A Lexical Resource for the Identification of "Weak Words" in German Specification Documents

Jennifer Krisch[1,2], Melanie Dick[2], Ronny Jauch[2] and Ulrich Heid[2]
[1]Daimler AG Stuttgart, [2]Universität Hildesheim –
Institute for Information Science and Natural Language Processing,
jennifer.krisch@daimler.com, {ronny.jauch, melanie.dick, ulrich.heid}@uni-hildesheim.de

Abstract

We report on the creation of a lexical resource for the identification of potentially unspecific or imprecise constructions in German requirements documentation from the car manufacturing industry. In requirements engineering, such expressions are called "weak words": they are not sufficiently precise to ensure an unambiguous interpretation by the contractual partners, who for the definition of their cooperation, typically rely on specification documents (Melchisedech, 2000); an example are dimension adjectives, such as *kurz* or *lang* ('short', 'long') which need to be modified by adverbials indicating the exact duration, size etc. Contrary to standard practice in requirements engineering, where the identification of such weak words is merely based on stopword lists, we identify weak uses in context, by querying annotated text. The queries are part of the resource, as they define the conditions when a word use is weak. We evaluate the recognition of weak uses on our development corpus and on an unseen evaluation corpus, reaching stable F_1-scores above 0.95.

Keywords: requirements engineering, weak words, corpus-based methods

1. Introduction

In the automotive industry, requirements specifications are used as a communication medium between contractor and supplier. Requirements specifications contain textual requirements which are "conditions or capabilities that must be met or possessed by a system or system component". These texts serve "to achieve a consensus among the stakeholders [...] to minimize the risk of delivering a system that does not meet the contractor's desires and needs" (Pohl and Rupp, 2011). Requirements texts should be unambiguous and easy to understand and they should allow consensual interpretation. The requirements are produced by many different authors (typically domain experts), without possibilities of using controlled language and author control systems; consequently, checks for ambiguities, for underspecified constructions etc. are needed before the requirements are made available to suppliers. Requirements that pose problems of ambiguity, underspecification or understandability are thus given back to the author, for correction, before they are made official. Our resource is part of a system for such checks.

A problem in the writing of requirements documents, which is well-known in the field of requirements engineering is the use of potentially vague adjectives. In requirements engineering they are called "weak words". Whether a weak word may lead to different interpretations depends on the syntactic and lexical context in which it is used. The class of weak words includes scalable adjectives (for example *klein* ('small')) and limit adjectives (for example: *unterschiedlich* ('different')), cf. Paradis (1997). Bierwisch (1989) further subdivides scalable adjectives into dimensional adjectives and evaluative adjectives. In this work we concentrate on dimensional and limit adjectives, as evalative adjectives are rare in our data.

Another example of a weak word is the dimensional adjective *lang* ('long'). When an author uses the weak word *long* in a context like *long time period*, readers differ about what *long* in this context actually means. However, not every use of a weak word lemma is necessarily imprecise or ambiguous. Weak words only lead to ambiguous requirements if they appear in certain contexts. In constructions like *a long cycle of 3 seconds* the weak word does not allow alternative interpretations and is thus not "problematic"; in such a case, the author of the requirement need not (and should not) be warned, while she/he must be invited to correct an imprecise statement like *a long cycle*.

Our objective is to identify uses of weak words which cause different possibilities of interpretation, as well as unproblematic uses. Our resource contains a set of context patterns for each potential weak word, both problematic and unproblematic. While the recognition of unproblematic contexts reduces the amount of "false alarms" in the feedback to authors, we distinguish two classes of problematic weak word uses and offer feedback to the authors accordingly: cases that must be changed ("defect") and cases where the author must verify that all necessary information is present in the document ("effect").

We also show how the approach to weak word analysis can be extended to grammatical phenomena. An evaluation on a total of over 6,400 sentences (4,065 in the development corpus, and 2,394 in an unseen evaluation corpus) has shown an F_1-score of 0.98 for "good" and of 0.89 for "bad" requirements.

