QUITA – Quantitative Index Text Analyzer

QUITA

Quantitative Index Text Analyzer

Miroslav Kubát
Vladimír Matlach
Radek Čech

2014
RAM-Verlag

Contents

1 Introduction

Quantitative experimental methods have been increasingly used in the humanities in recent years. We can hardly imagine the disciplines of social science, such as psychology, sociology or economics, without a quantitative approach. On the other hand, the majority of linguists, historians and especially literary critics are still refusing to use quantitative methods. One of the reasons is the fact that those researchers consider quantitative methods, and especially statistical methods, too difficult to apply to their field. QUITA (Quantitative Indicator Text Analyzer) is a tool which aims to help all people who try to analyse texts by quantitative methods.

QUITA is a program to enable researchers from various disciplines (linguistics, criticism, history, sociology, psychology, politics, biology, etc.) to analyse texts using quantitative methods. There are many indicators which measure various characteristics of text. Although the authors of QUITA focused mainly on indicators connected to the frequency structure of a text, there are also functions for several other characteristics. Since QUITA is designed especially for researchers outside quantitative linguistics, it includes functions for the most basic and common indicators.

Given that the main purpose of QUITA is to provide a user-friendly tool of quantitative text analysis for researchers without a deeper knowledge of quantitative linguistics, statistics or programming, QUITA also provides simple statistical comparisons and the ability to create charts. There is no need to use any additional software such as spreadsheet applications or special statistical programs. QUITA is therefore the program that combines all the essential parts of any quantitative research effort: obtaining results, statistical testing and graphical visualization.

The QUITA manual is written with step-by-step instructions. All tools are concisely described and accompanied by screen-shots. Every indicator is briefly presented (complete with references), and mathematically defined. There are also examples of computation and statistical comparison.

Although the manual provides users with all the essential information about QUITA, it was not possible to cover most topics in deeper detail. For this purpose, we highly recommend the book *Word frequency studies* (Popescu et al. 2009) which is a comprehensive overview about quantitative analysis using indicators based on the frequency structure of a text. The book *Aspects of Word Frequencies* (Popescu et al. 2009) is also well worth reading. Detailed examples of computing most indicators used in QUITA can be found in *Metody kvantitativní analýzy (nejen) básnických textů* (Čech et al. 2014).

Since we aim to help as many researchers as possible, QUITA is distributeed as freeware. Thus anyone can use QUITA without any restrictions. The latest version of the software is available on the website http://oltk.upol.cz/software. In published work, acknowledgement of QUITA would be appropriate and appreciated.

2 System requirements

Supported Systems: Windows XP, Windows Vista, Windows 7, Windows 8.
System requirements: NET Framework 3.5
Optional system requirements: Python, Perl

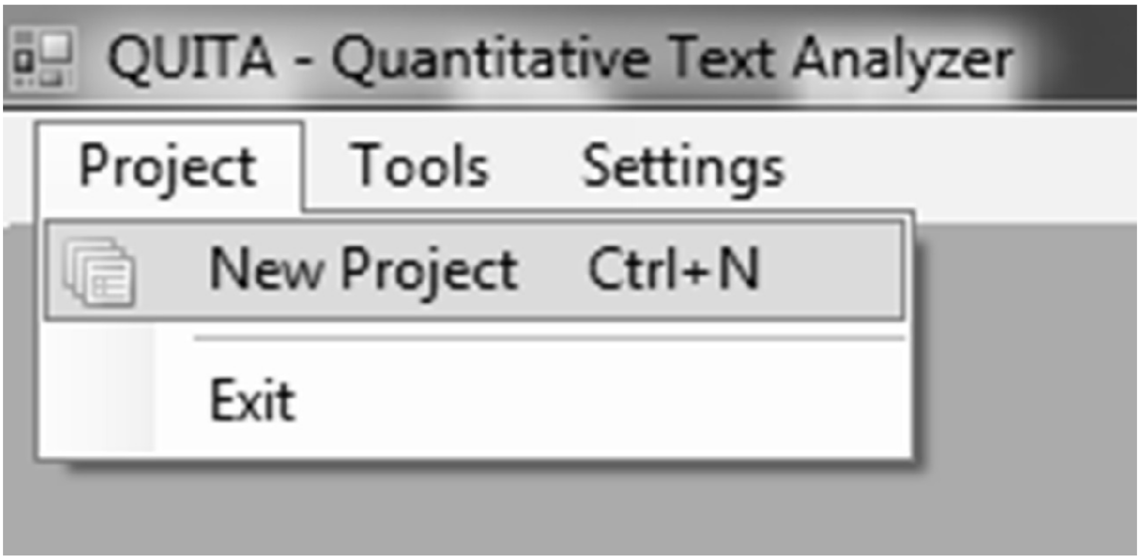
NOTE:
Users are automatically notified that it is necessary to install the system requirements with the download links.

3 Install and Starting the Application

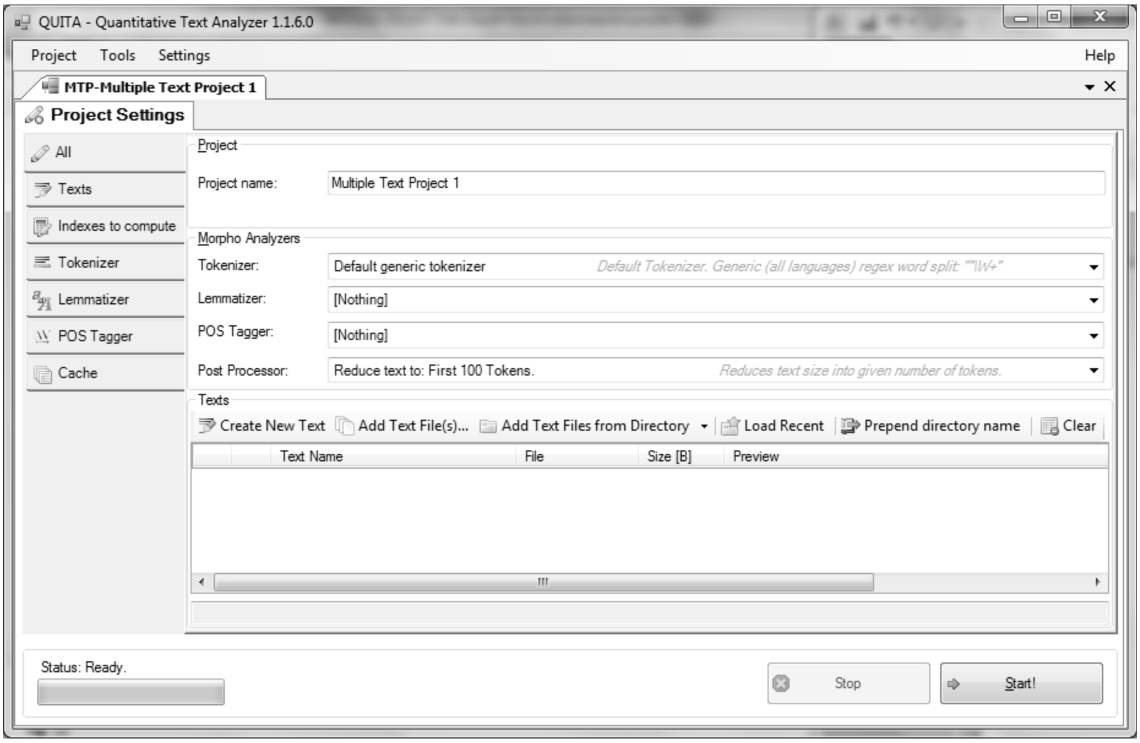
1. Download the latest installation file from the project website: http://oltk.upol.cz/software.
2. Open the installation file and follow the installation instructions.
3. Create a new shortcut to the QUITA application on the user's desktop.
4. Start the application by clicking the application shortcut or start QUITA directly by opening the QUITA executable file (QITA_OLTK.exe), which can be found in the installation folder.

4 Creating a New Project

To start a new project, click on “Project” and “New Project” or just press keyboard shortcut CTRL+N.



The New Project window should appear with the “Project Settings – All” card active.



Project

Project name: Multiple Text Project 1

Morpho Analyzers

Tokenizer: Default generic tokenizer *Default Tokenizer. Generic (all languages) regex word split: ""\W+"*

Lemmatizer: [Nothing]

POS Tagger: [Nothing]

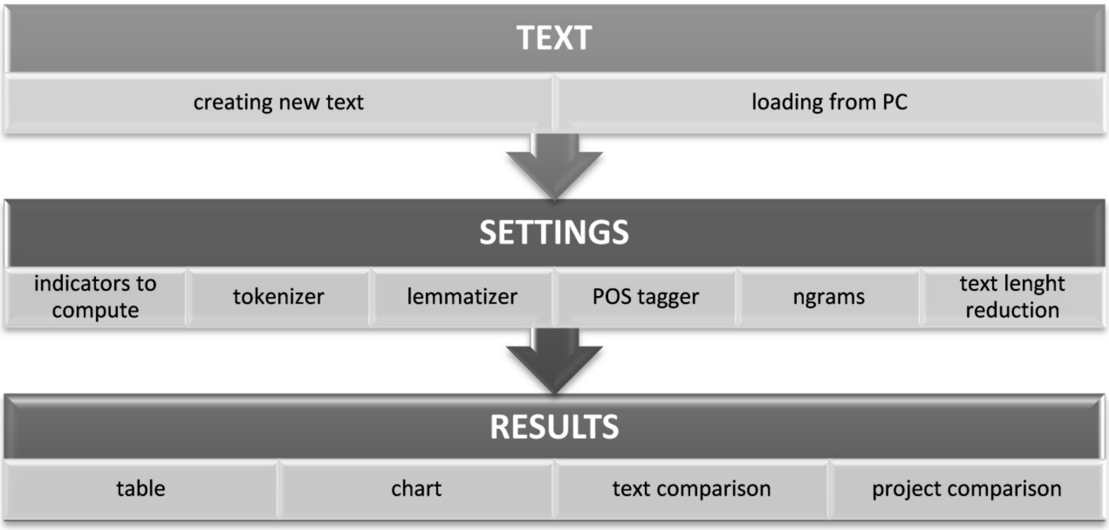
Post Processor: Reduce text to: First 100 Tokens. *Reduces text size into given number of tokens.*

Texts

		Text Name	File	Size [B]	Preview

Status: Ready.

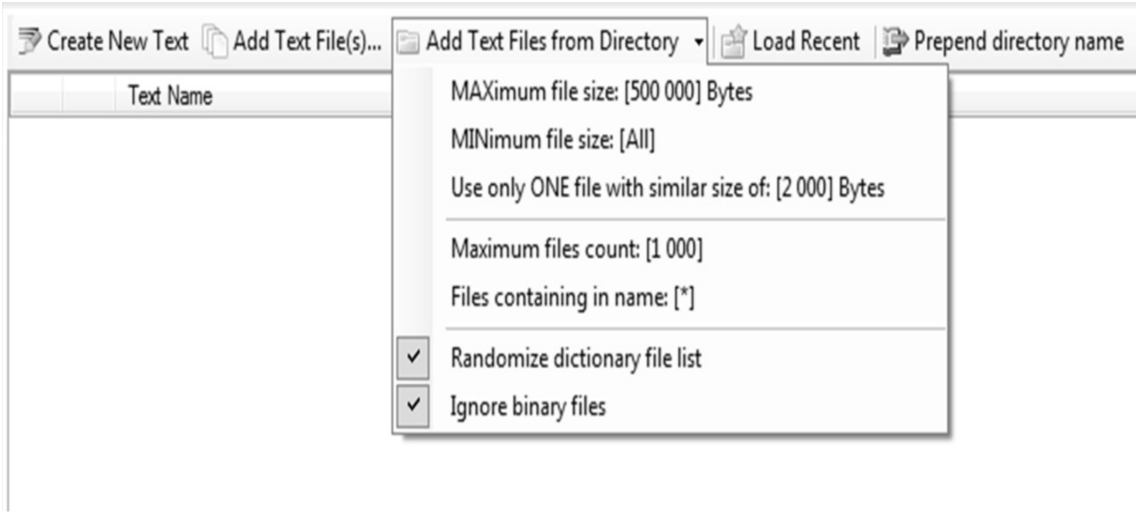
The card “Project Settings – All” displays a summary of the current project settings. Use the other cards below to see detailed settings of each particular setting for the project. This manual describes the following steps of project setup and (after processing) working with final results:



5 Creating and loading texts

New texts can be created or loaded in the main “Project Settings – All” card or, for better convenience and clarity, in the “Project Settings – Texts” card.

There are five options for how to input a text into the project:

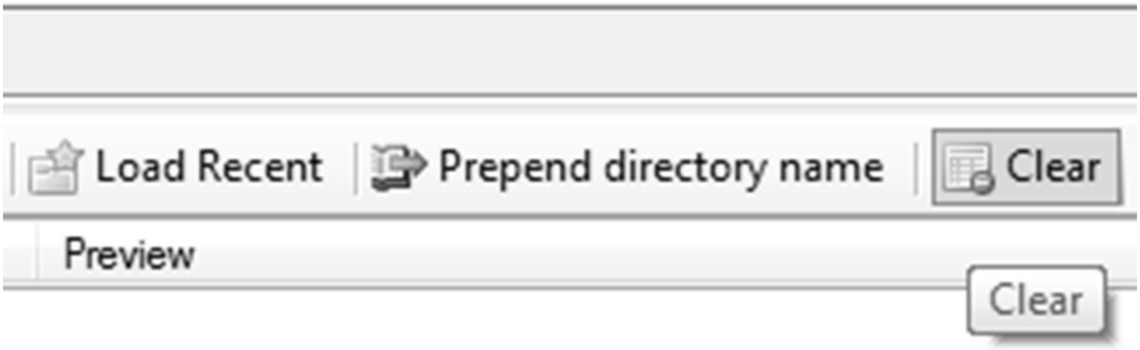


MAXimum file size: [500 000] Bytes
MINimum file size: [All]
Use only ONE file with similar size of: [2 000] Bytes
Maximum files count: [1 000]
Files containing in name: [*]
☑ Randomize dictionary file list
☑ Ignore binary files

- Click on “Create New Text” to write your own text.
- Click on “Add Text File(s)” to load a file or files from your drive.
- Click on “Add File(s) from Directory” to load all the files contained in a selected directory and all its sub-directories. By clicking on the arrow, you will get more choices for this option (mainly for limitation purposes):
    - “Maximum file size” and “Minimal file size” adjust the maximal and minimal file size in Bytes. Only files fitting the given size will be loaded.
    - “Use only one file with similar size of” specifies whether to load files whose size in Bytes is similar to the already loaded ones. This option helps to load files with unique sizes.
    - “Maximum files count” specifies the maximum number of files to load.
    - “Files containing in name” specifies the substring which has to be contained in the loaded file name.
    - “Randomize dictionary file list” prevents the system from loading files from a given dictionary in alphabetical order.
    - “Ignore binary files” specifies whether binary files (like images, sound files, executable files etc.) can be loaded or not. More about this feature below.
- Click on “Load Recent” for loading your recent text(s).
- Drag files with the mouse and drop them into the file list table.

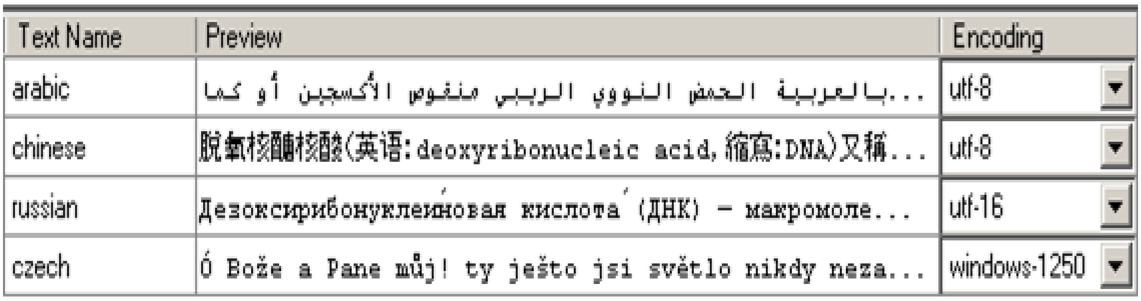
The “Prepend directory name” button adds a directory name in front of the loaded file name to help disambiguate file identity (e.g. when loading numbered text files from two different directories).

All texts loaded into the project are listed in the Texts table. All the texts together can be removed by clicking on the “Clear” button; then replace them with new texts. A single text can be removed by selecting it and then pressing the Delete button on the keyboard.



