Textractor: A Framework for Extracting Relevant Domain Concepts from Irregular Corporate Textual Datasets

Ashwin Ittoo, Laura Maruster, Hans Wortmann and Gosse Bouma

Faculty of Economics and Business, University of Groningen
9747 AE Groningen, The Netherlands
{r.a.ittoo,l.maruster, j.c.wortmann, g.bouma}@rug.nl

Abstract. Various information extraction (IE) systems for corporate usage exist. However, none of them target the product development and/or customer service domain, despite significant application potentials and benefits. This domain also poses new scientific challenges, such as the lack of external knowledge resources, and irregularities like ungrammatical constructs in textual data, which compromise successful information extraction. To address these issues, we describe the development of Textractor; an application for accurately extracting relevant concepts from irregular textual narratives in datasets of product development and/or customer service organizations. The extracted information can subsequently be fed to a host of business intelligence activities. We present novel algorithms, combining both statistical and linguistic approaches, for the accurate discovery of relevant domain concepts from highly irregular/ungrammatical texts. Evaluations on real-life corporate data revealed that Textractor extracts domain concepts, realized as single or multi-word terms in ungrammatical texts, with high precision.

Keywords: Natural Language processing, term extraction, information extraction, corporate industrial data, product development, customer service

1 Introduction

Many product development and customer service organizations are struggling with the rising number of customer complaints due to soft-failures. These failures arise from mismatches between products' specifications and customers' expectations. Previous studies [13] suggested that the information in product development and customer service data sources could provide insights on causes of soft-failures. However, this information is often expressed in natural language free-text. Its analysis requires natural language processing (NLP) techniques, such as Information Extraction (IE). For example, IE techniques could serve as a precursor to business intelligence (BI) by extracting relevant concepts pertaining to soft-failures from textual data, and thus, enable the development of better quality products.

Various IE systems exist for corporate usage [12]. However, none of them targeted the product development and/or customer service (PD-CS) domain despite the numerous application opportunities and benefits to be accrued. Our interest in this domain is also attributed to the scientific challenges that it poses to extant IE systems. A major challenge is the quasi-inexistence of knowledge resources, such as ontologies, upon which traditional IE systems relied to identify relevant concepts from text. Creating such resources in PD-CS organizations is impeded by the perpetually evolving product lines and business models. Existing resources are neither machine-readable (e.g. diagrams on paper), nor reliable since different organizational departments maintain conflicting domain conceptualizations due to their diverging business perspectives. Exploiting general knowledge repositories like Wikipedia is also not warranted due to the specific PD-CS terminologies. This is in stark contrast to traditional IE application areas, with readily available and authoritative resources such as Yahoo! Finance [12] or Unified Medical Language System (UMLS)[18] to support IE activities. Another challenge pertains to irregularities of PD-CS data that compromise traditional IE systems. One such irregularity is terminological variations due to subtle language patterns. For example, identifying domain concepts from semantically ambiguous phrases, as in "cooling fan" from "device cooling fan", is difficult especially in the absence of knowledge resources. Another type of inconsistency in the data which hinders IE is ungrammatical constructs. For example, the absence of sentence boundaries in "customer helpdesk collimator shutter" hinders the identification of the two distinct terms "customer helpdesk" and "collimator shutter". These difficulties are compounded by the presence of both valid and invalid multi-word terms, such as "processor connection cable" and "status check ok". Creating extraction rules to deal with these inconsistencies is not always viable. They do not guarantee the capture of all inconsistencies, and their hand-crafting is tedious. Another characteristic of PD-CS datasets that poses additional challenges to accurate term extraction is their multi-lingual contents generated by customers and engineers worldwide.

To address these issues, and enable PD-CS organizations to fully exploit IE capabilities in their business intelligence efforts, we develop and present a framework for extracting relevant concepts from corporate datasets with textual contents that exhibit the aforementioned irregularities. We realize our methodology in the Textractor term extraction application that we implemented as part of the DataFusion initiative[1]. DataFusion aims at facilitating the creation of better quality products by aligning customers' expectations to products' specifications.

