quanteda

Quantitative Analysis of Textual Data

15 July 2018

Kohei Watanabe
Waseda Institute for Advanced Study/
London School of Economics and Political Science

LSE SEDS Social and Economic Data Science

QI quanteda initiative

What is **quanteda**?

quanteda is a open-source quantitative text analysis tool in R.

- Stands for **qu**antitative **an**alysis **te**xtual **da**ta
- Prof. Kenneth Benoit (LSE) started developing in 2012 supported by the EU
- Developed for high consistency and performance
  - Many of the core functions are written in C++ with multithreading
  - Faster than other R packages (**tm** or **tidytext**) and even Python package (**gensim**)
  - Works with Asian languages texts (Chinese, Japanese and Korean)
- Stand-alone package but also used by other packages as a framework
  - **politeness**, **preText**, **sentometrics**, **stm** etc.

quanteda
Quantitative Analysis of Textual Data

Developers

Kenneth Benoit (LSE)

Kohei Watanabe (Waseda U)

Akitaka Matsuo (LSE)

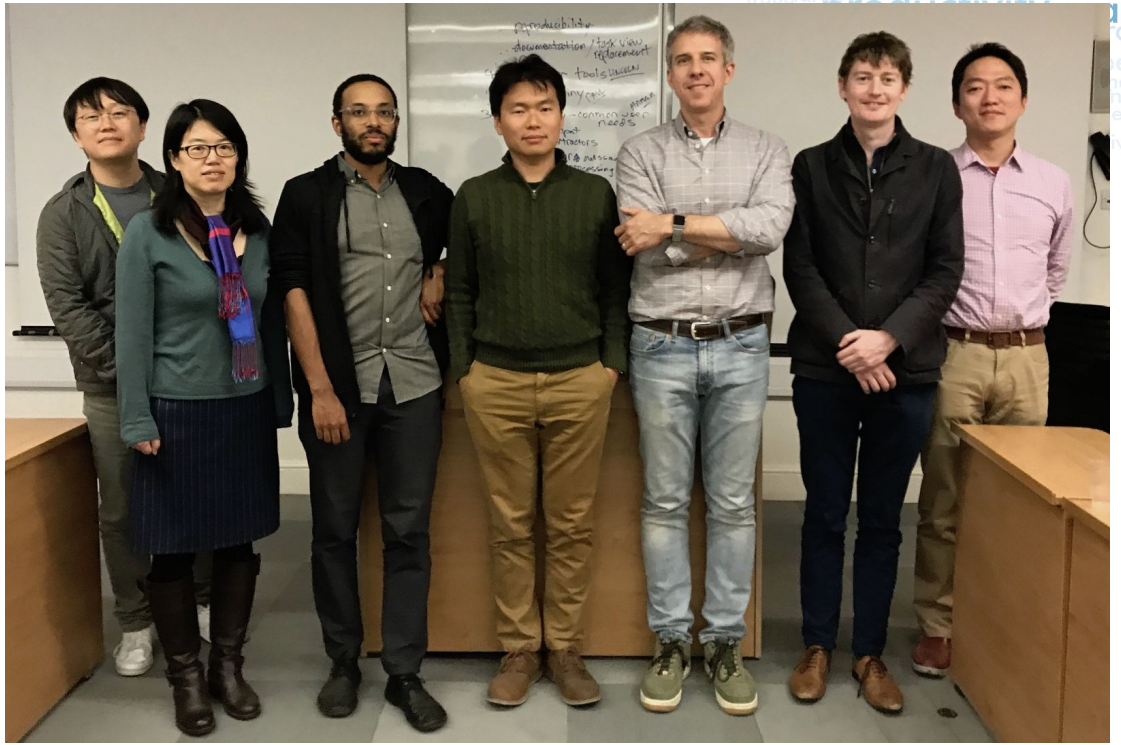
Haiyan Wang (De Beers)

Paul Nulty (Cambridge U)

Adam Obeng (Facebook)

Stefan Müller (Trinity College)

Benjamin Lauderdale (LSE)