5.1 Supported files and encodings

QUITA supports the usual plain text files in .txt format in various encodings. After a text file is loaded, QUITA attempts to detect a suitable text encoding for this file by several methods. However, the methods used for this task are mainly based on heuristics and may fail. The encoding of any loaded text can be changed by choosing a new encoding in the drop down menu on the right side of the file table. The new encoding is immediately applied to the text and its result can be seen in the Preview table cell. Each text in the project can have its own encoding and can be in any supported language (see the picture below).



Text Name	Preview	Encoding
arabic	بالعربية الحمض النووي الريبي منقوص الأكسجين أو كما...	utf-8 ▼
chinese	脱氧核醣核酸(英语:deoxyribonucleic acid，縮寫:DNA)又稱...	utf-8 ▼
russian	Дезоксирибонуклеи́новая кислота́ (ДНК) — макромоле...	utf-16 ▼
czech	Ó Bože a Pane můj! ty ješto jsi světlo nikdy neza...	windows-1250 ▼

All the usual line endings (Cr, Lf, CrLf) are supported.

6. Indicators to compute

In the “Project Settings – Indicators to compute”, you can tick all the indicators suitable for your purpose. Unticked indicators are not computed for the current project (except indicators which are in dependency with any ticked indicator(s)). However, a project can be processed again with a new selection of indicators.

Given that QUITA is aimed at researchers outside quantitative linguistics, there are only the most common and basic indicators to compute. Most indicators deal with the frequency structure of a text which is at the core of the QUITA aim. The indicators are therefore divided into two groups:

- **Frequency Structure indicators**
    - Type-Token Ratio (*TTR*)
    - *h*-point (*h*)
    - R_1
    - Repeat Rate (*RR*)
    - Relative Repeat Rate of McIntosh (RR_{mc})
    - Hapax Legomenon Percentage (*HL*)
    - Lambda (Λ)
    - Gini Coefficient (*G*)
    - R_4
    - Curve length (*L*)
    - Curve length R Indicator (*R*)
    - Entropy (*H*)
    - Adjusted Modulus (*A*)

- **Miscellaneous indicators**
    - Verb Distances (*VD*)
    - Activity (*Q*) & Descriptivity (*D*)
    - Writer’s View (α)
    - Average Tokens length (*ATL*)
    - Thematic Concentration (*TC*)
    - Secondary Thematic Concentration (S*TC*)
    - Proportional Thematic Concentration (*PTC*)

In the following chapters, every indicator is briefly presented, complete with references, and mathematically defined. Considering the manual is aimed at researchers without deeper mathematical skills, formulas for the indicators may not be comprehensible to everyone. Thus, each chapter is accompanied by an example. So an entire computation is described step by step in an example. For this purpose, two short English texts were selected, namely the second paragraph of the novel *Nineteen Eighty-Four* (Text 1) and the first two paragraphs of the novel *Animal Farm* (Text 2) both written by George Orwell. These very short texts were chosen in order to enable readers to understand the examples in detail. The texts have similar lengths, *Nineteen Eighty-Four* 179 and *Animal Farm* 202 words. The texts can be found in the appendix of the manual; there also are their frequency distributions. Given that these two texts are not appropriate for *TC*, *STC* and *PTC* indicators, two poems were chosen for this purpose, namely *I Said To Love* by Thomas Hardy (Text 3) and *The Two Nests* (Text 4) by Dora Sigerson. Both poems can also be found in the appendix of the manual.

All the indicators used in QUITA are influenced by text length. Although some of them reduce this impact, there is no indicator without any text length dependence. The results obtained express the magnitude of an indicator only when you take into account the text size. Thus, only texts with the same lengths can be compared. The simplest solution to this problem is to reduce texts to the first *n* words, for example, the first 1000 words. For this purpose, QUITA enables an option to reduce texts. On the other hand, this method is problematic from a linguistic viewpoint. Quantitative linguistics has been struggling with this issue for long time and there is still no final solution. To show how much indicators are influenced by text size, each one is accompanied by a graph with the results of a quite big corpus, namely 658 texts from various genres written by Czech writer Karel Čapek.

It is important to mention that everyone has to decide which units will be used in a computation. The most usual options are word-forms and lemmas (about other options see 7.2 Tokenizer). It cannot be said which one is better because it depends on the aim of your research, the language, the text size, etc. In this manual, only word-forms are considered as the basic unit in all the examples.

6.1. Frequency Structure Indicators

6.1.1. Type-Token Ratio (*TTR*)

The type-token ratio is a basic indicator of vocabulary richness. It is based on the simple ratio between the number of types and the number of tokens in a text. In general, type-token distinguishes a concept (or general class of things) from its particular instance. The difference can be seen in the following sequence of letters: *a b c d e a d d*. There are five types (a, b, c, d, e) and eight tokens. The type "*a*" has two tokens and the type "*d*" has three tokens, other types have just one token. This distinction can be applied at many levels of language (letters, phonemes, words, sentences, etc.). The resulting value of TTR shows how much the vocabulary varies (the more vocabulary variation in a text, the higher the TTR). The formula of TTR is as follows:

$$(6.1)\, TTR = \frac{V}{N}$$

V...number of types
N...number of tokens

As can be seen in figure 6.1 *TTR* is extremely influenced by text size. Another disadvantage is the fact that no statistical test is defined for a comparison between two texts. To summarize, the simple type-token ratio is the easiest way to measure vocabulary richness but it has several weaknesses at the same time.

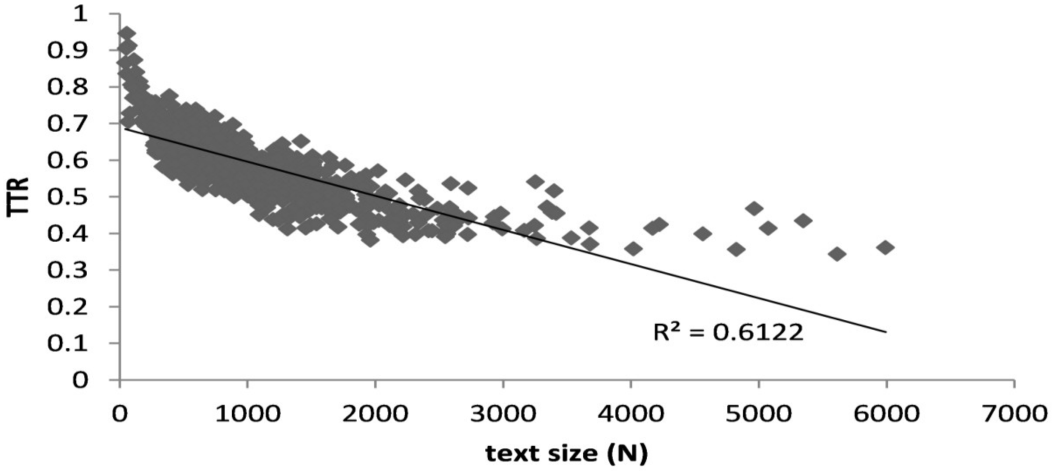


Figure 6.1. Text size impact on TTR in 658 Czech texts

Example

Text 1 has 179 tokens (*V*) and 119 types (*N*), Text 2 has 202 tokens and 121 types. The calculation using formula (6.1) is as follows:

$$TTR_{Text1} = \frac{V}{N} = \frac{119}{179} = 0.665$$

$$TTR_{Text2} = \frac{V}{N} = \frac{121}{202} = 0.59$$

Although, according to the results, Text 1 has richer vocabulary then Text 2, we are not able to decide whether the difference is significant due to the absence of a statistical test.

6.1.2. *h*-point (*h*)

The *h*-point was originally proposed by J. E. Hirsch for scientometrics (2005), and introduced into text analysis by Popescu (2009a). This point divides vocabulary into two groups (synsemantics and autosemantics). The *h*-point is a fuzzy boundary point on the curve where the rank is the same as the frequency ($r=f(r)$). There are various applications of the *h*-point in quantitative linguistics, especially those related to thematic concentration or vocabulary richness indicators.

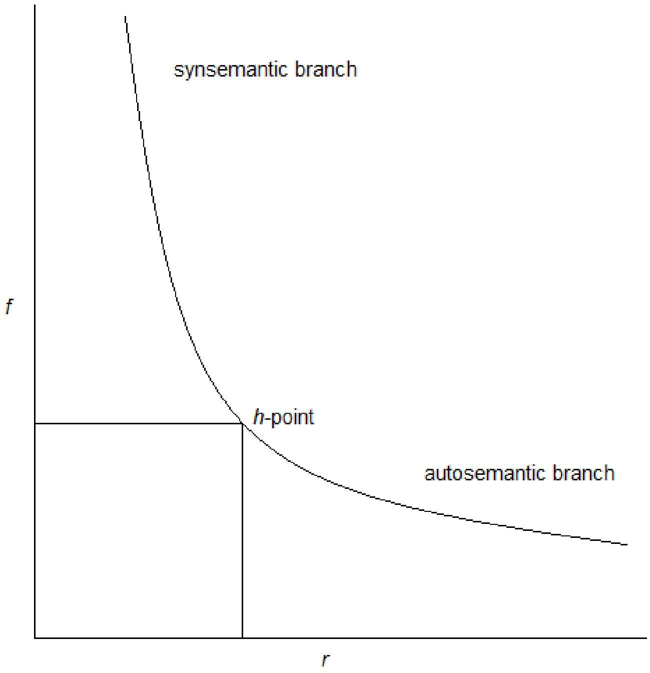


Figure 6.2. *h*-point in rank-frequency distribution

If the rank is not the same as the frequency, we can use the formula (6.2) to gain the h-point:

$$(6.2)\,h=\frac{f(r_1)(r_2-r_1)-[f(r_2)-f(r_1)]r_1}{r_2-r_1-[f(r_2)-f(r_1)]}=\frac{f(r_1)r_2-f(r_2)r_1}{r_2-r_1+f(r_1)-f(r_2)}$$

r…rank

f(r)…frequency of the rank

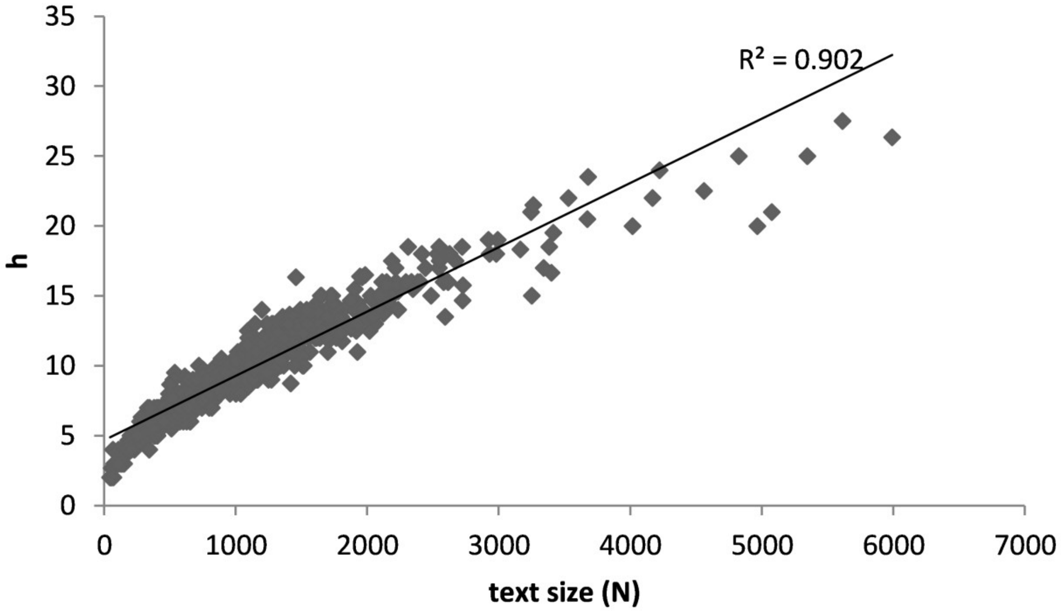


Figure 6.3. Text size impact on the *h*-point in 658 Czech texts

Example

The calculation of *h*-point is very easy in Text 1 because rank 5 has the same frequency 5, so the *h*-point is also 5.

Table 6.1
First 10 tokens in Text 1

Text 1		
Token	Rank	Frequenc y
the	1	16
it	2	7
was	3	7
of	4	7
and	5	5
a	6	5
for	7	3
you	8	3
at	9	3
face	10	3

The situation is substantially different in Text 2 because there is no rank with the same frequency.

Table 6.2
First 10 tokens in Text 2

Text 2		
Token	Rank	Frequency
the	1	20
to	2	9
was	3	8
had	4	7
he	5	4
as	6	4
a	7	4
in	8	4
of	9	4
jones	10	3

The computation of *h*-point using formula (6.2) is as follows.

$$h=\frac{f(r_1)r_2-f(r_2)r_1}{r_2-r_1+f(r_1)-f(r_2)}=\frac{7\cdot 5-4\cdot 4}{5-4+7-4}=4.75$$

6.1.3. *R*₁

R₁ is an indicator of vocabulary richness which is based on the *h*-point (*h*) (see 6.1.2 h-point (h)). This indicator reduces the impact of text length. The formula for calculating vocabulary richness indicator *R₁* is as follows:

$$(6.3)\, R_1=1-\left(F(h)-\frac{h^2}{2N}\right)=1-\left(\frac{\sum_{r=1}^{h} f_i}{N}-\frac{h^2}{2N}\right)$$

F(*h*)…cumulative relative frequency up to the *h*-point, i.e. it represents the *h*-coverage
h…*h*-point (see 6.1.2 h-point (h)).

The variance is defined as:

$$(6.4)\ Var(R_1) = \frac{F(h)[1 - F(h)]}{N}$$

The asymptotic *u*-test can be used for comparing two resulting values.

$$(6.5)\ u = \frac{\left|R_{1(1)} - R_{1(2)}\right|}{\sqrt{Var\,\text{¿¿¿}}}$$

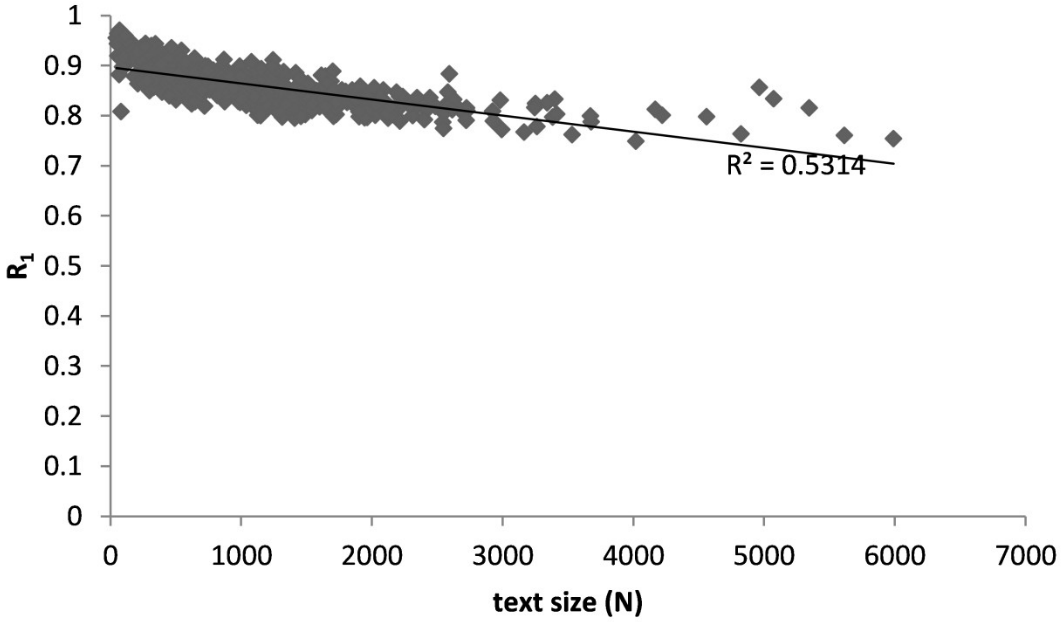


Figure 6.4. Text size impact on R_1 in 658 Czech texts

Example

To calculate R_1 it is necessary to know three variables, namely *N*, *h* and *F*(*h*). The text lengths (N_1=179, N_2=202) and *h*-points (h_1=5, h_2=4.75) are known from the example above; frequencies *F*(*h*) can be found in Table 6.3.

