CORRECTING ROMANIAN TYPESETTING MISTAKES BY USING REGULAR EXPRESSIONS

BECHERU, Petru-Ioan
Faculty of Mathematics and Computer Science[1]
University of Craiova, Romania
becheru.petru.ioan@gmail.com

Abstract

Romanian typesetting rules are heavily influenced by foreign ones. In this document we present "find and replace" regular expressions designed to correct some of the most common "flaws" from a text written by a neophyte in typesetting.

Keywords: *typesetting, punctuation, regular expression, kerning, find and replace, LaTeX.*

ACM Classification: I.2.7, I.7.2, J.7, K.5.2

1. Introduction

Typesetting rules are rules to graphically prepare a manuscript before sending it to print. They are compulsory for publishers and we find them in various national and international standards, technical manuals and recommendations.

A set of typesetting rules is a typesetting style. According to Zafiu [21], the Romanian style was influenced mainly by the French style and, to a lesser extent, by the German style; the most important rules being standardized by the national standardization organization.

Besides typos of some words, the most often encountered mistakes are the punctuation and kerning ones. The main role of punctuation and spacing is facilitating the reading of text, graphically marking speech and intonation [2].

2. Regular expressions

2.1. *About regular expressions*

Regular expressions are a means by which we can describe the context of a string in a friendly way.

[1] PhD. student. He works, also, as software developer for *Spiru Haret* University, Bucharest, Romania.

It is known that regular expressions describe and define regular languages, regular expressions being closely related to nondeterministic automata [11].

Regular expressions can be used for:

- finding whether a string has certain properties;
- searching for a substring with certain properties;
- lexical analysis.

2.2. *Find and replace*

Back references are a very important feature of the regular expressions. Using these references one can build regular expressions which, for example, find whether a text has two consecutive identical words: `([a-z]+)\s+\1` because the subexpression `([a-z]+)` finds a word, the character class `\s` with the quantifier `+` finds one or more spaces and `\1` finds the word found earlier between the first brackets.

These back references can be used in a new string where they are replaced with the corresponding substring. For example, suppose we have an initial string of characters that contain natural numbers. We want to change this string so that each number that is preceded and followed by space to be preceded and followed by `~`. To describe what we want, we will use:

- a regular expression to find the numbers:

  ```
  \s+([0-9]+)\s+
  ```

- a string that shows how to do the replacement:

  ```
  ␣~\1~␣
  ```

2.3. *Regular expressions in* vi

vi is one of the most widely used text editors, taking regular expressions support from *qed*—the first text editor [17] where regular expressions were implemented.

To perform "find and replace" operations, the *vi* editor must be switched in the *normal* state, then `%s/search/replace/options` is entered, where:

- `search` is the regular expression;
- `replace` is the string that shows how to do the replacement.

For example, the command `%s/a/b/g` will globally replace `'a'` with `'b'`.

In the following sections we will use *vi* syntax for regular expressions.

3. **The Cupertino effect**

The Cupertino effect is the tendency of the editing program to replace[2] misspelled or unknown words with inappropriate words.

[2] using the spell checker.

According to [22], the origin of "Cupertino" name for this effect is that an editing program replaced the English word "cooperation" with "Copertino"[3].

Global application of the rules below is not appropriate[4], so at every context where the rule may be applied we ask, using the `c` option, user's permission to do the replacement.

4. Typesetting in TEX

TEX typesetting language was created because Professor Knuth was "disappointed" by the second edition of the second volume of "Art of Computer Programming", edition that was printed using new photographic techniques [14].

TEX language, being originally designed for English, complies to the English style of typesetting, style that is strongly influenced by the German one.

The rules that we set out below are specific to traditional Romanian style. For kerning we use the following LATEX commands:

- `\,`—inextensible thin space (standard command);
- `\stretchthinspace`—thin expandable space (thinsp package);
- `\stretchthinthinspace`—hair[5] expandable space (thinsp package).

5. The rules

To obtain optimal results, the rules must be applied to a LATEX source file in the order given bellow.

Rule 1. The law [16] establishes the requirement to use the standardized Romanian character set for documents addressed to any individual or entity. Romanian characters are described in [13] and [15].

