which words with similar meanings are grouped together.

Initially thesauri were built manually as Roget's thesaurus [28] and later the lexical database WordNet [10], however this kind of semantic resource has some drawbacks as missing domain-specific senses or new words. Also, manually building a thesaurus is an expensive and time consuming task. Due to these drawbacks, many efforts to the automatic thesaurus construction have been presented, for instance [8, 16, 19, 20, 36].

Moreover, automatic thesaurus construction has been used in many applications. For example, in query expansion [6] associating related terms to each query term in an information retrieval system. It is also used in ontology enrichment, as presented by Castilho *et al.* [5] where related terms are associated to seeds of an ontology for the mapping of heterogeneous resources in the data privacy domain. Besides, it is also used to help students in a new language understanding [19], in document classification tasks [36], document clustering [25], and recommender systems [1].

According to Church and Hanks in [7], it is a common practice in linguistics to classify words not only on the basis of their meanings but also on the basis of their co-occurrence with other words. These co-occurrences can be within a certain limited distance in the context (using a window to generate these co-occurrences) or within syntactic relations (for instance, verb-object).

Turney and Pantel in [34], state that a manner to identify semantic similarity between terms is using the Vector Space Models (VSM) proposed by Salton *et al.* [30]. Initially the VSM should represent documents as points in a vector space. Each document is represented by a vector containing the terms of the document. Documents semantically similar must be close together and far apart when semantically distant. Later, Deerwester *et al.* [9] observed that this model could not only measure the documents similarity but also terms similarity, shifting the view of document vectors by a view of term vectors.

Although many methods have been presented to identify the semantic similarity between terms, it is still hard to identify which is the best approach to use. In this work we evaluate three approaches to the automatic thesaurus construction task based on the level of co-occurrence of the terms, and using VSM to compute the similarity distance.

Domain specialists evaluated the thesauri verifying the quality of the related terms generated by each approach. The main purpose of this evaluation is identifying which

IEEE IRI 2012, August 8-10, 2012, Las Vegas, Nevada, USA
978-1-4673-2284-3/12/$31.00 ©2012 IEEE

is the best method for the automatic thesaurus construction task. Knowing the best method for the automatic thesaurus construction, related terms can be generated to each concept of a domain ontology automating the process of ontology enrichment based on corpus. This enrichment using a thesaurus allows the mapping of semantic resources as presented in Castilho *et al*. [5].

The rest of the paper is organized as follows. Section 2 discusses related works. Section 3 gives a description of the resources used in our experiment, a freely available corpus, and a data privacy ontology which served as seeds to the thesauri generation. Section 4 describes the methods used for building thesauri according to their co-occurrence level. The evaluation process is presented in Section 5. Finally, the concluding remarks of the paper are presented in Section 6.

2. Related work

In this work, we compare three methods for automatic thesaurus construction task. They differ among themselves in the terms co-occurrence level considered in the extraction task. Gamallo and Bordag in [14] also compare methods of automatic thesaurus construction. In their work, the authors evaluate the usefulness of Singular Value Decomposition (SVD) to the similarity extraction task using comparable corpora. In that work, the authors argue that methods based on SVD are much less precise than other word space models for the task of extracting translation equivalents from comparable corpora. Besides that, Gamallo and Bordag discuss the computational efficiency of applying SVD.

The proponents of SVD, as Rapp in [27], say that a matrix reduced by SVD has the advantage that all subsequent similarity computations are much faster, since the final matrix representation is reduced from N original dimensions to $k < N$ dimensions. On the other hand, Gamallo and Bordag claim that an efficient data representation can easily outperform the computational efficiency obtained by SVD application. Also, they prove that the reduced matrix needs more memory space to be stored than other data structures, such as hash tables. To compare the effectiveness of applying SVD, Gamallo and Bordag evaluate it automatically using a parallel corpus, comparing similar terms extracted by second-order and third-order co-occurrences methods.

Gamallo and Bordag also verify the computational efficiency of SVD. Even the computational efficiency is an important issue to discuss when applying SVD, we believe that it was already fully discussed in that work. Unlike Gamallo and Bordag, in our work we added the first-order co-occurrence to be evaluated and instead of an automatic evaluation, we used domain specialists to verify the quality of the related terms generated by each thesaurus.