Table 6.3
The frequency lists of Text 1 and Text 2

Text 1			Text 2		
Token	Rank	Frequency	Token	Rank	Frequency
the	1	16	the	1	20
it	2	7	to	2	9
was	3	7	was	3	8

of	4	7	had	4	7
and	5	5	he	5	4
a	6	5	as	6	4
for	7	3	a	7	4
you	8	3	in	8	4
at	9	3	of	9	4
face	10	3	jones	10	3

$$R_{1(Text\,1)}=1-\left(\frac{\sum_{r=1}^{h} f_i}{N}-\frac{h^2}{2N}\right)=1-\left(\frac{16+7+7+7+5}{179}-\frac{5^2}{2\cdot 179}\right)=0.8352$$

$$R_{1(Text\,2)}=1-\left(\frac{\sum_{r=1}^{h} f_i}{N}-\frac{h^2}{2N}\right)=1-\left(\frac{20+9+8+7}{202}-\frac{4.75^2}{2\cdot 202}\right)=0.838$$

According to the obtained results Text 2 has higher vocabulary richness, but it must be discovered whether the difference (0.003) is significant. For this purpose we can use formula (6.5) but first, it is necessary to calculate the variances by formula (6.4).

$$Var\left(R_{1(1)}\right)=\frac{F(h)[1-F(h)]}{N}=\frac{0.2346\cdot(1-0.2346)}{179}=0.00100314$$

$$Var\left(R_{1(2)}\right)=\frac{F(h)[1-F(h)]}{N}=\frac{0.2178\cdot(1-0.2178)}{202}=0.00084338$$

All the relevant variables for the statistical test are known now, so we can use formula (6.5) to find out whether the two texts are significantly different.

$$u=\frac{\left|R_{1(1)}-R_{1(2)}\right|}{\sqrt{Var\,\text{¿¿¿}}}$$

Since the result 0.065 is lower than the threshold 1.96, it can be state that there is no significant difference. The threshold 1.96 is defined for the significance level 0.05.

6.1.4. Repeat Rate (*RR*)

The repeat rate shows the degree of vocabulary concentration in a text. In other words, this indicator measures vocabulary richness inversely: the higher *RR* is, the less vocabulary diversity a text has. Resulting values of *RR* are in the interval <1/*V*; 1>. The repeat rate is defined as:

$$(6.6)\, RR = \sum_{r=1}^{V} p_i^2$$

V…number of types
P$_i$…individual probabilities, we estimate them by means of relative frequencies as follows:

$$(6.7)\, p_i = \frac{f_1}{N}$$

f$_I$…absolute frequencies
N…number of tokens

The aforementioned formulas can be transformed into one:

$$(6.8)\, RR = \frac{1}{N^2} \sum_{r=1}^{V} f_i^2$$

A formula to obtain variances is defined as:

$$(6.9)\, Var(RR) = \frac{4}{N} \left(\sum_{r=1}^{V} p_r^3 - RR^2 \right)$$

The asymptotic *u*-test can be used for comparing two resulting values:

$$(6.10)\, u = \frac{|RR_1| - |RR_2|}{\sqrt{var(RR_1) + var(RR_2)}}$$

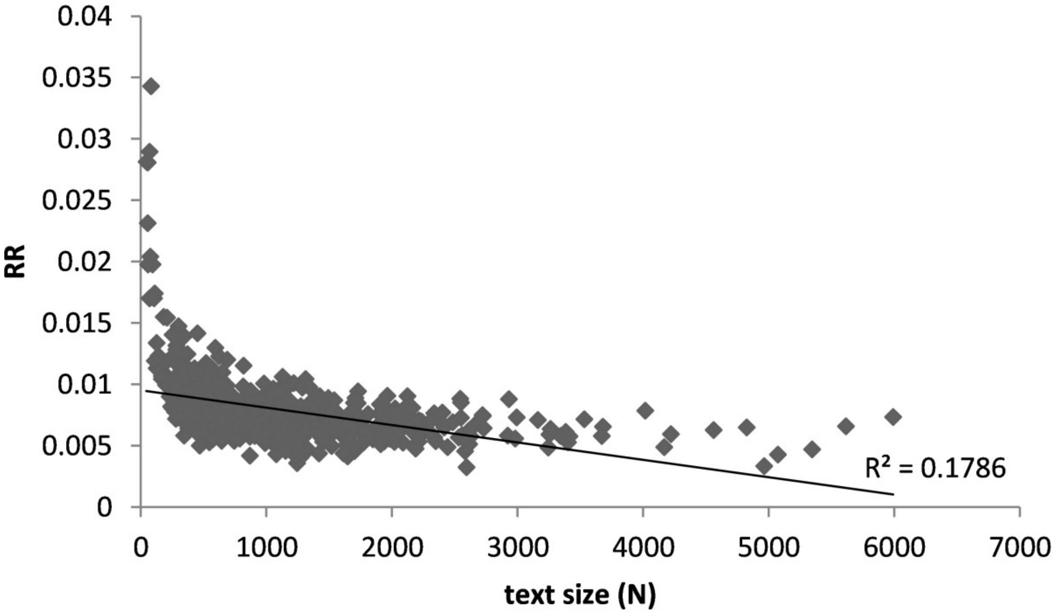


Figure 6.5. Text size impact on *RR* in 658 Czech texts

Example

First, we use formula (6.8) to obtain *RR* values.

$$RR_{Text\,1}=\frac{1}{N^2}\sum_{r=1}^{V} f_i^2=\frac{16^2+7^2+7^2+7^2+5^2+5^2+3^2+\ldots+1^2}{179^2}=0.0197$$

$$RR_{Text\,2}=\frac{1}{N^2}\sum_{r=1}^{V} f_i^2=\frac{20^2+9^2+8^2+7^2+4^2+4^2+4^2+\ldots+1^2}{202^2}=0.02147$$

Second, we gain variances using formula (6.9).

$$Var\left(RR_1\right)=\frac{4}{N}\left(\sum_{r=1}^{V} p_r^3-RR^2\right)$$

$$\text{¿}\,\frac{4}{179}\left(\frac{16^3+7^3+7^3+7^3+5^3+5^3+3^3+\ldots+1^3}{179^3}-0.0197^2\right)=0.00001341$$

$$Var\left(RR_2\right)=\frac{4}{N}\left(\sum_{r=1}^{V} p_r^3-RR^2\right)$$

$$\text{¿}\,\frac{4}{202}\left(\frac{20^3+9^3+8^3+7^3+4^3+4^3+4^3+\ldots+1^3}{202^3}-0.02147^2\right)=0.00001555$$

The last step is to discover whether the *RR* resulting values of the two texts differ significantly using formula (6.10).

$$u=\frac{|RR_1|-|RR_2|}{\sqrt{var\left(RR_1\right)+var\left(RR_2\right)}}=\frac{|0.0197|-|0.02147|}{\sqrt{0.00001341+0.00001555}}=0.32$$

Since the threshold value is 1.96, u<1.96 means that the difference between the two texts is not significant.

6.1.5. Relative Repeat Rate of McIntosh (*RR*$_{mc}$)

In order to compare the repeat rate with other indicators, the relative repeat rate *RR*$_{mc}$ was proposed by McIntosh. *RR*$_{mc}$ puts the results in the interval <0;1>. The formula is as follows:

$$\left(6.11\right)RR_{mc}=\frac{1-\sqrt{RR}}{1-1/\sqrt{V}}$$

RR…Repeat Rate (see 6.1.4 Repeat Rate (RR))
V…number of types

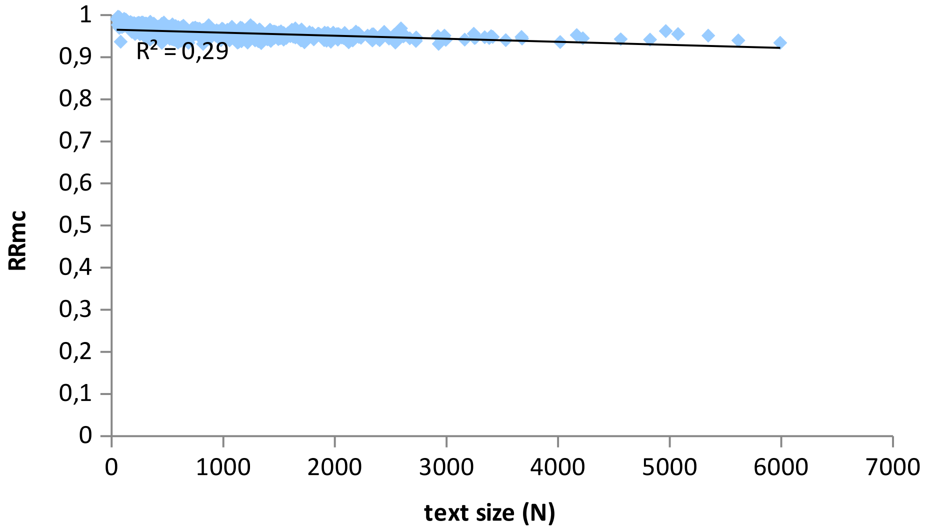


Figure 6.6. Text size impact on *RR*$_{mc}$ in 658 Czech texts

Example

$$RR_{mc(Text\,1)} = \frac{1-\sqrt{RR}}{1-1/\sqrt{V}} = \frac{1-\sqrt{0.0197}}{1-1/\sqrt{119}} = 0.946$$

$$RR_{mc(Text\,2)} = \frac{1-\sqrt{RR}}{1-1/\sqrt{V}} = \frac{1-\sqrt{0.02147}}{1-1/\sqrt{121}} = 0.939$$

6.1.6. Hapax Legomena Percentage (*HL*)

Hapax Legomena Percentage (*HL*) is a simple ratio between the number of tokens (N) and number of hapax legomena (N_h) in a text. The hapax legomena (sg. hapax legomenon) are the words that occur only once in a single text. The formula is as follows:

$$(6.12)\, HL = \frac{N_h}{N}$$

N…number of tokens
N_h…number of hapax legomena

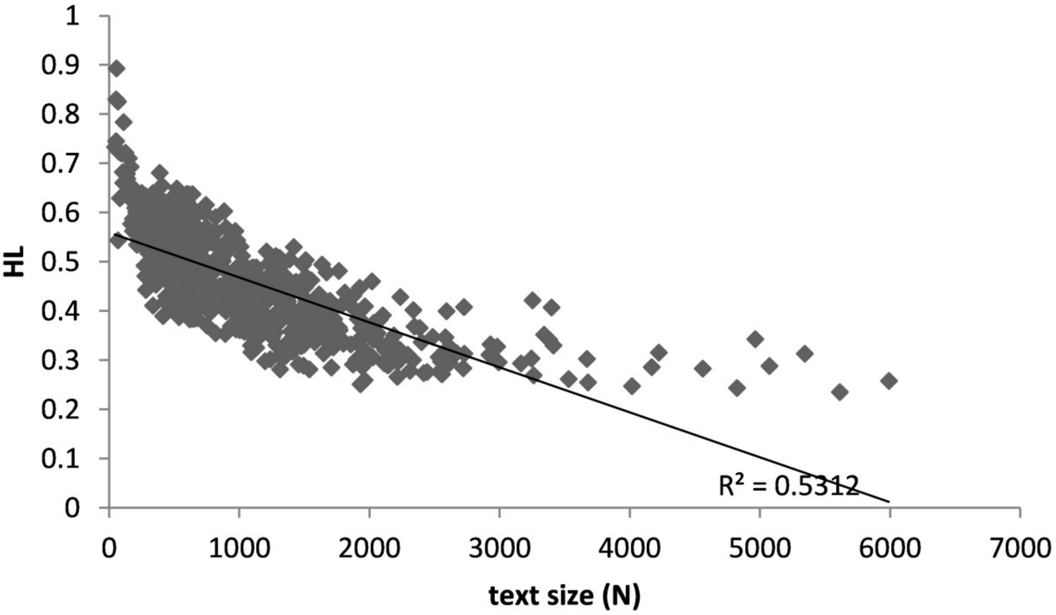


Figure 6.7. Text size impact on *HL* in 658 Czech texts

Example

$$HL_1=\frac{N_h}{N}=\frac{98}{179}=0.547$$

$$HL_2=\frac{N_h}{N}=\frac{92}{202}=0.455$$

6.1.7. Lambda (Λ)

The lambda (Λ) is an indicator which deals with a frequency structure of text. On the one hand, the lambda is related to vocabulary richness, and on the other hand, it takes into account the relationship between neighbouring frequencies. For example, we have three texts with the same length (*N*=20) and the same vocabulary size (*V*=10). So, the type-token ratio (*TTR*) (see 6.1.1 Type-Token Ratio (TTR)) is also the same (10/20=0.5). If we look at Table 6.4 and Figure 6.8, we can see that three hypothetical texts with the same *N* and *V* can have different relations between frequencies.

Table 6.4
The rank-frequency structures of three hypothetical texts

text 1		**text 2**		**text 3**	
rank	**frequency**	**rank**	**frequency**	**rank**	**frequency**
1	5	1	6	1	2
2	4	2	2	2	2
3	3	3	2	3	2
4	2	4	2	4	2
5	1	5	2	5	2
6	1	6	2	6	2
7	1	7	1	7	2
8	1	8	1	8	2
9	1	9	1	9	2
10	1	10	1	10	2

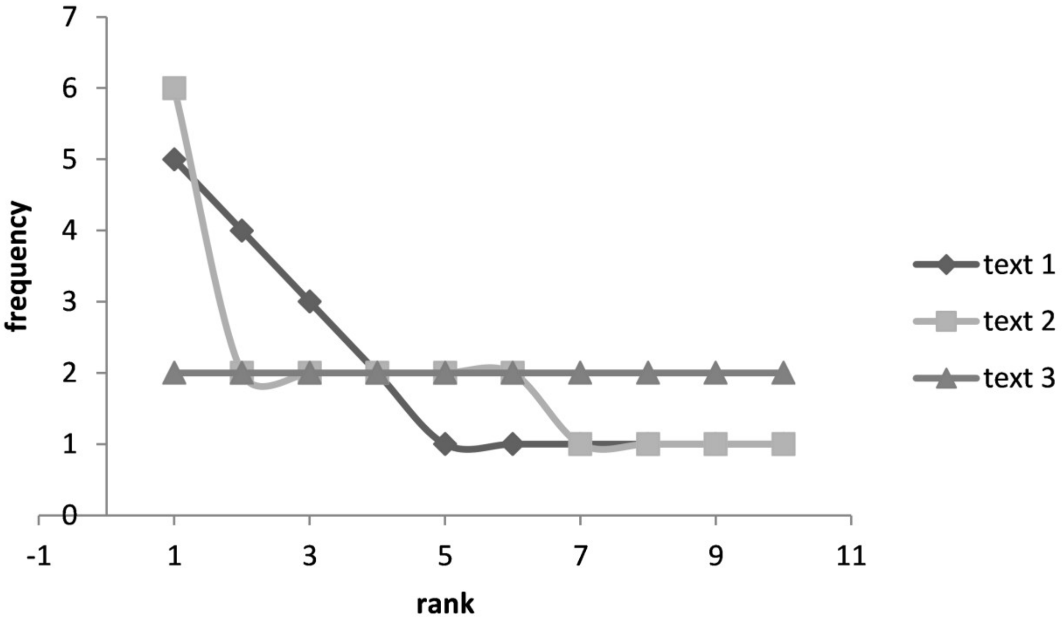


Figure 6.8. The rank-frequency structures of three hypothetical texts

From the example, it is obvious that the specific development of frequency structure is an important phenomenon which expresses an individual's way of writing. This indicator, therefore, is suitable for analyses of authorship, genres, etc. The formula for the computation is defined as:

$$(6.13)\ \Lambda = \frac{L\left(\log_{10} N\right)}{N}$$

N...length of the text (in tokens)
L...arc length of the rank-frequency distribution

The formula for L is as follows:

$$(6.14)\ L = \sum_{i=1}^{V-1} \left[\left(f_i - f_{i+1}\right)^2 + 1\right]^{1/2}$$

f_i…absolute frequencies
V...the highest rank.

