Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Procedia - Social and Behavioral Sciences 198 (2015) 377 – 385

Procedia
Social and Behavioral Sciences

7th International Conference on Corpus Linguistics: Current Work in Corpus Linguistics: Working with Traditionally-conceived Corpora and Beyond (CILC 2015)

DEXTER: Automatic Extraction of Domain-Specific Glossaries for Language Teaching

Carlos Periñán-Pascual*, Eva M. Mestre-Mestre

Universitat Politècnica de València, Applied Linguistics Department, Paranimf, 1, 46730 Gandia (Valencia), Spain

Abstract

Many researchers emphasize the importance of corpora in the design of Language-for-Specific-Purposes courses in higher education. However, identifying those lexical units which belong to a given specific domain is often a complex task for language teachers, where simple introspection or concordance analysis does not really become effective. The goal of this paper is to describe DEXTER, an open-access platform for data mining and terminology management, whose aim is not only the search, retrieval, exploration and analysis of texts in domain-specific corpora but also the automatic extraction of specialized words from the domain.

Peer-review under responsibility of Universidad de Valladolid, Facultad de Comercio.

Keywords: automatic term extraction; corpus; glossary; statistical measure; language for specific purposes; specialized domain; DEXTER

1. Introduction

Today, there is a variety of open-source corpus analysis software, e.g. IMS Open Corpus Workbench, PhiloLogic, Poliqarp or XAIRA, among many others. These tools, most of them aimed to be used for linguistic or lexicographic research, usually integrate a set of utilities which enable users to check word frequency, concordances and collocations. However, there are not many tools available that can really meet one of the main needs of terminographers, i.e. the automatic extraction of specialized lexical units from a corpus. The goal of this paper is to outline the main features of DEXTER (Discovering and EXtracting TERminology), an online multilingual

* Corresponding author. Tel.: +34962849433.
E-mail address: jopepas3@upv.es

Peer-review under responsibility of Universidad de Valladolid, Facultad de Comercio.
doi:10.1016/j.sbspro.2015.07.457

workbench for data mining and terminology management with non-structured text-based corpora[1]. Indeed, DEXTER consists of a suite of tools with different functionalities, such as corpus compilation and management, document indexation and retrieval, query elaboration, textual exploration and terminological extraction. It should be noted that the DEXTER term extractor is provided with a language- and discourse-independent processor which is connected to language-specific resources (e.g. lexical inventories and filters) and modules (e.g. stemmers and lemmatizers), resulting in an adaptable work environment. In this context, DEXTER provides a framework which integrates the characteristics of linguist-oriented text analysis tools with the enhanced efficiency of the terminology acquisition module for terminographers and knowledge engineers. Beyond research purposes, as an automatic term extraction (ATE) system, DEXTER can also be used to support language teachers to construct their own glossaries, containing unigrams, bigrams and trigrams, from small- and medium-sized specialized corpora compiled by themselves. The remainder of this paper is organized as follows: Section 2 describes the role that glossaries play in the design of Language-for-Specific-Purposes (LSP) courses; Section 3 briefly reviews the state of the art in ATE; Section 4 outlines the main features of DEXTER by exploring not only the term extraction metric but also those capabilities of interest to the language teacher; and finally, Section 5 presents the main conclusions.

2. Automatic term extraction for specialized domains

Traditionally, there have been three main approaches to ATE, i.e. linguistic, statistic and hybrid. The linguistic approach is typically performed by means of three consecutive tasks. First, words are tagged with their part-of-speech. Second, morphosyntactic patterns are used to capture acceptable surface realizations as term candidates; for example, the following pattern has been widely used to extract noun terms in English (Justeson and Katz, 1995):

$$((Adj|Noun)+|((Adj|Noun)*(NounPrep)?)(Adj|Noun)*)Noun$$

Third, a stop list of functional and generic words is applied. Thus, a stop list can remove false candidates such as *this thing* or *some day*, although these linguistic realizations match the pattern Adj + Noun. Many of the problems related to this approach are frequently related to issues such as language dependence and time-consuming labour, apart from the fact that pre-defined linguistic patterns only serve as an indicator of unithood.