In Section 2 we describe the data used; Section 3 summarizes the methodology both of the construction of context patterns ("rules") and of the subsequent evaluation. In Section 4, we report our evaluation results and in Section 5, we discuss an extension to grammatical phenomena. We conclude in Section 6. In Section 7, we give an outlook on future work.

2. Data and pre-processing

2.1. Corpus data

"Requirements for an automotive component, like an electronic control unit (ECU), cover many areas: functionality, performance, diagnosis functionality, material, environmental conditions, electromagnetic compatibility, and also process related facets like logistic processes and documentation" (Krisch and Houdek, 2015). The requirements specifications used for the development of our resource are retrieved from different domains: interior electronics, mechanical components, powertrain, telematics. From a broad range of requirements documents, we built a German and an English corpus. The German corpus consists of 213,550 sentences with 2,502,220 tokens which was divided into a development set with 101,935 sentences and an evaluation set with 111,615 sentences. The English corpus consists of 164,562 sentences and 2,240,839 tokens which was divided into two subcorpora as well. The development set consists of 86,004 sentences and the evaluation set of 78,558 sentences.

2.2. Linguistic corpus annotation

For the annotation of the corpora we used several tools. The requirements were exported from a standard requirements engineering database and saved in a format for tokenizing (Krisch, 2013). Further annotations were represented in the format defined in CoNLL-2009 (Hajič et al., 2009). We use a specific tokenizer (Krisch, 2013) and make use of *mate tools* (Bohnet, 2009; Björkelund et al., 2010) for lemmatization and parsing; for part-of-speech tagging and morphological tagging we used *MarMot* (Müller et al., 2013). The trainable tools were used without extra training on the specification texts, i.e. with the standard models acquired from news texts.

After the linguistic annotation, the resulting corpus was converted into a format which can be processed by the search and retrieval tool Corpus Workbench, CWB (Evert and Hardie, 2011; Evert, 2010; Krisch, 2013).

The corpus also contains word, sentence and requirement identifiers.

2.3. Weak word candidate list

We started from a weak word list provided by the car manufacturer; its lemmas can have weak uses. Standard technology in requirements engineering would use such lists as "stopword lists" and warn requirements authors against each sentence containing one of the lemmas (or one of their forms). We expanded this list by adding GermaNet synset members for each item; to keep the size of the candidate lists manageable, we first addressed the 55 candidates with a frequency of more than 10 in the development corpus.

3. Methodology

3.1. Rule development

We rely on contextual clues to find problematic uses of weak words as well as unproblematic uses. For example, adjectives such as *extrem(e), minimal, maximal* require a quantification to make up for a precise context (cf. (1) to (3)).

(1) [...] *eine extreme Bergfahrt mit mehr als 20% Steigung.*
[...] an extreme ascending drive with a gradient of more than 20%

(2) *Die mininmale Prüftemperatur beträgt 40°C.*
The minimum test temperature is 40°C.

(3) *Die maximale Dicke darf 3 mm nicht überschreiten.*
The maximum thickness shall not exceed 3 mm.

In the example cases, if the clue (here: the quantification) is present, the respective noun groups (e.g. *extreme Bergfahrt*) are unproblematic. Otherwise the author is invited to provide the necessary specification. Examples of insufficient specificity are given in (4) and (5):

(4) *Es muss eine **maximale** Eigenerwärmung erzeugt werden.*
A maximal self-warming must be provided.

(5) [...] *die Energiebereitstellung der []–Batterie bei tiefen Temperaturen ist **extrem** wichtig.*
[...] with low temperatures, the energy provision of the []–battery is extremely low.

Rule development is so far manual; while certain patterns are emerging as typical indicators for imprecise contexts (e.g. indefinite noun phrases, dimensional adjectives without quantification etc.), too few such clues seem to be generalizable for e.g. learning-based methods to be applied.

Given contextual variability and the three German word order models, rules are formulated as series of alternatives. They also keep track of "exceptions", e.g. to avoid that specialized terms lead to unwanted correction proposals: the term *leichtes Nutzfahrzeug* ('light utility vehicle'), for example, would otherwise fulfill the condition for being "corrected" (quantification of *leicht* lacking).

