

Detecting Inconsistencies of Natural Language Requirements in Satellite Ground Segment Domain

Sercan Çevikol

Fatma Başak Aydemir

Boğaziçi University

Istanbul, Turkey

{sercan.cevikol, basak.aydemir}@boun.edu.tr

Abstract

The ground segment constitutes the ground-based infrastructure necessary to support the operations of satellites, including the control of the spacecraft in orbit, and the acquisition, reception, processing and delivery of the data. Since the ground segment is one of the essential elements in satellite operations, the quality of the requirements are critically important for the success of the satellite missions. Similar to many other large-scale systems, requirements for the ground segment are documented in natural language, making them prone to ambiguity and vagueness, and making it difficult to check properties such as completeness and consistency. Due to these shortcomings, the review process of the requirements is expensive in terms of time and effort. Our aim is to provide automated support for detecting inconsistencies in the ground segment requirements. Our approach relies on natural language processing and machine learning techniques. Our plan is to validate our work on a real ground segment requirement set.

1 Introduction

Natural language (NL) is either the sole or the main complementary method to document requirements due to its convenience. However, it is prone to ambiguity and vagueness and it requires effort to check certain properties of the set of requirements such as completeness and consistency especially when the number of requirements is high. Natural language processing (NLP) based methods are employed to overcome these difficulties [DFFP18].

One domain with high numbers of requirements for a complex system is the ground segment of the satellites. The ground segment supports the ground functions required to meet the objectives of satellite missions. Main functions of the ground segment are:

- Acquiring, processing, and disseminating satellite data to the end users,
- Monitoring, controlling, and operating the satellites in-orbit,
- Archiving data and providing off-line retrieval from archive and user support services,
- Calibrating and validating the products, routinely monitoring the health status of the satellite instruments and the quality of the products.

The ground segment is an essential element for the success of a satellite mission. The staff at the ground segment are the first to detect any problems of a satellite and devise a solution. The ground segment stores the data collected by the satellite, transmits them, and ensures the proper functioning of the whole system. The successful implementation of a ground segment is a mission-critical process in the satellites domain for the ground segment is the central element of the whole network. As a result, the requirements for the ground segment is vigorously reviewed before the design and implementation phases to achieve the highest quality and consistency.

The requirements for the ground segment are written in NL by different teams. Due to the complexity of the system to be built, the set of requirements are big and complex in terms of dependencies. As with other set of requirements written in NL, there are imprecise or ambiguous sentences, and the lack of clarity increases the time spent for the review process. Maintenance of the requirements and relation management also impose challenges since adding a new requirement or modifying an existing one may easily contradict one of the other existing requirements, and this contradiction may remain undetected due to lack of explicit relations as in formal models.

Our long-term goal is to detect ambiguities and inconsistencies in ground segment requirements, although the techniques can be applied to other large-scale system requirements. In collaboration with EUMETSAT, we aim to reduce the time and effort spent to review the requirements for the ground segment which currently is a human intensive and expensive process that may take up-to four years. Our aim is to apply NLP and machine learning techniques to extract requirements and domain models, identifying ambiguities along the way and detecting inconsistencies by querying these models. Our plan is to use a real set of ground segment requirements of a meteorological satellite program. The set consists of more than 13000 requirements that also refer to several other support documents and will be used to train, test, and validate our techniques. In the requirements set, there are approximately 500 abbreviations, custom terms or product names, which are not part of any standard dictionary or reference model. Together with this custom domain reference model, the main challenge would be the scalability of the proposed methods due to the high number of requirements.

This paper is structured as follows. Section 2 presents the related work. Section 3 details our research plan. Finally, Section 4 concludes the paper.

2 Related Work

A domain model is a representation of conceptual entities or real-world objects in a domain of interest. Multiple approaches exist for extracting domain models or similar variants from requirements using extraction rules, but there are limited empirical results on industrial requirements. Arora *et al.* [ASBZ16] present a rule-based technique to extract domain models from NL requirements and apply it to an industrial case study.

NLP techniques have also been applied to detect requirement defects. Rosadini *et al.* identify quality defects of the requirements in the railway domain and incrementally tailor the existing rule-based NLP approaches to achieve a sufficient degree of accuracy. Bäumer and Geierhos [BG18] introduce a software system that helps end-users to create unambiguous and complete requirements descriptions by combining existing expert tools and controlling them using automatic compensation strategies.

Requirements management is another challenging requirements engineering activity for the large-scale systems as the requirements are documented into several specifications where each requirements document contains the specific knowledge for that document. It is difficult to aggregate the information spread in these documents in the later stages of the development. Schlutter *et al.* [SV18] build a NL pipeline that transforms a set of NL requirements into a knowledge representation graph to summarize and structure all concepts and relations contained in the requirements over all subsystem specifications was introduced and apply their technique in a case study in the automotive industry.