The statistical approach can be based on two different types of measures. On the one hand, lexical association measures, such as Chi-Square (Nagao, Mizutani, and Ikeda, 1976), Pointwise Mutual Information (Church and Hanks, 1990), T-score (Church et al., 1991), Dice coefficient (Smadja, 1993), Log-Likelihood Ratio (Dunning, 1994) or Jaccard similarity (Grefenstette, 1994), have been frequently used in ATE. The problem is that the above measures only serve to measure unithood, since they calculate the likelihood that two words can occur together. On the other hand, there are also some statistical measures for termhood, such as TF-IDF (Singhal, Salton, and Buckley, 1996), C-value (Frantzi and Ananiadou, 1996), Weirdness (Ahmad, Gillam, and Tostevin, 2000), Domain-Specificity (Park, Byrd, and Boguraev, 2002), and Domain-Pertinence and Domain-Consensus (Sclano and Velardi, 2007; Navigli and Velardi, 2002).

Finally, it should be noted that few ATE systems adopt a purely statistical approach. Instead, there is a general tendency to consider both linguistic and statistical properties for the identification and extraction of term candidates, where linguistic analysis is carried out before the application of statistical measures. One of the most popular examples of hybrid model is found in C-value/NC-value (Frantzi, Ananiadou, and Hideki, 2000), where the NC-Value incorporates contextual information into the C-Value, which focuses on nested terms.

DEXTER adopts a hybrid approach to ATE, where some linguistic constraints are applied before the statistical measures discover term candidates on the basis of stemmed ngrams (i.e. unigrams, bigrams and trigrams).

[1] DEXTER, which has been developed in C# with ASP.NET 4.0, is intended to be freely accessible from the FunGramKB website (www.fungramkb.com). Although only English and Spanish are currently supported in DEXTER, French, German and Italian are about to be included.

Considering that traditional linguistic patterns are quite restrictive, we chose not to perform full POS tagging. Instead, a set of shallow linguistic constraints supported by a stopword list was applied before term weighting. For example, one of these constraints served to filter out those trigrams containing a non-prepositional functional word in the mid-position. The next section presents a brief description of the statistical measure used in DEXTER.

3. An overview to DEXTER

3.1. The term extraction metric

SRC is the metric employed in DEXTER for the identification and extraction of term candidates. A detailed account of this user-adjustable composite metric is out of the scope of this paper. However, this section outlines the main components of this measure, which is grounded on the notions of salience (S), relevance (R) and cohesion (C), where the first two serve to measure termhood and the third determines the unithood of complex terms. The equation of SRC is as follows:

$$SRC(g) = termhood(g) + unithood(g)$$

$$termhood(g) = S(g) * \alpha + R(g) * \beta$$

$$unithood(g) = \begin{Bmatrix} 0, & \text{iff}|g| = 1 \\ C(g) * \gamma, & \text{iff}|g| > 1 \end{Bmatrix}$$

where g is a stemmed ngram, and the coefficients α, β and γ are the user-adjustable parameters, where α + β = 1 for unigrams and α + β + γ = 1 for complex ngrams. The three terminological features of SRC are briefly described in the next sections.

3.1.1. Term salience

The notion of salience is based on the automatic keyword extraction measure TF-IDF (cf. Salton, Wong, and Yang, 1975; Salton and Buckley, 1988; among many others), i.e. the weight of a term is determined by the relative frequency of the term in a certain document (or term frequency, i.e. TF) compared with the inverse proportion of that term in the entire document collection (or inverse document frequency, i.e. IDF). Thus, the salience of the stemmed ngram *g* in the document *d* is calculated by applying the following formula:

$$S_d(g) = TF(g) * IDF(g) * NORM(g)$$

$$TF(g) = \mathrm{f}_d(g)$$

$$IDF(g) = 1 + log\left(\frac{N_T}{\mathrm{df}(g)}\right), \textit{ where } \mathrm{df}(g) > 0$$

$$NORM(g) = \frac{1}{\sqrt{\sum_{g \in d} \left(TF(g) \times IDF(g) \right)^2}}$$

where $f_d(g)$ is the number of occurrences of g in d, N_T is the number of documents in the target corpus, and $df(g)$ is the number of documents in which the ngram appears in the target corpus. Apart from treating all documents as equally important regardless of their size, the cosine normalization factor makes the salience index range from 0 to 1. The salience of a given ngram with respect to the whole target corpus, and not just to a single document, is calculated as follows:

$$S(g) = \frac{\sum_{d \in CP_T} S_d(g)}{\sqrt{\sum_{g_j \in CP_T} \left(S(g_j) \right)^2}}$$