Other works compare automatic thesaurus construction methods on the basis of results obtained. Usually when a new method is proposed, the evaluation of this method is performed by using a gold standard corpus. The Test of English as a Foreign Language (TOEFL) multiple-choice synonym questions has been used in many works to evaluate their new methods. For instance, the work of Rapp [27] achieved 92.5 of precision in the task of identifying synonyms. Other proposed methods like Baroni and Lenci [2] achieved 91.3 of precision in the same task.

A more detailed table containing published results using TOEFL synonyms questions can be seen in ACLWiki[1]. Using this table, when a new method of automatic thesaurus construction is proposed, the results obtained by this new method can be compared with other methods already proposed. This type of evaluation has the drawback of using only one gold standard. Thus, if a brand new method is presented using another resource in its evaluation, it would be impractical to compare the results with methods which used the TOEFL multiple-choice synonym questions as a gold standard.

Another drawback of using TOEFL multiple-choice synonym questions as a gold standard is that a thesaurus can be designed to have other semantic related terms associated to its seeds besides synonyms. According to Budanitsky and Hirst in [4], two terms are semantically related if they have any kind of semantic relation. Thus, the semantically related terms in a thesaurus consider also hyperonym, hyponym or even antonym relations. As the TOEFL multiple-choice synonym questions is constructed using only synonym relations, the evaluation could not reflect the proper construction of the thesaurus.

This drawback can be overcome by the use of other resources, such as manually built gold standard thesaurus. Grefenstette in [16] used the Webster"s 7th Dictionary to evaluate the quality of the generated terms, Yang and Powers in [36] used a similarity measure based on WordNet, and in [37] a similarity measure based on Roget's thesaurus to evaluate the terms of a generated thesaurus. These types of evaluations allow measuring the degree of similarity between two terms covered by the gold standard. On the other hand, this type of evaluation does not cover all English terms, mainly domain terms which are very specific.

Our evaluation pays attention in the quality of the similarity extraction and unlike the related works presented in this paper, the evaluation was carried out by domain specialists, instead of using gold standards. The

[1] http://aclweb.org/aclwiki/

authors believe that this kind of evaluation overcome some of the drawbacks presented in the related works.

3. Resources

This section presents the resources used in this paper to compare methods of automatic thesaurus construction, a domain corpus and an ontology as follows.

3.1. Corpus

A domain corpus is used as a resource in which the methods for the thesaurus construction are applied. In a domain corpus we can find related terms according to each applied method. This resource can also be used in the evaluation process where the evaluator can see an excerpt of the text in which the related terms are used.

This corpus is an important source of knowledge on the domain of data privacy for projects involving the exchange of information. It was manually gathered from documents available in internet and all of them have public access. The whole corpus is composed of 100 documents containing legislation of various countries, and software industry guidelines for the data privacy domain. All documents are written in English and the ones which are from non-English native countries, the official translation to English were gathered. The documents can be accessed in an integrated visualization tool developed as part of our project[2].

3.2. Seeds to thesaurus

The construction of a thesaurus is primarily based on initial seed terms to which new related terms are grouped. These seeds can be simply the whole set of words in the corpus, building a relational graph where words are represented as nodes and relationships as edges [24]. Also, instead of taking all words as seeds to the thesaurus we can take only part of it, such as nouns and verbs [37], only nouns [38], or even terms of an ontology [5], which adds domain semantic relevance to the resulting thesaurus.

In this work we use the labels of the concepts of a domain ontology as seeds to the thesauri. This ontology was manually built based on the study of a data privacy accountability system constructed to verify privacy accountability compliance in projects [26]. The ontology was initially developed to support the domain of Data Privacy Regulation and Management, and aimed at expanding the accountability system. It takes into account a database of questions used to assess privacy risks [26] and the relevant terminology to assist the identification of laws and regulations involving exchange of information [3]. At the moment the ontology contain 248 concepts and can be viewed in our Visualization tool[3].

4. Automatic Thesaurus Construction

In this work we evaluate three automatic thesaurus construction methods considering different levels of the terms co-occurrence, that is, the degree of relations between words in the text. These relations are grouped in first, second, and third-order (or higher) co-occurrence, as explained below.

