Extracting Meronymy Relationships from Domain-Specific, Textual Corporate Databases

Ashwin Ittoo, Gosse Bouma, Laura Maruster, Hans Wortmann

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

Abstract. Various techniques for learning meronymy relationships from open-domain corpora exist. However, extracting meronymy relationships from domain-specific, textual corporate databases has been overlooked, despite numerous application opportunities particularly in domains like product development and/or customer service. These domains also pose new scientific challenges, such as the absence of elaborate knowledge resources, compromising the performance of supervised meronymy-learning algorithms. Furthermore, the domain-specific terminology of corporate texts makes it difficult to select appropriate seeds for minimally-supervised meronymy-learning algorithms. To address these issues, we develop and present a principled approach to extract accurate meronymy relationships from textual databases of product development and/or customer service organizations by leveraging on reliable meronymy lexico-syntactic patterns harvested from an open-domain corpus. Evaluations on real-life corporate databases indicate that our technique extracts precise meronymy relationships that provide valuable operational insights on causes of product failures and customer dissatisfaction. Our results also reveal that the types of some of the domain-specific meronymy relationships, extracted from the corporate data, cannot be conclusively and unambiguously classified under well-known taxonomies of relationships.

Keywords: Meronymy, part-whole relations, natural language processing.

1 Introduction

Meronymy is an important semantic relationship that exists between a part and its corresponding whole [2]. Approaches exist for automatically learning meronymy relationships [1,2,3,10,12] from open-domain corpora (e.g. SemCor) to support traditional natural-language-processing (NLP) applications like question-answering. However, none of them targeted textual databases in corporate domains, despite the numerous application opportunities. Product development and/or customer service (PD-CS) are such corporate domains in which meronymy is of fundamental importance as a central structuring principle in artifact design [12]. Meronymy relationships harvested from PD-CS textual databases could support activities like product quality assurance [8], and generating domain ontologies and bills-of-materials from product descriptions.

Our primary motivation in learning meronymy relationships from textual PD-CS databases is that they encode valuable operational knowledge that PD-CS organizations can exploit to improve product quality and ensure customer satisfaction. Meronymy relationships in the PD-CS domains are useful for uncovering causes of customer dissatisfaction that are implicitly expressed in complaint texts. For example, in "…dots *appear on* the screen…", the meronymy pattern "appear-on" expresses a customer's dissatisfaction at dots being shown *as part of* the screen display. These types of customer dissatisfaction causes, which are lexically realized with subtle, meronymy patterns (e.g. "appear-on", "available-on"), are harder to detect than those which are unequivocally expressed by customers in their complaints, such as "screen does not work". Meronymy relationships mined from PD-CS data also enable service engineers to efficiently diagnose product failures and devise remedial measures. For example, the meronymy pattern "located-at" in "switch *located at* panel 1 is broken" helps engineers to precisely identify defective components in products. Other meronymy relationships, as in "calibration is *part of* upgrade", provide pertinent information about actions performed by engineers and about service/warranty packages to management of PD-CS organizations.

Our interest in mining meronymy relationships from textual databases in corporate domains is also attributed to the challenges they pose to extant approaches. A major challenge in many corporate environments is the absence of readily-usable knowledge resources (e.g. WordNet) to support supervised meronymy learning approaches [2,3,12]. Minimally-supervised algorithms [1,10] alleviate the need for elaborate knowledge resources. Instead, they rely on a small initial set of part-whole instance pairs (e.g. engine-car), known as seeds, and extract the meronymy relationships that connect co-occurring instances in a corpus. However, defining seeds over corporate texts is challenging. It requires proficiency in the domain-terminology to ensure that selected seeds are valid part-whole instance pairs, and to deal with terminological variations due to multiple corporate stakeholders (e.g. management, engineers and customers) using different terms to refer to a single concept. Seed selection from domain-specific corporate texts also requires prior knowledge of the textual contents to ensure that the selected part-whole instances co-occur in sentences so that the meronymy relationships instantiated by these co-occurring instances can be mined. This is in stark contrast to traditional open-domain corpora, which facilitate seed selection by offering an abundance of archetypal part-whole pairs that can reasonably be assumed to co-occur in sentences (e.g. engine-car, grape-wine). These challenges in learning meronymy relationships from textual corporate databases are compounded by the wide variety of lexical constructs that encode meronymy [4].