Our major contributions in this paper are novel algorithms that, by applying linguistic and statistical approaches in an ensemble, accurately extract relevant domain concepts realized as terms of arbitrary length in irregular narrations. The extracted information can be fed to various BI models to support activities such as analyzing soft-failure causes. Textractor consists of independent modules, performing various tasks for successful IE, such as multi-language standardization, and data pre-processing. The modules are easily adaptable for other application domains, although we focused on PD-CS. Our framework, depicting various IE activities and Textractor's architecture, could serve as blueprints for organizations in their IE

[1] The DataFusion or "Merging of Incoherent Field Feedback Data into Prioritized Design Information" initiative is a collaboration between academia and industry, sponsored by the Dutch Ministry of Economic Affairs under the IOP-IPCR program.

endeavors. We evaluated Textractor on real-life industrial datasets provided by industrial partners in the DataFusion project. The high precision obtained during evaluation suggests that Textractor is indeed suitable for extracting relevant information from irregular corporate textual data, and that it can support various forms of BI.

This paper is organized as follows. Section 2 presents and compares related work. Our framework and the underlying methodology are discussed in Section 3. We present results of experimental evaluations in Section 4, before concluding and highlighting future work in Section 5.

2 Related Work

Term Extraction (TE) is a form of information extraction (IE) to automatically extract linguistic realizations of relevant concepts, i.e. terms, from domain text. Literature mentions three forms of TE approaches. Linguistic approaches [1] identify terms based on their linguistic properties (e.g. parts-of-speech). Statistical approaches employ techniques like mutual information [19], log-likelihood [19,22], and term frequency-inverse document frequency [9] for computing the saliency of terms. Hybrid approaches [7] combine linguistic and statistical techniques to accurately recognize terms from texts. Most current TE and general IE rely on knowledge resources like Wikipedia [21] or UMLS [10] to identify generic (e.g. Persons) or bio-medical concepts (e.g. genes) from specialized texts.

IE applications targeted at corporate usage have also been developed. The h-TechSight system [11] uses domain ontologies to extract information such as employment trends. Nexus [14] relies on the Politically Motivated Violent Events ontology to extract facts from online news articles. Xu et al. [22] use GermaNet for financial information extraction. MUSING [12] extracts information from corporate datasets and resources like Yahoo! Finance based on domain ontologies.

Compared to the classical application areas listed above, term extraction from product development and/or customer service (PD-CS) datasets presents new challenges. As already mentioned, existing knowledge resources, like ontologies, are quasi-inexistent. PD-CS datasets also exhibit more irregularities than those (e.g. bio-medical, finance data) traditionally targeted by existing IE systems. For example, inconsistent and ungrammatical language constructs are likelier in texts entered by (frustrated) customers or helpdesk officers with their personal jargon than in texts of bio-medical or financial experts. To address the challenges of term extraction from the product development and/or customer service domain, and to enable organizations fully exploit IE capabilities for business intelligence, we develop the Textractor application.

3 Textractor Framework

We now describe our methodology and its implementation in the Textractor application for extracting relevant domain concepts from corporate datasets with irregular textual contents. The various phases of our methodology are realized by independent but interoperable modules in Textractor's architecture, depicted in Figure 1 (Dotted objects are not part of Textractor, they depict the input and output). Textractor's design follows the Service-Oriented-Architecture (SOA) paradigm, with each of its modules providing specific services. They are discussed in the following sub-sections. As our major contributions, the Candidate Concept Identification, and Concept Selection and Extraction phases (modules) are treated more extensively. The modules were implemented as standalone and web-based applications in Java, Perl and (Java) Servlets running under the Tomcat server.