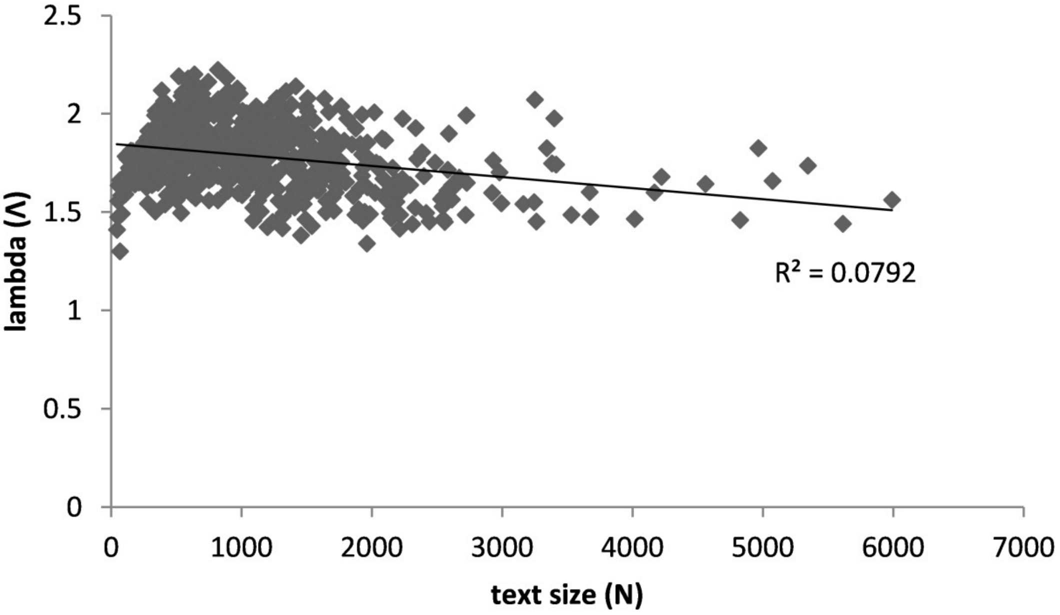


Figure 6.9. Text size impact on Lambda in 658 Czech texts

Example

Before computing lambda, it is necessary to obtain *L* using formula (6.14). The process is displayed in Table 6.5 and Table 6.6.

Table 6.5
Computation *L* in Text 1

word-form	***i***	***f***	$\left[\left(f_i - f_{i+1}\right)^2 + 1\right]^{1/2}$	**result**
the	1	16	$[(16-7)^2+1]^{1/2}$	9.055385138
it	2	7	$[(7-7)^2+1]^{1/2}$	1
was	3	7	$[(7-7)^2+1]^{1/2}$	1
of	4	7	$[(7-5)^2+1]^{1/2}$	2.236067977
and	5	5	$[(5-5)^2+1]^{1/2}$	1
a	6	5	$[(5-3)^2+1]^{1/2}$	2.236067977
for	7	3	$[(3-3)^2+1]^{1/2}$	1
you	8	3	$[(3-3)^2+1]^{1/2}$	1
at	9	3	$[(3-3)^2+1]^{1/2}$	1
face	10	3	$[(3-2)^2+1]^{1/2}$	1.414213562
times	11	2	$[(2-2)^2+1]^{1/2}$	1
had	12	2	$[(2-2)^2+1]^{1/2}$	1
enormous	13	2	$[(2-2)^2+1]^{1/2}$	1

on	14	2	$[(2-2)^2+1]^{1/2}$	1
with	15	2	$[(2-2)^2+1]^{1/2}$	1
wall	16	2	$[(2-2)^2+1]^{1/2}$	1
winston	17	2	$[(2-2)^2+1]^{1/2}$	1
about	18	2	$[(2-2)^2+1]^{1/2}$	1
lift	19	2	$[(2-2)^2+1]^{1/2}$	1
poster	20	2	$[(2-2)^2+1]^{1/2}$	1
one	21	2	$[(2-1)^2+1]^{1/2}$	1.414213562
ankle	22	1	$[(1-1)^2+1]^{1/2}$	1
hate	23	1	$[(1-1)^2+1]^{1/2}$	1
preparation	24	1	$[(1-1)^2+1]^{1/2}$	1
his	25	1	$[(1-1)^2+1]^{1/2}$	1
right	26	1	$[(1-1)^2+1]^{1/2}$	1
drive	27	1	$[(1-1)^2+1]^{1/2}$	1
economy	28	1	$[(1-1)^2+1]^{1/2}$	1
resting	29	1	$[(1-1)^2+1]^{1/2}$	1
in	30	1	$[(1-1)^2+1]^{1/2}$	1
went	31	1	$[(1-1)^2+1]^{1/2}$	1
slowly	32	1	$[(1-1)^2+1]^{1/2}$	1
thirty	33	1	$[(1-1)^2+1]^{1/2}$	1
nine	34	1	$[(1-1)^2+1]^{1/2}$	1
flights	35	1	$[(1-1)^2+1]^{1/2}$	1
up	36	1	$[(1-1)^2+1]^{1/2}$	1
who	37	1	$[(1-1)^2+1]^{1/2}$	1
seven	38	1	$[(1-1)^2+1]^{1/2}$	1
flat	39	1	$[(1-1)^2+1]^{1/2}$	1
week	40	1	$[(1-1)^2+1]^{1/2}$	1
above	41	1	$[(1-1)^2+1]^{1/2}$	1
varicose	42	1	$[(1-1)^2+1]^{1/2}$	1
ulcer	43	1	$[(1-1)^2+1]^{1/2}$	1
follow	44	1	$[(1-1)^2+1]^{1/2}$	1
when	45	1	$[(1-1)^2+1]^{1/2}$	1
move	46	1	$[(1-1)^2+1]^{1/2}$	1
contrived	47	1	$[(1-1)^2+1]^{1/2}$	1
that	48	1	$[(1-1)^2+1]^{1/2}$	1
eyes	49	1	$[(1-1)^2+1]^{1/2}$	1
big	50	1	$[(1-1)^2+1]^{1/2}$	1
caption	51	1	$[(1-1)^2+1]^{1/2}$	1
beneath	52	1	$[(1-1)^2+1]^{1/2}$	1
ran	53	1	$[(1-1)^2+1]^{1/2}$	1
brother	54	1	$[(1-1)^2+1]^{1/2}$	1
is	55	1	$[(1-1)^2+1]^{1/2}$	1
watching	56	1	$[(1-1)^2+1]^{1/2}$	1

landing	57	1	$[(1-1)^2+1]^{1/2}$	1
opposite	58	1	$[(1-1)^2+1]^{1/2}$	1
shaft	59	1	$[(1-1)^2+1]^{1/2}$	1
several	60	1	$[(1-1)^2+1]^{1/2}$	1
way	61	1	$[(1-1)^2+1]^{1/2}$	1
each	62	1	$[(1-1)^2+1]^{1/2}$	1
gazed	63	1	$[(1-1)^2+1]^{1/2}$	1
which	64	1	$[(1-1)^2+1]^{1/2}$	1
are	65	1	$[(1-1)^2+1]^{1/2}$	1
so	66	1	$[(1-1)^2+1]^{1/2}$	1
from	67	1	$[(1-1)^2+1]^{1/2}$	1
those	68	1	$[(1-1)^2+1]^{1/2}$	1
pictures	69	1	$[(1-1)^2+1]^{1/2}$	1
depicted	70	1	$[(1-1)^2+1]^{1/2}$	1
simply	71	1	$[(1-1)^2+1]^{1/2}$	1
an	72	1	$[(1-1)^2+1]^{1/2}$	1
been	73	1	$[(1-1)^2+1]^{1/2}$	1
tacked	74	1	$[(1-1)^2+1]^{1/2}$	1
to	75	1	$[(1-1)^2+1]^{1/2}$	1
wide	76	1	$[(1-1)^2+1]^{1/2}$	1
man	77	1	$[(1-1)^2+1]^{1/2}$	1
forty	78	1	$[(1-1)^2+1]^{1/2}$	1
more	79	1	$[(1-1)^2+1]^{1/2}$	1
than	80	1	$[(1-1)^2+1]^{1/2}$	1
metre	81	1	$[(1-1)^2+1]^{1/2}$	1
display	82	1	$[(1-1)^2+1]^{1/2}$	1
cabbage	83	1	$[(1-1)^2+1]^{1/2}$	1
old	84	1	$[(1-1)^2+1]^{1/2}$	1
rag	85	1	$[(1-1)^2+1]^{1/2}$	1
hallway	86	1	$[(1-1)^2+1]^{1/2}$	1
smelt	87	1	$[(1-1)^2+1]^{1/2}$	1
boiled	88	1	$[(1-1)^2+1]^{1/2}$	1
too	89	1	$[(1-1)^2+1]^{1/2}$	1
large	90	1	$[(1-1)^2+1]^{1/2}$	1
indoor	91	1	$[(1-1)^2+1]^{1/2}$	1
mats	92	1	$[(1-1)^2+1]^{1/2}$	1
end	93	1	$[(1-1)^2+1]^{1/2}$	1
coloured	94	1	$[(1-1)^2+1]^{1/2}$	1
present	95	1	$[(1-1)^2+1]^{1/2}$	1
electric	96	1	$[(1-1)^2+1]^{1/2}$	1
current	97	1	$[(1-1)^2+1]^{1/2}$	1
best	98	1	$[(1-1)^2+1]^{1/2}$	1
seldom	99	1	$[(1-1)^2+1]^{1/2}$	1

working	100	1	$[(1-1)^2+1]^{1/2}$	1
daylight	101	1	$[(1-1)^2+1]^{1/2}$	1
hours	102	1	$[(1-1)^2+1]^{1/2}$	1
part	103	1	$[(1-1)^2+1]^{1/2}$	1
cut	104	1	$[(1-1)^2+1]^{1/2}$	1
off	105	1	$[(1-1)^2+1]^{1/2}$	1
during	106	1	$[(1-1)^2+1]^{1/2}$	1
even	107	1	$[(1-1)^2+1]^{1/2}$	1
moustache	108	1	$[(1-1)^2+1]^{1/2}$	1
ruggedly	109	1	$[(1-1)^2+1]^{1/2}$	1
handsome	110	1	$[(1-1)^2+1]^{1/2}$	1
five	111	1	$[(1-1)^2+1]^{1/2}$	1
heavy	112	1	$[(1-1)^2+1]^{1/2}$	1
black	113	1	$[(1-1)^2+1]^{1/2}$	1
no	114	1	$[(1-1)^2+1]^{1/2}$	1
use	115	1	$[(1-1)^2+1]^{1/2}$	1
trying	116	1	$[(1-1)^2+1]^{1/2}$	1
features	117	1	$[(1-1)^2+1]^{1/2}$	1
made	118	1	$[(1-1)^2+1]^{1/2}$	1
stairs	119	1		
L_1= 129.3559482				

Table 6.6
Computation *L* in Text 2

word-form	***i***	***f***	$[(f_i - f_{i+1})^2+1]^{1/2}$	**result**
the	1	2	$[(20-9)^2+1]^{1/2}$	11.04536102
to	2	9	$[(9-8)^2+1]^{1/2}$	1.414213562
was	3	8	$[(8-7)^2+1]^{1/2}$	1.
had	4	7	$[(7-4)^2+1]^{1/2}$	3.16227766
he	5	4	$[(4-4)^2+1]^{1/2}$	1
as	6	4	$[(4-4)^2+1]^{1/2}$	1
a	7	4	$[(4-4)^2+1]^{1/2}$	1
in	8	4	$[(4-4)^2+1]^{1/2}$	1
of	9	4	$[(4-3)^2+1]^{1/2}$	1.414213562
jones	10	3	$[(3-3)^2+1]^{1/2}$	1
and	11	3	$[(3-3)^2+1]^{1/2}$	1
from	12	3	$[(3-3)^2+1]^{1/2}$	1
his	13	3	$[(3-3)^2+1]^{1/2}$	1
farm	14	3	$[(3-3)^2+1]^{1/2}$	1
that	15	3	$[(3-2)^2+1]^{1/2}$	1.414213562

been	16	2	$[(2-2)^2+1]^{1/2}$	1
light	17	2	$[(2-2)^2+1]^{1/2}$	1
it	18	2	$[(2-2)^2+1]^{1/2}$	1
so	19	2	$[(2-2)^2+1]^{1/2}$	1
side	20	2	$[(2-2)^2+1]^{1/2}$	1
way	21	2	$[(2-2)^2+1]^{1/2}$	1
on	22	2	$[(2-2)^2+1]^{1/2}$	1
night	23	2	$[(2-2)^2+1]^{1/2}$	1
all	24	2	$[(2-2)^2+1]^{1/2}$	1
major	25	2	$[(2-2)^2+1]^{1/2}$	1
mr	26	2	$[(2-2)^2+1]^{1/2}$	1
old	27	2	$[(2-2)^2+1]^{1/2}$	1
soon	28	2	$[(2-2)^2+1]^{1/2}$	1
out	29	2	$[(2-1)^2+1]^{1/2}$	1.414213562
dream	30	1	$[(1-1)^2+1]^{1/2}$	1
strange	31	1	$[(1-1)^2+1]^{1/2}$	1
prize	32	1	$[(1-1)^2+1]^{1/2}$	1
they	33	1	$[(1-1)^2+1]^{1/2}$	1
agreed	34	1	$[(1-1)^2+1]^{1/2}$	1
day	35	1	$[(1-1)^2+1]^{1/2}$	1
animals	36	1	$[(1-1)^2+1]^{1/2}$	1
wished	37	1	$[(1-1)^2+1]^{1/2}$	1
middle	38	1	$[(1-1)^2+1]^{1/2}$	1
white	39	1	$[(1-1)^2+1]^{1/2}$	1
communicat	40	1	$[(1-1)^2+1]^{1/2}$	1
previous	41	1	$[(1-1)^2+1]^{1/2}$	1
other	42	1	$[(1-1)^2+1]^{1/2}$	1
boar	43	1	$[(1-1)^2+1]^{1/2}$	1
should	44	1	$[(1-1)^2+1]^{1/2}$	1
ready	45	1	$[(1-1)^2+1]^{1/2}$	1
lose	46	1	$[(1-1)^2+1]^{1/2}$	1
an	47	1	$[(1-1)^2+1]^{1/2}$	1
regarded	48	1	$[(1-1)^2+1]^{1/2}$	1
everyone	49	1	$[(1-1)^2+1]^{1/2}$	1
quite	50	1	$[(1-1)^2+1]^{1/2}$	1
hour	51	1	$[(1-1)^2+1]^{1/2}$	1
hear	52	1	$[(1-1)^2+1]^{1/2}$	1
what	53	1	$[(1-1)^2+1]^{1/2}$	1
say	54	1	$[(1-1)^2+1]^{1/2}$	1

s	55	1	$[(1-1)^2+1]^{1/2}$	1
sleep	56	1	$[(1-1)^2+1]^{1/2}$	1
order	57	1	$[(1-1)^2+1]^{1/2}$	1
highly	58	1	$[(1-1)^2+1]^{1/2}$	1
safely	59	1	$[(1-1)^2+1]^{1/2}$	1
always	60	1	$[(1-1)^2+1]^{1/2}$	1
called	61	1	$[(1-1)^2+1]^{1/2}$	1
meet	62	1	$[(1-1)^2+1]^{1/2}$	1
big	63	1	$[(1-1)^2+1]^{1/2}$	1
barn	64	1	$[(1-1)^2+1]^{1/2}$	1
though	65	1	$[(1-1)^2+1]^{1/2}$	1
exhibited	66	1	$[(1-1)^2+1]^{1/2}$	1
willingdon	67	1	$[(1-1)^2+1]^{1/2}$	1
beauty	68	1	$[(1-1)^2+1]^{1/2}$	1
name	69	1	$[(1-1)^2+1]^{1/2}$	1
under	70	1	$[(1-1)^2+1]^{1/2}$	1
which	71	1	$[(1-1)^2+1]^{1/2}$	1
during	72	1	$[(1-1)^2+1]^{1/2}$	1
dancing	73	1	$[(1-1)^2+1]^{1/2}$	1
lurched	74	1	$[(1-1)^2+1]^{1/2}$	1
across	75	1	$[(1-1)^2+1]^{1/2}$	1
with	76	1	$[(1-1)^2+1]^{1/2}$	1
ring	77	1	$[(1-1)^2+1]^{1/2}$	1
lantern	78	1	$[(1-1)^2+1]^{1/2}$	1
boots	79	1	$[(1-1)^2+1]^{1/2}$	1
at	80	1	$[(1-1)^2+1]^{1/2}$	1
back	81	1	$[(1-1)^2+1]^{1/2}$	1
yard	82	1	$[(1-1)^2+1]^{1/2}$	1
kicked	83	1	$[(1-1)^2+1]^{1/2}$	1
off	84	1	$[(1-1)^2+1]^{1/2}$	1
houses	85	1	$[(1-1)^2+1]^{1/2}$	1
for	86	1	$[(1-1)^2+1]^{1/2}$	1
but	87	1	$[(1-1)^2+1]^{1/2}$	1
manor	88	1	$[(1-1)^2+1]^{1/2}$	1
locked	89	1	$[(1-1)^2+1]^{1/2}$	1
hen	90	1	$[(1-1)^2+1]^{1/2}$	1
shut	91	1	$[(1-1)^2+1]^{1/2}$	1
pop	92	1	$[(1-1)^2+1]^{1/2}$	1
holes	93	1	$[(1-1)^2+1]^{1/2}$	1

too	94	1	$[(1-1)^2+1]^{1/2}$	1
drunk	95	1	$[(1-1)^2+1]^{1/2}$	1
remember	96	1	$[(1-1)^2+1]^{1/2}$	1
door	97	1	$[(1-1)^2+1]^{1/2}$	1
went	98	1	$[(1-1)^2+1]^{1/2}$	1
there	99	1	$[(1-1)^2+1]^{1/2}$	1
stirring	10	1	$[(1-1)^2+1]^{1/2}$	1
already	10	1	$[(1-1)^2+1]^{1/2}$	1
snoring	10	1	$[(1-1)^2+1]^{1/2}$	1
bedroom	10	1	$[(1-1)^2+1]^{1/2}$	1
word	10	1	$[(1-1)^2+1]^{1/2}$	1
gone	10	1	$[(1-1)^2+1]^{1/2}$	1
round	10	1	$[(1-1)^2+1]^{1/2}$	1
fluttering	10	1	$[(1-1)^2+1]^{1/2}$	1
through	10	1	$[(1-1)^2+1]^{1/2}$	1
buildings	10	1	$[(1-1)^2+1]^{1/2}$	1
glass	110	1	$[(1-1)^2+1]^{1/2}$	1
beer	111	1	$[(1-1)^2+1]^{1/2}$	1
barrel	112	1	$[(1-1)^2+1]^{1/2}$	1
drew	113	1	$[(1-1)^2+1]^{1/2}$	1
himself	114	1	$[(1-1)^2+1]^{1/2}$	1
last	115	1	$[(1-1)^2+1]^{1/2}$	1
bed	116	1	$[(1-1)^2+1]^{1/2}$	1
where	117	1	$[(1-1)^2+1]^{1/2}$	1
mrs	118	1	$[(1-1)^2+1]^{1/2}$	1
scullery	119	1	$[(1-1)^2+1]^{1/2}$	1
made	12	1	$[(1-1)^2+1]^{1/2}$	1
up	12	1		
L $\boldsymbol{L_2}$**= 134.2787065**				