The rules attached to the lexical entries specify both acceptable ("ok") and truly problematic cases ("defect"). The third category ("effect"), is assigned to stylistically infelicitous contexts or to items where reference to a (potentially undefined) parameter is made. This tripartite classification also supports more vs. less rigid checking with more vs. less cases signaled to the authors.

Rules for all three cases would not need to be formulated explicitly, at least not in principle; in a strongly recall-oriented strategy focusing on problematic uses, we could restrict the resource to a modeling of acceptable cases only, and we could signal all other cases as problematic; but we would then lose the distinction of "defect" vs. "effect", as well as possibilities to suggest corrections based on the error types.

3.2. Evaluation

The development set served as a basis for the documentation of the weak words: all sentences with occurrences of weak word candidate lemmas (4,065 sentences) were extracted and manually classified into the three types ("ok", "effect", "defect"). These data served as a gold standard for the evaluation of the rule sets. We assessed both, rules for unproblematic cases and rules for the merged set of "effect" and "defect" cases, individually. We report individual precision, recall and F_1-measure figures for these sets, as well as for a random selection of individual weak words.

Another evaluation of the rule set was made on unseen data, the evaluation set (2,394 sentences with occurrences of weak word candidate lemmas). We calculated precision, recall and F_1-score (Manning and Schütze, 1999) for "good requirement" and "bad requirement", individually, adopting two different perspectives (cf. Figure 1). In order to

Weak word	Occurrences		Development corpus			Evaluation corpus		
	Develop	Evaluation	Precision	Recall	F_1-score	Precision	Recall	F_1-score
alternativ – 'alternative'	175	66	1.00	0.98	0.99	1.00	0.92	0.96
unterschiedlich – 'different'	164	184	1.00	0.82	0.90	0.97	0.95	0.97
besonderer – 'special'	92	55	1.00	1.00	1.00	0.95	0.88	0.89
sicher – 'safe'	189	75	1.00	1.00	1.00	1.00	1.00	1.00
aktuell – 'current'	498	131	1.00	1.00	1.00	0.91	0.81	0.86
weit(er) – 'more/further'	547	166	1.00	1.00	1.00	0.97	0.90	0.93
einfach – 'simple'	79	40	1.00	0.90	0.95	0.80	1.00	0.89
klein – 'small'	748	547	1.00	1.00	1.00	0.99	1.00	0.99
schnell – 'fast'	66	28	0.85	0.85	0.85	1.00	0.92	0.96

Table 1: Quantitative evaluation of the evaluation of "good requirement": evaluation vs. development corpus , number of occurrences, precision, recall and f-measure

Weak word	Occurrences		Development corpus			Evaluation corpus		
	Develop	Evaluation	Precision	Recall	F_1-score	Precision	Recall	F_1-score
alternativ – 'alternative'	175	66	0.98	1.00	0.99	0.78	1.00	0.88
unterschiedlich – 'different'	164	184	0.95	1.00	0.97	0.99	0.99	0.99
verschieden – 'different"	56	34	1.00	1.00	1.00	0.98	1.00	0.99
besonderer – 'special'	92	55	1.00	1.00	1.00	0.59	0.83	0.69
sicher – 'safe'	189	75	1.00	1.00	1.00	1.00	1.00	1.00
aktuell – 'current'	498	131	1.00	1.00	1.00	0.67	0.83	0.75
weit(er) – 'more/further'	547	166	1.00	1.00	1.00	0.67	0.88	0.76
einfach – 'simple'	79	40	0.97	1.00	0.98	1.00	0.96	0.98
klein – 'small'	748	547	1.00	1.00	1.00	1.00	0.54	0.75
schnell – 'fast'	66	28	0.85	0.85	0.85	0.94	1.00	0.97

Table 2: Overview (random) of the evaluation of the development corpus and the evaluation corpus (bad requirements)

calculate the precision of "good requirement" we use the true positive value.

