# The Myth of Bad Passive Voice and Weak Words
## An Empirical Investigation in the Automotive Industry

Jennifer Krisch
Research and Development, Daimler AG
Hanns-Klemm-Straße 45
71034 Böblingen
Germany
Email: jennifer.krisch@daimler.com

Frank Houdek
Research and Development, Daimler AG
Wilhelm-Runge-Straße 11
89013 Ulm
Germany
Email: frank.houdek@daimler.com

***Abstract*—In requirements engineering literature dealing with natural language specifications, we usually find writing rules like '*avoid passive voice*' or '*do not use weak words*'. Adhering to such rules should result in understandable and unambiguous requirements. Passive voice, especially when used without an explicit actor, is considered to result in incomplete requirements. The usage of weak words is considered to result in imprecise requirements that are hardly testable. But is the inversion of the claim correct, i.e. does the violation of the writing rules result in problematic specifications? At least in our environment (the passenger car development of Mercedes-Benz) we observe that authors often use passive voice, and there are many requirements containing weak words. To answer this question, we conducted an empirical investigation whose results we report in this paper. The results of this investigation are quite surprising: The use of passive voice, even when the actor is missing, is almost never problematic, as the missing information (the actor) can in most cases easily derived from the context (i.e. surrounding requirements or the general project context). The usage of weak words may be considered problematic in approximately 12% of all occurrences. For an automatic analysis on weak words linguistic patterns can be defined to detect these problematic occurrences.**

## I. Introduction

Natural language is still predominantly used to write requirements [1], [2], [3]. However, natural language can easily lead to underspecified or ambiguous requirements that are hard to understand or even completely incomprehensible. Regarding the fact that requirements are the basis for all subsequent development activities, it is very important that the requirements are understandable and unambiguous. Requirements engineering literature proposes requirement templates (like in [4]) and rules that should help formulating understandable and unambiguous requirements [5], [6], [7], [8], [9]. For example, there are rules instructing the writer to state only one action per requirement or to omit nominalizations. The rule to avoid passive voice is mentioned in almost every guideline. The problem when using passive voice is that it is easy for the author to omit the actor of a requirement. However, the information about what or who is the actor in the action stated in the requirement is a crucial one. Missing information opens the interpretation space for subsequent development activities like design, implementation, or testing. The guidelines advocate to use active voice instead of passive voice because it forces the author to name the actor. Another option is to use a '*by*'-phrase within the passive voice. When the author uses a '*by*'-phrase, the actor is stated in the subordinate clause. However, the guidelines advice to use the active voice rather than the passive voice even if it has a '*by*'-phrase. According to the guidelines, the use of passive voice without stating the actor is always a problem in requirements documents. The same applies to weak words. The guidelines state that every use of a weak word is problematic because the requirement gets underspecified. There are a lot of lists including weak words (for example in [8], [10]).

Following the writing rules helps to write good requirements, but is it also the case that the violation of the writing rules always result in problematic requirements? In our environment (the passenger car development of Mercedes-Benz) authors do often use passive voice and weak words. Does this mean that the requirements are of poor quality? In this contribution we concentrate on the question whether passive voice and weak words are really always leading to incomplete or incomprehensible requirements.

Automated analyses for detecting weak words do exist but they are usually word-list based and produce a lot of false positives. With a rule based approach the number of false positive can be reduced.

In Section II, we discuss the linguistic phenomena passive voice and weak words. Thereafter, in Section III we lay out the industrial setting of our work. Section IV presents the empirical investigation along with its results and finally in Section V the limitations of our research are discussed. In the last section, we present the conclusion and the planned future work.

## II. Background and Related Work

Writing rules are a common method for documenting natural language requirements. Following these rules is a good way to construct unambiguous requirements. In some cases the rules are written as "Don'ts", like '*Avoid comparative words*' or '*Avoid vague, subjective terms*' [8] and in some other cases they are written as "Do's" like '*Use active voice*' [5]. '*Avoid passive voice*' is one of the most frequently mentioned rules in requirements engineering literature [5], [6], [7], [8], [9].

The problem with the use of passive voice is that *"such sentences just say that some action is performed, but they do not say who performs the action"* [11], p. 181. The actor of a requirement is essentially important for further steps in the development process. If a reader scans a requirement and cannot find any information about who is to realize the requirement, respectively whether the requirement describes an automated process or whether he has to trigger the process, the reader possibly comes to a wrong interpretation and maybe does not realize the requirement that he should actually meet.