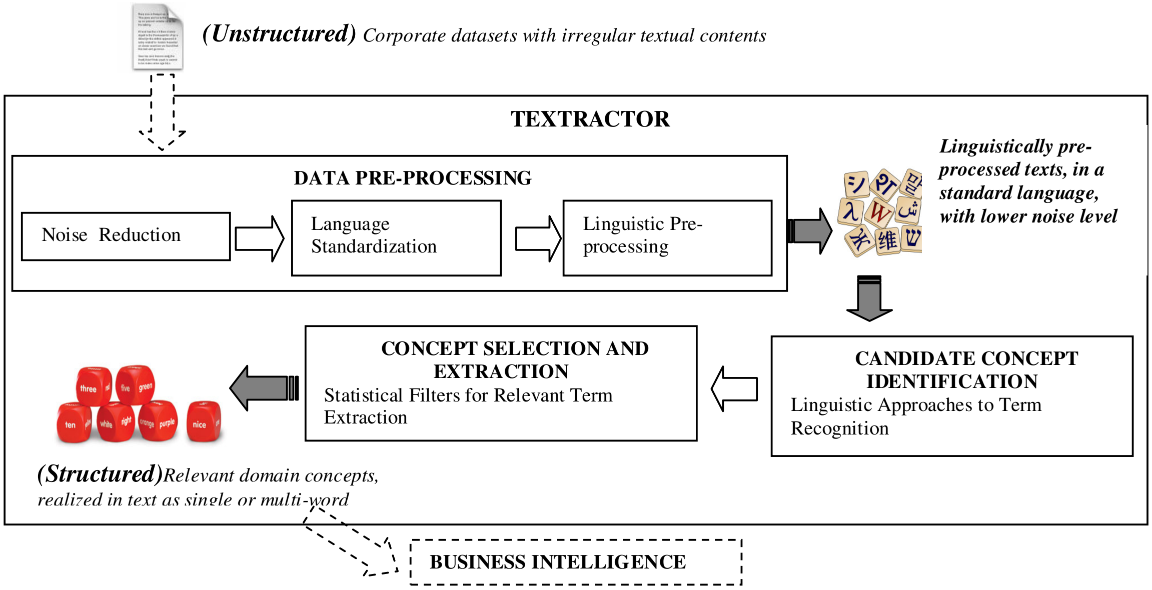

(Unstructured) Corporate datasets with irregular textual contents
TEXTRACTOR
DATA PRE-PROCESSING
Noise Reduction
Language Standardization
Linguistic Pre-processing
Linguistically pre-processed texts, in a standard language, with lower noise level
CANDIDATE CONCEPT IDENTIFICATION
Linguistic Approaches to Term Recognition
CONCEPT SELECTION AND EXTRACTION
Statistical Filters for Relevant Term Extraction
(Structured) Relevant domain concepts, realized in text as single or multi-word
BUSINESS INTELLIGENCE

Fig. 1. Textractor Architecture.

3.1 Data Pre-processing

Our approach starts with data pre-processing, which lowers the noise level, standardizes the languages for multi-lingual texts, and performs basic linguistic operations on the textual data.

Noise Reduction

We implemented wrapper-like techniques [3] based on regular expression rules to discard noisy entities from our texts. Entities that our rules target include identifiers like "<EOL>", which are automatically inserted by data-entry software, words with less than two letters, and irrelevant symbols entered by humans in their narrations (e.g. ",''{,},#...). The rules' authoring was done with domain experts so that only noisy entities are targeted, without dropping any relevant facts from the data.

Language Detection and Standardization

The narrations in our data were expressed in the major European and Asian languages. To facilitate information extraction, we implemented a language detection algorithm, based on [4], which automatically determines the language in which texts are written. Our algorithm predicts the language of textual records by computing their distances from category profiles that define various languages. The language with the smallest distance is predicted. While we maintain the core of the technique presented in [4], our algorithm only targets the languages (viz. English, Dutch, German, French, Spanish, Italian, and Portuguese) present in our dataset, and achieves higher accuracy than [4] on these selected few languages. It also annotates texts containing elements of more than one language with the label "inter-lingual". We omit further details of this algorithm in this paper.