4.1. First-order co-occurrences

First-order co-occurrence, also called direct co-occurrence, occurs when two terms appear in identical contexts [34]. The first-order co-occurrence is based on the J.R. Firth saying "*You shall know a word by the company it keeps*." [11]. For instance, on the one hand, *bank* co-occurs with words and expressions such as *money*, *notes*, *loan*, *account*, *investment*, *clerk*, *official*, *manager*, and so forth. On the other hand, we find *bank* co-occurring with *river*, *swim*, *boat*, and *east* depending on which meaning the word presents [17].

A thesaurus generated using first-order co-occurrences uses only statistical models, for instance, using a context-window [7, 20], clustering words [8], or even web-based [33]. In this work we applied the approach used by Kaji *et al*. [20] to build a thesaurus using first-order co-occurrences. This approach is composed by the following steps:

1. Tagging the corpus;
2. Extracting co-occurrence in a window;
3. Analyzing correlation.

The automatic thesaurus construction starts by tagging the corpus. We used the Stanford Log-linear Part-Of-Speech Tagger[4] since it is a tagger with high accuracy (97.24% on the Penn Treebank WSJ) [32]. This type of annotation is important because it allows us to identify the nouns, since our thesaurus is composed only by them.

In the second step a context length is defined, also called the size of the window, that is, the number of words surrounding the headword. The size of this window should accommodate a few sentences and not having a too large computational load. Then all the words in the context of every occurrence of a word *w* inside a bag are collected. That *bag of words* will represent the meaning of *w* [29].

As proposed by Kaji *et al*., this size should be between 20 and 50, thus we choose a window composed of 30

[2] http://www.cpca.pucrs.br/VisualizationTool/Resource/Corpus.html

[3] http://www.cpca.pucrs.br/VisualizationTool-TAT-evaluator3/Ontology.jsp

[4] http://nlp.stanford.edu/software/tagger.shtml

terms. To extract the terms in a window, we used the Ngram Statistical Package (NSP) [5]. This tool allows extracting co-occurrences of terms by choosing the size of the window.

The last step is the correlation analysis using Pointwise Mutual Information (PMI) [7] to compute the similarity between pairs of terms. As explained by Turney in [33], PMI provides a way to measure the degree of co-occurrence of two words by comparing the number of co-occurrences to the number of individual occurrences. This value is maximal when all occurrences are co-occurrences.

To compute the PMI we also used NSP. The pairs of terms were composed of a seed of the thesaurus and a related word (a term extracted in the window). The result is a list composed of seeds and the related terms ranked by its PMI value.

4.2. Second-order co-occurrences

Lemaire and Denhière in [23] define that two words are associated by means of second-order co-occurrence if they share at least one word context. This view is based on the Harris' distributional hypothesis [18] which states that words that occur in the same contexts tend to be similar.

The approach used in this work is based on the approach of Grefenstette in [16], that is, each word that is syntactically related to a noun is part of its syntactic context. Thus, each adjective, noun or verb that shares a syntactic relation with a noun is recorded as a context of this noun. The approach used in this work is composed by the following steps:

1. Parsing the corpus;
2. Extracting the syntactic contexts;
3. Computing similarity between pairs of terms.

The process starts by parsing the corpus to get the syntactic annotation of each phrase as well as the part-of-speech tag for each term. To parse the corpus we used the Stanford Parser [6], obtaining XML files containing the annotated documents. From these documents we extract the syntactic contexts for each noun as a triple <*relation, noun, relation_term*>. For example, in the phrase "*The privacy act regulates privacy.*" We initially extract the relations <*NN, act, privacy*>, <*SUBJ, privacy_act, regulates*>, and <*DOBJ, privacy, regulates*>. Thus, the first triple contains a relation between two nouns *NN*, where the noun *act* is modified by the noun *privacy*. The second triple contains a relation *SUBJ* indicating that the noun phrase *privacy_act* plays the role of a subject of the verb *regulates*.