These arguments for avoiding passive voice seem to be obvious. If the reader of a requirement considers every requirement isolated from the rest, it might cause confusion about who should realize the specified demands. Typically, the reader walks through the whole requirements document and can refer to the context or even the whole discourse. Context in this case means the adjacent requirements and discourse the whole requirements document with all its inherent requirements. Through context and discourse analysis many questions about who is to realize a requirement can be answered.

Often, requirements are used in a contractor-supplier relationship. In such a setting, requirements dealing with activities that should be performed during the development process (like '*An open issue list has to be maintained.*') are naturally addressing the supplier. The same is the case with product requirements (like '*If error xyz is detected, its occurrence shall be stored in the diagnosis memory.*'). These requirements are addressing the product the supplier is asked to develop. Clearly, this rule-of-thumb ("The supplier or his product has to fulfill each requirement") is not always true, but in practice, it is a good indicator.

Existing tools like *Congree Authoring Server*[1], *Language Tool*[2] and *Acrolinx*[3] can detect passive voice. They find passive voice by analyzing the syntax and the verb forms of a sentence. When one of these tools detects a use of passive voice, they raise a warning to the author.

Avoiding passive voice is a valuable approach for writing good requirements. Nevertheless it is obvious that using passive voice does not always lead to a problem. In many cases it is unproblematic. Information can be drawn through context, discourse, world knowledge and the background knowledge that requirements are addressed to the supplier.

The use of weak words in requirements documents is also often prohibited in guidelines. Weak words appear under several names; for example they are also called vague words or vague terms [8]. In this contribution we call them weak words. Typical examples are '*long*' and '*easily*'. Looking at the requirement '*If the button is pressed for a long time ...*', one could get the impression that '*long*' is always harmful to the quality of a requirement. But contemplating the requirement '*The vector must be 24 bits long*' one can see that the use of the word '*long*' does not always lead to an imprecise requirement. Only certain readings and contexts do that. For a good weak word analysis the linguistic context has to be taken into account. Literature only advises to use a word list with potential weak words. If one finds a word mentioned in a weak word list within a requirement, there is the need to change the requirement.

There are tools that automatically detect weak words, like *ReQualize* [12] and *DESIRe* [13]. These tools use the word-list based approach like suggested by the guidelines. When the tool detects a weak word that is in its weak word list, the tool returns a warning to the author. The linguistic context of the requirements is not taken into account in such an analysis.

To our knowledge there is no investigation about how well writing rules for requirements work in general.

## III. Industrial Setting

A large portion of automotive components is nowadays built by suppliers. Here, the main tasks of an automotive OEM like Daimler are writing requirements documents and later integrating and testing the delivered components. Requirements documents at Mercedes-Benz passenger car development typically range between 100 and 2000 pages [14]. Additionally, there are typically 30 to 300 supplementary documents that also contain requirements.

Requirements documents are mainly written in natural language, augmented with drawings, wiring diagrams, state charts, tables, etc., where appropriate. The used languages are German and English.

The basis for each Daimler requirements document is a voluminous template (approx. 150 pages). This template contains many standard requirements on various aspects of automotive components, like material, communication buses, diagnosis, environmental conditions, logistics, corrosion testing, AUTOSAR standard software, and many more. Out of these approx. 2000 standard requirements, the specification author has to select the relevant sections for his component.

Requirements documents are managed within the requirements management tool *IBM DOORS*. Every requirement has a unique ID, a type that classifies each individual statement as a heading, information, requirement, or predefinition. Statements classified as requirements are requested to be linked with one or more test cases.

Requirements for an automotive component, like an ECU (electronic control unit), cover many areas: functionality, performance, diagnosis functionality, material, environmental conditions, EMC (electromagnetic compatibility), and also process related facets like logistic processes and documentation.

Requirements regarding functionality are usually fine-grain: They describe, for instance, how the component should behave if dedicated signals arrive or sensor inputs are detected.

The requirements documents are used in a classical contractor-supplier relationship, i.e. after a tendering process, a single supplier is selected who is asked to develop and build a component according to the given requirements document. During the development process, typically updates of the

[1] www.congree.com
[2] www.languagetool.org
[3] www.acrolinx.de

requirements document are handed over to the supplier. The deltas between the versions of the requirements documents are treated as change request.