To address these issues, and support PD-CS organizations in creating better quality products, we develop and present in this paper a framework for automatically extracting accurate meronymy relationships from domain-specific, textual corporate databases. We realize our methodology in a prototype implemented as part of the DataFusion initiative[1]. DataFusion aims at facilitating product quality improvement

[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.

by using relevant information extracted from PD-CS databases to align customers' expectations to products' specifications.

Our core contribution is a principled approach to extract accurate meronymy relationships from domain-specific textual corporate databases. Our approach starts by harvesting reliable meronymy patterns from a large, open-domain corpus to circumvent the difficulties posed by domain-specific texts to relationship extraction. Targeting such a corpus also enables the wide-variety of meronymy patterns [4] to be learnt. The acquired patterns are then used to extract meronymy relationship triples from the domain-specific textual databases. To overcome the drawbacks of traditional surface-pattern representations, we formalize the patterns harvested from the open-domain corpus by using sophisticated syntactic structures. Results of evaluations performed on real-life databases provided by our industrial partners indicate that our approach accurately uncovers valuable insights on causes of customer complaints and product failures that were implicitly encoded in meronymy constructs in the data. As an ancillary contribution, we also show that some of the domain-specific relationships identified from the corporate data are not conclusively classifiable by well-known taxonomies of meronymy relationships such as the taxonomy of Winston et al. [13].

This paper is organized as follows. Section 2 presents and compares related work. We present our approach in Section 3. Experiments are described in Section 4, before concluding and highlighting areas of future work in Section 5.

2 Related Work

Winston et al. [13] (Winston) developed a taxonomy of meronymy relationships based on psycholinguistics experiments on the linguistic usage of the term "part of". They mention six types of meronymy relationships: component-integral (e.g. engine-car), member-collection (e.g. soldier-army), portion-mass (e.g. metre-kilometre), stuff-object (e.g. grape-wine), place-area (e.g. Groningen-Netherlands), and feature-activity (e.g. chewing-eating).

Algorithms to automatically acquire meronymy relationships from texts are either (fully-)supervised or minimally-supervised. In the supervised approaches presented in [2,3], meronymy relationships connecting WordNet instances are manually extracted from 200,000 sentences of the SemCor and L.A. Times corpora. The relationships are used to train a decision-tree classifier, which achieves a precision of 80.95% and a recall of 75.91% in predicting whether previously unseen constructs encode meronymy. The supervised algorithm in [12] relies on 503 part-whole pairs, acquired from specialized thesauri, to extract 91 reliable part-whole patterns from web documents with a precision of 74%.

Minimally-supervised approaches [1,10] do not require external knowledge resources. The algorithm in [10] uses part-whole instance pairs (e.g. city-region) as initial seeds to extract meronymy surface-patterns from the Acquaint (5,951,432 words) and Brown (313,590 words) corpora. An iterative procedure bootstraps the patterns, and uses them to induce new part-whole instances and patterns. The precisions reported over the Acquaint and Brown corpora are respectively 80% and 60%. However, this approach requires large numbers of surface-patterns to be manually authored, and fails to detect long-range dependencies (relationships) between words in text. The minimally-supervised technique in [1] uses six “whole” instances (e.g. school, car) and infers their corresponding “parts” (e.g. room, engine) from the North American News Corpus (1 million words) with an accuracy of 55%.

Compared to the open-domain corpora (e.g. Acquaint) targeted by the above approaches, domain-specific corporate texts present new challenges yet to be addressed. The absence of knowledge resources (e.g. ontologies) in corporate environments compromise the performance of supervised algorithms. Selecting appropriate seeds from domain-specific texts to support minimally-supervised meronymy mining algorithms is also challenging. Furthermore, the types of meronymy relationships mined from domain-specific, corporate texts could be different from those mentioned in existing taxonomies [13]. We address these challenges by developing and presenting, in the next section, our novel framework to extract accurate meronymy relationships from domain-specific, textual corporate databases. Although we focus on the product development and customer service (PD-CS) domains, our approach can be considered generic enough to be applied in other corporat contexts.