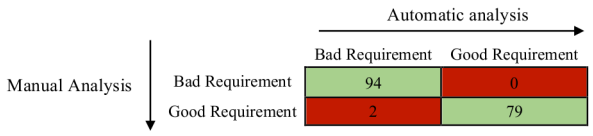

Figure 1: Automatic analysis – Manual Analysis

In Figure 1, this is the cell where the column and the row are both labeled "good requirement"; in the example case: 79. The false positive value is where the automatic analysis yields "good requirement", but the gold standard lists the context as a "bad requirement", in this case: 0. The precision value is thus:

$$\frac{79}{79+0} = 1.0$$

To compute recall, the false negative value is used where the rules produce "bad requirement" but the gold standard assigns "good requirement"; in this case: 2. So the recall value is:

$$\frac{79}{79+2} = 0.9753$$

In order to get the values for the "bad requirement" perspective, we inverse the perspective of "good requirement". Thus, for "bad requirement" the true positive value is 94, there are 2 false positives and 0 false negatives.

3.3. Sample results for weak words

Table 2 shows the same data for the perspective of bad requirements, ("effect" and "defect" taken together).

Table 3 and Table 4 show evaluation results over all evaluated weak words. The data indicate that the rule output is almost identical to the manual categorization. These results are partly due to the nature of the textual data: they are stylistically relatively homogeneous, including some repetition or sentences with small differences only. In part the results are also due to rather specific rules.

4. Quantitative evaluation

Table 1 shows data for good requirements. The first column contains ten randomly selected weak words, the next column is divided into two subcolumns ("Develop" and "Evaluation") showing the number of occurrences of each weak word in either corpus. The third and fourth column contain precision, recall and F_1-scores for each corpus.

Occurrences		Development corpus			Evaluation corpus		
Develop	**Evaluation**	**Precision**	**Recall**	**F_1-score**	**Precision**	**Recall**	**F_1-score**
4,065	2,394	0.99	0.99	0.99	0.97	0.98	0.94

Table 3: Precision, recall and F1-score over all evaluated weak words (good requirements)

Occurrences		Development corpus			Evaluation corpus		
Develop	**Evaluation**	**Precision**	**Recall**	**F_1-score**	**Precision**	**Recall**	**F_1-score**
4,065	2,394	0.97	0.95	0.96	0.86	0.93	0.88

Table 4: Precision, recall and F1-score over all evaluated weak words (bad requirements)

5. Extension to vague grammatical constructions

We have applied the methods described above to grammatical constructions with lexical indicators. An example are passive constructions (for English requirements, see Krisch and Houdek (2015)). Another one are German subjunctive forms, in particular *sollte(n)* ('should').
These forms may be used as "polite" replacements for *muss* ('must'), but in the requirement context they are ambiguous between an obligation and an option (like *must* vs. *may*). If these *sollte*-forms are used, a requirement is not specific enough, as the degree to which it is legally binding remains unclear.

The word *sollte* is however not always problematic: *sollte(n)* can also be used in verb-first conditional clauses without conjunction (*sollte die Lampe aufleuchten, so...* – 'should the lamp flash on, then...', instead of *wenn die Lampe aufleuchtet*, 'if the lamp flashes on') or, with an infinitive, as a substitute of an indicative word form in a standard conditional clause. An automated analysis must distinguish these cases from the problematic uses of *sollte*. To this end, the dependency annotation produced by *mate* is consulted: if *sollte* is the main verb (or coordinated with it), the sentence is problematic. Only these constructions must be presented for a correction. If there is a coordination and the word *sollte* is not a dependent of the conjunction (i.e. not at main clause level), the use of *sollte* is not critical and therefore the requirement is not shown to the author for revision.
We analyzed 687 requirements with the word *sollte*. 278 requirements were correctly classified as bad requirements ("must" vs. "should") and 401 were correctly classified as good requirements. Only eight requirements were misclassified. One misclassified requirement for example contains a full sentence within a parenthesis, and another one contains an embedded main clause.