Berry and Kamsties [BKK03] introduce categories of ambiguities that are relevant to requirements engineering, including lexical, syntactic or structural, semantic, and pragmatic. Dalpiaz *et al.* [DSL18] study the synergy between humans' analytic capabilities and natural language processing to identify terminological ambiguity defects quickly. This study confirms the conventional wisdom, which is identifying terminological ambiguities is time consuming, even when supported by a tool and it is hard to determine whether a near-synonym may challenge the correct development of a system.

Contents of a requirements specification document can not be considered as requirements only. The document also includes information such as constraints and domain assumptions. Vogelsang and Winkler [WV16] introduce an approach to automatically classify the content elements of a natural language requirements specification document as "requirement" or "information" using convolutional neural networks with a high precision.

3 Research Plan

Our research goal is to reduce the time and effort spent to review ground segment requirements to detect ambiguities and inconsistencies in the set of requirements. Our plan is to employ NLP and machine learning techniques to extract domain and requirements models from the NL requirements and check for inconsistencies using these models. We also use NLP techniques to identify ambiguities.

Although there has been several studies in the past to detect the requirement defects in the industry, there are few large-scale case studies concerning applications of NLP for defect detection. In our research, we aim to focus on applying NLP to a large set of real industrial requirements using methods to extract domain models.

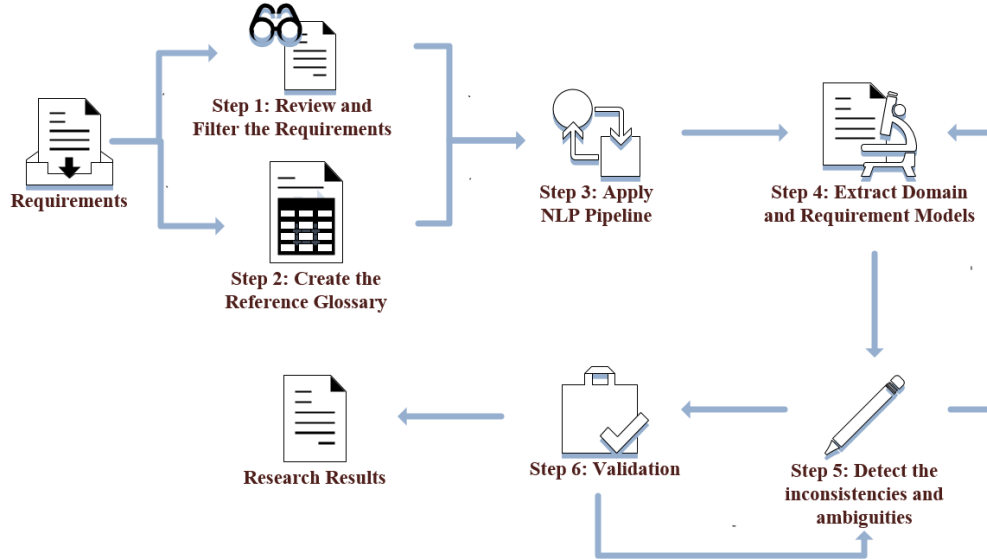


Figure 1: Overview of our research plan

Figure 1 captures the main steps of our research plan. Below we discuss each step in detail.

- Step 1. Review and Filter the Requirements: The requirements of the ground segment is stored in a requirements management tool and also distributed in multiple documents with additional information. The initial step is to filter the necessary and relevant information and format the requirements. Tables, images, and charts are discarded as well.
- Step 2. Creating the reference glossary: The requirement specifications of the ground segments include not only space specific terms, but also many custom abbreviations of the products, instruments or terms used in the space programs. Due to the high number of custom abbreviations and terms, the usage of generic reference models are quite limited, therefore we need to establish a custom reference glossary. Due to the nondisclosure agreement with EUMETSAT, we are not able to publish the glossary which has approximately 500 custom terms. Since processing this jargon is a barrier against using existing general-purpose NLP libraries, we identify specific terms, document identifications, abbreviations, and acronyms to assist future steps.
- Step 3. Apply NLP Pipeline: Our plan is to apply an NLP pipeline to create a domain model and requirements model in the next step.

Our NLP include the following steps.

- We identify and differentiate the requirements from the information notes in the requirements specification. Our current plan is to evaluate the approached proposed by Winkler and Vogelsang [WV16] and adopt it if it yields to similar results in our data set and improve the approach where necessary.
- We divide the requirements into separate tokens, such as words, numbers, spaces (tokenizing) and relate each token to a part-of-speech, such as noun, verb, adjective (part-of-speech tagging).