3 Methodology for Meronymy Relationships Extraction

Our proposed methodology to learn meronymy relationships from domain-specific textual corporate databases consists of three major phases: Pattern Induction, Meronymy Pattern Selection and Meronymy Relationships Extraction, as depicted in Figure 1 (dotted objects represent inputs and outputs).

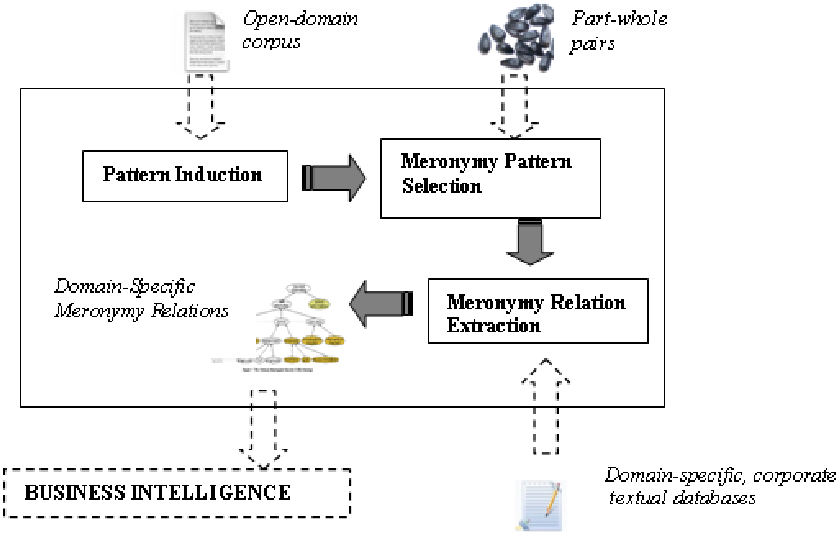

Open-domain corpus
Part-whole pairs
Pattern Induction
Meronymy Pattern Selection
Meronymy Relation Extraction
Domain-Specific Meronymy Relations
BUSINESS INTELLIGENCE
Domain-specific, corporate textual databases

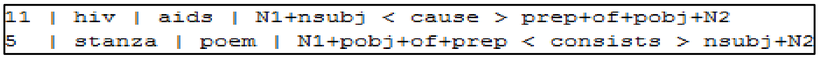
Fig. 1. Phases underlying the proposed methodology.

The Pattern Induction phase (Section 3.1) induces lexico-syntactic patterns from a large, open-domain and broad-coverage corpus known as the “learning-corpus”. Our rationale for initially targeting such a corpus is that it offers abundant typical and co-occurring part-whole instances (e.g. engine-car) that are suitable seeds for minimally-supervised meronymy learning algorithms. Thus, it circumvents the difficulties of seed selection over domain-specific corporate texts. Targeting a large corpus, following the predication that “it is useful to have more data than better data” [6], also captures the wide variety of patterns that encode meronymy. A minimally-supervised algorithm in the Meronymy Pattern Selection stage (Section 3.2) determines which of the induced patterns express meronymy. The Meronymy Relationships Extraction phase (Section 3.3) then uses these patterns to extract meronymy relationships from the domain-specific texts.

3.1 Pattern Induction

Pattern induction starts by syntactically parsing the sentences of the learning-corpus to derive their parse-trees. Co-occurring instances (entities) in the sentences are then detected based on their parts-of-speech (PoS) by term-recognition filters [7]. Instead of representing the relationships between co-occurring instances with traditional surface-patterns as in [10], we adopt linguistically-sophisticated dependency paths. A dependency path is the shortest sequence of lexico-syntactic elements, i.e. the shortest lexico-syntactic pattern, connecting instances in their parse-trees. Dependency paths abstract from surface texts, and alleviate the manual authoring of large numbers of surface-patterns. They formally characterize relationships by capturing long-range dependencies between words regardless of position and distance in surface texts [11].

