Text Tokenisation Using `unitok`

Jan Michelfeit, Jan Pomikálek, Vít Suchomel

Natural Language Processing Centre
Faculty of Informatics
Masaryk University

and
Lexical Computing Ltd.

5 December 2014

Introduction

The aim of this work was to develop a tokeniser

- fast – able to process big data in billion word sized corpora,
- reliable – robust to deal with messy web data,
- universal – allowing at least basic support for all writing systems utilizing a whitespace to separate words,
- easy to maintain – adding new tokenisation rules or making corrections based on evaluation should be straightforward,
- text stream operating – text in, tokenised text (one token per line) out,
- reversible – the tokenised output must contain all information needed for reconstructing the original text.

Implementation

- Python script utilising the `re` library and operating on a text stream.
- The input text is decoded to unicode, normalised ('*&*', whitespace), scanned for sequences forming tokens, the tokens are separated by line breaks and the result vertical is encoded into the original encoding.
- Sequences of letters, numbers, marks, punctuation, and symbols are clustered together. SGML markup is preserved. URLs, e-mail addreses, DNS domains, IP addresses, general abbreviations are recognized.
- Predefined language specific rules: clitics ('*d'accord*'), abbreviations ('*např.*'), or special character rules (Unicode `0780-07bf` = Maldivian script Thaana).

Reversibility of tokenisation

- A ‘glue’ XML element inserted between tokens not separated by a space in the input data.
- Plaintext → `unitok` → vertical → `vert2plain` → plaintext.

```
The
"
<g/>
end
<g/>
"
<g/>
.
```

Evaluation – tokenising speed and output size

Language	tool	output tokens	rel tok	duration	tok/s	rel tok/s
English	Unitok	207,806,261	100%	6,880 s	30,200	100%
	TTWrapper	200,122,178	−3.70%	2,380 s	84,100	+178%
	Freeling	215,790,562	+3.84%	2,670 s	80,800	+168%
Spanish	Unitok	196,385,184	100%	6,250 s	31,400	100%
	TTWrapper	204,867,056	+4.32%	2,260 s	90,600	+188%
	Freeling	201,413,272	+2.56%	2,040 s	98,700	+214%
German	Unitok	171,354,427	100%	5,530 s	31,000	100%
	TTWrapper	179,120,243	+4.53%	2,360 s	75,900	+145%
French	Unitok	202,542,294	100%	6,400 s	31,600	100%
	TTWrapper	242,965,328	+20.0%	2,870 s	84,700	+168%
	Freeling	211,517,995	+4.43%	2,300 s	92,000	+191%
Russian	Unitok	98,343,308	100%	3,170 s	31,023	100%
	Freeling	102,565,908	+4.29%	1,450 s	70,800	+128%
Czech	Unitok	183,986,726		5,960 s	30,900	

Evaluation

Comments

- noticeable difference between the tools in the number of output tokens,
- `unitok` was the slowest of the three tools but still quite sufficient for fast processing of large text data,
- Freeling (with proper settings) recognises numbers, dates, even named entities (turned off) – might be useful,
- `unitok` and TTWrapper deal well with internet mess (still problems in recognising some emoticons).

We use

- Freeling for Spanish, Protuguese, Catalan,
- `unitok` for other languages with spaces between words,
- specialised tools for other languages (e.g. Stanford Segmenter for Chiese).

Conclusion

- `unitok` is a tokeniser for texts with spaces between words.
- It has been successfully used for tokenising large web corpora. [Jakubíček et al: The tenten corpus family, 2014]
- The main benefits:
  - good coverage of various sequences of characters, especially web phenomena,
  - normalisation of messy control or whitespace characters,
  - reversibility of the tokenised output,
  - extensibility by language specific rules (similarly to TTWrapper).

The Problem of Words

- What is a word, what is a sentence? Problems of Tokenisation. [Grefenstette, 1994]
- Our approach: corpus search – concordancer and other corpus inspection tools. What tokens are expected to figure in the corpus based analysis (such as word frequency lists, collocations, thesaurus)?
- The users search for sequences of letters.
- Sequences of numbers, marks, punctuation, symbols, separators and other characters should be clustered together in order to be counted as single tokens in corpus statistics.