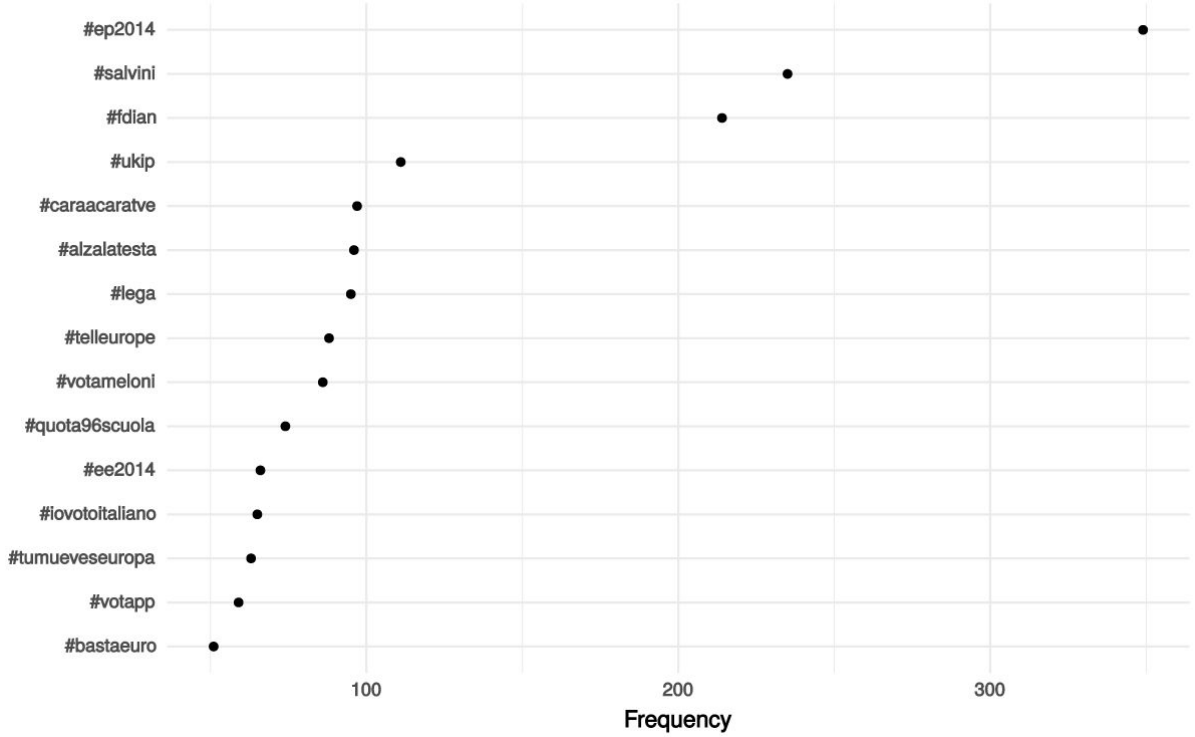
What is quantitative text analysis?

Application of NLP in social sciences and humanities in the context “big data”.

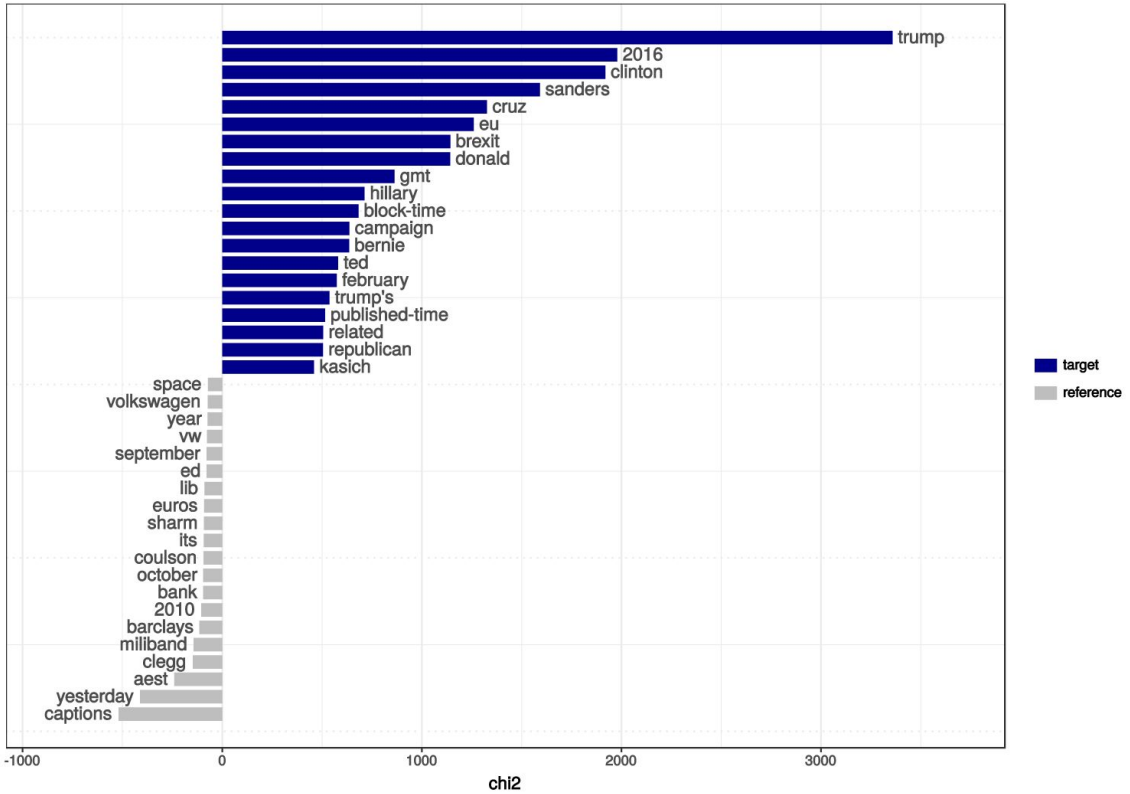
- Political Science
  - Estimate political ideology through election manifestos
  - Gauge complexity of political speeches
  - Monitor people’s response to election campaign on social media
- Media studies
  - Detect political bias in news articles
  - Extract mentions of political actors in new articles
- (Digital) humanity
  - Identify real authors of novels
  - Classify books according to the themes