Linguistic Pre-processing

Although NLP tools for other languages exist, in this work, we restrict ourselves to English texts for ease of evaluation by the authors and industrial partners. Also, we believe that formal grammars in English and their associated NLP tools are more mature than those in other languages.

We apply standard NLP techniques to segment our textual records into individual sentences, to tokenize the sentences, and to determine the parts-of-speech (PoS)-tags of these tokens. PoS-tags are later used to identify candidate terms (Section 3.2). We also determine the lemmas (roots) of inflected words-forms. PoS-tagging and lemmatization were achieved with the Stanford maximum entropy tagger and morphological analyzer [17].

The output of this phase is linguistically processed textual records, in a standard language, and with lower noise levels. The data is now amenable for term identification and extraction.

3.2 Candidate Concept Identification

This phase of our methodology recognizes terms in text based on contiguous sequences of parts-of-speech (PoS)-tags, called term signatures.

Various term signatures are mentioned in literature. However, existing term signatures are susceptible to PoS-tagging errors, which are common in our corporate datasets due to their irregular texts with subtle language patterns and malformed sentences. We noted that in semantically ambiguous phrases, some nouns were wrongly tagged by standard tools [5,17] as progressive-verbs (VBG). This led to inaccurate term identification by existing signatures as they always look for sequences of adjectives and nouns. For instance, "device cooling fan" was PoS-tagged as "device/N cooling/VBG fan/N", and misled existing term signatures to inaccurately suggest "device" and "fan" as terms, instead of "cooling fan". As a brute-force countermeasure, we devised a PoS-signature that considers as terms, any phrase with a progressive verb (VBG) or adjective (A) that precedes nouns (N). A simplified version is shown in the regular expression of equation (1). Note that ?,+, and * are regular expression cardinality operators, and =~ is the "matching" operator. Experimental evaluations (Section 4.2) reveal that our signature accurately identifies candidate terms even in the presence of PoS-tag errors.

$$\text{term} =\sim \ (\text{VBG?})\ (\text{A*})(\text{N+}). \qquad (1)$$

Another limitation of term signatures is their inability to recognize valid terms, particularly if term (or sentence) boundaries are not explicit. For example, the ungrammatical construct "customer helpdesk collimator shutter" contains two terms, "customer helpdesk" and "collimator shutter" in two sentences, which are not explicitly demarcated by boundaries. PoS-tagging yields the (correct) sequence "N N N N", which causes term signatures to wrongly suggest "customer helpdesk collimator shutter" as a term, instead of "customer helpdesk" and "collimator shutter". Dealing with such intricacies and other invalid terms requires statistical analyses.

3.3 Concept Selection and Extraction

This stage applies two statistical filters in an ensemble to address the short-comings of the previous linguistic filter, to discriminate between valid and invalid terms, and to measure the terms' relevancy.

The first statistical filter in our ensemble is based on the term frequency-inverse document frequency (tf-idf) [15] metric. It determines the relevancy of a candidate term *t* in a document (i.e. textual record of our dataset) *d* as

$$\text{tf} - \text{idf}\,(\text{t}_\text{d}) = \text{f}_\text{t} \times \log\left(\frac{\text{N}}{\text{Df}_\text{t}}\right) \qquad (2)$$

where f_t is the frequency of *t* in *d*, Df_t is the number of records containing *t*, and *N* is the total number of records.

Higher scores are assigned to terms occurring frequently in a few documents. Terms are considered relevant (i.e. they designate salient domain concepts) if their scores exceed experimentally set thresholds. We used tf-idf to determine relevant single-word terms.

Identifying relevant multi-word terms requires measuring the collocation strength between their individual lexical elements (word tokens). To this aim, we applied the cubed mutual information technique (MI3), which was shown to achieve highest precision and recall compared to other collocation measures [6,19]. MI3 computes the collocation strength between words *x* and *y* in a term "*x y"* as

$$\mathrm{MI3(t)} = \log \frac{\left(\frac{\mathrm{F(x,y)}}{\mathrm{N}}\right)^3}{\frac{\sum \mathrm{F(x)}}{\mathrm{N}} \times \frac{\sum \mathrm{F(y)}}{\mathrm{N}}} \tag{3}$$

where *F(x,y)* is the co-occurrence frequency of words *x* and *y*, *F(x)* is the frequency of *x* and *N* is the number of candidate multi-word terms identified by the Candidate Concept Identification phase.