To increase the number of syntactic contexts, we also consider the co-requirement phenomenon, which is not considered in Grefenstette's work. According to Gamallo *et al*. in [13] the co-requirement, also called co-compositionality in [14], occurs when two words impose linguistic requirements on each other. Thus, in a Head-Dependent syntactic dependency not only the Head imposes a linguistic constraint in the Dependent, but also the Dependent imposes linguistic requirements on the Head in return. This assumption believes that the higher is the number of syntactic contexts to describe a word behavior, the higher is the reliability of the results when comparing it with other words. Using this assumption in our previous example, the context <*NN, privacy, act*> is created, based on the context <*NN, act, privacy*>.

Another difference with the Grefenstette's work is the assumption that nouns play different roles when they are related to a verb as subject or, as direct or indirect object. This is the same assumption made by Gasperin and de Lima in [15]. Making this assumption we extracted syntactic contexts as presented in the example above.

The syntactic contexts extraction builds a VSM where the rows contain the nouns, the columns contain the terms of each related syntactic context, and each cell represents the frequency of a noun and its syntactic context.

The last step consists of calculating the similarity between row vectors of the space. Similar row vectors in this word–context space indicate similar word meanings. A module of Lingua Toolkit package [7] was used to calculate the similarity between these row vectors. In this module, we choose the weighted Jaccard measure [31], as Grefenstette did in his work. The result was a list of nouns and their related terms ranked by the similarity with the noun.

4.3. Third-order (or higher) co-occurrence

Gamallo and Bordag in [14] explain third-order or higher co-occurrences as co-occurrences between words that do not co-occur in the corpus with the same words (or lexical-syntactic contexts) but between words that can be related through further indirect co-occurrences. This type of co-occurrence can be obtained by means of applying SVD methods. Thus, SVD methods try to represent a more abstract and generic word space which tries to capture higher-order associations by inducing a latent (hidden) structure that does not rely on word co-occurrences attested in the corpus.

SVD methods are based on linear algebra and use mathematical operation on a term-document or term-context matrix. Deerwester *et al*. in [9] used SVD as the main application to Information Retrieval, calling it as

[5] http://www.d.umn.edu/~tpederse/nsp.html

[6] http://nlp.stanford.edu/software/lex-parser.shtml

[7] http://gramatica.usc.es/~gamallo/thesaurus/index.htm

Latent Semantic Indexing (LSI), but also mentioned that truncated SVD can also be applied to word similarity. The approach pointed out by Deerwester *et al*. was evaluated by Landauer and Dumais in [22]. The latter applied SVD to word similarity, calling it as Latent Semantic Analysis (LSA), on the TOEFL multiple-choice synonym questions achieving a human-level performance.

According to Turney in [33], SVD methods decompose a matrix X into a product of three matrices $U\Sigma V^T$, where U and V are in column orthonormal form and Σ is a diagonal matrix of singular values. Besides, V^T is the transpose matrix of V, that is, V is reflected over its main diagonal.

If X is of rank r, then Σ is also of rank r. Let Σ_k, where $k < r$, be the matrix produced by removing from Σ the r - k columns and rows with the smallest singular values, and let U_k and V_k be the matrices produced by removing the corresponding columns from U and V. The matrix $U_k\Sigma_kV_k^T$ is the matrix of rank k that best approximates the original matrix X. Limiting the number of latent dimensions ($k < r$) forces a greater correspondence between words and contexts that do not appear in the corpus. This new matrix can be viewed as a "smoothed" or "compressed" version of the original matrix X.

In this work, the authors follow the steps used by Yang and Powers in [37], applying SVD over three different context-matrices. The whole process is composed by the follow steps:

1. Parsing the corpus;
2. Extracting syntactic relations;
3. Applying Singular Value Decomposition;
4. Computing similarity between pair of terms.

As the second-order co-occurrence, the first step is to parse the whole corpus using Stanford Parser. After parsing the corpus we extract the syntactic relations, building three matrices containing different syntactic contexts. The first one, called matrix *AN*, contain relations between nouns and nouns, and between nouns and adjectives. The second one, called matrix *SV*, contain relations between nouns and verbs, when the nouns play the role as subjects. The last one, called matrix *VO*, contain relations between nouns and verbs, when the nouns play the role as direct objects or indirect objects. Yang and Powers also build a matrix called *RV*, which contain relations between verbs, but as the intention of this work is find relations to nouns, we decided not to build this matrix.