3.1.2. Term relevance

In DEXTER, the salience of a term is combined with a measure which quantifies the relevance of the ngram through the contrastive analysis between the target corpus and a reference corpus. As an adaptation of Ahmad, Gillam, and Tostevin's weirdness (2000), relevance is calculated as follows[2]:

$$R(g)^{''} = \frac{P_T(g)}{P_R(g)}$$

$$P_T(g) = \frac{f_T(g)}{|CP_T|}, \text{iff } |g| = 1; otherwise, P_T(g) = \frac{\sqrt[|g|]{\prod_{k_i \in g} f_T(k_i)}}{|CP_T|}$$

$$P_R(g) = \frac{f_R(g)}{|CP_R|}, \text{iff } |g| = 1; otherwise, P_R(g) = \frac{\sqrt[|g|]{\prod_{k_i \in g} f_R(k_i)}}{|CP_R|}$$

where $f_T(g)$ and $f_R(g)$ represent the frequency of the stemmed ngram g in the target corpus and the reference corpus respectively, $f_T(k)$ and $f_R(k)$ represent the frequency of a given unigram in g with respect to target corpus and reference corpus respectively, $|CP_T|$ and $|CP_R|$ represent the total number of words in the target corpus and the reference corpus respectively, and $|g|$ is the number of lexical items included in the ngram. If the ngram does not occur in the reference corpus, then $f_R(g) = 1$. In the case of complex candidates, relevance is calculated on the basis

[2] DEXTER uses the *British National Corpus* (BNC) and *Corpus de Referencia del Español Actual* (CREA) as the corpora of reference for English and Spanish respectively.

of the geometric mean of each lexical item within the candidate. Finally, the relevance index is normalized with the following equation:

$$R(g) = 1 - \frac{1}{\log_2\left(2 + R(g)''\right)}$$

3.1.3. Term cohesion

Cohesion is aimed to quantify the degree of stability of bigrams and trigrams. As an adaptation of Park, Byrd and Boguraev's metric (2002), cohesion is calculated as follows:

$$C(g)'' = \frac{f_T(g)}{\sqrt[|g|]{\prod_{k_i \in g} f_T(k_i)}} \times F, \text{ iff } |g| > 1$$

$$F = \begin{cases} 1, & \text{iff } f_T(g) = 1 \\ \log_2(f_T(g)), & \text{iff } f_T(g) > 1 \end{cases}$$

where $f_T(g)$ is the frequency of the stemmed ngram g in the target corpus, $f_T(k)$ is the frequency of a given unigram in g with respect to the target corpus, and $|g|$ is the number of unigrams in g. Here geometric mean also smooths a frequency distribution where extreme values are present. Finally, cohesion values are normalized similarly to those of relevance:

$$C(g) = 1 - \frac{1}{\log_2\left(2 + C(g)''\right)}, \text{ iff } |g| > 1$$

3.2. Considerations on corpus design

As noted by Hunston (2008), a corpus designed for research purposes needs more careful consideration and planning than a small corpus compiled by a language teacher for pedagogical purposes. In the spectrum between careful planning and random sampling, however, an ATE workbench such as DEXTER always requires some guidelines for the design and compilation of a specialized corpus with the aim to provide meaningful results after term weighting. These guidelines refer to both the quantity and quality of the corpus, and more particularly to the three issues which are usually taken into account when designing a corpus: size, representativeness and balance. In this section, we follow the sample case of an English-for-Specific-Purposes (ESP) teacher who intends to develop a glossary on "Electronics for Computer Hardware".

DEXTER has been devised to build small and medium-sized corpora. In fact, this is logically what is expected from such a setting, where "any corpus an individual researcher or practitioner, such as a teacher of ESP or EAP

[English for Academic Purposes], will be able to construct will necessarily be small" (Koester, 2010: 67). Indeed, large corpora have less relevance in the field of LSP:

> The large corpus (…) provides either too much data across too large a spectrum, or too little focused data, to be directly helpful to learners with specific learning purposes. (Tribble, 2001: 132)

