Towards a Dataset for Natural Language Requirements Processing

Alessio Ferrari[1], Giorgio O. Spagnolo[1], and Stefania Gnesi[1]

Institute of Information Science and Technologies (ISTI)
of the Italian National Research Council (CNR), Pisa, Italy
[alessio.ferrari,spagnolo,stefania.gnesi]@isti.cnr.it

Abstract. [**Context and motivation**] The current breakthrough of natural language processing (NLP) techniques can provide the requirements engineering (RE) community with powerful tools that can help addressing specific tasks of natural language (NL) requirements analysis, such as traceability, ambiguity detection and requirements classification, to name a few. [**Question/problem**] However, modern NLP techniques are mainly *statistical*, and need large NL requirements datasets, to support appropriate training, test and validation of the techniques. The RE community has experimented with NLP since long time, but datasets were often proprietary, or limited to few software projects for which requirements were publicly available. Hence, replication of the experiments and generalization have always been an issue. [**Principal idea/results**] Our near future commitment is to provide a publicly available NL requirements dataset. [**Contribution**] To this end, we are collecting requirements documents from the Web, and we are representing them in a common XML format. In this paper, we present the current version of the dataset, together with our agenda concerning formatting, extension, and annotation of the dataset.

1 Introduction

As well known, requirements are normally expressed with the most human of the communication codes, which is natural language (NL) [12]. In recent years, natural language processing (NLP) technologies have seen a rapid growth, and our ability of addressing common NLP tasks, such as concept categorisation, synonyms detection, semantic relatedness evaluation, *etc.*, have radically improved [11]. Therefore, we would like the capabilities of modern NLP technologies to be shared also by the requirements engineering (RE) community. However, recent techniques are mainly machine learning methods, which are *statistical* in nature [14], and require large datasets to properly work. Hence, extensive requirements datasets are needed to effectively exploit these technologies in RE [7].

Several works were performed in RE, which used real-world NL requirements to address specific tasks. In particular, works were performed on functional and non-functional requirements categorization [2, 13], traceability [4, 9, 16], detection of equivalent requirements [6], ambiguity detection [8, 10, 17, 18] and model

Copyright 2017 for this paper by its authors. Copying permitted for private and academic purposes.

synthesis [15]. Notwithstanding the value of these works, the majority of them share one or both of these weaknesses: (a) experiments are hard to reproduce; (b) results cannot be considered general. To our understanding, the only dataset that was used by more than one work (e.g., by Gervasi and Zowghi [9], and by Sultanov and Hayes [16]) is the NASA CM-1 dataset, which concerns a scientific instrument to be carried on board a satellite. However, since the dataset is focused on a specific project, one needs to experiment with other datasets to expect generality from the results. Tjong and Berry [17] use a dataset of seven publicly available industrial requirements documents. Formats vary from *.pdf* to *.doc*, and, to replicate the experiments, one need to have some uniform format – e.g., plain text, XML – to be sure that pre-processing of the files did not alter the original data. Some efforts on the direction of having a common dataset for NLP in RE are ongoing within the TRACE-LAB project [3]. In this case, the focus is on the specific task of requirements traceability. Overall, to our knowledge, a large dataset of NL requirements documents from different sources, different domains, for different tasks, and in a uniform format is not available yet.

To address this benchmark gap, and at the same time be able to exploit the capability of modern NLP techniques, our commitment is to define a publicly available NL requirements dataset. To this end, we have currently retrieved 79 requirements documents from the Web. The documents cover multiple domains, have different degrees of abstraction, and range from product standards, to international project deliverables, to university projects. We also defined a general XML schema file (XSD) to represent these different documents in a uniform format. Our short-term goal will be mapping the original requirements documents to this common format, and share the resulting XML files. Our long-term goal, which requires the contribution of the RE community, will be annotating the dataset for the different tasks that are relevant in RE.