Requirements within a requirements document are usually addressed to the supplier (in case of process requirements) or to the component, the supplier is asked to build (in case of product requirements).

## IV. Empirical Investigation

### A. Goals and Theses

The main goal of this study is to investigate whether the violation of the writing rules '*Avoid passive voice*' and '*Do not use weak words*' results in bad-quality requirements. The main theses of this contribution are:

- T1 The use of passive voice in requirements documents is not always problematic.
- T2 The use of weak words in requirements documents is not always problematic.
- T2 An automated weak word analysis can be improved when using a rule based approach instead of a word-list based approach.

### B. Research Questions

According to the theses, we deviate the following research questions:

- R1 To what extend does the use of passive voice lead to underspecified requirements (i.e. which percentage of passive voice occurrences has to be considered problematic)?
  - R1.1 When regarding the requirements isolated from the other requirements.
  - R1.2 When regarding the whole requirements document.
- R2 To what extend does the use of weak words lead to underspecified requirements (i.e. which percentage of weak word occurrences has to be considered problematic)?
- R3 To which extend is it possible to improve an automated weak word analysis by using a rule based approach (i.e. how many false positives can be reduced)?

### C. Data Sets and Data Preprocessing

In this study, we used four original Daimler requirements documents from recent car projects from the interior electronics domain and telematics domain. All documents have been written in English. Table I gives an overview on the size of the four documents. Column 2 and 3 present the size in pages and DOORS objects, respectively. Column 4 shows the number of non-standard requirements, i.e. all requirements that had not been derived from Daimler specification templates. We refused the standard requirements as they are used in many requirements documents and would bias the results of our investigation.

All in all, we considered 4729 non-standard requirements for the weak word analysis. Then we eliminated objects that did not contain complete sentences or were too short (e.g. statements within bullet lists). All in all we identified 1171 relevant requirements written in complete sentences (column 5). These were the basis for the passive voice analysis. Out of these, as shown in Figure 1, 460 requirements (= 39%) contained only active voice, 303 requirements (= 26%) contained only passive voice and 408 requirements contained both, active and passive voice (= 35%). For the analysis on the passive voice we used the requirements that only contained passive voice and the requirements that contained both, active and passive voice. Altogether 711 requirements have been considered for the passive voice analysis.

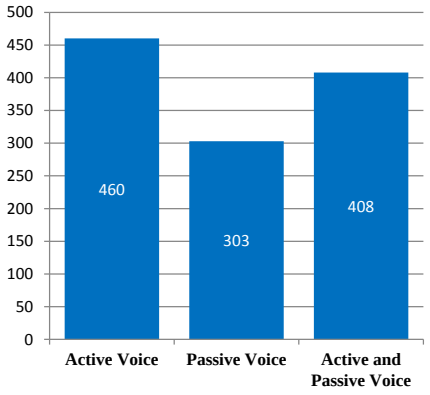


Fig. 1. Distribution of Voice in 1171 Requirements

### D. Analysis and Results

Both analyses were performed manually by reading and rating the requirements. For each requirement the evaluator had to make the decision whether the requirement is `OK` or `problematic`, with respect to the investigated phenomenon.

*1) Passive Voice:* As described before, we analyzed 711 requirements that contain passive voice and rated them. One person performed the analysis on 711 requirements.

To guarantee the rating quality of the person who did the complete analysis, we carried out an experiment to calculate Fleiss' Kappa [15]. Fleiss' Kappa is a measure that calculates whether evaluators have a good rating coverage. This experiment was carried out with three evaluators. One evaluator was the person who did the analysis on all requirements. The other two evaluators were persons who also work on requirements documents. The three persons rated 150 requirements out of the 711 requirements in two ways: The first analysis was to contemplate these 150 requirements without discourse, that means the requirements were regarded isolated from the rest of the requirements document. For each requirement each evaluator had to decide whether the requirement is unambiguous in terms of the passive voice (category: `OK`) or whether the requirement is vague (category: `problematic`) due to the passive voice. The second analysis was to regard the requirements with the whole discourse and to decide if the requirement is unambiguous or not due to the use of passive voice.