Moreover, specialized corpora do not need to be "as large as more general corpora to yield reliable results"; since specialized corpora are "carefully targeted, they are more likely to reliably represent a particular register or genre than general corpora" (Koester, 2010: 68-69). In DEXTER, there is no constraint in the number of documents a corpus can contain, but each document in the corpus can have a maximum of 25,000 tokens. On average, DEXTER could process a corpus containing 200 documents without compromising on processing speed. Indeed, DEXTER has been successfully tested with an imported version of GLOBALCRIMETERM (Felices Lago, and Ureña Gómez-Moreno, 2012; Periñán-Pascual and Arcas Túnez, 2014), a corpus on the legal subdomain of organized crime and terrorism which contains more than 600 documents and 4 million and a half tokens. Since "there seems to be general agreement that a corpus of up to 250,000 words can be considered as *small*" (Flowerdew, 2004: 19), we can convincingly describe our corpora as small and medium-sized.

When looking at the issue of corpus size, the first idea that usually comes to mind is that the corpus should be as large as possible, being in line with the traditional view that the larger the corpus is, the better: "small is not beautiful; it is simply a limitation" (Sinclair, 2004: 189). However, "a huge corpus does not necessarily 'represent' a language or a variety of a language any better than a smaller corpus" (Kennedy, 1998: 68). Therefore, "what is more important than the actual size of the corpus is how well it is designed and that it is 'representative'" (Koester, 2010: 68), that is, "a corpus is 'representative' to the extent that findings based on its contents can be generalized to a larger hypothetical corpus" (Leech, 1991: 27). Biber (1993: 243) described representativeness as "the extent to which a sample includes the full range of variability in a population". In this way, "a corpus is usually intended to be a microcosm of a larger phenomenon" (Hunston, 2008: 160). In our example, this "larger phenomenon" would be the whole body of technical literature that students will be exposed to in real-life situations. Therefore, texts for our sample corpus should be extracted from textbooks, technical documentation or user manuals, among other sources.

Finally, there should be an equal distribution of the texts in the composition of the corpus with respect to the types of documents, that is, the balance of text types in the corpus is addressed by having similar amounts of text from different sources. For example, the proportion of samples from textbooks should be equal to the proportion of samples from user manuals. Therefore, balance is closely connected to size, but it also takes into account representativeness. The corpus should also have an equal distribution of texts with respect to the different subtopics. In our sample corpus, for example, there should be no unbalance between the number of documents describing electronic components and the number of installation and troubleshooting guides. However, it should be noticed that text sampling is essential in DEXTER. Whereas some texts in the corpus can be whole documents, others can be only parts of documents (i.e. text samples). Otherwise, one or two very large documents could finally result in a disproportionate composition of the corpus, since "a corpus of a million words or so cannot afford to include whole books which might be up to 100,000 words in length" (Hunston, 2008: 165). Therefore, instead of having a whole book about electronics as one single document, it would be advisable to split the book into many text samples, where each sample could be treated as a cohesive, thematically-consistent document. In conclusion, the reliability of term weighting will definitely be subject to considerations on the quantity and quality of data in the corpus.

3.3. Displaying and validating term candidates

After corpus compilation, term extraction and weighting, we use the same interface in DEXTER for terminology exploration and validation. On the one hand, users can browse the terms of the corpus domain, together with their weights for SRC, S, R, C and the total frequency, from the stemmed ngrams extracted by DEXTER. Moreover, SRC values can be displayed by means of a bar chart, which can be saved as a PNG image. For practical reasons, the user can also get a TXT list of the lemmatized tokens associated to every stemmed ngram.

We can read the context of the term by clicking on a given ngram after having determined the number and length of fragments to be retrieved[3]. Unlike other similar applications, we chose to contextualize the token within a paragraph, instead of displaying the output in KWIC format. As shown in Figure 1, it is not unusual to find several instances of the same term in one single paragraph, allowing us to understand term usage better when these tokens are visualized together. In fact, this co-occurrence of word-forms frequently happens with DEXTER, since the documents of the corpus are indexed by Lucene.Net (McCandless, Hatcher, and Gospodnetic, 2010), an open source library used in information retrieval. As noted by these researchers, a naïve approach to search a certain word in a large collection of documents would be to sequentially scan each file for the given word, but this would generate a bottleneck in case of processing a large set of documents. A more effective approach is based on indexation, and more particularly on creating an inverted index, i.e. a data structure analogous to an index at the end of a book, which lets you quickly locate pages where certain topic words appear[4]. Thus, when the user selects an ngram, this internally takes the form of a query, and DEXTER returns those snippets with the best similarity score: the higher the score is, the more likely for humans to judge the fragment as relevant. Users can also have access to the whole document from where a snippet has been extracted.