After building the matrices, the value of the frequency of each context *freq(Xi,j)* in each matrix is changed into its information form, using *log(freq(Xi,j)+1),* retaining sparsity (0→0) [22]. To apply the SVD in each matrix we used a Python module called SparseSVD[8] that wraps SVDLIBC, a library for sparse Singular Value Decomposition. As Yang and Powers, we also established 250 as a fixed size of the compressed semantic space since it comes from the idea that among the singular values, the first 20 components account for around 50% of the variance, and the first 250 components for over 75% [37]. The results are smoothed matrices AN_{250}, SV_{250}, and VO_{250}.

The last step consists of applying a similarity measure on the rows of the smoothed matrices, calculating the relation degree between nouns. As Young and Powers, we applied the Cosine similarity measure over the matrices. This similarity measure is also a module of Lingua Toolkit. The result is a list of nouns and their related terms ranked by the similarity with the noun.

5. Evaluation

As discussed in Section 2, many ways to evaluate thesaurus have been proposed. Each way has its strengths and weaknesses. In this work, we do a manual evaluation of the generated thesauri. This type of evaluation considers the human judgment of each relation found between seed and related term, and it is not restricted to synonyms.

The evaluation was performed by three domain specialists. The first one is a researcher in a systems security lab, focused on working to create trustworthy information system environments. The second one is a technical leader in secure products development. The last one is a technology strategist that holds degrees in Law and in Computer Science and is currently pursuing a Masters degree in Corporate Governance Law.

The first step of the evaluation process consists of choosing a set of seeds to be evaluated. As a manually evaluation of a thesaurus is very time consuming to our set of seeds, we reduced this set to 9 seeds. For each seed the top 10 most similar related terms for each method were chosen to be part of the evaluation, resulting in a total of 251 relations, since some of them appeared in more than one list.

The domain specialists decided which seeds should compose the reduced evaluation set. This decision was based on their judgment about the importance of the term to the domain. The set of seeds consists of the terms: *children*, *consent*, *customer*, *data subject*, *marketing*, *notice*, *personal data*, *personal information*, and *regulation*.

After that, the relations between seeds and generated related terms produced by the three methods were evaluated. Using the evaluation interface[9], the evaluators

[8] http://pypi.python.org/pypi/sparsesvd

[9] http://www.cpca.pucrs.br/VisualizationTool-TAT-evaluator3/Thesaurus.jsp

can point out the similarity of the terms and rank the similar ones according to other similar terms found for that seed. The interface also contains a concordancer, helping domain specialists to understand the context of the related term. At any time, the evaluator may select a term of the thesaurus and choose to view this term where it appears in the corpus, using the concordancer.

5.1. Results

The first analyze verify the number of related terms assigned as similar by the domain specialists for each method. To compute the number of similar related terms we sum the number of related terms assigned as similar. Table 1 presents the number of related terms identified as similar by the domain specialists. Thus, for example, the seed *children* has a total of 12 related terms generated by the thesaurus using the first-order method, considering the judgment of all three judges.

Table 1. Number of related terms identified as similar by the domain specialists to each method

Seed	1Order	2Order	3Order
children	12	12	15
consent	15	18	21
customer	11	23	19
data subject	12	26	16
marketing	16	16	7
notice	17	13	21
personal data	18	26	12
personal information	6	23	11
regulation	18	24	18
Total:	125	181	140

In this table, the second-order method has a higher score than the other methods, having almost all seeds the largest number of related terms.

Considering the similarity ranking assigned by the domain specialists, we observe the distribution of the related terms to each position of the thesauri. Figure 1 presents this distribution where the horizontal axis represents the position of the term in the ranking, in which the most similar terms are classified in the first positions of the rank. The vertical axis represents the sum of terms obtained by each thesaurus to each position ranked by domain specialists. For example, the position 1 of the ranking has 5 related terms generated by the method of first-order co-occurrence, 16 related terms generated by the second-order co-occurrence, and 9 related terms generated by the third-order (or higher) co-occurrence.

In this experiment the thesaurus generated by the second-order co-occurrence method has more similar related terms in the first positions of the rank. It means that besides this method generates the largest number of terms, it also generates the most similar ones. The method using third-order (or higher) co-occurrence also has good results in the first positions of the ranking, decreasing the efficiency around the tenth position. The method using first-order co-occurrence has the worst results in the first positions of the ranking, meaning that this thesaurus doesn’t find the best related terms.

