Preface

When we search with Google search engine information on rule-based language technology, we get publications such as *Why did rule-based language processing finally fail*[1] and *Rule-based information extraction is dead!*[2] The writer of the former article fails to understand the current development phase of rule-based language technology and is ready to state that the technology has failed. The heading of the latter article is interesting. When we open the article, we get the full heading *Rule-Based Information Extraction is Dead! Long Live Rule-Based Information Extraction Systems!* The first part of the title gives the impression that the writer really thinks that the approach is dead, although he means the opposite. Copying the announcement styles of the British Empire when the king or queen has passed away, may turn out to be disastrous, because people usually see on the listing only the beginning of a long title.

Why have we decided to compile this book? Is it in defence of the withering technique?

There are two reasons for this. First, although rule-based language technology has a long history, much longer than the statistical or neural language technology, there is no such publication that would describe the main approaches using this theoretical background. Therefore, we decided to invite the key researchers using rule-based approaches to describe their systems in detail and in such a language that also those who are not initiated into the systems themselves could get a picture of how the systems work. This was at least our aim. It is up to the reader to judge whether we have succeeded in it.

The second reason for writing the book is that we want to show that the rule-based approaches suit also to languages with scarce resources, and that a large number or applications can be constructed using rule-based technology.

We are also concerned about the current trend on investing financial resources for such technologies, which automatically leave out at least 95 percentage of world's languages. Africa, for example, with rapidly increasing population, is in danger of getting marginalised in terms of language technology, unless rule-based technology is taken as the basis in developing language applications for African languages.

Rule-based technology is capable of giving prestige to such local languages, which currently are under strong stress as dying languages, because statistical methods do not suit to them due to insufficient language resources.

The book has two sections. In the first section, there are descriptions of such platforms, which have been developing rule-based systems for a number of years. Although the use of different programming languages might give the impression that the approaches are very different, they struggle with the same problems, each searching for solutions to them. The second section includes descriptions of specific problems. It also contains articles on various applications for meeting the needs of the public.

---

[1] https://medium.com/voice-tech-podcast/why-did-rule-based-natural-language-processing-finally-fail-ff5e3eae16e8
[2] https://aclanthology.org/D13-1079/

Among the development platforms described here, two platforms constitute the main branches of rule-based language technology.

The oldest of these three is the two-level description of languages using finite-state methods. The system included the two-level rule component, which made it possible to reduce the size of the lexicon, because the rules took care of the morphophonological changes. The first implementations were published in the 1980'ies. The finite-state methods were further developed in Xerox Research Centre in France. Also a new rule system was developed as an alternative for two-level rules. The result of the work by Xerox is known as Xerox Finite-State Tools, or in short, *xfst*. This tool package was later re-implemented in open domain with the name *foma*. Xfst was further developed in the HFST tool package, described in this book. Also GiellaLT, described in this book, has made extensive use of this technology. Salama makes use of the original implementation of the two-level description.

Grammatical Framework differs from the other platforms mainly in that it splits the structure of the language into two levels. On the deep level, all languages have a common structure. The description of individual languages takes this deep level as starting point and branches out into individual surface level languages. In principle, all languages form a network, where a well-described language can be translated to any other language of the network. A large number of languages have already been included into the system, although many of them only to limited extent.

In addition to the two main branches of rule-based language technology, the book also contains descriptions of platforms that are extensions or applications of the main platforms. One of them is GiellaLT Infrastructure, developed mainly in Tromssa, Norway. It has put emphasis on developing language tools for minority languages, thus representing the majority of world's languages. Instead of developing its own basic development platforms, it makes use of such platforms as lexc and *twolc* in morphological description, as well as of Constraint Grammar in disambiguation and syntactic mapping, making all these platforms in open access. Giella LT has also been working on the keyboard problem that is an obstacle in working with many minority languages with non-standard characters. In machine translation, GiellaLT relies on the Apertium solution, converting the analysis result to the standards required by Apertium. In all, GiellaLT addresses various needs of minority language communities, taking the needs of language users into focus.

Apertium focuses primarily on machine translation. It was originally designed for translation between closely related languages, mainly those in Spain and Portugal, but over the years the scope has expanded to contain a large number of very different languages. Its technical and linguistic modules are strictly in open access, which makes it possible to use the components also in other projects. Apertium makes use of rule-based technology, such as *HFST*, *LexC* (also known as *lexc*), *lexd*, and the original *lttoolbox*. Also Constraint Grammar rules can be part of Apertium applications. Because Apertium translation systems are based on modularity, various modules can be transferred to other applications.

Constraint Grammar (CG) is a platform designed originally for morphological disambiguation and syntactic mapping. Its open access application is described in this book. It has been continually developed, and new features have been added to it. Because CG provides a powerful set of possibilities for constraining rule applications, it has also

been widely used for other purposes, such as semantic disambiguation, isolation of multi-word expressions, and for adding linguistic tags for facilitating machine translation. Many systems described in this book have included CG as part of the infrastructure. The open access CG development environment has been developed in Denmark. The work has been extended also to computer assisted language learning (CALL), and machine translation.

Salama differs from the other platforms in two respects. It does not get financial support from public funds, and for this reason its applications cannot be made open access. It is also mainly one-man business. Instead of expanding the system to many languages, it concentrates on exploring various possibilities in machine translation between morphologically complex and linguistically different languages, such as Swahili, English and Finnish. Salama also puts emphasis on other user applications, such as dictionary compilation, accurate information retrieval, and language learning.

There are also other approaches to rule-based language technology. However, for a number of reasons they are not included into this book. Part of explanation is that they concentrate only to a specific phase in computational language processing.

The second part of the book contains descriptions of specific problems in rule-based language technology. They also contain solutions to problems that for a long time have been difficult to solve. In a sense, chapters in this section contain latest inventions in rule-based language technology.

One chapter deals with solutions for handling various types of multi-word expressions. The question of how to handle such words that are not listed in the lexicon is discussed in two chapters. There is also a description of a simplified system for writing two-level rules, needed especially for languages with many phonological and morphological alternations. One chapter describes experiences of working with several languages, thus providing useful information for less experienced developers. Finally, there is a chapter estimating the suitability of rule-based technology for African languages, which constitute a major market for this technology.

The chapters of the book were peer-reviewed by at least two such reviewers, who have no co-publications with the author(s).

28.3. 2023
Arvi Hurskainen, Kimmo Koskenniemi, and Tommi Pirinen
Editors