quanteda
Quantitative Analysis of Textual Data

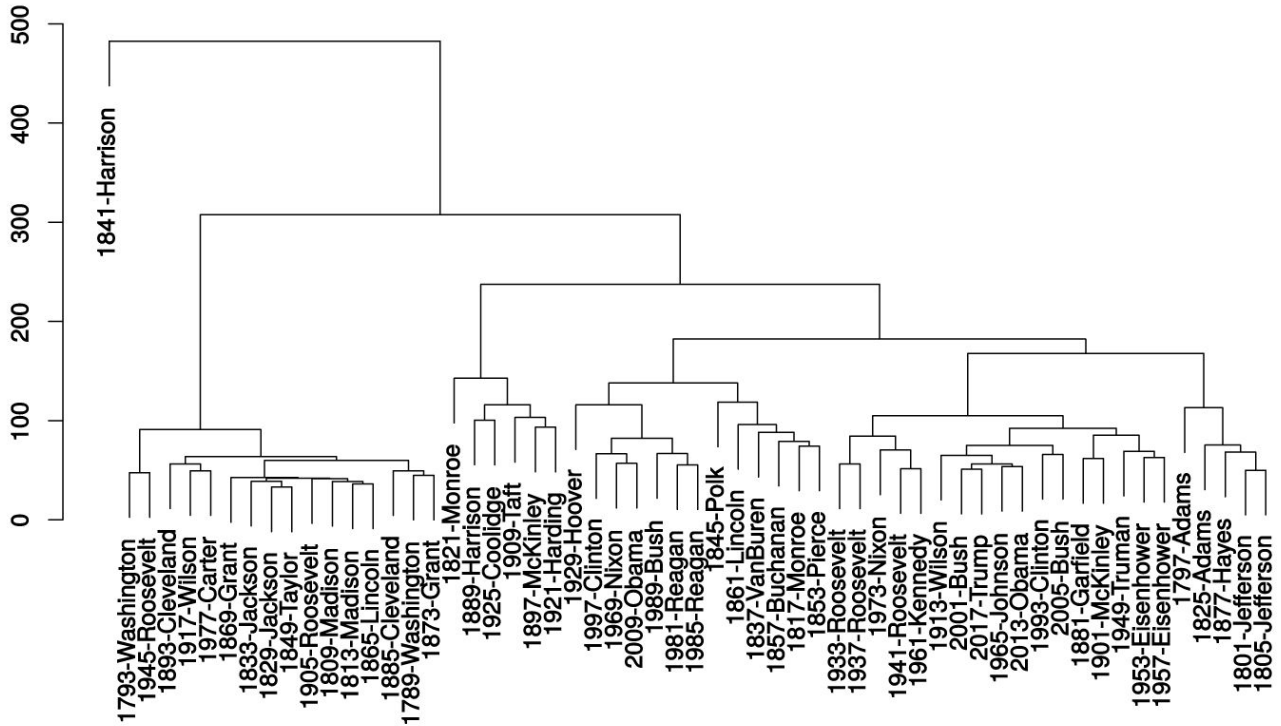
Simple frequency



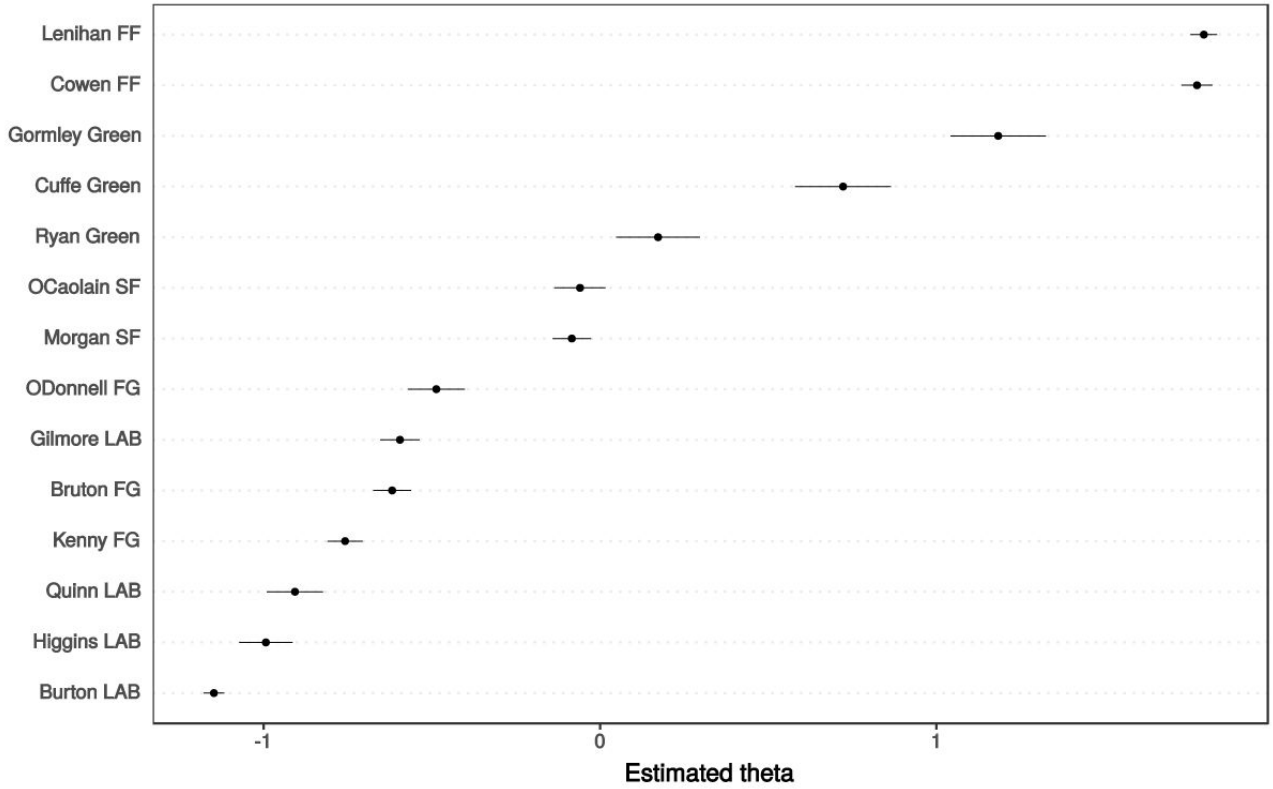
Relative frequency (keyness)