The remainder of the paper is structured as follows. In Sect. 2, we list the requirements documents that we retrieved from the Web. In Sect. 3, we discuss our agenda, and the specific challenges that we expect to have to face in our current effort for the RE community. Finally, Sect. 4 provides final remarks.

2 Publicly Available Requirements Documents

The first stage of our work concerned the identification of publicly available requirements documents from the Web. To this end, we queried Google with the OR-linked keywords *Requirements Documents*, *Requirements Specification*, *System Specification*, *Software Specification*, *SRS*, and we selected those links that pointed to requirements documents. Our search led to the identification of 79 documents. The whole dataset can be downloaded from our Web-site [1]. We inspected each document, and labelled it according to the following main fields, plus additional ones, which provide some first-stage qualitative information.

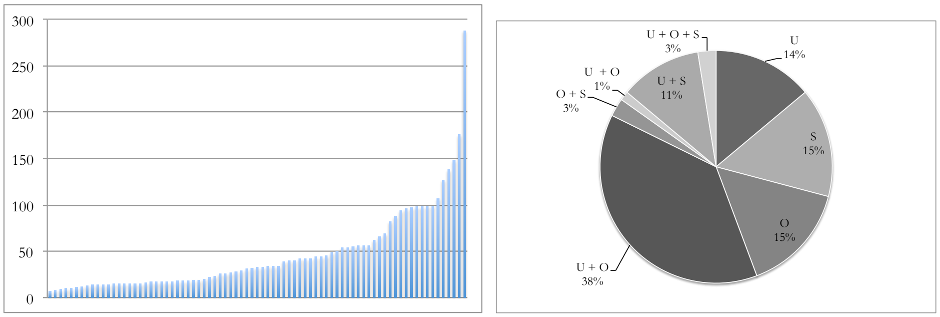
- **Doc Name:** an alphanumerical ID that identifies the document.

[1] http://fmt.isti.cnr.it/nlreqdataset/

- **Pages:** a number indicating the number of pages of the document.
- **Level:** a letter indicating the degree of abstraction of the requirements. Can be H = *high-level* requirements, or L = *low-level* requirements. The judgment was subjectively given according to the following rationale. If further refinement of the document was required before the system could be implemented, we labelled the document with H. If the content of the document was ready for implementation, we labelled it with L.
- **Structure:** a letter, or combinations of letters, indicating how requirements are structurally expressed. Can be: S = *structured*: if the requirements are expressed in a structured format, as, e.g., use-cases; U = *unstructured*: if requirements are expressed as unstructured NL descriptions; O = *one statement*: if each requirement is expressed in a single NL statement. If mixed ways of expressing requirements were used – e.g., if in the same document, we found both structured requirements (S) and unstructured ones (U) –, we combined the letters with the + operator (i.e., S + U).
- **Source:** a letter indicating if the source of the requirements is a University (U), or an Public/Private Organization (I). Documents tagged with U normally include case studies, or excercises. Documents tagged with I include industrial strength requirements.

A complete table that summarizes all the requirements documents, and all the fields, is available from our Web-site. Here, we show some statistics on the different fields. These statistics are not meant to be a formal evaluation of the generality and balance of the dataset, but are oriented to give a flavor of what can be found in our repository at this stage of our project.

Pages (Fig. 1a) We have a maximum of 288 pages, a minimum of 7 pages, an average of 47 pages, with a quite high standard deviation of 45 pages. This indicates a strong variability of the dataset in terms of length. More accurate indicators of the documents' length (e.g., number of requirements) will be provided when all the documents will be formatted in XML.



(a) Pages

(b) Structure

Fig. 1: Statistics on the current dataset.

Structure (Fig. 1b) The majority of the documents include a combination of unstructured content and requirements expressed in one sentence (U + O, 38%). Document with uniform formats – i.e., U, S, or O – are equally distributed, with about 15% of the documents for each class. Less represented are the other composite classes. However, the dataset appears already quite general and balanced for what concerns the structure of the requirements.