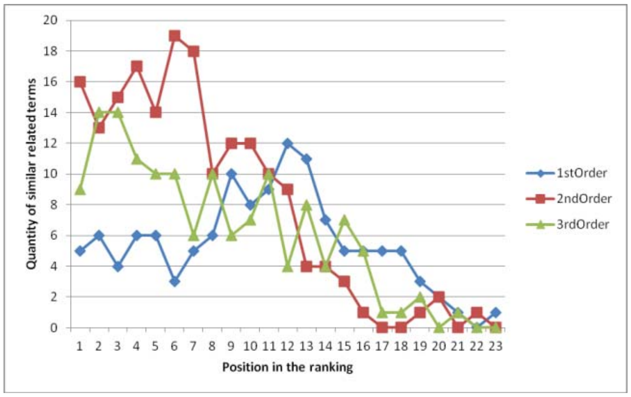

Figure 1. Distribution of related terms in each position of the thesauri

Figure 2 presents a cumulative chart of the number of similar related terms, that is, the number of related terms increases according to the position in the ranking. Thus, the position in the ranking do not consider only the number of similar related terms in that position, but also the accumulated quantity of similar related terms assigned before it. For example, considering the second-order co-occurrence method in the 3th position, the chart presents 44 similar related terms. These 44 similar related terms represent the sum of the 16 terms in the 1st position, the 13 in the 2nd position, and the 15 in the 3th position.

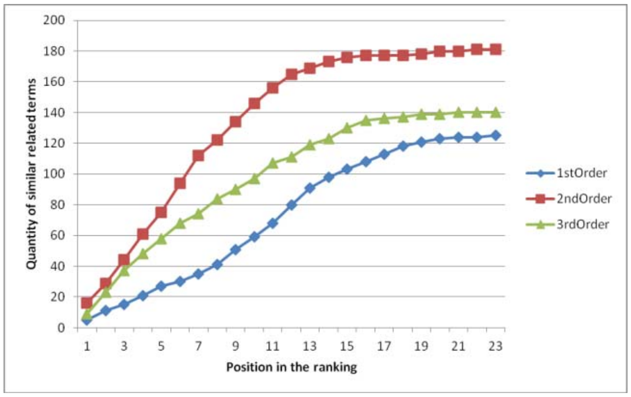

Figure 2. Distribution of the cumulated related terms in each position of the thesauri

In the chart presented in Figure 2 the thesaurus generated by the second-order co-occurrence method overcomes the other two methods in the number of similar terms to each position of the ranking. Also, the first-order co-occurrence method has the worst results when comparing with the other ones.

To verify the reliability of the agreement among the evaluators we used the Fleiss' Kappa measure [12]. The result of this measure was $\kappa = 0.33$, which corresponds to a fair agreement. Intending to get a more robust reference, we consider the term similar only when at least two evaluators agree with its similarity. This analysis can help to assess the quality of the terms generated by each method. This is based on the idea that the similarity is more objective when recognized by more than one evaluator.

Table 2 presents the number of terms generated by each method of automatic thesaurus construction according to at least two evaluators. In this table, the second-order co-occurrence method keeps generating the major quantity of similar related terms. Also, the first-order method held the lowest number of related terms assigned as similar by the domain specialists.

Table 2. Number of related terms identified as similar by at least two domain specialists to each method

Seed	1Order	2Order	3Order
children	3	3	5
consent	5	5	7
customer	3	7	7
data subject	4	10	4
marketing	6	6	2
notice	6	3	6
personal data	7	9	3
personal information	2	8	3
regulation	6	9	6
Total:	42	60	43

Analyzing the results, the statement of Gamallo and Bordag in [14] which says "the benefits of finding third-order or more co-occurrences using SVD are overruled by the decrease of second-order similarity, whose contribution for the overall similarity is crucial", is confirmed. As an example we can cite the relation between the seed *personal data* and the related term *national identification number*, which has only one syntactic context in common. Using the second-order co-occurrence method, this relation did not appear in a list containing the first 100 similar related terms. After applying SVD the term *national identification number* appeared in the first 10 most related terms. On the other hand, terms like *health information* and *PII* (acronym for *Personal Identifiable Information*), which were identified by domain specialists as more similar than *national identification number*, came out of the list containing the first 10 most similar terms.