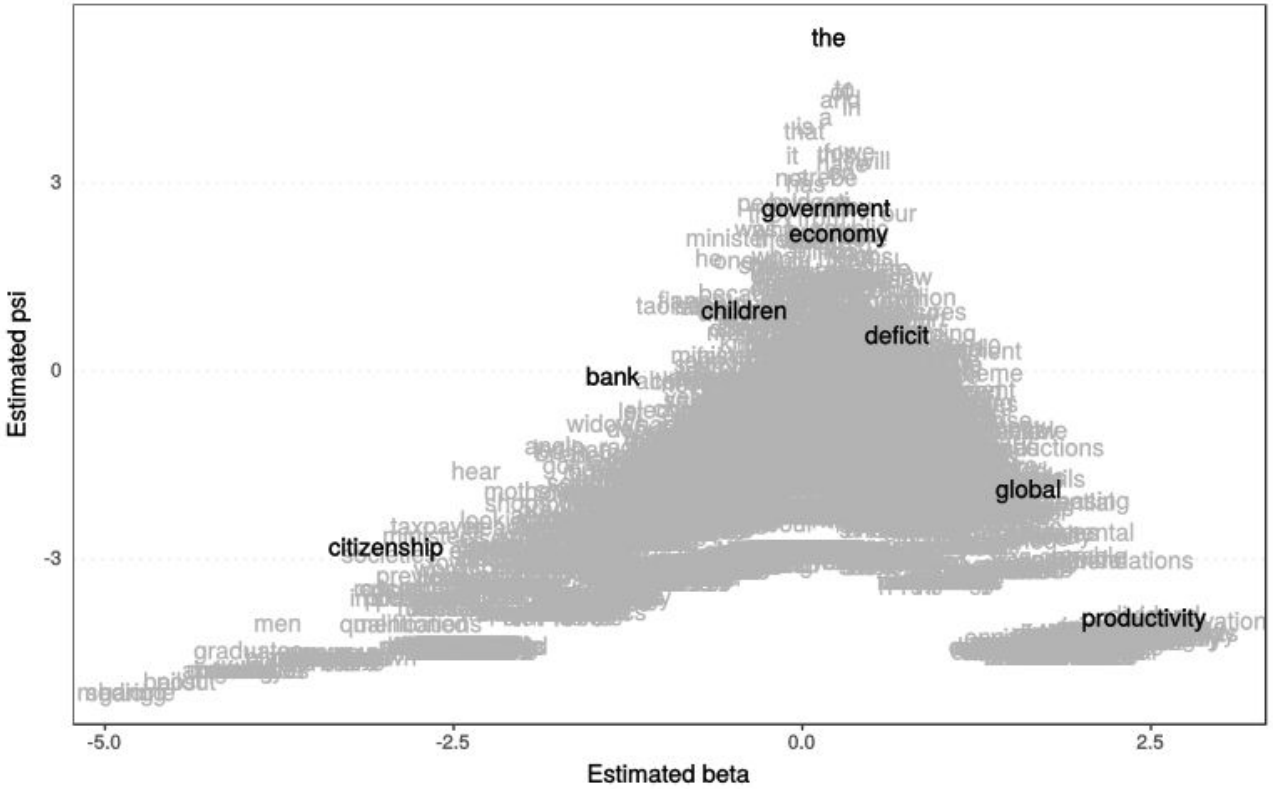
Document similarity



Document scaling (wordfish)



Estimated word parameters (wordfish)



Potential industrial applications

Quantitative text analysis is not limited to social scientific research.

- Investors
  - Discover prospective companies to invest based on their financial reports
  - Predict future stock prices through analysis of financial newswires
- Marketers
  - Identify the most important issues in customer satisfaction surveys
  - Monitor reputation of brands or companies through analysis of social media
- Recruiters
  - Find good job applicants through analysis of their cover letters or essays

quanteda
Quantitative Analysis of Textual Data

Functions and objects

quanteda
Quantitative Analysis of Textual Data

Functions for corpus

A **corpus** object contains texts with document-level variables.