Level and Source Concerning the level of the requirements – not shown in the pictures – we have a dominance of high-level (H) requirements, with 71% of the documents classified with H, and 29% of them classified with L. Overall, more low-level requirements documents shall be added to the dataset to increase the balance. Concerning the source of the requirements, we have 62% of the documents coming from Public/Private Organizations (I), and 38% from Universities (U). Additional industrial requirements are needed, since each company has its specific jargon [7], and, although the dataset includes documents from companies, it does not cover all the potential writing styles.

3 Agenda and Challenges

The work that we are sharing in this paper is at its early stages. Here, we discuss our agenda, and the related challenges that we expect to face in the near future.

1. **Extracting the Text** A first step towards a dataset in a uniform XML format[2] is the extraction of the text from the documents. Tools for text extraction from *.doc*, *.pdf* and other formats are available[3]. However, the extraction is never fully clean, and some manual post-processing is required, to extract XML meta-data from the text (e.g., requirements ID), and to deal with other conversion issues. Therefore, we expect to combine automated text extraction techniques with manual work, to have a high-quality dataset in which only clean and informative text and meta-data are included.
2. **Annotating the Dataset** Even though we would already have all our requirements documents in a clean and uniform format, this would not be sufficient. Indeed, for each specific RE task, manual annotations have to be provided for the requirements, in order to use the documents as training, test and validation sets, for supervised machine-learning algorithms, or as *gold standards* (i.e., benchmarks) for unsupervised algorithms [14]. For example, if one has to address a classification task, in which requirements are classified based on their functional topic, we have to go through each requirement, and manually associate a topic to it (e.g., *train braking*, *man machine interface*, for a document of the railway domain). In this way, we can train a supervised classifier on a sub-set of the data, and evaluate its performance on the remaining sub-set. Similarly, we can train a clustering

[2] The generic XSD, together with requirements document examples in XML, is available through our Web-site.

[3] See, for example, TEXTRACT: `https://goo.gl/38NF7Z`

algorithm, and check its ability of identifying clusters of topics against the gold standard of topic classes (i.e., the manually annotated requirements). Of course, work-around solutions can be found, using contextual information – as e.g., title of the paragraphs – for the specific task of functional topic classification. On the other hand, for other tasks, as, e.g., ambiguity, we do not see other options rather than manually identifying ambiguous requirements. The annotation work is expected to involve the whole RE community interested in NLP. Indeed, requirements annotation requires a relevant effort, and domain-specific knowledge is needed to evaluate the requirements. A single human annotator is often not acceptable, and one have to compare the annotations of different subjects, and compute their inter-annotator agreement (e.g., through Cohen's kappa [5]), as common in NLP. However, we expect that researchers interested in a specific task will annotate our dataset for their task, also defining shared annotation schemes to be reused by the community. Concerning the format to be used for annotations, we recommend to use GATE[4], which is already used within the RE community [1].

3. **Updating and Extending the Dataset** From our preliminary analysis, we have already seen that our dataset is partially unbalanced for what concerns, e.g., the level of the requirements. Therefore, we expect to surf the Web to find additional requirements documents, as well as to include documents described within the RE literature. On the other hand, we encourage researchers and companies to share their documents with us, and contribute to our challenge. For a task such as traceability, the issue of dataset extension is more tricky. Indeed, to identify requirements traces between requirements at different levels of abstraction, as performed, e.g., in [9], we need high-level and low-level requirements belonging to the same project. At this stage, our dataset includes only few documents belonging to the same project.
4. **API Definition** To provide researchers with a easy-to-use dataset, we also need to develop appropriate APIs (Application Program Interfaces) to access the XML files, and extract both text and meta-data. Luckily, given the XSD file, technologies like JAXB (Java Architecture for XML Binding)[5], can automatically create Java classes directly from the XSD. In this way, one can easily access XSD-compliant XML files. Our work is therefore reduced to the definition of more high-level APIs, to, e.g., compute statistics from the dataset, which can work on top of JAXB.