The output of this phase is a set of lexico-syntactic patterns, the instances they sub-categorize (connect) and statistics about the occurrence/co-occurrence frequencies as harvested from the learning-corpus. Sample patterns, instance pairs they connect and the co-occurrence frequencies of the pairs and patterns are in the 4th, 3rd, 2nd and 1st columns of Figure 2 (N1 and N2 are generic markers to represent the actual instances). These patterns denote various types of relationships (including meronymy) between their instance pairs. The 1st pattern in Figure 2 depicts a “cause-of” relationship between instances “hiv” and “aids” in the learning-corpus, as in “hiv (is the) cause of aids”. The 2nd pattern denotes a meronymy relationship between “stanza” and “poem”, as in “poem consists of stanza”.



```
11 | hiv | aids | N1+nsubj < cause > prep+of+pobj+N2
5  | stanza | poem | N1+pobj+of+prep < consists > nsubj+N2
```

Fig. 2. Sample lexico-syntactic patterns extracted, representing various semantic relationships.

3.2 Meronymy Pattern Selection

The Meronymy Pattern Selection phase uses a minimally-supervised algorithm [10] that takes as input typical part-whole instance pairs (seeds), e.g. engine-car, to determine which of the previously acquired lexico-syntactic patterns express meronymy. Our algorithm considers a pattern to encode meronymy if it sub-categorizes any of the seeds in the learning-corpus. We rank the inferred meronymy patterns according to their reliability scores *r(p)*, computed by equation (1). It measures the reliability of a pattern *p* in expressing meronymy as its average strength of association with part-whole instance pairs *i*, weighted by the reliability *r(i)* of these instance pairs. Initially, the reliability of the seeds is set to 1 (i.e. r(i) =1). In equation (1), pmi(i,p) is the point-wise mutual information between an instance pair *i=x-y* (e.g. *i*= engine-car) and a meronymy pattern *p* (e.g. "consists of"). It is calculated by equation (2), where |x,p,y| is the probability that *p* sub-categorizes *x* and *y*, and * represents any character.

$$r(p) = \frac{\sum_{i \in I} \left(\frac{pmi\ (i,p)}{\max\ _{pmi}} \times r(i) \right)}{|\ I\ |} \tag{1}$$

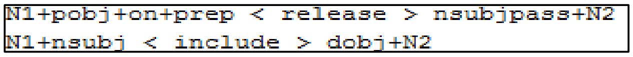
$$pmi(i,p) = \log \frac{|\ x,p,y\ |}{|\ x,*,y\ ||\ *,p,*\ |} \tag{2}$$

After calculating their reliability, the top-*k* most reliable meronymy patterns are bootstrapped, and used to induce new part-whole instance pairs from the learning-corpus. The reliability of these instance pairs, r(*i)*, analogous to the patterns' reliability, is computed using equation (3), where |P| is the set of top-*k* meronymy patterns selected earlier.

$$r(i) = \frac{\sum_{p \in P} \left(\frac{pmi\ (i,p)}{\max\ _{pmi}} \times r(p) \right)}{|\ P\ |} \tag{3}$$

The top-*m* most reliable part-whole instance pairs are then bootstrapped for inferring other meronymy patterns. This recursive procedure of learning reliable meronymy patterns from reliable part-whole instance pairs is repeated until *t* patterns are extracted. The values of *k*, *m* and *t* are experimentally determined.

This phase yields a set of reliable meronymy-encoding lexico-syntactic patterns. Two example patterns, which read as "N2 released on N1" and "N1 includes N2" are shown in Figure 3. N1 and N2 are generic slot-fillers, respectively representing the "whole" (e.g. car) and the "part"(e.g. engine) instances.



```
N1+pobj+on+prep < release > nsubjpass+N2
N1+nsubj < include > dobj+N2
```

Fig. 3. Reliable meronymy-encoding lexico-syntactic patterns identified.

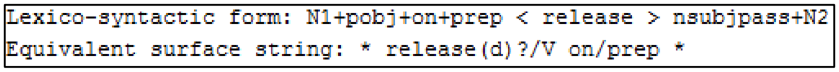
3.3 Meronymy Relationships Extraction

The Meronymy Relationships Extraction stage extracts meronymy relationships as triples from the domain-specific, textual corporate databases. Each triple consists of an instance pair, in the corporate data, that is sub-categorized by any of the reliable meronymy patterns harvested earlier.