Once the values of *L* are obtained, we can use them in formula (6.13) to obtain lambda.

$$\Lambda_1 = \frac{L(\log_{10} N)}{N} = \frac{129.3559482(\log_{10} 179)}{179} = 1.628$$

$$\Lambda_2 = \frac{L(\log_{10} N)}{N} = \frac{134.2787065(\log_{10} 202)}{202} = 1.5325$$

6.1.8. Gini Coefficient (*G*)

The Gini coefficient is a well-known measure of statistical dispersion, used especially in economics. In linguistics, *G* is applicable to text analysis as one of many tools for vocabulary richness measurement. The Gini coefficient is based on the Lorenz curve.

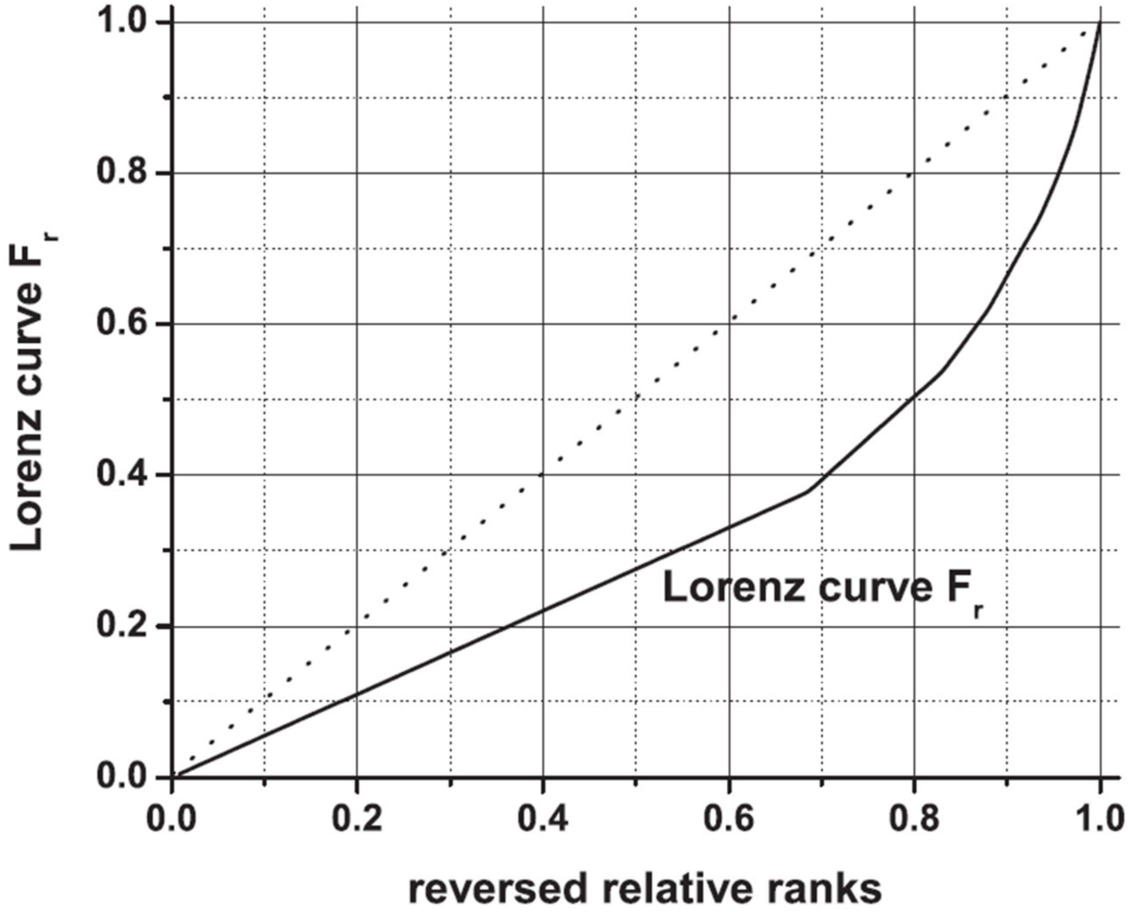


Figure 6.10. Lorenz curve

To create the Lorenz curve, two changes in a frequency distribution of tokens must be made. First, rank must be assigned in a reverse order (i.e. the smallest frequency obtains rank 1, the next equal or greater obtains rank 2, etc. Second, ranks and frequencies must be relativized to put variables in the interval <0; 1> (rank: *rr*=*r*/*V*; frequency: *pr* = *fr*/*N*).

If each word would occur exactly once, the sequence <*rr*,*pr*> would be a straight line practically between [0,0] and [1,1]. In this case, a text has maximal vocabulary richness. The Gini coefficient is the distance between the diagonal and the sequence of cumulative frequencies (the Lorenz curve). *G* is defined as:

$$(6.15) G=\frac{1}{V}\left(V+1-\frac{2}{N}\sum_{r=1}^{V} rf(r)\right)=\frac{1}{V}(V+1-2m_1^{'})$$

$$(6.16) m_1=\frac{\sum_{r=1}^{V} rf(r)}{N}$$

V…number of types
N…number of tokens
r…rank
f(r)…individual frequency
m_1…average frequency distribution

The variance is defined as:

$$(6.17) Var(G)=\frac{4}{V^2}Var\left(m_1^{'}\right)=\frac{4m_2}{V^2N}$$

$$(6.18) m_2=\frac{1}{N}\sum_{i=1}^{V}\left(r_i-m_1^{'}\right)^2 f(r_i)$$

M_2…variance of frequency distribution

For a comparison of two texts we can use the asymptotic *u*-test:

$$(6.19) u=\frac{\left|G_1-G_2\right|}{\sqrt{Var\left(G_1\right)+Var\left(G_2\right)}}$$

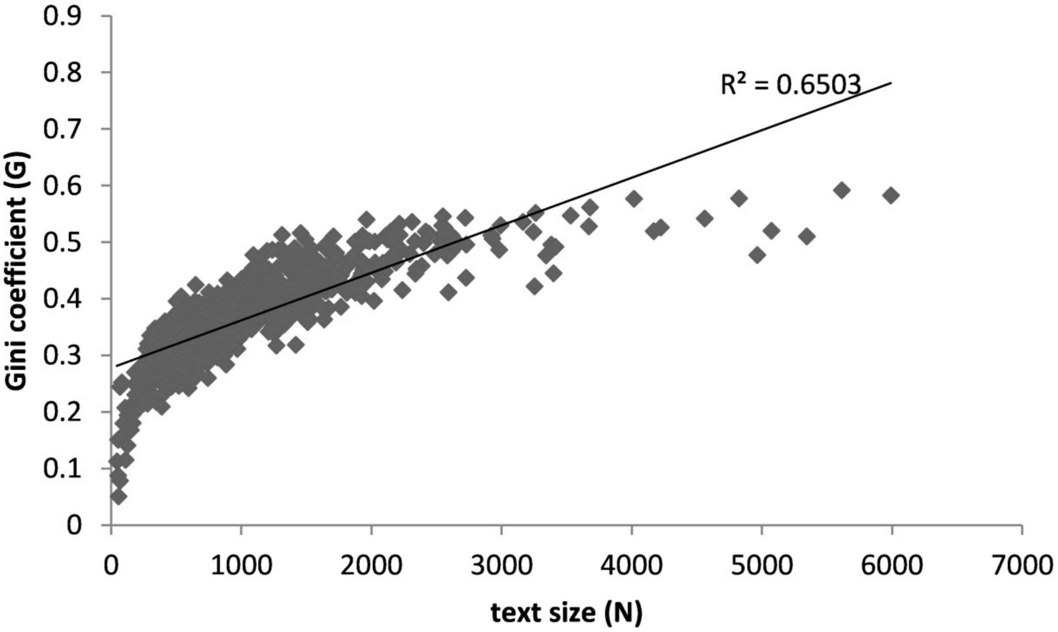


Figure 6.11. Text size impact on *G* in 658 Czech texts

Example

The first step is to obtain m_1 by formula (6.16)

$$m_{1(Text1)} = \frac{\sum_{r=1}^{V} rf(r)}{N} = \frac{16 \cdot 1 + 7 \cdot 2 + 7 \cdot 3 + 7 \cdot 4 + 5 \cdot 5 + \ldots + 1 \cdot 119}{179} = 41.88268156$$

$$m_{1(Text2)} = \frac{\sum_{r=1}^{V} rf(r)}{N} = \frac{20 \cdot 1 + 9 \cdot 2 + 8 \cdot 3 + 7 \cdot 4 + 4 \cdot 5 + \ldots + 1 \cdot 121}{202} = 39.75742574$$

When we have m_1 values, we can compute *G* using formula (6.15)

$$G_{Text1} = \frac{1}{V}\left(V + 1 - 2m_1^{'}\right) = \frac{1}{119}(119 + 1 - 2 \cdot 41.88268156) = 0.3045$$

$$G_{Text2} = \frac{1}{V}\left(V + 1 - 2m_1^{'}\right) = \frac{1}{121}(121 + 1 - 2 \cdot 39.75742574) = 0.3511$$

Although we know the Gini coefficient, we must find out whether the differences between two texts are significant. For this purpose we can use *u*-test (formula 6.19), but it is necessary to know the variances (formula 6.17) and m_2 values (formula 6.18).

$$m_{2(Text1)}=\frac{1}{N}\sum_{i=1}^{V}\left(r_i-m_1'\right)^2 f(r_i)=\frac{(1-41.8827)^2\cdot 16+(2-41.8827)^2\cdot 7+(3-41.8827)^2\cdot 7+\ldots+(119-41.8827)}{179}$$

$$m_{2(Text2)}=\frac{1}{N}\sum_{i=1}^{V}\left(r_i-m_1'\right)^2 f(r_i)=\frac{(1-39.7574)^2\cdot 20+(2-39.7574)^2\cdot 9+(3-39.7574)^2\cdot 8+\ldots+(121-39.7574)}{202}$$

$$Var\left(G_1\right)=\frac{4m_2}{V^2N}=\frac{4\cdot 1445.6564}{119^2\cdot 179}=0.002281277$$

$$Var\left(G_2\right)=\frac{4m_2}{V^2N}=\frac{4\cdot 1429.55}{121^2\cdot 202}=0.001933469$$

$$u=\frac{\left|G_1-G_2\right|}{\sqrt{Var\left(G_1\right)+Var\left(G_2\right)}}=\frac{\left|0.3045-0.3511\right|}{\sqrt{0.002281277+0.001933469}}=0.72$$

Since the threshold is 1.96, U<1.96 means that there is no significant difference between two texts.

6.1.9. Indicator R_4

R_4 is one of many ways of calculating vocabulary richness of a text. In fact, this indicator is the reversed Gini coefficient (*G)* (see 6.1.8 Gini Coefficient (G)). R_4 is therefore defined as:

$$(6.20)\, R_4=1-G$$

$G\ldots$Gini coefficient (see 6.1.8 Gini Coefficient (G))

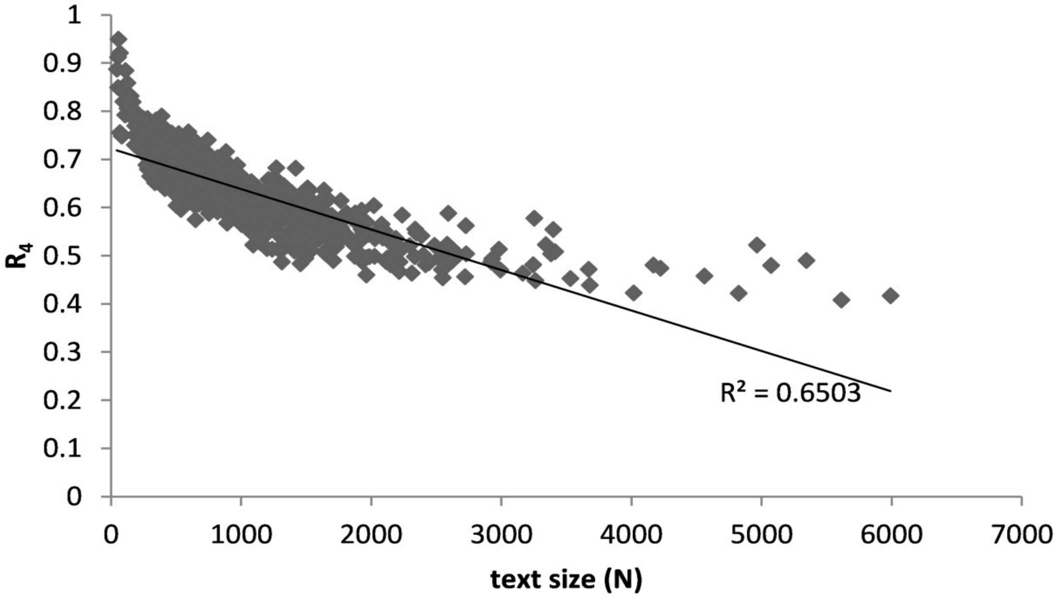


Figure 6.12. Text size impact on R_4 in 658 Czech texts

Example

$R_{4(Text1)} = 1 - G = 1 - 0.3045 = 0.6955$

$$R_{4(Text2)} = 1 - G = 1 - 0.3511 = 0.6489$$

6.1.10. Curve length (*L*)

Many vocabulary richness indicators are based on the curve of rank-frequency distribution, like this:

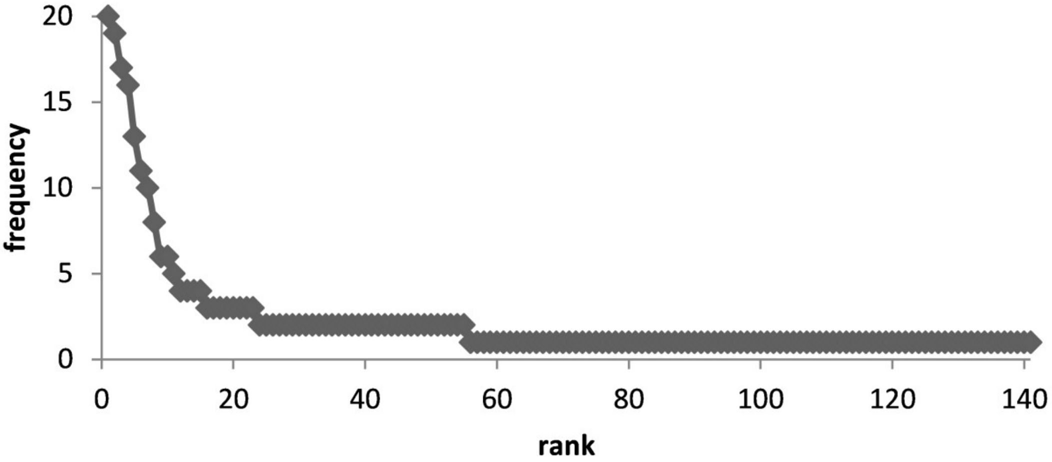


Figure 6.13. Example of typical rank-frequency distribution

The length of the curve is defined as the sum of the Euclidean distances (D_r) between all adjacent points on the curve:

$$(6.21)\, L=\sum_{r=1}^{V-1} D_r=\sum_{r=1}^{V-1} \sqrt{\left(f(r)-f(r+1)\right)^2+1}$$

f…individual frequency
r…individual rank

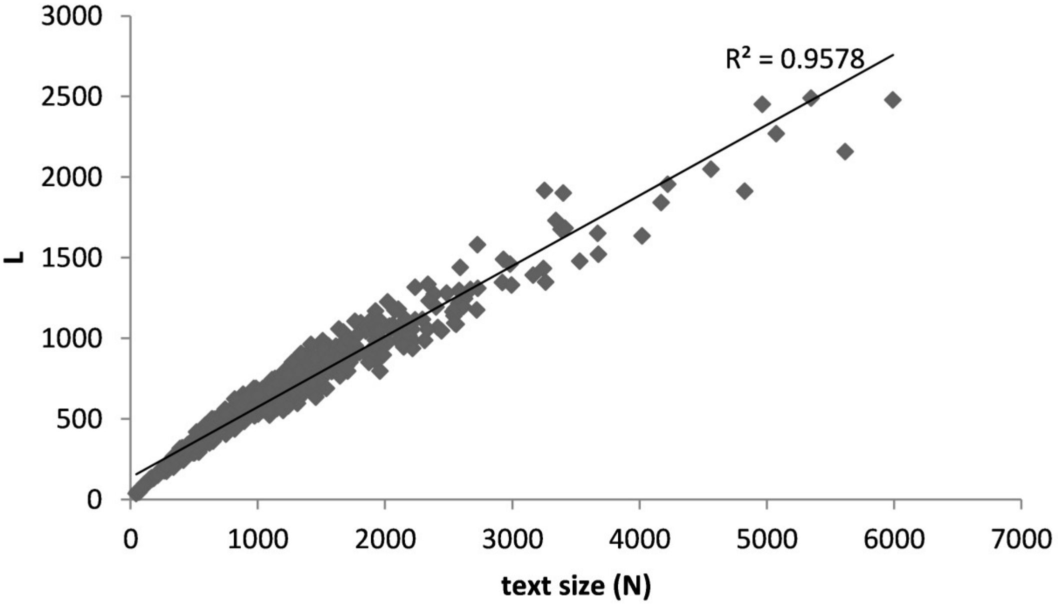


Figure 6.14. Text size impact on *L* in 658 Czech texts

Example

We obtain the curve length of our two texts by formula (6.21)

$$L_1=\sum_{r=1}^{V-1} \sqrt{\left(f(r)-f(r+1)\right)^2+1}=\sqrt{(16-7)^2+1}+\sqrt{(7-7)^2+1}+\sqrt{(7-7)^2+1}+\sqrt{(7-5)^2+1}+\ldots+\sqrt{(1-1)^2+1}=129.3$$

$$L_2=\sum_{r=1}^{V-1} \sqrt{\left(f(r)-f(r+1)\right)^2+1}=\sqrt{(20-9)^2+1}+\sqrt{(9-8)^2+1}+\sqrt{(8-7)^2+1}+\sqrt{(7-4)^2+1}+\ldots+\sqrt{(1-1)^2+1}=134.2$$

6.1.11. Curve length R Indicator (*R*)

This indicator of vocabulary richness is directly derived from the curve length (*L*) (see 6.1.10 Curve length (L)). In fact, *R* is the ratio of the curve length above the *h*-point (*L_h*) (see 6.1.2 h-point (h)) to the whole curve length (*L*). The formula is as follows:

$$(6.22)\, R = 1 - \frac{L_h}{L}$$

$$(6.23)\, L_h = \sum_{r=1}^{h} \sqrt{[f(r) - f(r+1)]^2 + 1}$$

L_h…curve length above *h*-point (see 6.1.2 h-point (h))
L…curve length (see 6.1.10 Curve length (L))
f…individual frequency
r…individual rank

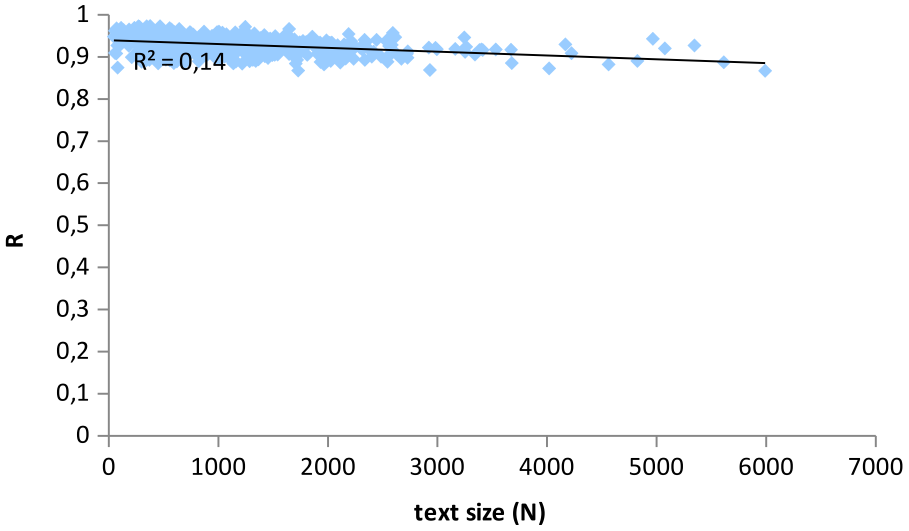


Figure 6.15. Text size impact on *R* in 658 Czech texts

Example

The first step is to calculate the curve length above *h*-point by formula (6.23)

$$L_{h(Text1)} = \sum_{r=1}^{h} \sqrt{[f(r) - f(r+1)]^2 + 1} = \text{¿} \sqrt{(16-7)^2+1} + \sqrt{(7-7)^2+1} + \sqrt{(7-7)^2+1} + \sqrt{(7-5)^2+1} = 14.29145\,\text{¿}$$

$$L_{h(Text2)} = \sum_{r=1}^{h} \sqrt{[f(r) - f(r+1)]^2 + 1} = \text{¿} \sqrt{(20-9)^2+1} + \sqrt{(9-8)^2+1} + \sqrt{(8-7)^2+1} + \sqrt{(7-4)^2} = 18.03607\,\text{¿}$$

The second step is to obtain *R* values using formula (6.22)

$$R_{Text1}=1-\frac{L_h}{L}=1-\frac{14.29145}{129.3559}=0.8895$$

$$R_{Text2}=1-\frac{L_h}{L}=1-\frac{18.03607}{134.2787}=0.8657$$

6.1.12. Entropy (*H*)

The term “entropy” is used in many sciences and there are several definitions with different meanings. In general, entropy measures diversity or uncertainty. In linguistics, entropy expresses how much the vocabulary of a text is concentrated. The smaller the *H* is, the more the vocabulary is concentrated and the smaller the vocabulary richness is. For example, if a text consists of 100 tokens and only one type, all frequencies are concentrated in one word and *H*=0. The most common formula of entropy in linguistics was defined by Shannon:

$$(6.24)\,H=-\sum_{i=1}^{K} p_i\, ld\, p_i$$

$$(6.25)\,p_i=\frac{f_i}{N}$$

P_i…individual probabilities (estimated by relative frequencies)
K…inventory size
ld…logarithm to the base 2
f_i…absolute frequency

$$(6.26)\,H=\log_2 N-\frac{1}{N}\sum_{r=1}^{V} f_i \log_2 f_i$$

The equation of the variance is as follows:

$$(6.27)\,Var(H)=\frac{1}{N}\left(\sum_{r=1}^{V} p_i(\log_2 p_i)^2-H^2\right)=\frac{1}{N}\left(\sum_{r=1}^{V}\frac{f_i}{N}\left(\log_2\left(\frac{f_i}{N}\right)\right)^2-H^2\right)$$

For comparison of two texts, we can use the asymptotic *u*-test:

$$(6.28)\,u=\frac{|H_1-H_2|}{\sqrt{Var(H_1)+Var(H_2)}}$$

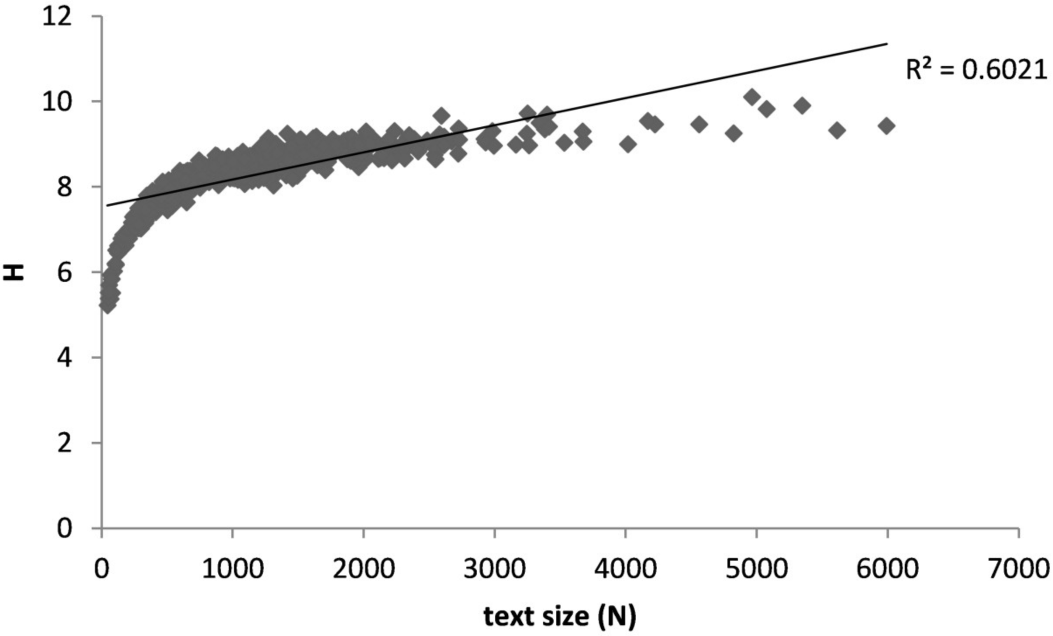


Figure 6.16. Text size impact on H in 658 Czech texts

Example

Entropy values can be directly obtained using just one formula (6.26).

$$H_{Text1} = \log_2 N - \frac{1}{N}\sum_{r=1}^{V} f_i \log_2 f_i$$

$$\text{¿}\log_2 179 - 16\cdot\log_2 16 + \text{¿}\,7\cdot\log_2 7 + \text{¿}\,7\cdot\log_2 \frac{7+\text{¿}\ldots+1\cdot\log_2 1}{179} = 6.438043\text{¿¿¿}$$

$$H_{Text2} = \log_2 N - \frac{1}{N}\sum_{r=1}^{V} f_i \log_2 f_i$$

$$\text{¿}\log_2 202 - 20\cdot\log_2 20 + \text{¿}\,9\cdot\log_2 9 + \text{¿}\,8\cdot\log_2 \frac{8+\text{¿}\ldots+1\cdot\log_2 1}{202} = 6.395099\text{¿¿¿}$$

The statistical comparison of two texts can be performed by the u-test (6.28), but we must firstly know the variances using formula (6.27).

$$Var_{Text1}(H) = \frac{1}{N}\left(\sum_{r=1}^{V} p_i(\log_2 p_i)^2 - H^2\right) = \frac{1}{N}\left(\sum_{r=1}^{V} \frac{f_i}{N}\left(\log_2\left(\frac{f_i}{N}\right)\right)^2 - H^2\right) = \Bigg|\frac{1}{179}\Bigg[\frac{16}{179}\left(\log_2\frac{16}{179}\right)^2 + \frac{7}{179}\Bigg(\log_2\frac{7}{17}$$

$$Var_{Text2}(H)=\frac{1}{N}\left(\sum_{r=1}^{V} p_i(\log_2 p_i)^2-H^2\right)=\frac{1}{N}\left(\sum_{r=1}^{V}\frac{f_i}{N}\left(\log_2\left(\frac{f_i}{N}\right)\right)^2-H^2\right)=\left|\frac{1}{202}\left[\frac{20}{202}\left(\log_2\frac{20}{202}\right)^2+\frac{9}{202}\left(\log_2\frac{9}{202}\right.\right.\right.$$

$$u=\frac{|H_1-H_2|}{\sqrt{Var(H_1)+Var(H_2)}}=\frac{|6.438043-6.395099|}{\sqrt{0.010355601+0.010345126}}=0.027$$

Given that the result of the *u*-test (0.027) is smaller than the threshold (1.96), the difference between the two texts is not significant.

6.1.13. Adjusted Modulus (*A*)

The adjusted modulus (*A*) is a frequency structure indicator which is supposed to be independent of text length. Nevertheless it can be seen in Figure 6.17 that this indicator is also influenced by text size. The formula of the adjusted modulus is defined as:

$$(6.29)\, A=\frac{M}{\log_{10} N}$$

$$(6.30)\, M=\left(\left(\frac{f(1)}{h}\right)^2+\left(\frac{V}{h}\right)^2\right)^{1/2}=\frac{1}{h}(f(1)^2+V^2)^{1/2}$$

h…*h*-point (see 6.1.2 h-point (h))
f(1)…the frequency of the most frequent word
V…vocabulary size
M…modulus

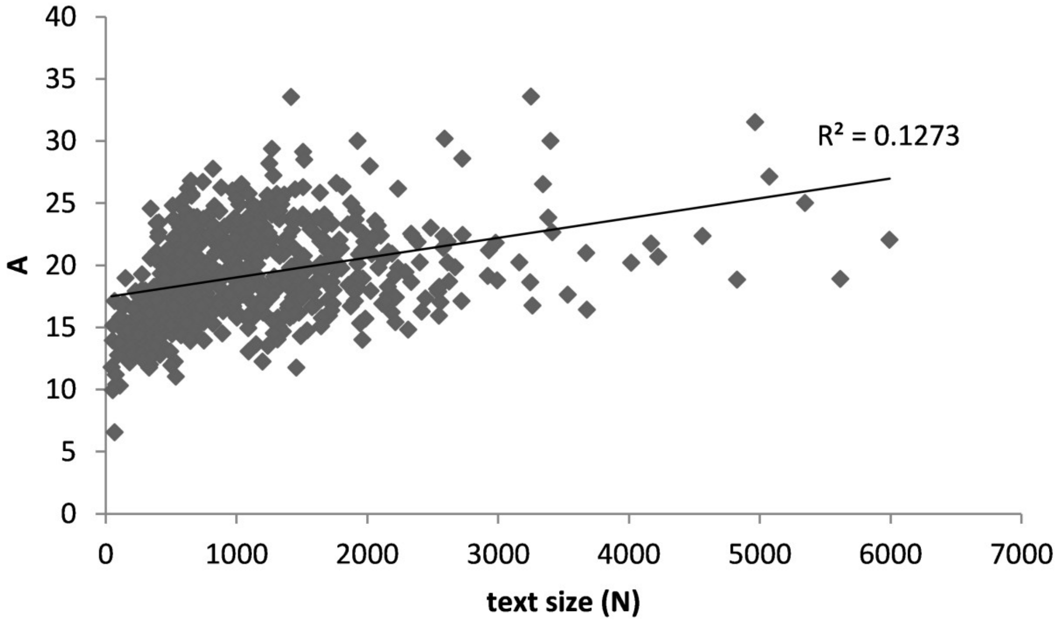


Figure 6.17. Text size impact on *A* in 658 Czech texts

Example

The calculation of the adjusted modulus consists of two steps. We must first obtain the values of the modulus (*M*) by formula (6.30) and then we can use formula 6.29 to calculate the adjusted modulus.

$$M_{Text1}=\frac{1}{h}\left(f(1)^2+V^2\right)^{1/2}=\frac{\left(16^2+119^2\right)^{1/2}}{5}=24.01416249$$

$$M_{Text2}=\frac{1}{h}\left(f(1)^2+V^2\right)^{1/2}=\frac{\left(20^2+121^2\right)^{1/2}}{4.75}=25.81931678$$

$$A_{Text1}=\frac{M}{\log_{10}N}=\frac{24.01416249}{\log_{10}179}=10.6594$$

$$A_{Text1}=\frac{M}{\log_{10}N}=\frac{25.81931678}{\log_{10}202}=11.1997$$

6.2. Miscellaneous indicators

6.2.1. Verb Distances (*VD*)

This indicator counts how many tokens on average are between two successive verbs. To obtain the results of *VD* it is necessary to select an appropriate POS (part of speech) tagger (see 7.4 POS Tagger) in the project settings.