On the other hand, with regard to term validation, DEXTER is quite different from other ATE systems, since term validation is here guided not only by the probabilistic weight of the terms but also by a set of lexical filters (i.e. functional, basic and advanced). A brief description of these filters is presented below:

1. The functional filter contains the stems corresponding to articles, pronouns, prepositions, conjunctions, contractions (e.g. *'s, 've*) and auxiliary verbs (e.g. *be, do, have*, *get*), as well as Arabic and Roman numerals and common abbreviations used in documents (e.g., *i.e., c.f., etc., et al*).

2. The basic filter contains the stems of words typically found at a beginner or intermediate level in foreign language learning. More particularly, these stems were derived from *Easier English Basic Dictionary* (Collin, 2004) in the case of English, and from Bustos (2001) in the case of Spanish. In both cases, the inventory also includes the irregular word forms of those verbs present in the lexical resources.

3. The advanced filter contains the stems of words typically found at an advanced level in foreign language learning. More particularly, these stems were derived from *Collins COBUILD English Language Dictionary* (Sinclair, 1987) in the case of English, and from *Diccionario Salamanca* (Gutiérrez Cuadrado, 1996) in the case of Spanish. In both cases, the inventory also includes the irregular word forms of those verbs present in the lexical resources.

These lexical filters are actually applied as stopword lists, since those candidate terms which match any item in these lists are automatically hidden for validation.

4. Conclusions

Language teachers instructing in a given area of specialized knowledge need quick and reliable access to terminology. In fact, it is very useful for them to create their own domain-specific glossaries, based on documents which were written to meet professional needs in the field of expertise. In this regard, DEXTER can help language teachers discover specialized vocabulary for LSP courses from small- and medium-sized customized corpora by means of a suite of tools with different functionalities, such as corpus compilation and management, document indexation and retrieval, query elaboration, textual exploration and terminological extraction. As an ATE system,

[3] We can retrieve a maximum of 80 snippets for each term, with a maximum of 400 characters per fragment.

[4] McCandless, Hatcher, and Gospodnetic (2010: appendix B) provided a detailed account of the structure of Lucene's inverted index.

shallow linguistic patterns filter term candidates in the form of unigrams, bigrams and trigrams before term weighting, which is based on a user-adjustable composite metric that we call SRC, because it captures the terminological notions of salience, relevance and cohesion. Although currently used for English and Spanish, the modular architecture of DEXTER makes its work environment be easily adaptable to corpora in many other languages.

5. Acknowledgement

Financial support for this research has been provided by the DGI, Spanish Ministry of Education and Science, grant FFI2011-29798-C02-01.

References

Ahmad, K., Gillam, L., and Tostevin, L. (2000). Weirdness Indexing for Logical Document Extrapolation and Retrieval (WILDER).. In E. M Voorhees, and D. K Harman (Eds.), *Proceedings of the 8th Text Retrieval Conference (TREC-8)* (pp. 717-724). Washington: National Institute of Standards and Technology.

Biber, D. (1993). Representativeness in corpus design. *Literary and Linguistic Computing, 8 (4)*, 243-257.

Bustos, J. M. (2001). Definición de glosarios léxicos del español: Niveles inicial e intermedio. *Enseñanza, 19*, 35 - 72.

Church, K. W., and Hanks, P. (1990). Word Association Norms, Mutual Information and Lexicography. *Computational Linguistics 6 (1)*, 22 - 29.

Church, K. W., Gale, W. Hanks, P., and Hindle, D. (1991). Using Statistics in Lexical Analysis. In U. Zernik (Ed.), *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon* (pp. 115-164). Hillsdale, NJ: Lawrence Erlbaum.

Collin, P. (2004). *Easier English Basic Dictionary*. London: Bloomsbury.

Dunning, T. (1994). Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics 19 (1)*: 61 - 74.

Felices Lago, A., and Ureña Gómez-Moreno, P. (2012). Fundamentos metodológicos de la creación subontológica en FunGramKB. *Onomázein 26*, 49-67.