6. Concluding remarks

This work proposed to study three different methods for the automatic thesaurus construction task based on the relations of co-occurrence between terms. The method with higher score will be used to generate related terms to a domain ontology, automating the process of ontology enrichment.

Using a domain corpus and the concepts of a data privacy ontology as seeds to the thesauri, we generated three thesaurus using first, second and third (or higher) order co-occurrence methods. Three domain specialists evaluated the terms generated by each thesauri assigning them as similar or not similar and ranking the most similar terms to the seed. Analyzing the results, the second-order co-occurrence method has a higher score when compared with the other two methods. It gets a higher score in quantity of related terms assign as similar by the domain specialists as well as the best ranked related terms as similar to the seeds.

Finally, analyzing the related terms generated by the second and the third (or higher) order methods, we confirm the statement of Gamallo and Bordag in [14] which says that when applying SVD the second-order similarity decrease to the increasing of the third (or higher) order similarity.

Future research will focus on using WordNet to refine the meaning of each concept of the ontology. Using WordNet will allow us to discover the type of relation (hyperonym, synonym, meronym, so on) between an ontology concept and a related term.

7. Acknowledgement

This work is partially supported by CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil.

8. References

[1] V.M.P. Anick, V. Murthi, and S. Sebastian, "Similar term discovery using web search", In: Proceedings of LREC'08, 2008, pp. 1209-1213.

[2] M. Baroni, and A. Lenci, "Concepts and properties in word space", *Italian Journal of Linguistics*, 20(1), 2008, pp. 55–88.

[3] M. Bruckschen, C. Northfleet, D.M. Silva, P. Bridi, R. Granada, R. Vieira, P, Rao, and T. Sander, "Named Entity recognition in the legal domain for ontology population", In: LREC'10, 2010, pp. 16-21.

[4] A. Budanitsky, and G. Hirst, "Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures", In Proceedings of the Workshop on WordNet and Other Lexical Resources, Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2001), 2001, pp. 29–24.

[5] F.M.B.M. Castilho, R. Granada, R. Vieira, T. Sander, and P. Rao, “Ontology Enrichment Based on the Mapping of Knowledge Resources for Data Privacy Management”, In: ONTOBRAS-MOST 2011, 2011, pp. 85-96.

[6] L. Chen, and S. Chen, “A New Approach for Automatic Thesaurus Construction and Query Expansion for Document Retrieval”, *International Journal of Information and Management Sciences*, vol. 18-4, 2007, pp. 299-315.

[7] K.W. Church, and P. Hanks, “Word association norms, mutual information, and lexicography”, *Computational Linguistics*, 16(1), 1990, pp. 22-29.

[8] C.J. Crouch, and B. Yang, “Experiments in automatic statistical thesaurus construction”, In: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, 1992, pp. 77–88.

[9] S. Deerwester, S.T. Dumais, G.W. Furmas, T.K. Landauer, and R. Harshman, “Indexing by Latent Semantic Analysis”, *Journal of the American Society for Information Science*, 41(6), 1990, pp. 391-407.

[10] C. Felbaum. “Wordnet, an Electronic Lexical Database”. Cambridge: MIT Press, 1998, 445p.

[11] J.R. Firth, “A synopsis of linguistic theory 1930-1955”, *Studies in Linguistic Analysis*, 1957, pp. 1-32.

[12] J.L. Fleiss, “Measuring nominal scale agreement among many raters”, *Psychological Bulletin*, 76(5), 1971, pp. 378–382.

[13] P. Gamallo, A. Agustini, and G. Lopes, “Clustering syntactic positions with similar semantic requirements”, *Computational Linguistics*, 31(1), 2005, pp. 107–146.

[14] P. Gamallo, and S. Bordag, “Is singular value decomposition useful for word similarity extraction?” *Language Resources and Evaluation*, 45(2), 2011, pp. 95-119.

[15] C.V. Gasperin, V.L.S. de Lima, “Experiments on extracting semantic relations from syntactic relations”, In: Proceedings of the 4th international conference on Computational linguistics and intelligent text processing, 2003, pp. 314-324.