`corpus()`	Construct a corpus
`corpus_reshape()`	Recast the document units of a corpus
`corpus_sample()`	Randomly sample documents from a corpus
`corpus_segment()`	Segment texts into component elements
`corpus_subset()`	Extract a subset of a corpus
`corpus_trim()`	Remove sentences based on their token lengths or a pattern match

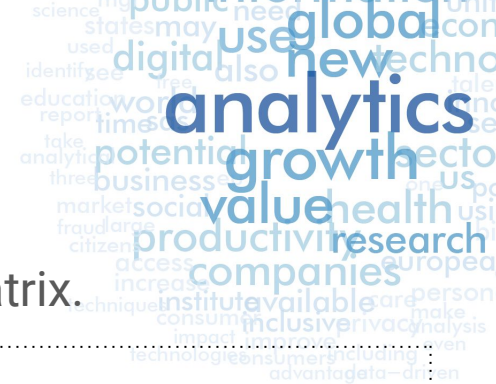
quanteda
Quantitative Analysis of Textual Data

Functions for tokens

A **tokens** object contains individual words or symbols as tokens.

`tokens()`	Tokenize a set of texts
`tokens_compound()`	Convert token sequences into compound tokens
`tokens_lookup()`	Apply a dictionary to a tokens object
`tokens_select()`,`tokens_remove()`	Select or remove tokens from a tokens object
`tokens_ngrams()`,`tokens_skipgrams()`	Create ngrams and skipgrams from tokens
`tokens_tolower()`, `tokens_toupper()`	Convert the case of tokens
`tokens_wordstem()`	Stem the terms in an object

Functions for document-feature matrix



A **dfm** object contains frequencies of words or symbols in a matrix.

`dfm()`	Create a document-feature matrix
`dfm_group()`	Recombine a dfm by grouping on a variable
`dfm_lookup()`	Apply a dictionary to a dfm
`dfm_select(), dfm_remove()`	Select features from a dfm or fcm
`dfm_trim()`	Trim a dfm using frequency threshold-based feature selection
`dfm_weight()`	Weight a dfm, including full SMART scheme, tf-idf, etc. in a dfm
`dfm_wordstem()`	Stem the features in a dfm
`fcm()`	Create a feature co-occurrence matrix

Statistical analytic functions

textstat_*() functions perform statistical analysis of textual data.

`textstat_collocations()`	Calculate collocation statistics
`textstat_dist()`	Distance computation between documents or features
`textstat_keyness()`	Calculate keyness statistics
`textstat_lexdiv()`	Calculate lexical diversity
`textstat_readability()`	Calculate readability
`textstat_simil()`	Similarity computation between documents or features

Machine learning functions

textmodel_*() functions perform machine learning on textual data.

`textmodel_ca()`	Correspondence analysis of a document-feature matrix
`textmodel_lsa()`	Latent semantic analysis of a document-feature matrix
`textmodel_nb()`	Naive Bayes (multinomial, Bernoulli) classifier for texts
`textmodel_wordfish()`	Slapin and Proksch (2008) text scaling model
`textmodel_wordscores()`	Laver, Benoit and Garry (2003) text scaling
`textmodel_affinity()`	Perry and Benoit (2017) class affinity scaling
`covert()`	Interface to other R packages (**topicmodels**, **stm** etc.)

Visualization functions

textplot_*() functions plots textual data.

`textplot_scale1d()`	Plot a fitted scaling model
`textplot_wordcloud()`	Plot features as a wordcloud
`textplot_xray()`	Plot the dispersion of key word(s)
`textplot_keyness()`	Plot association of words with target v. reference set

Asian-language support: Japanese

Japanese segmentation without morphological analysis tool (e.g. Mecab).

```
> txt_jp <- "政治とは社会に対して全体的な影響を及ぼし、社会で生きるひとりひとりの人の人
生にも様々な影響を及ぼす複雑な領域である。"
> quanteda::tokens(txt_jp)
tokens from 1 document.
text1 :
 [1] "政治"       "と"          "は"          "社会"        "に対して"
 [6] "全体"       "的"          "な"          "影響"        "を"
[11] "及"         "ぼ"          "し"          "、"          "社会"
[16] "で"         "生きる"      "ひとりひとり" "の"          "人"
[21] "の"         "人生"        "に"          "も"          "様々"
[26] "な"         "影響"        "を"          "及ぼす"      "複雑"
[31] "な"         "領域"        "で"          "ある"        "。"
```

quanteda
Quantitative Analysis of Textual Data

Asian-language support: Chinese

Chinese segmentation without morphological analysis (e.g. jieba)

```
> txt_cn <- "政治是各种团體进行集体决策的一个过程，也是各种团體或个人为了各自的領域所结
成的特定关系，尤指對於某一政治實體的統治，例如統治一個國家，亦指對於一國內外事務之監督
與管制。"
> quanteda::tokens(txt_cn)
tokens from 1 document.
text1 :
 [1] "政治"   "是"     "各种"   "团"     "體"     "进行"   "集体"   "决策"
 [9] "的"     "一个"   "过程"   "，"     "也是"   "各种"   "团"     "體"
[17] "或"     "个人"   "为了"   "各自"   "的"     "領域"   "所"     "结成"
[25] "的"     "特定"   "关系"   "，"     "尤"     "指"     "對於"   "某一"
[33] "政治"   "實體"   "的"     "統治"   "，"     "例如"   "統治"   "一個"
[41] "國家"   "，"     "亦"     "指"     "對於"   "一"     "國內外" "事務"
[49] "之"     "監督"   "與"     "管制"   "。"
```