Collocation techniques are designed only for 2-word terms (bi-grams). They are unable to compute the collocation strength for longer terms (general n-grams, n>2), like “disk image power supply unit”, which are prominent in corporate domains. To address this issue, we present an innovative algorithm based on dynamic programming to calculate the collocation strength of a term *t*, consisting of an arbitrary number of words (n-gram, n >=2). A simplified version of our algorithm is listed below.

```
Procedure collocation_mi3 (Term t)
1. n = length of t;
2. if n == 2 then
3.    score=MI3(t); //note: according to equation (3)
4.    add (t,score) to hash_n; //note: n=term’s length
5. else
6.    sTermSet= get subterms of t with length m=2…(n-1);
7.    for each sTerm of length m, element of sTermSet
8.        if hash_m contains(sTerm)then
9.           score+=retrieve score of sTerm from hash_m;
10.       else
11.          score += collocation_mi3(sTerm);
12.   score = score/(size of sTermSet);
13.   add (t,score) to hash_n;
```

Our iterative procedure starts by computing the collocation scores between the elements of bi-grams using MI3. Bi-grams and their scores are indexed in a look-up (hash) table (lines 3-4). In each subsequent iterations, we consider terms with one additional word. For example, tri-grams (3-word terms) are processed after bi-grams. To deal with general n-word terms (n>2), we reformulate the statement that “terms are composed of words” [16] to posit that n-word terms (n>2) are composed of sub-terms, each of which can consist of at least 2 and at most (n-1) words. Our algorithm operates upon the premise that if a multi-word term *t* is relevant, its sub-terms must also be relevant. Thus, given any n-word term (n>2), our algorithm first decomposes them into sub-terms (line 6). For example, sub-terms of “control rack power supply” are “control rack”, “control power”, “control supply”, “rack power”, “rack supply”, “power supply”, “control rack power”, and “rack power supply”. Next, lookup tables of previous iterations are inspected to determine whether the sub-terms’ collocations are already computed (line 8). If so, the scores are retrieved and accumulated (line 9). Otherwise, we compute the scores of the sub-terms in a recursive manner (line 11).

Finally, the collocation score of an n-word term (n>2) is calculated by normalizing the accumulated collocations of its sub-terms (line 12). The term and its score are stored in a look-up table, which will be inspected in future iterations when processing longer terms (e.g. with n+1 words). We consider (multi-word) terms as domain relevant if their collocation scores are higher than an experimentally set threshold. Experiments (Section 4.3) reveal that our algorithm not only accurately identifies valid multi-words terms of arbitrary length, but also discards invalid ones. It is also able to identify individual terms from ungrammatical constructs, which for example, lack explicit term boundaries.

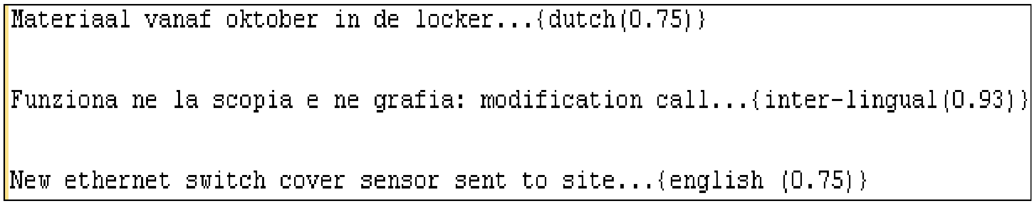
The output of this stage is a set of salient domain concepts, manifested in text as single-word or multi-word terms with relevancy and collocation scores above experimentally set thresholds.