We applied standard linguistic pre-processing to the domain-specific texts by segmenting them into individual sentences, tokenizing the sentences, and determining the parts-of-speech (PoS) and lemmas of the word-tokens. The sentences are not syntactically parsed since they are short and ungrammatical, as in “image not available on console”, and do not involve long-range dependencies between terms (e.g. “image”, “console”) and patterns (e.g. “available-on”). Furthermore, errors in the syntactic parse-trees of ungrammatical sentences can compromise our overall performance in mining meronymy relationships.

We identify instances (terms) from the domain-specific texts using the Textractor algorithm in [5], and we select the most frequent ones as domain-relevant instances. (Term identification is not discussed further in this paper).

Next, we re-write the previously acquired meronymy lexico-syntactic patterns (Section 3.2) into their equivalent surface-strings to facilitate their detection in the corporate texts, which were not syntactically parsed. Our automatic re-writing procedure starts at the patterns’ subject, indicated by “nsubj” or “nsubjpass”. It then collects the patterns’ roots, enclosed in “<” and “>”, and prepositional modifiers, indicated by “prep”, to generate the corresponding surface-strings. Figure 4 shows a meronymy lexico-syntactic pattern and its equivalent surface-string. In this example, “V” is the PoS for verbs, and the regular-expression operator “?” makes the token “d” optional to cope with inflected verb forms (e.g. past-tense of regular verbs).



```
Lexico-syntactic form: N1+pobj+on+prep < release > nsubjpass+N2
Equivalent surface string: * release(d)?/V on/prep *
```

Fig. 4. Lexico-syntactic pattern and its surface-string equivalent.

Finally, we extract occurrences of the meronymy patterns and the instance pairs that they connect in the domain-specific texts as meronymy relationship triples.

4 Experimental Evaluation

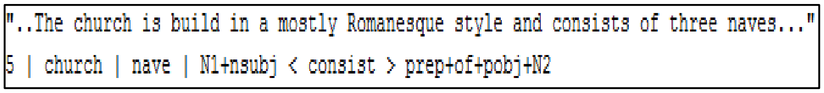
We conducted experiments to evaluate the performance of our approach in extracting meronymy relationships from real-life databases of our industrial partners. The data contained 143,255 textual narratives of customer complaints captured at helpdesks, and repair actions of service engineers.

4.1 Pattern Induction

We chose the English Wikipedia texts as our learning-corpus to infer reliable meronymy patterns that are subsequently used to extract meronymy relationship triples from the domain-specific databases. With two million articles in 18 Gb of texts, encompassing a broad spectrum of topics [9], the Wikipedia corpus satisfies the desiderata of being large, open-domain and broad-coverage. It offers abundant general part-whole instances (e.g. engine-car) that co-occur in its sentences, and thus, facilitates seeds selection for minimally-supervised meronymy mining algorithms. Its broad-coverage also ensures that it captures the wide variety of meronymy patterns.

We parsed a copy of the English Wikipedia corpus [14] (about 400 million words) using the Stanford parser [15], and extracted 2,018,587 distinct lexico-syntactic patterns, and 6,683,784 distinct instance pairs. Statistics about their occurrence and co-occurrence frequencies were also computed.

Figure 5 shows a Wikipedia sentence describing various relationships between instances (terms) "church", "Romanesque style", and "naves". The bottom row (last column) depicts the corresponding lexico-syntactic pattern that we derived to formalize the relationship between instances "church" and "naves". Lemmatized instances, and their co-occurrence frequency with the pattern are respectively in the 3rd, 2nd and 1st columns. N1 and N2 are generic slot-fillers for the "part" and "whole" instances. As can be seen, our lexico-syntactic patterns concisely encode the semantic relationships between instance pairs regardless of their distance and position in texts.



```
"..The church is build in a mostly Romanesque style and consists of three naves..."
5 | church | nave | N1+nsubj < consist > prep+of+pobj+N2
```

Fig. 5. Wikipedia sentence and corresponding dependency path extracted.