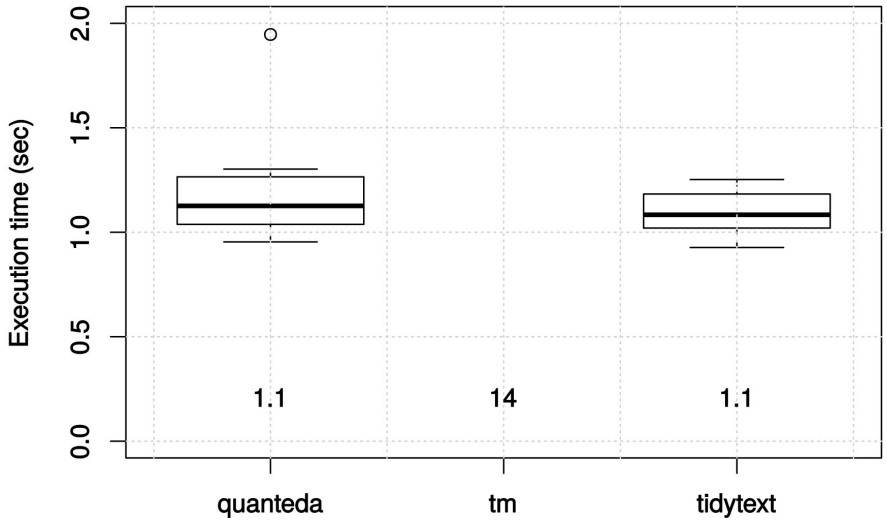
quanteda
Quantitative Analysis of Textual Data

Performance

quanteda
Quantitative Analysis of Textual Data

Tokenization

Tokenization using **stringi** to fully support Unicode



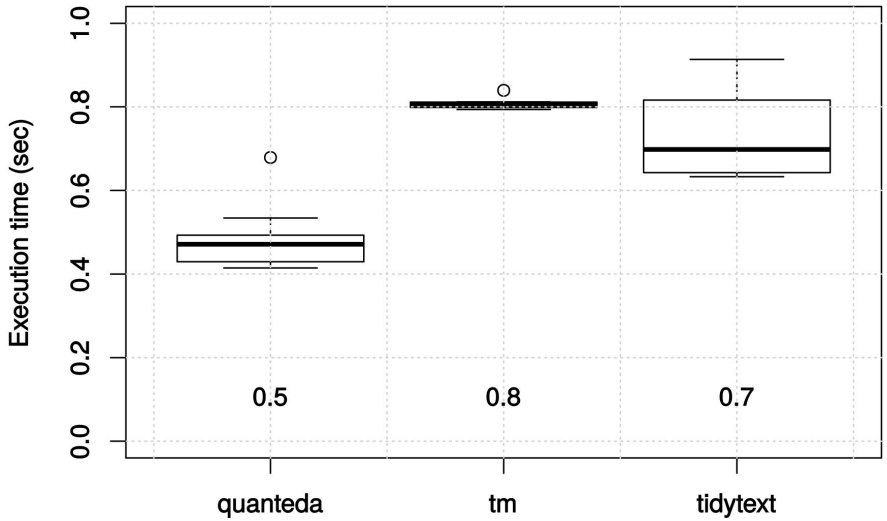

```
# quanteda commands

txt <- readLines("guardian.txt")
corp <- corpus(txt)
toks <- tokens(corp,
               what = "fastestword")
```

The gorpus contains 6,000 full-text Guardian news articles (10MB)

quanteda
Quantitative Analysis of Textual Data

Remove stopwords

Selection of tokens or sequences of tokens (multi-word expressions)



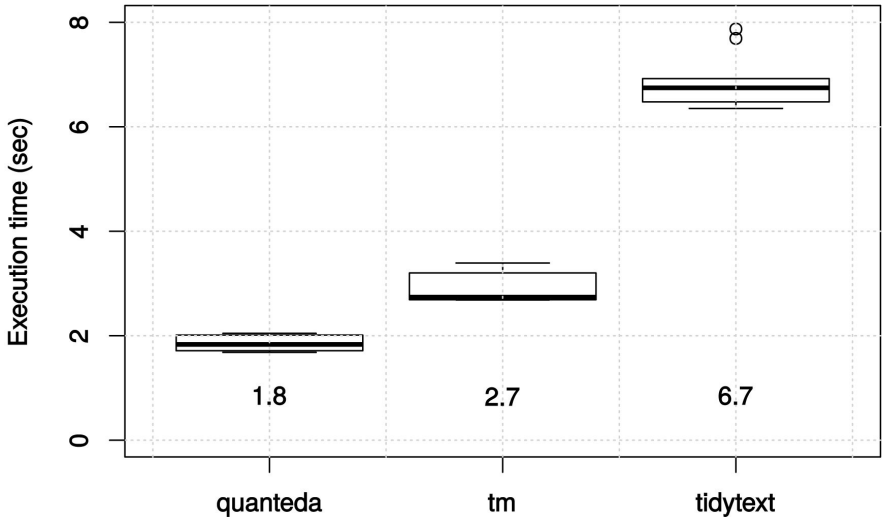

```
# quanteda commands

toks2 <- tokens_remove(toks, stopwords())
```