TABLE I
INFORMATION ABOUT THE SIZE OF THE DATA SET

| Requirements Document | Pages | DOORS objects | Non-Standard Requirements | Full-sentence Requirements |
|---|---|---|---|---|
| Powernet Management System | 439 | 5364 | 2856 | 706 |
| Communication Module | 245 | 2169 | 875 | 270 |
| Headphone Rear Seat | 75 | 1867 | 498 | 102 |
| Wireless Audio Module | 119 | 1916 | 500 | 93 |
| $\sum$ | 878 | 11316 | 4729 | 1171 |

Only regarding the isolated requirements, 85 out of the 150 requirements were rated as `OK` by all evaluators. The fact that the actors of the requirements are missing was not considered problematic in these cases. 46 requirements were rated as `problematic` because of a missing actor by all evaluators. 8 requirements were rated as `problematic` by two evaluators and rated as `OK` by one evaluator. Furthermore 11 requirements were rated as `problematic` by one evaluator and rated as `OK` by two evaluators.

The requirements which were rated as `problematic` were almost always about signal transformations and signal transmissions. Contemplating the requirement '*The CAN signals from engine and alternator are converted to some bitwise information for the other functions*' and the requirement '*If during that time the car is unlocked again, X is set to 1 and the function waits for the car to be locked again*' without any additional information from the context, it is not possible to resolve whether the supplier is to realize the requirements or whether they are just an explanation. Furthermore there is no information about whether the signals are consumed or produced. Another problematic requirement is '*The information if the car is locked (inverted signal U) is stored in a flag, needing 1 Bit of memory*'. Here again, it is not obvious whether the supplier is to realize the requirement or whether it is just an additional information.

Ratings differed, for example, in the following requirement: '*All controls of the user interface shall be in a position where they are not touched accidentally when putting the headphones on or off*'. For two evaluators it was obvious that the supplier is to realize this requirement. One evaluator categorized the requirement as `problematic`. For him, it was not that obvious as for the other evaluators that the supplier should realize the requirement. Ratings sometimes also differed because of bad grammatical structures which caused additional confusion and misinterpretation.

Counting the agreements and disagreements of the evaluators we get the following results: in 131 cases the three evaluators agreed, in 19 cases the analysis results differ. With these results we could calculate Fleiss' Kappa [15]. Our Kappa was 0.81, which means an almost perfect agreement.

By including discourse information the agreement was even higher. Only one requirement was rated as `problematic` by two evaluators. This requirement was '*Corrosion tests are carried out according to the xyz standards which are used*'. In the whole requirements document there was no information about who should do these corrosion tests. One evaluator rated the requirement as `OK` because the preceding requirements were all addressed to the supplier. But this interpretation is more a guess than a given fact. All other requirements were categorized as `OK` by all evaluators. The requirements which were rated as `problematic` in the analysis without regarding the discourse of the requirements are rated here as `OK` because in the requirements document the evaluators could find tables that illustrate the signal transmissions and transformations. Furthermore, information that helped to resolve the actor could be found in the surrounding requirements. Since the evaluators' opinions only differed in case of one requirement we got a high Kappa of 0.99 in this case.

After calculating Fleiss' Kappa we could be sure that the analysis of one person over 711 requirements is meaningful. These 711 requirements were rated in the same way as in the analysis with three evaluators. In the first step the requirements were regarded without discourse and in the second step with discourse. The results (in Table II) showed that without regarding the discourse 475 requirements were rated as `OK` and 236 requirements were rated as `problematic`. Here again the use of passive voice does not always lead to problematic requirements. The number of unproblematic requirements (66,8%) is about twice as large as the number of problematic requirements (33,2%). Out of the 475 `OK` rated requirements, only 126 contained a '*by*'-phrase. This result again emphasizes that leaving out the actor is not always harmful for the quality of a requirement.

Regarding the discourse of the requirements only two requirements are rated as problematic. In almost all requirements the actor could be resolved by the reader although he was not directly specified. Only 0,28% of the occurrences are problematic when regarding the whole requirements document.

TABLE II
RATINGS OF 711 PASSIVE VOICE REQUIREMENTS

| | Without Discourse | With Discourse |
|---|---|---|
| `OK` | 475 | 709 |
| `Problematic` | 236 | 2 |

To summarize, regarding isolated requirements can be problematic in terms of passive voice but this is not always the case. Contemplating the discourse of the requirement often helps to solve the question who is to act. Here, either the surrounding requirements provide the missing information

(like signal tables) or the fact that development projects in the automotive industry are usually conducted in contractor-supplier relations help to infer the missing actor.