4 Conclusion

NLP consists of two fundamental ingredients: algorithms and data. The NLP community can provide the RE community with advanced *algorithms* for text processing. However, we cannot use these powerful tools, unless we take the burden of providing the *data*. This paper presents a first step towards the definition of a dataset for natural language requirements processing. To perform the next

[4] `https://gate.ac.uk`

[5] `http://www.oracle.com/technetwork/articles/javase/index-140168.html`

steps, we need the contribution of the whole RE community interested in NLP, especially for what concerns the *annotation* of the requirements for specific tasks, and the definition of reusable annotation schemes.

References

1. Arora, C., Sabetzadeh, M., Briand, L., Zimmer, F.: Automated checking of conformance to requirements templates using natural language processing. IEEE TSE 41(10), 944–968 (2015)
2. Casamayor, A., Godoy, D., Campo, M.: Functional grouping of natural language requirements for assistance in architectural software design. KBS 30, 78–86 (2012)
3. Cleland-Huang, J., Czauderna, A., Dekhtyar, A., Gotel, O., Hayes, J.H., Keenan, E., Leach, G., Maletic, J., Poshyvanyk, D., Shin, Y., et al.: Grand challenges, benchmarks, and tracelab: developing infrastructure for the software traceability research community. In: TEFSE'11. pp. 17–23. ACM (2011)
4. Cleland-Huang, J., Czauderna, A., Gibiec, M., Emenecker, J.: A machine learning approach for tracing regulatory codes to product specific requirements. In: ICSE (1). pp. 155–164. ACM (2010)
5. Cohen, J.: Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. Psychological bulletin 70(4), 213 (1968)
6. Falessi, D., Cantone, G., Canfora, G.: Empirical principles and an industrial case study in retrieving equivalent requirements via natural language processing techniques. IEEE TSE 39(1), 18–44 (2013)
7. Ferrari, A., Dell'Orletta, F., Esuli, A., Gervasi, V., Gnesi, S.: Natural Language Requirements Processing: a 4D Vision. IEEE Software (to appear) (2017)
8. Ferrari, A., Lipari, G., Gnesi, S., Spagnolo, G.O.: Pragmatic ambiguity detection in natural language requirements. In: AIRE'14. pp. 1–8. IEEE (2014)
9. Gervasi, V., Zowghi, D.: Supporting traceability through affinity mining. In: RE'14. pp. 143–152. IEEE (2014)
10. Gleich, B., Creighton, O., Kof, L.: Ambiguity detection: Towards a tool explaining ambiguity sources. In: REFSQ'10, pp. 218–232. Springer (2010)
11. Goth, G.: Deep or shallow, NLP is breaking out. Communications of the ACM 59(3), 13–16 (2016)
12. Kassab, M., Neill, C., Laplante, P.: State of practice in requirements engineering: contemporary data. Innovations in Systems and Software Engineering 10(4), 235–241 (2014)
13. Knauss, E., Ott, D.: (semi-) automatic categorization of natural language requirements. In: REFSQ, pp. 39–54. Springer (2014)
14. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT Press (2003)
15. Robeer, M., Lucassen, G., van der Werf, J.M.E., Dalpiaz, F., Brinkkemper, S.: Automated extraction of conceptual models from user stories via nlp. In: RE'16. pp. 196–205. IEEE (2016)
16. Sultanov, H., Hayes, J.H.: Application of reinforcement learning to requirements engineering: requirements tracing. In: RE'13. pp. 52–61. IEEE (2013)
17. Tjong, S.F., Berry, D.M.: The design of sreea prototype potential ambiguity finder for requirements specifications and lessons learned. In: REFSQ'13, pp. 80–95. Springer (2013)
18. Yang, H., De Roeck, A., Gervasi, V., Willis, A., Nuseibeh, B.: Analysing anaphoric ambiguity in natural language requirements. REJ 16(3), 163–189 (2011)