Stopwords contain 175 English function words from the stopwords() package

quanteda
Quantitative Analysis of Textual Data

Document-feature matrix

Tokenization and document-feature matrix construction using **Matrix**

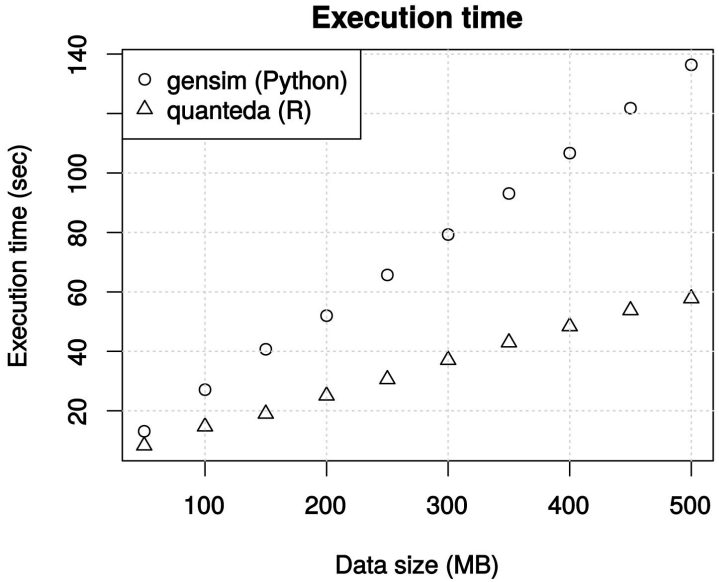


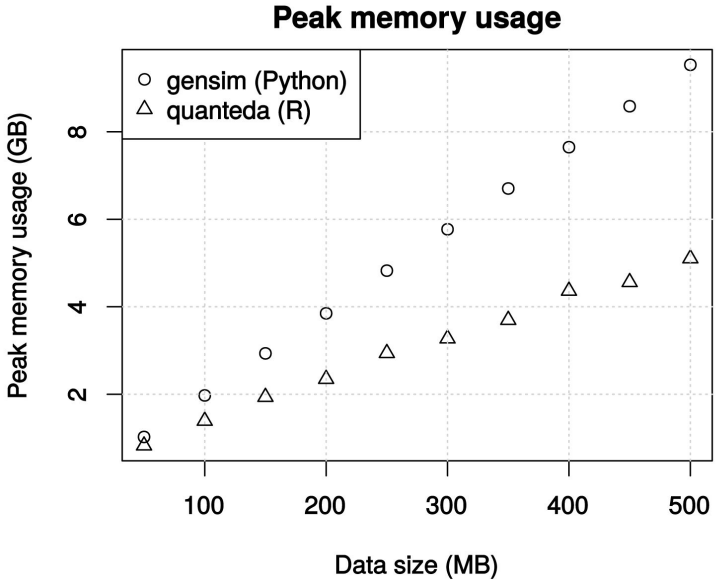

```
# quanteda commands

mt <- dfm(corp, what = "fastestword")
```

quanteda
Quantitative Analysis of Textual Data

Comparison with Python

2x more efficient than Python in speed and memory

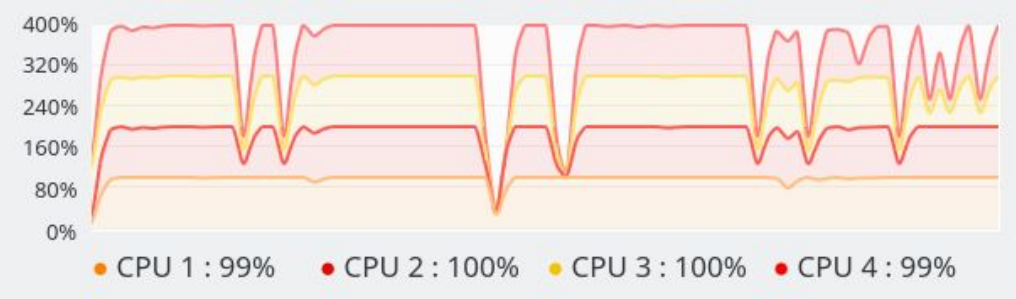




Test code is available at https://koheiw.net/?p=468

Secrets of high performance

- Data serialization
  - Tokens are serialized to reduce memory usage by 50 to 70%
  - Speeds up all the basic operations
- Parallel computation
  - Many of the tokens_*() and textstat_*() functions are parallelized using **RcppParallel**
  - Parallelization in C++ is much more efficient than in R