4.2 Meronymy Pattern Selection

We implemented a minimally-supervised meronymy learning algorithm similar to [10], and defined a seed set of 143 typical part-whole instance pairs that are likely to co-occur in Wikipedia sentences. Seeds were equally distributed across the six types of meronymy relationships mentioned in [13]. Examples include engine-car, wine-grape, director-board, m-km, municipality-town, and paying-shopping. In each of its iteration, our algorithm bootstraps the top-*k* most reliable meronymy patterns to induce new part-whole instance pairs, and the top-*m* most reliable part-whole instance pairs to infer new meronymy patterns, until *t* patterns are extracted. In our experiments, we set $k = |P| + 5$ and $m=|I|+20$, where |P| and |I| are respectively the number of patterns and instance pairs from the previous iterations. The largest set of most reliable meronymy patterns (i.e. optimal precision and recall) was obtained in the 45th iteration, which yielded 162 patterns (i.e. t=162). Patterns introduced in subsequent iterations were noisy and irrelevant. Incrementing *k* with more than 5 patterns and *m* with more than 20 instance pairs in each iteration resulted in smaller sets of reliable patterns. Smaller values for the pattern and instance increments did not have significant effects on the performance. However, the largest set of most reliable meronymy patterns was then obtained in later iterations. Table 1 shows the five most reliable meronymy lexico-syntactic patterns inferred by our approach from Wikipedia, and their possible linguistic interpretations.

Table 1. Inferred meronymy lexico-syntactic patterns and their interpretations.

Meronymy Pattern	**Linguistic Interpretation**

N1+nsubj < include > dobj+N2	Whole *includes* Part
N1+nsubj < contain > dobj+N2	Whole *contains* Part
N1+nsubj < consist > prep+of+pobj+N2	Whole *consists-of* Part
N1+pobj+on+prep <release> nsubjpass+N2	Part *released-on* Whole
N1+pobj+in+prep < find > nsubjpass+N2	Part *found-in* Whole

4.3 Meronymy Relationships Extraction

We pre-processed the domain-specific corporate texts to determine the parts-of-speech tags and lemmas of the word-tokens using the Stanford tagger and morphological analyzer [15]. The most frequently occurring terms were then identified as relevant domain instances using the Textractor algorithm in [5]. We also automatically transformed our meronymy lexico-syntactic patterns (Section 4.2) into their equivalent surface-strings, as described in Section 3.3, to facilitate their detection in the domain-specific texts, which were not syntactically parsed.

Out of our 162 distinct meronymy patterns, 63 were found to connect the domain-specific instance pairs in the corporate texts. The instance pairs-patterns combinations yielded 10,195 domain-specific meronymy triples that we extracted. Examples are in Table 2. Meronymy patterns are shown in the 2nd column. The 1st column indicates the patterns' occurrence frequency in the domain-specific texts. Meronymy triples extracted, of the form <*meronymy pattern, part-instance, whole-instance*>, are in column 3. Relationships that cannot be conclusively classified in existing taxonomies of meronymy relationships are marked with "*". Lexical manifestations of the patterns and part-whole instance pairs in the corporate texts are illustrated in the last column.

Table 2. Sample domain-specific meronymy triples extracted from corporate data.

Freq (%)	**Pattern**	**Triple**	**Example**
71-75	Available-on	<available-on, image , monitor>*	Image not *available on* monitor
	Show-in	<show-in, artifact, image>*	Artifact *shown in* image
66-70	Include	<include, calibration, corrective action >	Corrective action *includes* calibration
	Perform-in	<perform-in, reboot, configuring>	Reboot *performed in* configuring
51-65	Locate-in	<locate-in, adaptor board, pc>	Adaptor board l*ocated in* PC
	Find-in	<find-in, blown fuse, settop box>	Blown fuse *found in* settop box
	Come-from	<come-from, noise, generator>*	Noise *comes from* generator

1-30	Reach	<reach, c-arm, table base>	C-arm unable to *reach* table base
	Release-on	<release-on, software upgrade, processor>*	Software upgrade *released on* processor