According to [3], at "short distance" [18] below the letters S/T comma is placed to form the letters Ș/Ț. Surprisingly, this is recognized by foreigners [19, p. 327], [5, p. 294, p. 304], but is not well known in Romania because of the spreading of software "wrongly" using the cedilla "that is used under *c* in other languages" [3] to form *ç*.

- `%s/ţ/ț/g`
- `%s/ş/ș/g`
- `%s/Ţ/Ț/g`
- `%s/Ş/Ș/g`

[3] Home town of Apple and Hewlett-Packard.

[4] As we will show, in Romanian, the character that separates the decimal part from the fractional part is comma, not point. From this we can deduct the rule that, if we encounter [no].[no], we will replace the point with a comma. But these rules can not always apply, because sometimes the point between the two numbers is not the fractional separator, it is just a number separator, like in dentistry where it is used to identify teeth.

[5] Very thin.

Rule 2. According to [2, p. 79, p. 81], [4, 15, 18], [1, ¶3.6.2], in Romanian language quotation marks are „. . . ” (99 down, 99 up) or «. . . ».

Since many users don't have the computer configured for the Romanian language, the most common error is the use of quotation marks according to English rules “. . . ”.

```
%s/“/„/g
```

Rule 3. Inside words the letter â is used instead of î. Attention to the compound words, such as *“neîncetat”*, which is written with î!

```
%s/\([^0-9()␣]\)î\([^0-9␣]\)/\1â\2/gc
```

Rule 4. Space should not be before, but after the comma or period [12].

```
%s/\s*\([,.]\)\(\(jpg\)\|\(png\)\)\@!
\s*\([^0-9},␣~)]\)/\1␣\5/gc
```

Rule 5. Before an opening parenthesis one space is required; a very thin space is required after it [12], [9, p. 48].

```
%s/\s*[(]\s*\(\(\\stretchthinthinspace\)\|
\([0-9]+[)]\)\|\([0-9]+-*[0-9]+[)]\)\)\@!
/␣(\\stretchthinthinspace␣/gc
```

Rule 6. Before a closing parenthesis a very thin space is required; one space is required after it [12], [9, p. 48].

```
%s/\(\(\\stretchthinthinspace\)|\([0-9]+[)]\)\|
\([0-9]+-*[0-9]+[)]\)\)\@<!\s*[)]\s*
/\\stretchthinthinspace␣)␣/gc
```

Rule 7. No space after a closing parenthesis if it is followed by ,.;:!? [12].

```
%s/[)]\s\+\([,.;:!?]\)/)\1/gc
```

Rule 8. Before : an inextensible thin space is required; one space is required after it [12], [9, p. 59].

```
%s/\s*\(\\,\)\@<!:\s*/\\,:␣/gc
```

Rule 9. Before ;!? a thin space is required; one space is required after [12], [6, p. 50].

```
%s/\s*\(\\stretchthinspace\)\@<!\([;!?]\)\s*
/\\stretchthinspace␣\2␣/gc
```

Rule 10. The decimal marker is the comma, not the period [8, ¶2.1.2], [7, ¶5.3.4].

```
%s/\([0-9]\+\)[.]\([0-9]\+\)/\1,\2/gc
```

Rule 11. Do not hyphenate at dash [3]. Replace the dash character (Unicode 0x002D) with the "non-breaking hyphen" (Unicode 0x2011).

```
%s/\([a-zșțăîâ]\)-\([a-zșțăîâ]\)/\1-\2/gc
```

Rule 12. En dashes should be used to show a range [3, p. XCII], [20, ¶ 6.83], [5, p. 80].

```
%s/\([0-9]\+\(,[0-9]\+\)\?\)\s*[---]
\s*\([0-9]\+\(,[0-9]\+\)\?\)/\1--\3/gc
```

Rule 13. A non-breaking [10, p. 142] thin space is always used to separate the unit from the number [8, ¶4.3], [7, ¶5.3.3, ¶5.3.7].

```
%s/\([0-9]\+\(,[0-9]\+\)\?\)\s*
\(mm\|nm\|cmc\|cm\|ml\|mg\|N\|\%\|msec\|ms\)
/\1\\stretchthinthinspace␣\3/gc
```

Rule 14. The numbers, usually those longer than four digits [7, ¶5.3.4], should be divided into groups of three digits by a thin space, in order to facilitate reading [8, ¶2.1.3].

```
%s/\([0-9]\)\([0-9]\{3\}\)\([^0-9]\)/\1\\,\2\3/gc
```