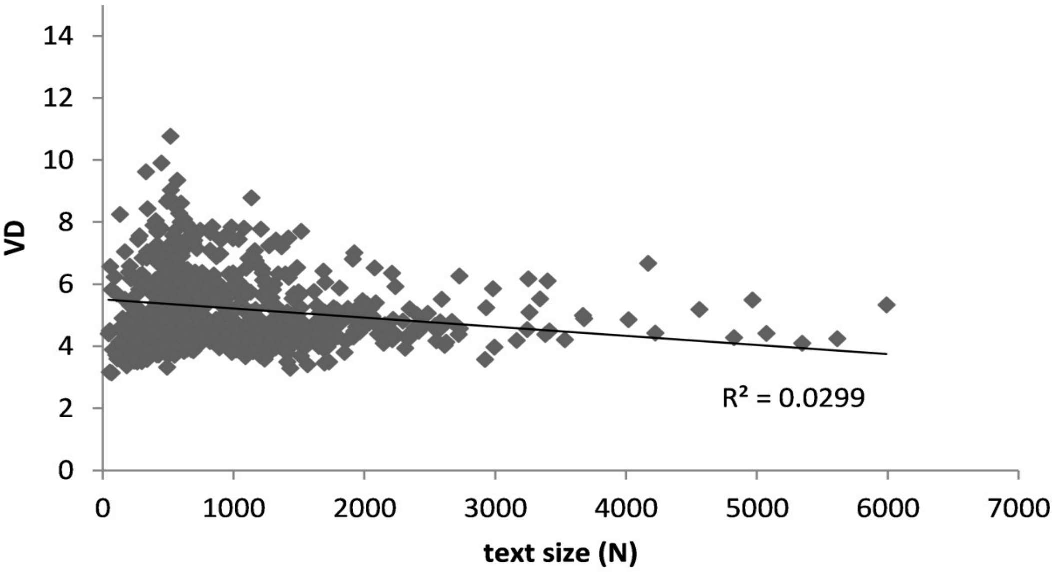


Figure 6.18. Text size impact on *VD* in 658 Czech texts

Example

Since QUITA has to recognize verbs in a text to compute verb distances, you must choose one of the available POS taggers before making the calculation. Given that our two texts are in English, there is only the NLTK POS tagger (see 7.4 POS Tagger) available. QUITA counts average gaps (in tokens) between two successive verbs in a text. In our case, we obtain the following results:

$VD_{Text\ 1}$=6.357

$VD_{Text\ 2}$=5.432

6.2.2. Activity (*Q*) & Descriptivity (*D)*

The concept of the activity and the descriptivity measurement in a text is very simple. The activity is represented by verbs and the descriptivity is represented by adjectives. The formula for activity (*Q*) is as follows:

$$(6.31)\, Q=\frac{V}{V+A}$$

V…number of verbs

A…number of adjectives

For a comparison of two texts we can use the *u*-test:

$$(6.32)\, u = \frac{|Q_1 - Q_2|}{\sqrt{Q_1 Q_2}\sqrt{\frac{1}{V_1} + \frac{1}{A_1} + \frac{1}{A_2} + \frac{1}{V_2}}}$$

The descriptivity (*D*) is just reversed value of the activity (*Q)*. *D* is therefore defined as:

$$(6.33)\, D = 1 - Q$$

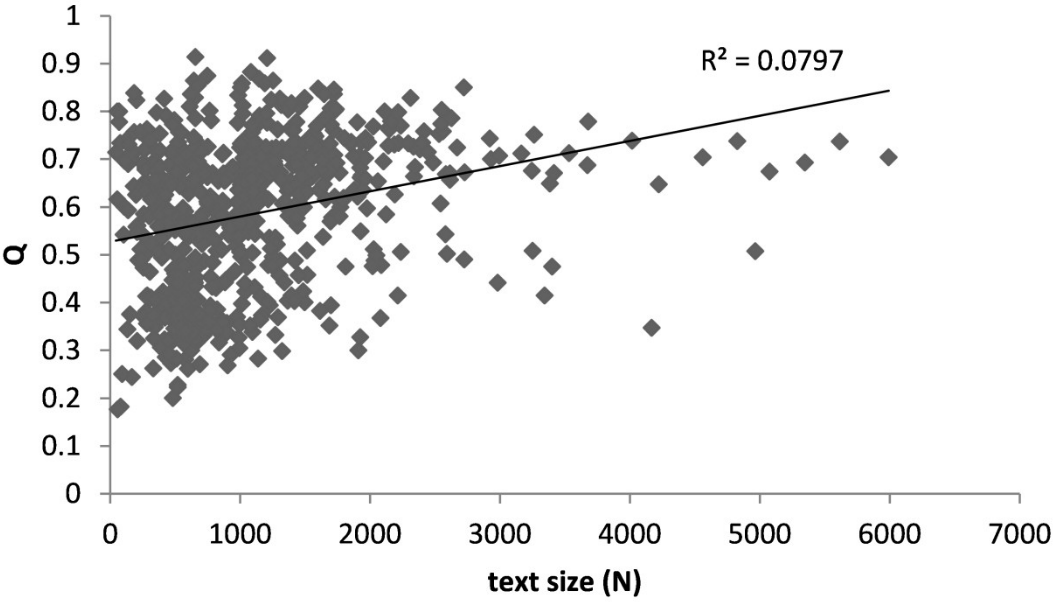


Figure 6.19. Text size impact on *Q* in 658 Czech texts

Example

The calculation by formula 6.31 is very simple; we need just to know the number of verbs and the number of adjectives in a text.

$$Q_{Text\,1} = \frac{V}{V + A} = \frac{26}{26 + 14} = 0.65$$

$$Q_{Text\,2} = \frac{V}{V + A} = \frac{35}{35 + 8} = 0.814$$

The difference obtained between two texts can be statistically compared by the *u*-test using formula 6.32.

$$u=\frac{|Q_1-Q_2|}{\sqrt{Q_1Q_2}\sqrt{\frac{1}{V_1}+\frac{1}{A_1}+\frac{1}{A_2}+\frac{1}{V_2}}}=\frac{|0.65-0.814|}{\sqrt{0.68\cdot0.814}\sqrt{\frac{1}{26}+\frac{1}{14}+\frac{1}{35}+\frac{1}{8}}}=0.43$$

Descriptivity can be directly calculated from the results obtained for activity by formula (6.33).

$$D_{Text1}=1-Q=1-0.65=0.35$$
$$D_{Text2}=1-Q=1-0.814=0.186$$

6.2.3. Writer's View (*a*)

The writer's view is an indicator connected to the golden ratio. It is supposed that each author of any text must abide by some universal law, namely the golden ratio. The writer's view is defined by the angle between the *h*-point (see 6.1.2 h-point (h)) and the ends of the rank-frequency distribution. The results should approximate to the value of the golden ratio (φ 1.618).

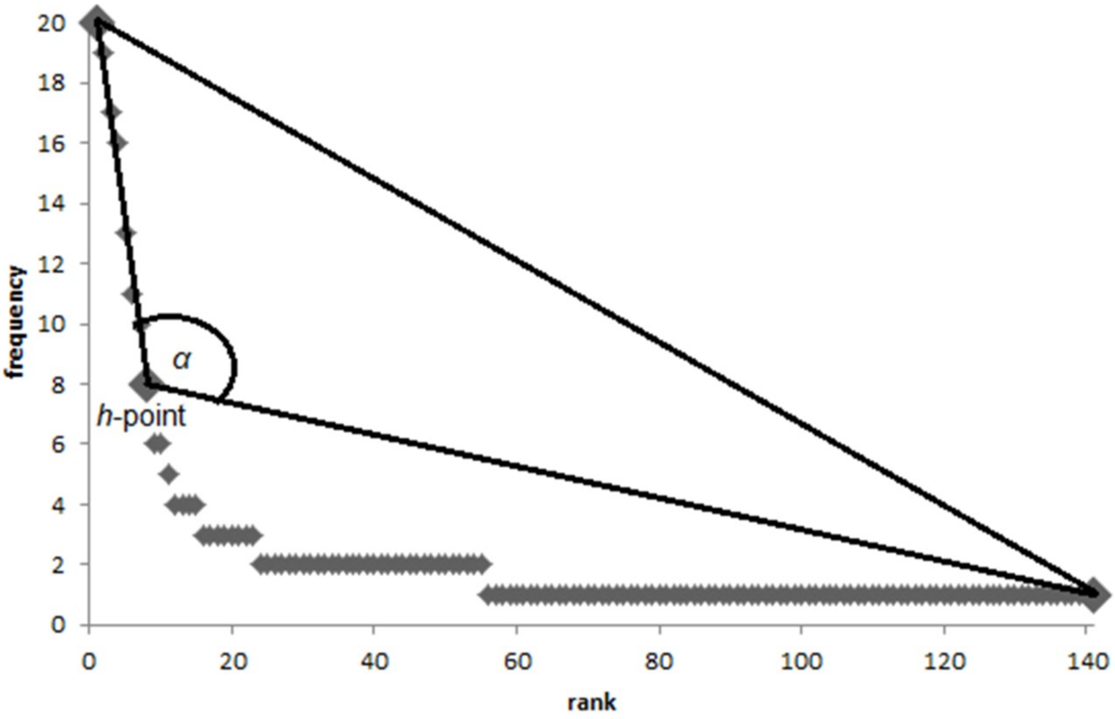


Figure 6.20. Example of writer's view in rank-frequency distribution

The equation of the writer's view is as follows:

$$(6.34)\cos\alpha=\frac{-[(h-1)(f_1-h)+(h-1)(V-h)]}{[(h-1)^2+(f_1-h)^2]^{1/2}[(h-1)^2+(V-h)^2]^{1/2}}$$

h…*h*-point (see 6.1.2 h-point (h))
f₁…the highest frequency
V...number of types

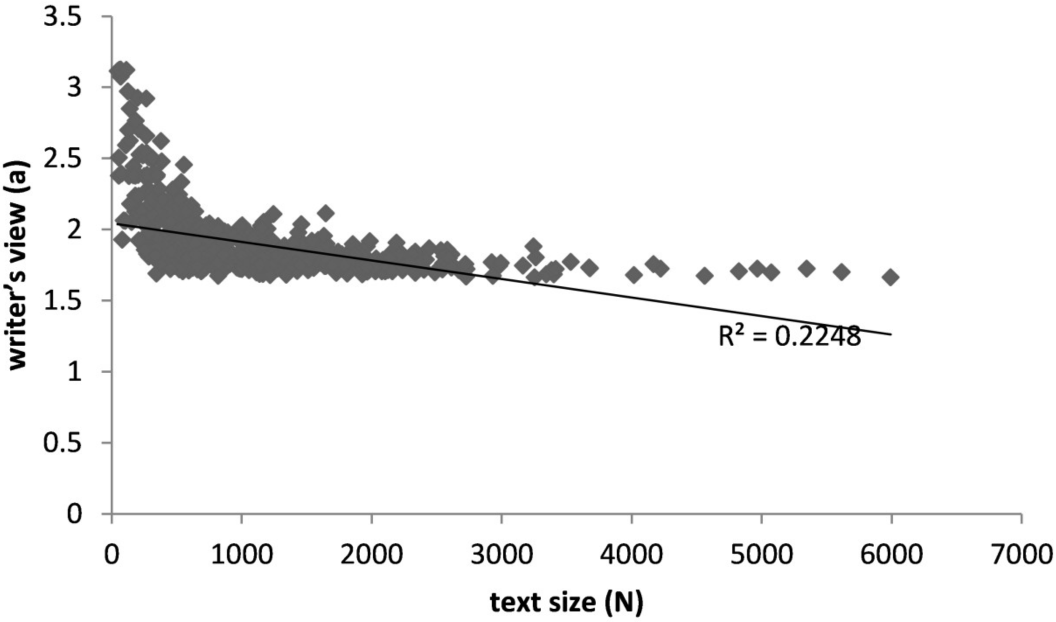


Figure 6.21. Text size impact on Writer's view in 658 Czech texts

Example

Although the writer's view is defined by one formula (6.34), it is necessary to do two steps because the results must be converted from degrees to radians.

$$\cos\alpha_{Text1}=\frac{-\left[(h-1)(f_1-h)+(h-1)(V-h)\right]}{\left[(h-1)^2+(f_1-h)^2\right]^{\frac{1}{2}}\left[(h-1)^2+(V-h)^2\right]^{\frac{1}{2}}}=\frac{-\left[(5-1)(16-5)+(5-1)(119-5)\right]}{\left[(5-1)^2+(16-5)^2\right]^{\frac{1}{2}}\left[(5-1)^2+(119-5)^2\right]^{\frac{1}{2}}}=-0.37448781$$

$$\cos\alpha_{Text2}=\frac{-[(h-1)(f_1-h)+(h-1)(V-h)]}{[(h-1)^2+(f_1-h)^2]^{1/2}[(h-1)^2+(V-h)^2]^{1/2}}=\frac{-[(4.75-1)(20-4.75)+(4.75-1)(121-4.75)]}{[(4.75-1)^2+(20-4.75)^2]^{1/2}[(4.75-1)^2+(121-4.75)^2}$$

We must convert the results of cos *a* to radians. Firstly we compute *a* from cosine (*a*) using a calculator. Finally we can convert degrees to radians.

$$a_{Text1} = \frac{\alpha \cdot \pi}{180} = \frac{111.9926603 \cdot \pi}{180} = 1.9546$$

$$a_{Text2} = \frac{\alpha \cdot \pi}{180} = \frac{105.6626356 \cdot \pi}{180} = 1.8442$$

6.2.4. Average Token length (*ATL)*

The resulting value simply shows the arithmetic mean of the lengths of tokens. *ATL* is defined as:

$$(6.35)\; ATL = \frac{1}{N} \sum_{i=1}^{N} x_i$$

N…number of tokens
x…individual length

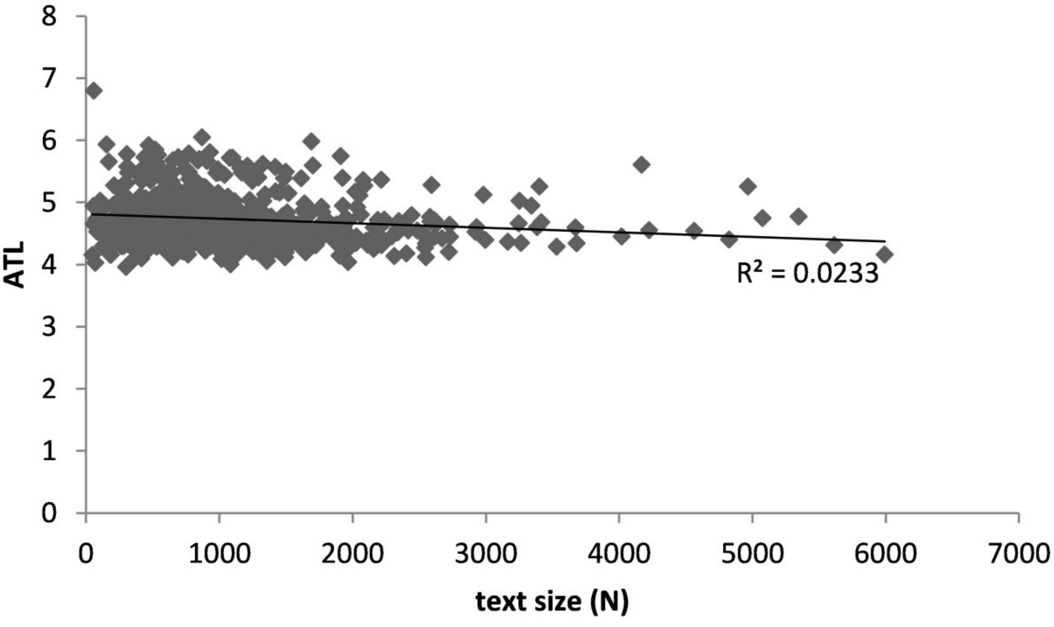


Figure 6.22. Text size impact on *ATL* in 658 Czech texts

Example

$$ATL_{Text1}=\frac{1}{n}\sum_{i=1}^{n}x_i=\frac{3+3+3+3+3+3+3+3+3+\ldots+6}{179}=4.252$$

$$ATL_{Text2}=\frac{1}{n}\sum_{i=1}^{n}x_i=\frac{3+3+3+3+3+3+3+3+3+\ldots+2}{202}=3.975$$

6.2.5. Thematic Concentration (*TC*)

The thematic concentration is a topic focusing measurement of a text. Each author of any text focuses on some topic which is represented by several autosemantic words. *TC* measures how much a text is concentrated on some topic: the bigger *TC* is, the more concentrated on some issue a text is.