The most frequently extracted domain-specific meronymy relationships involved patterns like “appear-on/in”, “show-in” and “available-on”. They accounted for 71-75% of our extracted triples. These types of relationships, between intangible parts (e.g. image) and their wholes, are not defined in Winston’s taxonomy of meronymy relationships [13]. They are relevant to PD-CS organizations as they enable the identification of product malfunctioning that are implicitly expressed in customer complaint texts, for example in “Horizontal line *appears on* screen”. The high frequency of such relationships can be attributed to the contents of the data we investigated, which pertained to video/imaging equipment. The next most frequent relationships that we identified were realized with patterns such as “include” and “perform-in”. They occurred in 66-70% of our triples, and related activities (processes) to their constituent phases (steps). These relationships correspond to the “Feature-Activity” relationship type in Winston’s taxonomy. They provide pertinent information about repair actions of engineers, and about service/warranty packages to management of PD-CS organizations, as in “Reboot *performed in* configuring”. Mereotopological relationships (i.e. 3-D containment) [8] were also frequent in our data, constituting around 51-65% of the extracted triples. These relationships, which exist between parts and their containers/regions, were manifested with patterns like “find in/on/at”, “contains”, and “consist-of”. They can be classified under the “Component-Integral” relationship type of Winston’s taxonomy. Mereotopological relationships, such as “Blown fuse *found in* settop box”, enable PD-CS engineers in precisely identifying product components that fail, and in efficiently devising corresponding remedial measures. Meronymy relationships involving patterns like “reach”, “come-from”, “release-on” and “incorporate-in” were rarer, accounting for at most 30% of our extracted triples. The meronymy pattern “reach” was found to relate discrete (physical) parts to their wholes. These relationships are classifiable under the “Component-Integral” type of Winston’s taxonomy. They are suitable in PD-CS organizations for determining causes of product failures, as in “C-arm unable to *reach* table base”. Meronymy relationships involving the pattern “come-from” related parts to their originating wholes. These relationships are not classifiable in Winston’s taxonomy, and are useful in identifying sources of customer dissatisfaction as in “Noise *comes from* generator“. Patterns like “incorporate-in” or “release-on” were found to relate intangible information artifacts (e.g. software) in the domain-specific texts. These relationships do not correspond to any of the types mentioned in Winston’s taxonomy. They provide pertinent information on patches or upgrades released in existing software applications, and can be applied in software versioning in PD-CS organizations.

The remaining 99 (out of 162) open-domain meronymy patterns were not found in our domain-specific, corporate texts since they are unlikely to occur in narratives of customer complaints and repair actions of engineers in PD-CS databases. Examples of such patterns with zero frequency are “divide-into”, “character-from”, “publish-in”, “member in/of”, “add-to”, “record in/on”, and “collection-of”.

We manually evaluated 2500 of the extracted meronymy relationship triples with the help of industrial domain experts since no gold-standard knowledge resources were available. Our evaluation sample consisted of relationships which were identified with a frequency of at least 30% from the domain-specific texts. The relationships were chosen such that they were equally distributed across the various frequency ranges of Table 2. Less frequent relationships, i.e. with frequency below than 30%, (e.g. <*released-on, software upgrade, processor*>) were not taken into account since they were very precise, and could positively bias our evaluation results. We calculated the precision of the meronymy triples according to equation (4), where *true_positive* is the number of valid domain-specific meronymy triples that our approach identified, and *false_positive* is the number of triples suggested by our approach, but deemed invalid by the domain experts.

$$\text{Precision} = \frac{\text{true\_positive}}{\text{true\_positive} + \text{false\_positive}} \qquad \textbf{(4)}$$

Our manual evaluations identified 2023 *true_positives*, and 477 *false_positives.* The majority of *false_positives* involved the pattern “make-in”, which did not always encode meronymy as in “monitor *made in* factory”. The overall precision of our approach was thus 81%. This result compares favorably with the precisions reported by state-of-the-art techniques that mine meronymy relationships from open-domain corpora, such as Pantel’s [10] 80%, Girju’s [2] 81% and van Hage’s [12] 74%. We did not compute the recall measure as the number of valid meronymy relationships in the corporate databases was unknown. However, we can expect a reasonably high recall score since the patterns used to extract the meronymy relationships from the domain-specific texts were harvested from a much larger corpus.