[16] G. Grefenstette, *Explorations in automatic thesaurus discovery*, Kluwer Academic Publishers Norwell, 1994, 306 p.

[17] P. Hanks, “Definitions and Explanations”, Looking Up: An Account of the COBUILD Project in Lexical Computing, Collins, London and Glasgow, 1987, 192 p.

[18] Z.S. Harris, “Distributional structure”, *Words*, 10(23), 1954, pp. 146-162.

[19] M. Heilman, and M. Eskenazi, “Application of Automatic Thesaurus Extraction for Computer Generation of Vocabulary Questions”, In: Proceedings of the SLaTE Workshop on Speech and Language Technology in Education, 2007, pp. 65-68.

[20] H. Kaji, Y. Morimoto, T. Aizono, and N. Yamasaki, “Corpus dependent association thesauri for information retrieval”, In: Proceedings of the 18th Conference on Computational Linguistics, 2000, pp 404-410.

[21] A. Kilgarriff, and C. Yallop, “What's in a thesaurus”, In: Proceedings of the Second LREC, 2000, pp. 1371-1379.

[22] T.K. Landauer, and S.T. Dumais, “A solution to Plato’s problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge”, *Psychological Review*, 104(2), 1997, pp. 211-240.

[23] B. Lemaire, and G. Denhière, “Effects of high-order co-occurrences on word semantic similarity”, *Current Psychology Letters*, 18(1), 2006, pp. 1-12.

[24] L. Michelbacher, F. Laws, B. Dorow, U. Heid, and H. Schütze, “Building a cross-lingual relatedness thesaurus using a graph similarity measure”, In: Proceedings of LREC’10, 2010.

[25] P. Pantel, and D. Lin, “Discovering word senses from text”, In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002, pp. 613–619.

[26] S. Pearson, P. Rao, T. Sander, A. Parry, A. Paull, S. Patruni, V. Dandamudi-Ratnakar, and P. Sharma, “Scalable, Accountable Privacy Management for Large Organizations”, In: Proceedings of EDOCW 2009, 2009, pp. 168-175.

[27] R. Rapp, “A freely available automatically generated thesaurus of related words”, In: Proceedings of LREC’04, 2004, pp. 395-398.

[28] P. M. Roget, B. Sears, *Thesaurus of English words: so classified and arranged as to facilitate the expression of ideas and assist in literary composition,* Gould and Lincol, 1868, 468p.

[29] M. Ruiz-Casado, E. Alfonseca, and P. Castells, “Using context-window overlapping in synonym discovery and ontology extension”, In: Proceedings of RANLP-2005, 2005.

[30] G. Salton, A. Wong, and C.S. Yang, “A vector space model for automatic indexing”, *Communications of the ACM*, 18(11), 1975, pp. 613–620.

[31] T.T. Tanimoto, “An elementary mathematical theory of classification”, Technical report, IBM Research, 1958, 238p.

[32] K. Toutanova, D. Klein, C. Manning, and Y. Singer, “Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network”, In: Proceedings of HLT-NAACL 2003, 2003, pp. 252-259.

[33] P.D. Turney, “Mining the Web for synonyms: PMI-IR versus LSA on TOEFL”, In: Proceedings of the Twelfth European Conference on Machine Learning - ECML-01, 2001, pp. 491-502.

[34] P.D. Turney, and P. Pantel, “From frequency to meaning”, *Journal of Artificial Intelligence Research*, 37(1), 2010, pp. 141-188.

[35] H. Xu, and B. Yu, "Automatic thesaurus construction for spam filtering using revised back propagation neural network", Expert Systems with Applications, 37(1), 2010, pp. 18-23.

[36] D. Yang, and D.M.W. Powers, “Measuring semantic similarity in the taxonomy of WordNet”, In: Proceedings of the Twenty-eighth Australasian Conference on Computer Science, 2005, pp. 315-322.

[37] D. Yang, and D.M.W. Powers, “Automatic thesaurus construction”, In: Proceedings of the 31st Australasian conference on Computer science - ACSC '08, 2008, pp. 147-156.

[38] P. Wang, J. Hu, H.J. Zeng, and Z. Chen. “Using Wikipedia knowledge to improve text classification”. Knowledge and Information Systems, 19(3), 2009, pp. 265-28.