6. Conclusion

We have presented a resource for the automatic checking of requirements documentation with respect to weak words and grammatical constructions which are not sufficiently precise to be unambiguously or fully specifically interpretable. The resource includes lexical items and rules for identifying both problematic and unproblematic cases in pos-tagged, lemmatized and dependency parsed text. The resource currently covers around 30 different weak word lemmas and 400 contextual identification rules. Another ca. 15 weak word lemmas have been analyzed, but no rules were written, as none of the contexts in the corpora proved to cause problems of interpretation.

7. Future Work

The resource is still under development, and the methodology applied is being tested on English data. The English weak word list contains 45 items of which 35 have a frequency over 50 occurences in the 1.4 million word corpus. As mostly adjective and adverb readings *(slow – slowly)* have the same properties, pairs of adverbs and adjectives can be treated together as one item.

First evaluation results on a smaller sample of English data are encouraging (see Table 5 and Table 6). The development of a similar resource for English will allow us to assess to what extent generalizations over German and English are possible. If applicable, we use these generalizations as modules to partially automate the process of rule writing for the English data.

Weak word	Occurrences		Development corpus			Evaluation corpus		
	Develop	Evaluation	Precision	Recall	F_1-score	Precision	Recall	F_1-score
adequate / adequately	49	37	1.00	0.83	0.91	1.00	0.80	0.89
bad / badly	64	40	0.95	0.95	0.95	1.00	0.79	0.88
regular / regularly	107	70	1.00	0.78	0.87	1.00	0.57	0.73
fast	121	105	1.00	0.93	0.96	0.95	0.95	0.95

Table 5: Quantitative evaluation of the evaluation of "good requirement" on English data

Weak word	Occurrences		Development corpus			Evaluation corpus		
	Develop	Evaluation	Precision	Recall	F_1-score	Precision	Recall	F_1-score
adequate / adequately	49	37	0.78	1.00	0.88	0.88	1.00	0.94
bad / badly	64	40	0.92	0.92	0.92	0.67	1.00	0.80
regular / regularly	107	70	0.88	1.00	0.94	0.85	1.00	0.92
fast	121	105	0.91	1.00	0.95	0.85	0.92	0.88

Table 6: First evaluation results of the development corpus and the evaluation corpus (bad requirements) on English data

8. Bibliographical References

Bierwisch, M. (1989). The semantics of gradation. *Dimensional adjectives: Grammatical Structure and Conceptual Interpretation*.

Björkelund, A., Bohnet, B., Hafdell, L., and Nugues, P. (2010). A high-performance syntactic and semantic dependency parser. In *Coling 2010: Demonstrations*, pages 33–36, Beijing, China, August. Coling 2010 Organizing Committee.

Bohnet, B. (2009). Efficient Parsing of Syntactic and Semantic Dependency Structures. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 67–72. Association for Computational Linguistics.

Evert, S. and Hardie, A. (2011). Twenty-first century corpus workbench: Updating a query architecture for the new millennium. In *Proceedings of the Corpus Linguistics 2011 conference*. University of Birmingham.

Evert, S. (2010). The IMS Open Corpus Workbench (CWB) CQP Query Language Tutorial.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., and Zhang, Y. (2009). The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, CoNLL '09, pages 1–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

Krisch, J. and Houdek, F. (2015). The Myth of Bad Passive Voice and Weak Words – An Empirical Investigation in the Automotive Industry. In *23rd International IEEE 2015, Requirements Engineering Conference (RE)*, pages 344–351. IEEE.

Krisch, J. (2013). Identifikation kritischer Weak-Words aufgrund ihres Satzkontextes in Anforderungsdokumenten. Diploma Thesis, University of Stuttgart.

Manning, C. D. and Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press.

Melchisedech, R. (2000). *Verwaltung und Prüfung natürlichsprachlicher Spezifikationen*. Ph.D. Thesis, University of Stuttgart, Shaker: Aachen, Germany.

Müller, T., Schmid, H., and Schütze, H. (2013). Efficient Higher-Order CRFs for Morphological Tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA, October. Association for Computational Linguistics.

Paradis, C. (1997). *Degree modifiers of adjectives in spoken British English*. Lund University Press.

Pohl, K. and Rupp, C. (2011). *Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam-foundation level-IREB compliant*. Rocky Nook, Inc.