4 Evaluation

Experiments were conducted on real-life corporate data provided by our industrial partners to gauge the performance of Textractor. The datasets contained 143,255 text records of customer complaints captured at helpdesks, and repair actions of service engineers. The contents exhibited typical peculiarities of corporate data, namely high noise level, multi-lingual narratives, subtle language patterns with nested terms, ungrammatical constructs, and valid and invalid multi-word terms. We only report the evaluation of the language detector, candidate concept identification and concept selection and extraction phases, emphasizing on the latter two major contributions..

4.1 Language Detector

Around 65% of our corpus consisted of English texts. Italian documents constituted around 20%, while the remaining 15% of the corpus was almost equally distributed among documents in the French, Dutch, German, Spanish and Portuguese languages. We found out that, on average, our detector correctly predicted the language of a given text with an accuracy of 95%. Sample outputs are in Figure 2, with the first narration correctly identified as Dutch, the second one as “inter-lingual” due to the presence of both English and Italian words, and the last one as English.



```
Materiaal vanaf oktober in de locker...{dutch(0.75)}

Funziona ne la scopia e ne grafia: modification call...{inter-lingual(0.93)}

New ethernet switch cover sensor sent to site...{english (0.75)}
```

Fig. 2. Sample output of language detector

4.2 Candidate Concept Identification

The PoS-tag pattern that we propose as signature for term recognition (Section 3.2) was less susceptible to PoS-tagging errors in identifying terms from ill-formed sentences. For example, despite the PoS-tagging errors in “device/N cooling/VBG fan/N”, it correctly induced “cooling fan” as a candidate domain concept. Other similar examples are listed in Table 1. PoS-tag errors are marked with *.

Table 1. Correctly induced candidate terms in the presence of PoS-tag errors.

Original Phrase (with PoS-tags)	Candidate Term Identified
unit/N viewing/VBG* console/N	viewing console
testing/VBG archiving/VBG* device/N	archiving device
italy/N flickering/VBG* monitor/N	flickering monitor

However, our term signature fails to recognize terms from ungrammatical sentences without boundaries, such as the terms “customer helpdesk” and “collimator shutter” from “customer helpdesk collimator shutter”. Such intricacies are dealt with in the next statistical filtering stage.

4.3 Concept Selection and Extraction

Table 2 illustrates some results of our 2-stage statistical filtering (Section 3.3). Pertinent single-word terms were identified based on their relevancy scores using tf-idf. Relevancy scores for multi-word terms (i.e. n-word terms, n>=2) were computed using our dynamic programming algorithm based on MI3. The maximum tf-idf score (single-word term) we obtained was 84.32, while for MI3 (multi-word term) the maximum score was 143.43

Table 2. Sample multi-word terms extracted by statistical filter.

Term	Term Length	Score: tf-idf and MI3
headset	1	33.71
collimator shutter	2	26.75
customer helpdesk	2	30.45
cpu circuit breaker	3	44.56
control rack power supply	4	33.21
video tube window cover	4	33.67
audio console keyboard circuit board	5	30.50
disk image power supply unit	5	25.91
Status check ok	invalid	0
quality ppl_gb ppl_gb	invalid	0
customer helpdesk collimator shutter	invalid	0

The single and multi-word terms, in Table 2, extracted by our technique, indicate that our approach successfully identifies relevant domain terms of arbitrary lengths, and does not suffer from limitations of traditional statistical techniques that are intended only for 2-word terms. Our technique also separates individual terms embedded in ungrammatical texts that lack sentence/term boundaries. For example, it assigns significantly higher scores to the terms “customer helpdesk” and “collimator shutter” than to “customer helpdesk collimator shutter”, the latter being an ungrammatical sentence in which the terms appear. Thus, based on relevancy scores, “customer helpdesk” and “collimator shutter” are suggested as domain terms, while “customer helpdesk collimator shutter” is discarded. Other invalid terms, like “quality ppl_gb ppl_gb” are also discarded due to their low relevancy scores as computed by our technique.