*2) Weak Words:* For the weak word analysis we used a Daimler internal list of weak words [16] that contains 103 words. Table III shows all weak words that occurred at least 10 times in the 4729 analyzed requirements. Those words that had a count of zero or a count of one to nine are shown in Table IV. For the weak words that occurred less than 10 times it is hard to evaluate how critical they really are. Therefore we do not present the number of problems in these cases. In Table III the absolute number of each weak word and the number of problematic occurrences of each weak word are shown. Based on the absolute numbers of occurrences of each weak word and the numbers of problematic occurrences we could calculate the percentage of problematic usages. The result is that only 12% of the occurrences of weak words are causing problems in our requirements documents.

In the following, we show to what purpose weak words are used and how harmful they are. This is done using the weak word '*long/longer*', '*good*' and '*special*'.

*Long/Longer*: Table III shows that 91 instances of '*long/longer*' were found in our four requirements documents. 5 instances were categorized as `problematic`. A usage of '*long*' that leads to an underspecified requirement is for example '*It determines the SOC by the voltage if the standby time is long enough*'. The term '*long enough*' is not precise because there is no temporal expression that further describes '*long*'.

Another underspecified usage is '*The charging is interrupted, if a no-charge condition applies longer than a few seconds*'. But the use of this weak word is not always problematic as one can see in the requirement '*The evaluation takes as long as the charging is not finished*'. '*As long as*' is a precise temporal expression and is therefore not a term that must be replaced by another construction. Summarized that means that the weak word '*long*' does not always lead to an underspecified requirement. It depends on the context of the word or on the context of the whole requirement.

*Good*: The word *good* occurred 10 times in our requirements documents. A precise usage of '*good*' is for example a description like '*X = 1 means a good charging condition*'. Descriptions like these are a common way to define weak words and give them an unambiguous meaning. But '*good*' is not always used in a precise way: '*In a PreSafe condition the car takes actions to prevent an accident or at least safe the passengers from injuries as good as possible*'. Clearly, '*as good as possible*' is not precise enough. In this special case, we face an information that is wrongly classified as a requirement.

We want to add here that our requirements documents are not exclusively written by native English speakers. In fact, it would be better to use '*well*' instead of '*good*' but that does not change the fact that the requirement is ambiguous since there would still be room for interpretation triggered by the use of a weak word. All in all, we detected 3 critical occurrences of '*good*'. In this case the summary again is that the use of '*good*' does not always lead to ambiguous requirements neither does it always lead to unambiguous requirements.

*Special*: The word '*special*' occurred 20 times and only in 1 case we found it to be `problematic`. The critical construction is '*The algorithm catches, whenever special conditions are fulfilled*'. There is no further description about what these '*special conditions*' are, therefore we categorized the requirement as `problematic`. The rest of the usages

TABLE III
WEAK-WORDS WITH A COUNT OF $\geq 10$

| Weak-Word | $\sum$ | $\sum$ Problems | $\sum$ Problems in % |
|---|---|---|---|
| about | 40 | 0 | 0 % |
| actual | 85 | 0 | 0 % |
| another | 33 | 2 | 6,06 % |
| appropriate | 15 | 7 | 46,67 % |
| certain | 46 | 40 | 86,96 % |
| clear/clearly | 26 | 3 | 11,54 % |
| close | 10 | 0 | 0 % |
| corresponding | 40 | 0 | 0 % |
| detailed | 28 | 11 | 39,29 % |
| different | 49 | 0 | 0 % |
| easily | 10 | 1 | 10,00 % |
| else | 135 | 0 | 0 % |
| further | 58 | 1 | 1,72 % |
| good | 10 | 3 | 30,00 % |
| high/higher | 123 | 41 | 33,33 % |
| less | 13 | 1 | 7,69 % |
| long/longer | 91 | 5 | 5,49 % |
| more | 46 | 2 | 4,35 % |
| other | 87 | 10 | 11,49 % |
| possible | 174 | 10 | 5,75 % |
| short | 22 | 5 | 22,73 % |
| special | 20 | 1 | 5,00 % |
| specific | 24 | 3 | 12,50 % |
| well | 26 | 0 | 0 % |