- We perform several analyses: morphological analysis to explore and analyze the structure of the words, such as inflections or derivations; semantic analysis to identify and label the roles of the words in the sentences, i.e. who did what to whom; and context analysis to understand the context that a word, phrase, or sentence appears in to understand what the requirement is about.
- Step 4. Extracting the domain and requirements models: As the requirement set is quite large, it is difficult to visualize, get an overall view, or analyze the requirements. We plan to generate models to benefit from the formal methods to detect inconsistencies. For the requirements model, we first focus on identifying related requirements, for example refinements of a requirement. At this step, we plan to explore both rule-based and machine learning based approaches such as text mining and active learning to generate the models. Due to the size of the requirements set, it is challenging to define extraction rules. We need to define and implement preliminary analysis on the requirements, such as frequency analysis on the usage of the words to find out the key words. After the extraction rules are defined, we generate the domain model.
- Step 5. Detection: This step focuses on identifying defects on linguistic patterns [BKK03] as well as logical contradictions between requirements [GZ05] and find the inconsistencies and ambiguities using the models we establish in Step 4. The details of this step will form when the entities and relations used in these models are finalized. Our goal is to analyse the models check certain properties. For example, violation of a property set for a parent requirement by the aggregation on refinement requirements is a common inconsistency in the ground segment and at the end of this step, our approach should highlight such inconsistencies.
- Step 6. Validation. For validation purposes, we derive our real data from top-level customer requirements for implementation by industry, which describe the ground segment requirements of a meteorological satellite programme. The data set consists of an overall ground segment requirement specification. The requirement set also refers to 19 other applicable documents (i.e. other requirements or standards where requirements refer and shall comply with) and 28 interface specification requirements. Therefore, in total, the package consists of multiple documents with thousands of requirements. We will apply our approach first to a smaller set derived from the requirements, and will then extend the scope and implement our approach on a bigger part of the requirement specification working in close collaborations with the owners of the requirements to validate the results.

4 Conclusions

Satellite ground segment is a domain where the requirements are written in NL by multiple teams, distributed in different documents, are high in volume. Many technical terms, acronyms, and abbreviations are used in the requirements. Such characteristics pose a challenge for the requirement review process that aims excellence due to the significant role of ground segment in the success of a satellite mission. In order to support human-centric review process we propose an NLP powered research-line to detect inconsistencies and ambiguities in requirements automatically.

Our planned research activity mainly concerns

- applying NLP processing techniques to parse and tokenize requirements,
- generating domain and requirements model extractions from NL requirements,
- analyzing the models to detect inconsistencies.
- validating our approach with an industrial case study

Throughout this process, we will employ NLP techniques to detect flaws in the requirement set and highlight them for the human experts to reduce the time and effort spent to review the requirements. A natural future step is to propose solutions to get rid of ambiguities and resolve inconsistencies, which is currently beyond the scope of our work.

5 Acknowledgments

We gratefully acknowledge the support of EUMETSAT, the European Organization for Exploitation of Meteorological Satellites by providing the requirements documentation.

References

- [ASBZ16] Chetan Arora, Mehrdad Sabetzadeh, Lionel C. Briand, and Frank Zimmer. Extracting domain models from natural-language requirements: approach and industrial evaluation. In *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems, Saint-Malo, France, October 2-7, 2016*, pages 250–260. ACM, 2016.
- [BG18] Frederik Simon Bäumer and Michaela Geierhos. Flexible ambiguity resolution and incompleteness detection in requirements descriptions via an indicator-based configuration of text analysis pipelines. In *51st Hawaii International Conference on System Sciences, HICSS 2018, Hilton Waikoloa Village, Hawaii, USA, January 3-6, 2018*, 2018.
- [BKK03] Daniel M. Berry, Erik Kamsties, and Michael M. Krieger. *From Contract Drafting to Software Specification: Linguistic Sources of Ambiguity, A Handbook*. , 2003.
- [DFFP18] Fabiano Dalpiaz, Alessio Ferrari, Xavier Franch, and Cristina Palomares. Natural language processing for requirements engineering: The best is yet to come. *IEEE Software*, 35(5):115–119, 2018.
- [DSL18] Fabiano Dalpiaz, Ivor Van Der Schalk, and Garm Lucassen. Pinpointing ambiguity and incompleteness in requirements engineering via information visualization and NLP. In *Requirements Engineering: Foundation for Software Quality - 24th International Working Conference, REFSQ 2018, Utrecht, The Netherlands, March 19-22, 2018, Proceedings*, pages 119–135. Springer, 2018.
- [GZ05] Vincenzo Gervasi and Didar Zowghi. Reasoning about inconsistencies in natural language requirements. *ACM Trans. Softw. Eng. Methodol.*, 14(3):277–330, 2005.
- [SV18] Aaron Schlutter and Andreas Vogelsang. Knowledge representation of requirements documents using natural language processing. In *Joint Proceedings of REFSQ-2018 Workshops, Doctoral Symposium, Live Studies Track, and Poster Track co-located with the 23rd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2018), Utrecht, The Netherlands, March 19, 2018.*, volume 2075 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018.
- [WV16] Jonas Winkler and Andreas Vogelsang. Automatic classification of requirements based on convolutional neural networks. In *24th IEEE International Requirements Engineering Conference, RE 2016, Beijing, China, September 12-16, 2016*, pages 39–45, 2016.