Evaluation was manually performed by domain experts from our industrial partners as no external resources for gold-standards were available. The precision [12], *P*, of Textractor in extracting relevant single and multi-word terms was measured according to equation (4) as the percentage of extracted terms that are considered domain relevant.

$$P = \frac{true\_positive}{true\_positive + false\_positive} \quad \textbf{(4)}$$

where *true_postive* is the number of relevant domain terms identified by Textractor and confirmed by the domain experts, and *false_positive* is the number of terms suggested by Textractor but deemed irrelevant by the experts. The highest precision obtained was 91.5%. with the threshold of the tf-idf filter set to 25 (i.e. only considering single-word terms with tf-idf score > 25) and that for the MI3 filter set to 15 (i.e. only considering muti-word terms with scores > 15). In the absence of any gold-standard, we selected a small sub-corpus, with 500 known entities in the domain, realized as both single and multi-word terms, and computed the recall score, *R*, of our technique as

$$R = \frac{true\_positive}{true\_positive + false\_negative} \quad \textbf{(5)}$$

, where false_negative is the number of known relevant domain concepts that our approach could not detect. By lowering thresholds for both the tf-idf and MI3 filters to 5, we achieved a nearly perfect recall of 99%. However, as expected, these low thresholds entailed significant precision losses. We thus computed the F-score to determine the optimal trade-off between precision and recall as

$$F - score = \frac{2 \times P \times R}{P + R} \quad \textbf{(6)}$$

The maximum F-score obtained was 87.7 %, with the thresholds for the tf-idf and MI3 filters set to 23 and 10 respectively, which resulted in a precision of 85.3% and recall of 90.4. Our scores, obtained in extracting terms from noisy, irregular corporate texts, are comparable to other related work, such as [12], which reports average precision, recall and F-measure of respectively 85.6, 93.6 and 84% in company/country profile information extraction, [22], which reports a precision of 61% for financial term extraction, and [14] with precision ranging from 28-100% for news event extraction.

5 Conclusion and Future Work

We have described the design and implementation of Textractor, an application to extract relevant domain concepts, realized as single or multi word terms, from text. Textractor enables business organizations to fully exploit capabilities of information extraction (IE) in order to overcome the difficulties in uncovering critical knowledge hidden within textual data to subsequently support business intelligence activities.

Unlike previous information extraction (IE) systems, Textractor does not rely on external resources (e.g. ontologies). It employs novel algorithms to accurately extract relevant domain concepts, realized as terms of arbitrary lengths, from corporate datasets. Our algorithms efficiently overcome the challenges posed by textual contents of corporate datasets such as terminological variations, subtle language patterns, ungrammatical constructs and the presence of valid and invalid multi-words terms. The high precision obtained during experimental evaluation by domain experts illustrates Textractor's suitability as a pre-cursor to business intelligence activities in corporate settings, especially in the domain of product development and/or customer service.

Future work will extract relations between the concepts identified by Textractor. We will then learn ontologies, which are crucial for semantically integrating heterogeneous, but complementing, data sources into a comprehensive basis. Ontologies facilitate the effective access and usage of information, so that organizations induce more meaningful insights from their business intelligence activities. In the domain of product development and customer service, ontology-based integration could lead to the discovery of soft-failures causes and a better understanding of customer complaints. This knowledge will allow organizations develop better quality products and ensure financial returns. Although we focused on a specific corporate domain in this paper, our algorithms are generic to be applicable in other corporate or even "open" domains, especially to deal with extracting multi-word terms from ungrammatical texts, such as from online forums and blogs.

Acknowledgement
This work is being carried out as part of the project "Merging of Incoherent Field Feedback Data into Prioritized Design Information (DataFusion)", sponsored by the Dutch Ministry of Economic Affairs under the IOP IPCR program.