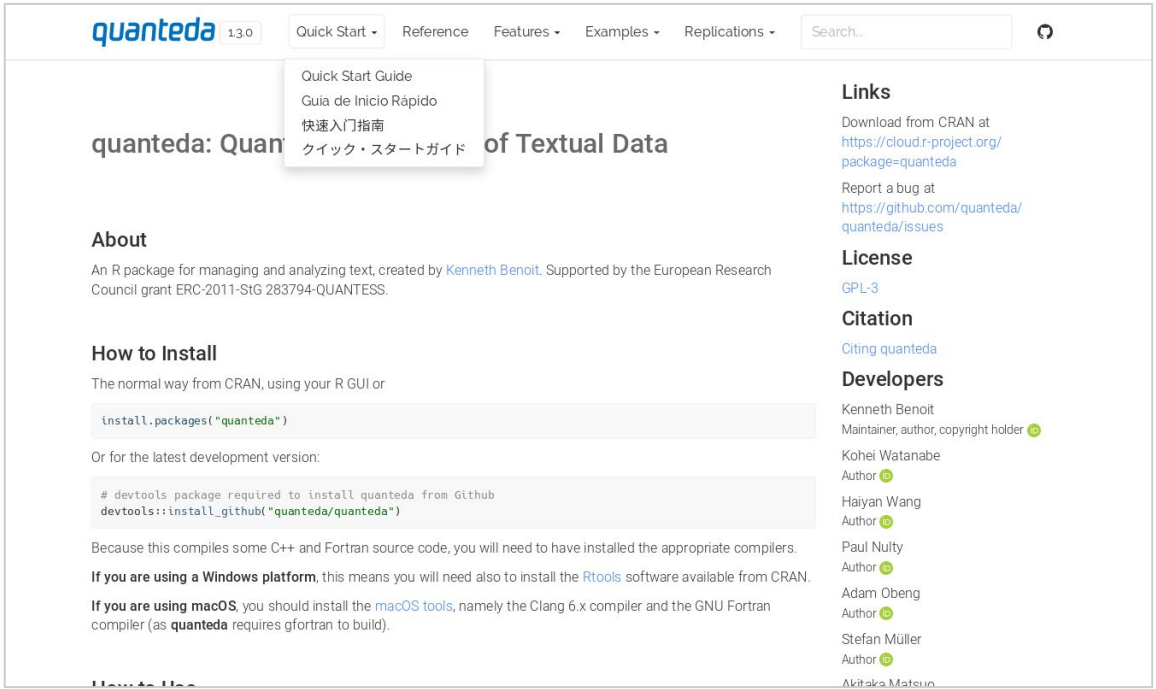
quanteda
Quantitative Analysis of Textual Data

Additional resources

quanteda
Quantitative Analysis of Textual Data

Documentation website

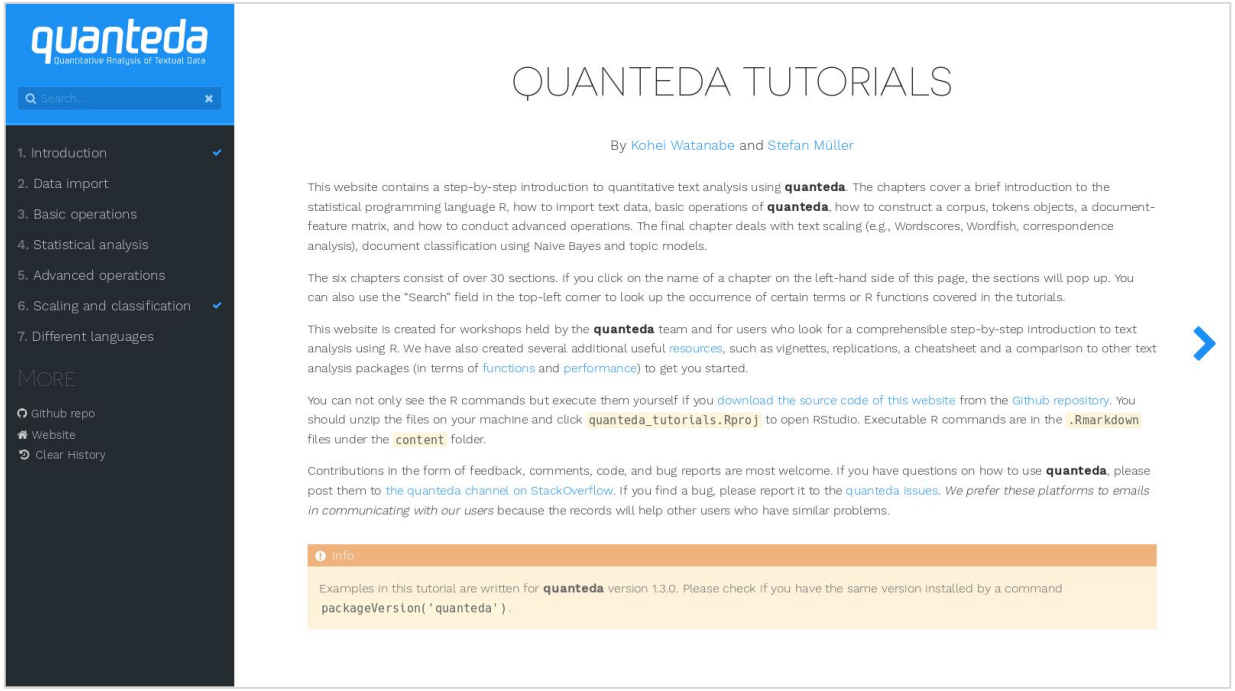
https://docs.quanteda.io



quanteda
Quantitative Analysis of Textual Data

Tutorials website and workshops

https://tutorials.quanteda.io



quanteda
Quantitative Analysis of Textual Data

Contact us

We are happy to help you with quantitative text analysis.

- Contacts us
  - Kohei Watanabe (Tokyo, koheiw@quanteda.org)
  - Kenneth Benoit (London, kbenoit@quanteda.org)
- Previous workshops
  - Japan (Waseda U, Kobe U, Tokyo U)
  - Germany (Berlin Social Science Center)
  - Israel (Tel Aviv U)
  - UK (LSE, Alan Turing Institute)
  - Hungary (Hungarian Academy of Science)
  - More....

LSE SEDS Social and Economic Data Science

QI quanteda initiative