5 Conclusion and Future Work

We have described the design and implementation of a principled approach to learn precise meronymy relationships from domain-specific, textual, corporate databases. Our approach efficiently addresses the challenges of meronymy relationships extraction from domain-specific corporate texts by leveraging on linguistically sophisticated meronymy lexico-syntactic patterns harvested from a large, open-domain corpus. Evaluations on real-life, industrial databases indicate that our approach uncovers, with high precision, valuable insights on causes of customer complaints and product failures that are implicitly encoded in meronymy constructs. Our results also reveal that the types of some of the domain-specific meronymy relationships extracted from corporate data cannot be unambiguously classified in Winston’s well-known taxonomy of meronymy relationships. Future work will involve learning ontologies from the extracted information to semantically integrate heterogeneous, but complementing, data sources to support business intelligence activities. We will also investigate the extraction of other semantic relationships that are relevant in the product development and/or customer service domains, such as “caused-by” to discover the “causes” of product failures. Our other research efforts will be dedicated towards a deeper examination of the identified domain-specific meronymy relationships that could not be classified in Winston’s taxonomy.

Acknowledgements
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. Berland, M., Charniak, E.: Finding parts in very large corpora. In: 37th Annual Meeting of the Association for Computational Linguistics, pp. 57--64. University of Maryland (1999)
2. Girju,R., Badulescu, A., Moldovan,D.: Automatic Discovery of Part-Whole Relations. Computational Linguistics. 32, 83--135 (2006)
3. Girju,R., Badulescu, A., Moldovan,D.: Learning semantic constraints for the automatic discovery of part-whole relations. In: Conference of the NAACL on HLT, pp. 1--8. Association for Computational Linguistics, Morristown, NJ (2003)
4. Iris, M., Litowitz, B., Evens, M.: Problems with part-whole Relation. In: Evens, M. W. (eds) Relational Models of the Lexicon: Representing Knowledge in Semantic Networks. pp. 261--288. Cambridge University Press, Cambridge (1988)
5. Ittoo, A., Maruster L., Wortmann H., Bouma, G.: Textractor: A Framework for Extracting Relevant Domain Concepts from Irregular Corporate Textual Datasets. In: Abramowicz, W., Tolksdorf, R. (eds.) BIS 2010. LNBIP, vol. 47, pp.71--82. Springer, Heidelberg (2010)
6. Jijkoun, V., de Rijke, M., Mur, J.: Information extraction for question answering: improving recall through syntactic patterns. In 20th Intl Conference on Computational Linguistics. Association for Computational Linguistics, Morristown, NJ, USA
7. Justeson, J., Katz, S.M.: Technical terminology: some linguistic properties and an algorithm for identification in text. Natural Language Engineering,1,9--27 (1995)
8. Keet, C.M., Artale, A.: Representing and reasoning over a taxonomy of part-whole relations. Appl. Ontol. 3, 91--110(2008)
9. Medelyan, O., Milne, D., Legg, C., Witten, I.H.: Mining meaning from Wikipedia. International Journal of Human-Computer Studies. 67, 716--754 (2009)
10. Pantel, P., Pennacchiotti, M.: Espresso: leveraging generic patterns for automatically harvesting semantic relations. In: 21st Intl Conference on Computational Linguistics, pp.113--220. Association for Computational Linguistics, Morristown, NJ, USA (2006).
11. Shen, D., Kruijff, K. G-J., Klakow, D.: Exploring syntactic relation patterns for Question-Answering. In: Dale, R., Wong, K-F., Su, J., Oi, Y.W. (eds). IJCNLP 2005. LNAI, vol. 3651, pp. 507--518. Springer, Heidelberg (2005)
12. van Hage, W.R., Kolb H., Schreiber, G.: A method for learning part-whole relations. In: Cruz, F.I., Decker, S., Allemang, D., Preist., Schwabe, D., Mika, P., Uschold, M., Aroyo, L. (eds.) The Semantic Web-ISWC 2006. LNCS, vol. 4273, pp. 723--735. Springer, Heiderlberg (2006)
13. Winston, M., Chaffin, R., Hermann, D.: A taxonomy of part-whole relations. Cognitive Science. 11, 417--444 (1987)
14. English Wikipedia (2007-08-02), ISLA, University of Amsterdam, http://ilps.science.uva.nl/WikiXML/
15. Stanford Natural Language Processing Group, http://nlp.stanford.edu/software/index.shtml