References

1. Ananiadou, S.: A methodology for automatic term recognition. In: 15th Conference on Computational Linguistics, pp. 1034--1038.Association for Computational Linguistics, Morristown, NJ, USA (1994)

2. Bourigault, D.: Surface grammatical analysis for the extraction of terminological noun phrases. In: 14th Conference on Computational Linguistics, pp. 977-- 981. Association for Computational Linguistics, Morristown, NJ, USA (1992)
3. Buitelaar, P., Cimiano, P., Frank, A., Hartung, M., Racioppa, S.: Ontology-based Information Extraction and Integration from Heterogeneous Data Sources. International Journal of Human Computer Studies. 66, 759--788 (2008)
4. Cavnar, W. B., Trenkle, J. M.: N-Gram-Based Text Categorization. In: Third Annual Symposium on Document Analysis and Information Retrieval, pp. 161--175. UNLV Publications/Reprographics, Las Vegas, NV, USA (1994)
5. Cunningham, H., Maynard, D., Bontcheva, K., Tablan,V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: 40th Anniversary Meeting of the Association for Computational Linguistics (2002)
6. Daille, B., Gaussier, E., Lange, J.M.: Towards automatic extraction of monolingual and bilingual terminology. In: 15th conference on Computational Linguistics, pp 515--521. Association for Computational Linguistics, Morristown, NJ, USA (1994)
7. Frantzi ,K.T., Ananiadou, S.: Extracting nested collocations. In: 16th Conference on Computational Linguistics, pp 41--46. Association for Computational Linguistics, Morristown, NJ, USA (1996)
8. Justeson, J., Katz, S.M.: Technical terminology: some linguistic properties and an algorithm for identification in text. Natural Language Engineering,1,9--27 (1995)
9. Koyama, T., Kageura, K.: Term extraction using verb co-occurrence. In: 3rd International Workshop on Computational Terminology (2004)
10. Maynard, D., Ananiadou, S.: Identifying terms by their family and friends. In: 18th conference on Computational Linguistics, pp 530--536. Association for Computational Linguistics, Morristown, NJ, USA (2000)
11. Maynard, D., Yankova, M., Kourakis, A., Kokossis, A.: Ontology-based information extraction for market monitoring and technology watch. In ESWC Workshop End User Aspects of the Semantic Web", Heraklion, Crete (2005)
12. Maynard, D., Saggion, H., Yankova, M., Bontcheva, K., Peters, W.: Natural Language Technology for Information Integration in Business Intelligence. In: Abramowicz, W. (ed) BIS 2007. LNCS, vol. 4439, pp.366—380. Springer, Heidelberg (2007)
13. Petkova, V.: An analysis of field feedback in consumer electronics industry. PhD thesis, Eindhoven University of Technology
14. Piskorski, J. Tanev, H., Oezden-Wennerberg, P.: Extracting Violent Events from On-line News for Ontology Population. In: Abramowicz, W. (ed) BIS 2007. LNCS, vol. 4439, pp.287--300. Springer, Heidelberg (2007)
15. Salton, G.: Developments in automatic text retrieval. Science, 974--979 (1991)
16. Schone, P., Jurafsky,D.: Is knowledge-free induction of multiword unit dictionary headwords a solved problem? In Lee,L., Harman, D. (eds). Conference on Empirical Methods in Natural Language Processing, pp 100—108 (2001)
17. Stanford Tagger, http://nlp.stanford.edu/software/index.shtml
18. Unified Medical Language System (UMLS), http://www.nlm.nih.gov/research/umls/
19. Vivaldi, J., Rodriguez, H.: Improving term extraction by combining different techniques. Terminology. 7, 31--48 (2001)
20. Wright, S.E., Budin, G.: Term Selection: The Initial Phase of Terminology Management. Handbook of Terminology Management, 1, pp. 13 -- 23 (1997)
21. Wu, F., Weld, D.,S.: Autonomously semantifying Wikipedia. In: sixteenth ACM conference on Conference on information and knowledge management, pp. 41--50. ACM, New York, USA (2007)
22. Xu, F., Kurz, D., Piskorski, J., Schmeier, S.: Term Extraction and Mining of Term Relations from Unrestricted Texts in the Financial Domain. In: Abramowicz, W. (ed) Businesss Information Systems. Proceedings of BIS 2002, Poznan, Poland (2002)