Flowerdew, L. (2004). The Argument for Using English Specialized Corpora to Understand Academic and Professional Language. In U. Connor, and T. A. Upton (Eds.), *Discourse in the Professions. Perspectives from Corpus Linguistics* (pp. 11-33). Amsterdam/Philadelphia: John Benjamins.

Frantzi, K., and Ananiadou, S. (1996). Extracting Nested Collocations. *Proceedings of the 16th International Conference on Computational Linguistics* (pp. 41-46). Morristown: Association for Computational Linguistics.

Frantzi, K., Ananiadou, S., and Hideki, M. (2000). Automatic Recognition of Multi-Word Terms: the C-Value/NC-Value Method. *International Journal of Digital Libraries 3 (2)*, 115 - 130.

Grefenstette, G. (1994). *Explorations in Automatic Thesaurus Discovery*. Boston: Kluwer Academic.

Gutiérrez Cuadrado, J. (1996). *Diccionario Salamanca de la lengua española*. Madrid: Santillana-Universidad de Salamanca.

Hunston, S. (2008). Collection Strategies and Design Decisions. In A Lüdeling, and M. Kytö (Eds.), *Corpus Linguistics: An International Handbook*. Volume 1 (pp. 154-168). Berlin: de Gruyter.

Justeson, J. S., and Katz, S. M. (1995). Technical Terminology: Some Linguistic Properties and an Algorithm for Identification in Text. *Natural Language Engineering 1 (1)*, 9 - 27.

Kennedy, G. D. (1998). *An Introduction to Corpus Linguistics*. London: Longman.

Koester, A. (2010). Building Small Specialized Corpora. In A. O'Keeffe, and M. McCarthy (Eds.), *The Routledge Handbook of Corpus Linguistics* (pp. 66-79). London: Routledge.

Leech, G. (1991). The State of the Art in Corpus Linguistics. In K. Aijmer, and B. Altenberg (Eds.), *English Corpus Linguistics* (pp. 8-29) London: Longman.

McCandless, M., Hatcher, E., and Gospodnetic, O. (2010). *Lucene in action.* Greenwich: Manning.

Nagao, M., Mizutani, M., and Ikeda, H. (1976). An Automated Method of the Extraction of Important Words from Japanese Scientific Documents. *Transactions of Information Processing Society of Japan 17 (2)*, 110 - 117.

Navigli, R, and Velardi, P. (2002). Semantic Interpretation of Terminological Strings. *Proceedings of the 6th International Conference on Terminology and Knowledge Engineering* (pp. 95-100). Berlin-Heidelberg: Springer.

Park, Y., Byrd, R. J., and Boguraev, B. K. (2002). Automatic Glossary Extraction: Beyond Terminology Identification. In *Proceedings of the 19th International Conference on Computational Linguistics*, Volume 1 (pp. 1-7). Stroudsburg, PA: Association for Computational Linguistics.

Periñán-Pascual, C., and Arcas Túnez, F. (2014). La ingeniería del conocimiento en el dominio legal: La construcción de una Ontología Satélite en FunGramKB. *Revista Signos: Estudios de Lingüística 47 (84)*, 113 - 139.

Salton, G., and Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management 24 (5)*, 513 - 523.

Salton, G., Wong, A., and Yang, C. (1975). A Vector Space Model for Automatic Indexing. *Communications of the ACM 18 (11)*, 613 - 620.

Sclano, F., and Velardi, P. (2007). TermExtractor: A Web Application to Learn the Common Terminology of Interest Groups and Research Communities. In *Proceedings of the 9th Conference on Terminology and Artificial Intelligence*, Sophia Antinopolis.

Sinclair, J. M. (Ed.) (1987). *Collins COBUILD English Language Dictionary*. Londres: HarperCollins.

Sinclair, J. (2004). *Trust the Text: Language, Corpus and Discourse*. London: Routledge.

Singhal, A., Salton, G. and Buckley, C. (1996). Length Normalization in Degraded Text Collections. In *Proceedings of the 5th Annual Symposium on Document Analysis and Information Retrieval* (pp. 149-162). Las Vegas: University of Nevada.

Smadja, F. (1993). Retrieving Collocations from Text: Xtract. *Computational Linguistics 19 (1)*, 143 - 178.

Tribble, C. (2001). Corpora and Corpus Analysis: New Windows on Academic Writing. In J. Flowerdew (Ed.), *Academic Discourse* (pp. 131-149). Harlow: Addison Wesley Longman.