Rule 15. The emdash character should be surrounded by spaces [12], [9, p. 49].

```
%s/\(\s*\)[---]\+\(\s\+\)\([␣]---[␣]\)\@<!/␣---␣/gc
```

Rule 16. No separation for slash / .

```
%s/\s\+[/]\s\+/\//gc
```

Rule 17. A big letter should not be the last letter of a line.

```
%s/\([␣~{}]\)\([A-Z]\)\s/\1\2\~/gc
```

6. References

1. Academia Republicii Socialiste România, *Cărţi şi broşuri: prezentarea redacţională,* STAS 8660–82, "Institutul român de standardizare", 03 1982.
2. Academia Română – Institutul de Lingvistică "Iorgu Iordan", *Îndreptar ortografic, ortoepic şi de punctuaţie*, "Univers Enciclopedic", Bucureşti, România, fifth edition, 2001.
3. Academia Română – Institutul de Lingvistică "Iorgu Iordan – Al. Rosetti", *DOOM — Dicţionarul ortografic, ortoepic şi morfologic al limbii române*, "Univers Enciclopedic", Bucureşti, România, second edition, 2005.
4. Academia Română, *Punctuaţia limbii române, I. Ghilimelele,* online, http://www.academiaromana.ro/com2006/doc/ghilimele.doc, 2006.
5. Bringhurst, R., *The elements of typographic style*, "Hartley & Marks", Publishers, Point Roberts, Washington, United States of America, second edition, 2004.

6. Brun, M.A., *Manuel pratique et abrégé de la typographie française*, Bruxelles, 1826.
7. Bureau international des poids et mesures, *Le Système international d'unités (SI)*. "STEDI Media", Paris, France, 8 edition, may 2006.
8. Consiliul culturii şi educaţiei socialiste, *Reguli pentru scrierea şi tipărirea notaţiilor în fizică şi matematică*. STAS 1508 – 81, Institutul român de standardizare, 11 1981.
9. Fournier, H., *Traité de la typographie*, "P. J. de Mat", Bruxelles, 1826.
10. Guéry, L., *Dictionnaire des règles typographiques*, En française dans le texte. "Victoires Éditions", Paris, France, quatrième edition, 2010.
11. Hopcroft, J.E., Motwani, R. and Ullman J.D., *Introduction to Automata Theory, Languages and Computation*, "Prentice Hall", 2006.
12. Imprimerie nationale, *Lexique des règles typographiques: en usage à l'imprimerie nationale*, "Imprimerie nationale", France, sixième edition, 2008.
13. ISO/IEC, *Tehnologia informaţiei. Set de caractere grafice codate pe un singur octet. Partea 16: Alfabetul latin nr. 10.* SR ISO/CEI 8859 – 16:2006, "International Organization for Standardization, Geneva, Switzerland", 2006.
14. Knuth, D.E., *Digital typography*, CSLI lecture notes. "CSLI Publications", 1999.
15. Ministerul Comunicaţiilor şi Tehnologiei Informaţiei, *Ordin cu privire la utilizarea codării standardizate a seturilor de caractere în documentele în formă electronică* "Monitorul Oficial al României", 174(842), 11–13, 10 2006.
16. Parlamentul României, *Lege privind utilizarea codificării standardizate a setului de caractere în documentele în formă electronică,* "Monitorul Oficial al României", 174(443), 4–5, 2006.
17. Ritchie, D., Thompson, K.L., *QED text editor,* Technical report, "Bell Laboratories", June 22 1970.
18. Sala, M., *Răspuns nr. 1010/09.10.2003,* Technical report, Institutul de Lingvistică "Iorgu Iordan – Al. Rosetti", http://secarica.ro/html/s-uri_si_t-uri.html, 2003.
19. Syropoulos, A., Antonis Tsolomitis, A. and Sofroniou, N., *Digital Typography using LaTeX*, "Springer-Verlag", New York, 2002.
20. University of Chicago, *The Chicago Manual of Style*, "University of Chicago Press", 16th edition, 2010.
21. Zafiu, R., *Din istoria punctuaţiei. . . ,* "România literară", 39(48), 1 decembrie 2006.
22. ***, *The word: Cupertino effect*, "New Scientist", 178(2632), 62, December, 2007.