An interactive environment for creating and validating syntactic rules

Panagiotis Bouros, Aggeliki Fotopoulou, Nicholas Glaros
Institute for Language and Speech Processing (ILSP), GR-151 25, Athens, Greece {pbour, afotop, nglaros}@ilsp.gr

1 Introduction
Syntactic analysis is a key component in many Natural Language Processing applications. This is especially true when considering advanced spelling checkers, where the usage of contextual rules at the syntax level can significantly improve the spelling error detection and correction capability of such systems. The advantage of the contextual approach over the isolated-word approach becomes more clear in morphologically rich languages, in which it is very likely that a spelling error can occur due to a missing morphological rule. We developed an XML-based syntax kernel using Symfonia that can be used to extend the spell checker to include contextual information. The monitor procedure relies on the speller’s built-in lexicon for the lexical analysis and on the Rules Kernel for syntactic analysis, in order to generate a detailed report.

2 Objectives - Specifications
The main purpose of the work presented in this paper is to provide a supportive environment for fast generating a consistent set of syntactic rules optimized for advanced spelling checking engines. The developed environment provides a supportive environment for fast generating a robust spelling checker as much as possible. The system allows the user to create, view, edit, test, and update rules in real-time. The user can also monitor and report on the usage of a subset of the rules, integrated into the kernel, across real unformatted text. The system also takes advantage of the lexicon of Symfonia in order to perform the additional grammatical and lexical analysis required.

3 Architecture

Figure 1: System architecture

Figure 2: Main screen

Figure 3: Rule tree

Figure 4: Checking procedure settings

Figure 5: Report structure

4 Working Environment - Functionalities

4.1 Rule Handling
Rule-handling mainly pertains to the management of the Rules Kernel component, i.e., addition of new rules, editing of the definition and of the status of an existing rule or simply its removal from the kernel. We present the following functionalities:

1. Create a new rule. In order to create a new syntactic rule, the user can take advantage of the syntax tree representation presented in Figure 3. Each rule is focused on a single lexical item. The user defines the description, the lexical item, and the context of a syntactic rule. For this purpose, the system provides a monitor functionality for the evaluation of Rules Kernel while being on text documents. The system also takes advantage of the lexicon of Symfonia in order to perform the additional grammatical and lexical analysis required.

2. Edit an existing rule. The procedure of editing an existing rule is alike to the one of creating a new rule as it reproduces the tree representation of the rule (Figure 3).

3. Remove an existing rule. Removal of an existing rule can be done with the respective menu option or toolbar icon located in the main screen (Figure 2).

4. Disable/enable an existing rule. By default, the status of a new rule is set to enabled. The status can be altered from the main screen in Figure 2 either to disabled or enabled.

5. Export of existing rules. If the system will produce a report of the check and a document containing the erroneous sentences. Figure 5 depicts the environment that introduces the syntax of the text, as well as details on the application of the user selected subset of rules, in order to identity or handle potential misspelling, conflicts, etc.

4.2 Monitor
Efficient syntactic rules-based spell checking leads to the problem of generating a consistent set of syntactic rules that on the one hand optimize the performance of the spell checker engine and on the other constitute a consistent set of rules. For this purpose, the system provides a monitor functionality for the evaluation of Rules Kernel while being on text documents. The system also takes advantage of the lexicon of Symfonia in order to perform the additional grammatical and lexical analysis required.

5 Real-World scenario
Let us assume that we wish to solve the ambiguity between the greek words for “more” and “which” “περ” and “ποιος”. Although these two words have the same phonetic transcription “pjo”, the first one is an adverb and the second one a pronoun. We create a syntactic rule with the following environment:

Lexi1 LexiX Lexi2

If Lexi1 is characterized by the ambiguity “περ” - “ποιος” and Lexi2 is an article and LexiX is either an adjective or a noun or an adverb, then LexiX is an adverb, i.e., “περος”.

The previous rule resolves the ambiguity by rendering LexiX as an adverb. We can also define another rule for specifying that LexiX should be the pronoun, i.e., “ποιος”. The environment of the required rule would be:

Lexi1 LexiX Lexi1 Lexi4 Lexi2

Lexi1 is “περ” if Lexi1 is an article, Lexi2 an adjective, Lexi3 a noun, Lexi4 a particle and Lexi5 a verb. In addition, some or all of Lexi1, Lexi2, Lexi3 and Lexi4 can be missing.

1 The terms lexis (in plural) is used in this text to denote the set of grammatical characteristics of a word – On the other hand, words are simply the tokens of a sentence.