TABLE IV
WEAK WORDS WITH A COUNT OF $< 10$

| | |
|---|---|
| 0 counts | accurate, actually, approximately, better, common, commonly, easy, easier, especially, large, larger, often, particular, particularly, several, shorter, shortly, similar, simple, simpler, simply, slightly, slow, slowly, small, smaller, specially, sure, surely, |
| 1-9 counts | basically, big, bigger, brief, briefly, certainly, clearer, closely, closer, definite, definitely, distinct, distinctly, fine, fundamental, fundamentally, huge, huger, largely, little, multiple, narrow, narrower, narrowly, obvious, obviously, optimal, optimally, perhaps, possibly, quite, quietly, reasonable, reasonably, similarly, slight, slighter, slower, specifically, superior, surer, tall, taller, thorough, thoroughly, tight, tightly |

were not critical. These were of different types. We found requirements that are only informations like in '*No special reset handling required*', requirements in which '*special*' was a term and set in quotation marks '*The modes are "standard mode", "standby/sleep" and "special function"*' and requirements in which the meaning of '*special*' is mentioned in the same requirement in a subordinate clause '*In very special cases it might be acceptable that the supplier replies with a qualified status message, e.g. in case of a complex problem during investigation*'. To summarize, for the word '*special*' rules for good and bad usages can be examined as was already determined for the other weak words.

Weak words can indeed be a problem in requirements documents. They cause underspecification of a requirement, but not in every case. The context of the words has to be considered. Sometimes the sentence itself provides sufficient context, sometimes the context of the whole requirements document has to be considered. In descriptive expressions and phrases that describe the requirement in more detail, for example in a subordinate clause or when terms occur in quotation marks, it is fine to use weak words.

### *E. Improved Automated Weak Word Analysis*

In [17], we developed an automated weak word analysis (for German words) that does not only use word-lists but also includes the linguistic context of the words. Context in this sense means only the context in the sentence itself, not context over several requirements. We defined rules that are based on computational linguistic methods like lemmatizing, tagging and parsing [18], [19]. These rules define in which contexts the use of the respective weak word is `OK`. We derive the `OK`-rules from a large corpus containing requirements with more than a million words origining from more than 40 German requirements documents.

The result was that the number of unnecessary warnings to the requirements document author can be significantly reduced in contrast to a word-list based analysis. The analysis was developed in a first step for the words '*lang*', '*kurz*', '*schnell*', '*mal*' (in English: '*long*', '*short*', '*fast*' and '*times*' (like in '*two times*' or '*multiple times*')). The analysis of additional weak words is in progress.

For English weak words it is possible to do an automated analysis like in [17]. For automatic analysis it is sufficient to define good contexts like in the German analysis.

In the previous section we regarded the weak word '*long/longer*' and showed some good and some bad contexts of it. We identified rules for good contexts of '*long/longer*' that are shown in Table V. To use these rules, a requirement needs to be tokenized (segmentation of the text in words), lemmatized (building the base form of the words: '*walks* $\Rightarrow$ *walk*'), tagged (annotating part of speech and morphological information like gender or number), and finally parsed (annotating dependencies and sentence functions like subject) [19]. '*CARD*' (cardinal number), '*VV*' (verb), '*NN*' (noun) and '*NE*' (noun) which are used in the rule set are part of speech (POS) tags. They are used to define general rules that are not only word dependent. Although the words are written in lower case that might not imply that only constructions in lower case are being analyzed. The rules apply to words in lower case and words in upper case, too.

TABLE V
RULES FOR GOOD CONTEXTS OF '*long*'

| No. | Rule |
|---|---|
| 1 | (for\|not)$^+$ longer than CARD |
| 2 | (for\|not)$^+$ longer than [A-Z]$^+$_ [A-Za-z]$^+$ |
| 3 | as long as |
| 4 | lemma.be no(t) longer (be) VV.* |
| 5 | long (NN/NE)$^+$ lemma.be CARD |
| 6 | longer than the (NN/NE)$^+$ |

The first rule is composed of words, a regular expression and a POS-Tag. This rule defines that a construction like '*for longer than 2 seconds*' is a good usage of '*longer*' and needs therefore not be shown to the requirements document author for checking. The second rule is almost the same as the first. It only differs in the last part. Instead of the number in the first rule, it contains a regular expression that represents a parameter. Parameters in our requirements documents have a special structure and they are defined at the beginning of the document and could therefore be categorized as a good context. An instance for Rule 2 could be for example '*not longer than ABC_HUZ*'. Rule 3 is to read as it is. The construction '*as long as*' is always unproblematic. Rule 4 includes constructions that start with a word that has the lemma '*be*' (that means '*is*' or '*are*') which is followed by a negation, the weak word itself, an optional '*be*' (really the word, not the lemma), and some kind of verb. That would be for example '*... is not longer allowed*'. Rule 5 represents descriptions like in '*A long parking time is 2 hours*'. The last rule represents constructions like '*longer than the parking time*'. Here we assume that the definite article '*the*' implies that somewhere in the document there must be a definition of the term '*parking time*'. In this case, the author has to check whether the term '*parking time*' is defined. If not he should add a definition. But the requirement itself does not need to be revised. If such a requirement appears with an indefinite article, a checking tool should raise a warning because the indefinite article implies that the term is mentioned for the first time in the requirements document.

Having defined such rules, the approach is as follows:

Given a rule set $R$ for weak word $w$ and requirement $req$:

```
bool problematic = true
if contains (req, w) then
    for every r_i in R do
        if applies (r_i, req) then problematic = false
if problematic == true then raise warning
```

That means we only show those requirements to the requirements document author that do not match a positive context rule.

Using an approach like this, the number of unnecessary warnings to the author can be significantly reduced. Using a word-list based approach the system would give 91 warnings to the author when analyzing our four requirements documents. Out of these, 86 would be irrelevant. Using a context based analysis the number of warnings can be reduced to a minimum, in this case 5 warnings. By reviewing more requirements documents, it is likely that some more rules can be identified (see also Table VI).

TABLE VI
RESULTS OF '*long*'-RULES

| | | Rule-based Analysis | |
|---|---|---|---|
| | | Vague Req | Precise Req |
| Manual Analysis | Vague Req | 5 | 0 (false negatives) |
| | Precise Req | 0 (false positives) | 86 |

From our work with German weak works we found that it is not possible to define rules for every weak word in such a way that the number of unnecessary warnings ("false positives") can be reduced to zero. Nevertheless the number of unnecessary warnings can be reduced significantly compared to a word-list based approach. In rare situations, this approach suppresses valid warnings ("false negatives"). However, from our work with requirements authors we learned that it is better not to identify all negative occurrences than presenting too many false positives – more than, let's say, 25% of false positives significantly reduces the acceptance of an automatic weak word detection, leading to refusal of supporting tools.

## V. THREATS TO VALIDITY

We identified some threats to validity during our experiments.

One threat to the internal validity is the subjectivity of the ratings in our analyses. In the experiment for calculating Fleiss' Kappa this subjective character is revealed as there were some disagreements between the three evaluators. As we had a high Kappa, we can assume that this subjective character does not have a substantial effect on our result.

There are also some external threats to validity. The experiments were performed exclusively on Daimler internal requirements documents and can therefore not support any statement about the behaviour of the phenomena passive voice and weak words in requirements documents in general. Besides, in our analysis we assume that the requirements documents are addressed to a supplier and that he is the one supposed to fulfill the requirements. Furthermore we assume that in the requirements documents signal tables can be found which reveal information about whether signals are produced or consumed and whether the supplier is asked to realize the requirements. We assume that the results can be transferred to industries with similar characteristics, i.e. contractor-supplier relationship and fine-grain technical requirements.

Another threat to external validity is that we only investigated the use of passive voice in requirements documents written in English. We cannot make a statement about whether the use of passive voice is harmless to the exact definition of requirements in other languages.

For the analysis on weak words we did not calculate Fleiss' Kappa. The ambiguities that are triggered by weak words are more obvious than for passive voice and could easily be classified into good and bad contexts.

Moreover, we did the classification only on four requirements documents. For weak words that had a count of 9 or less we need to check more requirements documents to define well-founded rules.

## VI. CONCLUSION AND FUTURE WORK

Our analysis showed that the use of passive voice rarely causes problems. Context and discourse analysis as well as expert knowledge often help to identify the actor of a requirement reliably. Regarding a requirement isolated from the rest of the requirements document leads to more problems (33,2%) than regarding it in the context of the whole discourse (0,28%). It cannot be stated that the use of passive voice is never problematic. But it is by far not as harmful as commonly assumed in requirements engineering literature, at least in contractor-supplier relationships. The number of problematic occurrences is too low for defining suitable rules for an automated analysis.

Weak words are also not necessarily problematic, in our case only in 12%. Their context has to be taken into account.

Our results do only apply for the requirements documents from Daimler, so far. For future work it would be useful to find out whether the results do also apply to requirements documents from other fields besides the automotive industry.

In our future work we will also concentrate on defining new rules for weak words in German and in English. An analysis on the use of passive voice in German requirements documents of Daimler is also planned as well as a study for examining whether German passive voice behaves similar to English passive voice in requirements documents.

There are many other linguistic phenomena mentioned in requirements engineering literature which we could also investigate. We are in the process of doing this for nominalizations [20]. First results show that their use is not problematic in many cases. Again, context and discourse information and of course expert knowledge help to resolve ambiguities.

## ACKNOWLEDGMENT

We would like to thank Anna Paola Brehm for her help analyzing the data.

## REFERENCES

[1] M. Kassab, C. Neill, and P. Laplante, "State of practice in requirements engineering: contemporary data," *Innovations in Systems and Software Engineering*, vol. 10, no. 4, pp. 235–241, 2014.

[2] S. Adam, C. Wünch, and M. Koch, "Results report re-kompass [Ergebnisbericht RE-Kompass]," Fraunhofer IESE, Tech. Rep., 2013, (in German).

[3] L. Mich, M. Franch, and P. Novi Inverardi, "Market research for requirements analysis using linguistic tools," *Requirements Engineering*, vol. 9, no. 1, pp. 40–56, 2004.

[4] Die SOPHISTen, *Patterns for all situations [Schablonen für alle Fälle]*. SOPHIST GmbH, 2013, (in German).

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

[6] tekom, *Rule based writing – English for German-speaking authors [Regelbasiertes Schreiben – Englisch für deutschsprachige Autoren]*, 2014, (in German).

[7] C. Rupp, *Requirements engineering and management: professional, iterative requirements analysis for practice [Requirements-Engineering und-Management: professionelle, iterative Anforderungsanalyse für die Praxis]*. Hanser Verlag, 2009, (in German).

[8] K. Wiegers, *Software Requirements*. Microsoft Press, 1999.

[9] ASD, "Simplified technical english," *Specification ASD-STE 100*, 2007.

[10] R. Melchisedech, "Management and inspection of natural language requirement specifications [Verwaltung und Prüfung natürlichsprachlicher Spezifikationen]," Ph.D. dissertation, University of Stuttgart, 2000, (in German).

[11] L. Kof, "Treatment of passive voice and conjunctions in use case documents," in *Natural Language Processing and Information Systems*. Springer, 2007, pp. 181–192.

[12] M. Heidenreich, "Metrics and tool support for the inspection of requirements [Metriken und Werkzeugunterstützung zur Überprüfung von Anforderungen]," *OBJEKTspektrum*, June 2010, (in German). [Online]. Available: http://www.martin-heidenreich.com/download/scripte/Artikel-ObjektSpektrum-RE.pdf

[13] F. Stöckel, P. Stolz, I. Uddin, and L. Endriss, "Desire® dynamic expert system for improving requirements," 2009.

[14] F. Houdek, "Challenges in automotive requirements engineering," *Industrial Presentations by REFSQ*, 2010.

[15] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological bulletin*, vol. 76, no. 5, p. 378, 1971.

[16] Daimler, "List of weak words," 2015, property of Daimler AG.

[17] J. Krisch, "Identification of critical weak words in requirement specifications [Identifikation kritischer Weak-Words aufgrund ihres Satzkontextes in Anforderungsdokumenten]," Diploma Thesis, University of Stuttgart, 2013, (in German).

[18] B. Bohnet, "Efficient parsing of syntactic and semantic dependency structures," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*. Association for Computational Linguistics, 2009, pp. 67–72.

[19] R. Mitkov, *The Oxford handbook of computational linguistics*. Oxford University Press, 2005.

[20] J. Keim, "Identification of problematic nominalizations in natural language requirement specifications [Identifikation problematischer Substantivierungen in natürlichsprachigen Anforderungsdokumenten]," Bachelor Thesis, University of Karlsruhe, 2015, (in German).