The thematic concentration is based on the *h*-point (*h*) (see 6.1.2 h-point (h)) which divides vocabulary into synsemantics and autosemantics. Nevertheless, among synsemantics often occur several autosemantics. We can therefore consider these autosemantics above the *h*-point as some anomaly. These autosemantics are considered as thematic words of a text. *TC* is the sum of thematic weights (TW) of the individual thematic words. The thematic weight is defined as the distance between the *h*-point and the rank of a word above the *h*-point multiplied by its frequency f (r'):

$$(6.36)\,TW_{word}=2\frac{(h-r')f(r')}{h(h-1)f(1)}$$

$$(6.37)\,TC=\sum_{r'=1}^{T}2\frac{(h-r')f(r')}{h(h-1)f(1)}$$

r'…rank of autosemantic word above *h*-point
h…*h*-point (see 6.1.2 h-point (h))
T…number of thematic words

The variance of *TC* is defined as:

$$(6.38)\,Var(TC)=\left[\frac{2}{h(h-1)f(1)}\right]^2 n\,m_{2r'}$$

$$(6.39)\,m_{2r'}=\frac{\sum_{r'=1}^{T}\left(r'-m_{1r'}\right)^2 f(r')}{\sum_{r'=1}^{T}f(r')}$$

$$(6.40)\, m_{1r'} = \frac{\sum r' f(r')}{\sum f(r')}$$

n…sum of frequencies of the autosemantics

For a comparison of two texts we can use the asymptotic *u*-test:

$$(6.41)\, u = \frac{|TC_1 - TC_2|}{\sqrt{Var(TC_1) + Var(TC_2)}}$$

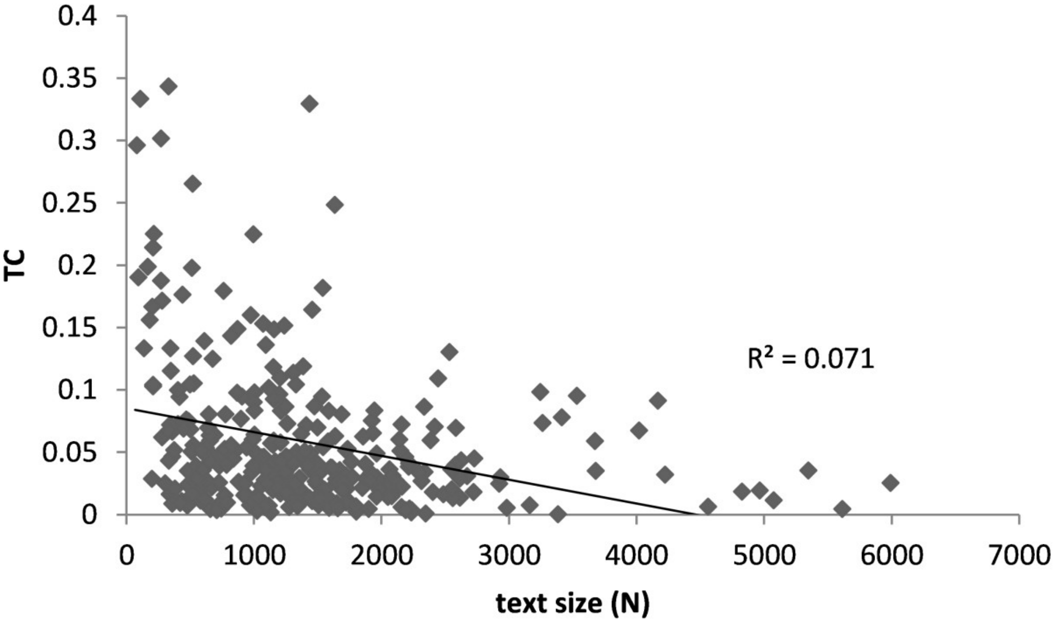


Figure 6.23. Text size impact on Thematic concentration in 658 Czech texts

Example

Given that Text 1 and Text 2 are not appropriate for *TC*, we used two poems, namely *I Said To Love* (Text 3) by Thomas Hardy and *The Two Nests* (Text 4) by Dora Sigerson. The texts can be found in the appendix of the manual.

Table 6.7
The rank-frequency distribution of Text 3

Text 3, h=5.5			
Token	Rank	Average rank	Frequency
to	1	1	8
said	**2**	**2.5**	**7**

i	3	2.5	7
the	4	4.5	6
love	**5**	**4.5**	**6**
we	6	6.5	5
thee	7	6.5	5
of	8	9	4
not	9	9	4
thou	10	9	4
now	11	11.5	3
in	12	11.5	3

Table 6.8
The rank-frequency distribution of Text 4

Text 4, h=6			
Token	Rank	Average rank	Frequency
her	1	1.5	16
the	2	1.5	16
wise	**3**	**3.5**	**8**
thrush	**4**	**3.5**	**8**
she	5	5	7
in	6	6	6
foolish	7	9.5	4
young	8	9.5	4
on	9	9.5	4
were	10	9.5	4
but	11	9.5	4
and	12	9.5	4
a	13	14.5	3
all	14	14.5	3
pretty	15	14.5	3
to	16	14.5	3

The first step is to find the *h*-point (see 6.1.2 h-point (h)).

$$h_{Text\,3}=\frac{f(r_1)r_2-f(r_2)r_1}{r_2-r_1+f(r_1)-f(r_2)}=\frac{6\cdot 6-5\cdot 5}{6-5+6-5}=5.5$$

In Text 4 *r*=*f* (*r*), so *h*=6.

When we have *h* values, we must find thematic words above the *h*-point in the word lists. We usually consider nouns, verbs and adjectives to be thematic words, but one can use also other parts of speech. There is also a problem with verbs which have only grammatical function. So everybody must decide which words would be considered to be thematic words. Given that some lemmatizers in QUITA are not able to separate verbs with grammatical function from verbs with content function, the software considers all verbs as thematic words. It is therefore necessary to calculate the thematic concentration by hand sometimes.

The next step is to compute *TC* using formula (6.37).

$$TC_{Text\,3}=2\sum_{r'=1}^{T}\frac{(h-r')f(r')}{h(h-1)f(1)}=2\frac{(5.5-2.5)7}{5.5(5.5-1)8}+2\frac{(5.5-4.5)6}{5.5(5.5-1)8}=0.242424$$

$$TC_{Text\,4}=2\sum_{r'=1}^{T}\frac{(h-r')f(r')}{h(h-1)f(1)}=2\frac{(6-3.5)8}{6(6-1)16}+2\frac{(6-3.5)8}{6(6-1)16}=0.166667$$

It is necessary to compute the variance for the final statistical test:

$$Var(TC)_{Text3}=\left[\frac{2}{h(h-1)f(1)}\right]^2 n\,m_{2r'}=\left[\frac{2}{5.5(5.5-1)8}\right]^2 13\cdot 0.994083=0.00131855$$

$$m_{2r'Text3}=\frac{\sum_{r'=1}^{T}(r'-m_{1r'})^2 f(r')}{\sum_{r'=1}^{T}f(r')}=\frac{(2.5-3.423077)^2 7+(4.5-3.423077)^2 6}{7+6}=0.994083$$

$$m_{1r'Text3}=\frac{\sum r' f(r')}{\sum f(r')}=\frac{2.5\cdot 7+4.5\cdot 6}{7+6}=3.423077$$

$$Var(TC)_{Text\,4}=\left[\frac{2}{h(h-1)f(1)}\right]^2 n\,m_{2r'}=\left[\frac{2}{6(6-1)1}\right]^2 16\cdot 0=0.$$

$$m_{2r'Text4}=\frac{\sum_{r'=1}^{T}(r'-m_{1r'})^2 f(r')}{\sum_{r'=1}^{T}f(r')}=\frac{(3.5-3.5)^2 8+(3.5-3.5)^2 8}{8+8}=0$$

$$m_{1r'Text4}=\frac{\sum r'f(r')}{\sum f(r')}=\frac{3.5\cdot 8+3.5\cdot 8}{8+8}=3.5$$

The last step is to discover whether the resulting *TC* values of the two texts differ significantly from each other using formula (6.41).

$$u=\frac{|TC_1-TC_2|}{\sqrt{Var(TC_1)+Var(TC_2)}}=\frac{|0.242424-0.166667|}{\sqrt{0.00131855+0}}=2.09$$

Since the threshold value is 1.96, u>1.96 means that the difference between the two texts is significant.

6.2.6. Secondary Thematic Concentration (S*TC*)

The secondary thematic concentration is almost the same indicator as the thematic concentration above (see 6.2.5 Thematic Concentration (TC)). The computation process differs only in the *h*-point (see 6.1.2 h-point (h)) which is multiplied by 2.

Thematic concentration struggles with a problem of quite often null results. It means that there is no autosemantic word above the *h*-point in many texts, so the results of thematic concentration must be 0. This fact complicates the use of the indicator. Given that secondary thematic concentration covers a bigger number of words, there is a higher probability that the result will not be 0.

$$(6.42)\, STC=\sum_{r'=1}^{2h}\frac{(2h-r')f(r')}{h(2h-1)f(1)}$$

The variance of *STC* is defined as:

$$(6.43)\, Var(STC)=\frac{nm_{2r'}}{[h(2h-1)f(1)]^2}$$

$$(6.44)\, m_{2r'}=\frac{\sum_{r'=1}^{T}(r'-m_{1r'})^2 f(r')}{\sum_{r'=1}^{T} f(r')}$$

$$(6.45)\, m_{1r'}=\frac{\sum r'f(r')}{\sum f(r')}$$

For a comparison of the two texts we can use the asymptotic *u*-test:

$$(6.46)\,u=\frac{\left|STC_1-STC_2\right|}{\sqrt{Var\left(STC_1\right)+Var\left(STC_2\right)}}$$

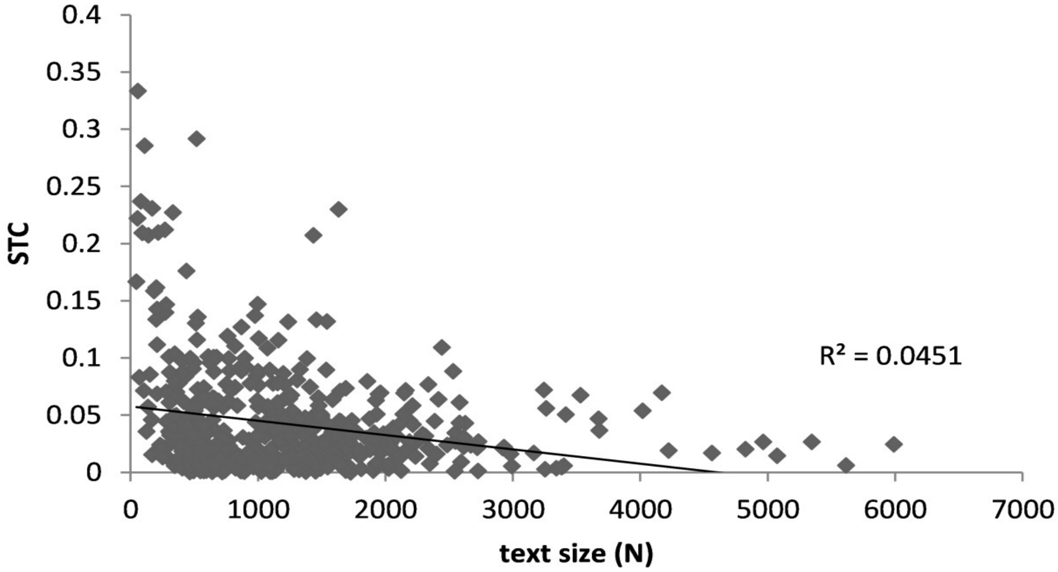


Figure 6.24. Text size impact on Secondary thematic concentration in 658 Czech texts

Example

We used the same texts (Text 3 and Text 4) as in the *TC* example above. So we do not have to compute the *h*-points again. The problem with choosing thematic words from the word list is also discussed in the chapter above.

Table 6.9
The rank-frequency distribution of Text 3

Text 3, h=5.5			
Token	Rank	Averag e rank	Frequency
to	1	1	8
said	**2**	**2.5**	**7**
i	3	2.5	7
the	4	4.5	6
love	**5**	**4.5**	**6**
we	6	6.5	5
thee	7	6.5	5
of	8	9	4

not	9	9	4
thou	10	9	4
now	11	11.5	3
in	12	11.5	3

Table 6.10
The rank-frequency distribution of Text 4

Text 4, h=6			
Token	Rank	Average rank	Frequency
her	1	1.5	16
the	2	1.5	16
wise	**3**	**3.5**	**8**
thrush	**4**	**3.5**	**8**
she	5	5	7
in	6	6	6
foolish	**7**	**9.5**	**4**
young	**8**	**9.5**	**4**
on	9	9.5	4
were	10	9.5	4
but	11	9.5	4
and	12	9.5	4
a	13	14.5	3
all	14	14.5	3
pretty	15	14.5	3
to	16	14.5	3

Firstly, we use formula (6.42) to obtain *STC* values.

$$STC_{Text\,3}=\sum_{r^{'}=1}^{2h}\frac{(2h-r^{'})f(r^{'})}{h(2h-1)f(1)}=\frac{(2\cdot 5.5-2.5)7}{5.5(2\cdot 5.5-1)8}+\frac{(2\cdot 5.5-4.5)6}{5.5(2\cdot 5.5-1)8}=0.223864$$

$$STC_{Text\,4}=\sum_{r^{'}=1}^{2h}\frac{(2h-r^{'})f(r^{'})}{h(2h-1)f(1)}=\frac{(2\cdot 6-3.5)8}{6(2\cdot 6-1)16}+\frac{(2\cdot 6-3.5)8}{6(2\cdot 6-1)16}+\frac{(2\cdot 6-9.5)4}{6(2\cdot 6-1)16}+\frac{(2\cdot 6-9.5)4}{6(2\cdot 6-1)16}=0.147727$$

It is necessary to compute the variance for the final statistical test:

$$Var(STC)_{Text\,3}=\frac{nm_{2r'}}{[h(2h-1)f(1)]^2}=\frac{13\cdot 0.994083}{[5.5(2\cdot 5.5-1)8]^2}=0.000066751$$

$$m_{2r'Text3}=\frac{\sum_{r'=1}^{T}\left(r'-m_{1r'}\right)^2 f\left(r'\right)}{\sum_{r'=1}^{T} f\left(r'\right)}=\frac{(2.5-3.423077)^2 7+(4.5-3.423077)^2 6}{7+6}=0.994083$$

$$m_{1r'Text3}=\frac{\sum r' f\left(r'\right)}{\sum f\left(r'\right)}=\frac{2.5\cdot 7+4.5\cdot 6}{7+6}=3.423077$$

$$Var\left(STC\right)_{Text4}=\frac{nm_{2r'}}{\left[h(2h-1)f(1)\right]^2}=\frac{24\cdot 8}{\left[6(2\cdot 6-1)16\right]^2}=0.000172176$$

$$m_{2r'Text4}=\frac{\sum_{r'=1}^{T}\left(r'-m_{1r'}\right)^2 f\left(r'\right)}{\sum_{r'=1}^{T} f\left(r'\right)}=\frac{(3.5-5.5)^2 8+(3.5-5.5)^2 8+(9.5-5.5)^2 4+(9.5-5.5)^2 4}{8+8+4+4}=8$$

$$m_{1r'Text4}=\frac{\sum r' f\left(r'\right)}{\sum f\left(r'\right)}=\frac{3.5\cdot 8+3.5\cdot 8+9.5\cdot 4+9.5\cdot 4}{8+8+4+4}=5.5$$

The last step is to discover whether the S*TC* resulting values of the two texts differ significantly from each other using formula (6.46).

$$u=\frac{\left|STC_1-STC_2\right|}{\sqrt{Var\left(STC_1\right)+Var\left(STC_2\right)}}=\frac{\left|0.223864-0.147727\right|}{\sqrt{0.000066751+0.000172176}}=4.93$$

Since the threshold value is 1.96, u>1.96 means that the difference between the two texts is significant.

6.2.7. Proportional Thematic Concentration (*PTC*)

The proportional thematic concentration is a different way of approaching thematic concentration. *PTC* is also based on the *h*-point and the index is computed as the proportion of thematic words in the pre-*h*-domain. The formula is as follows:

$$(6.47)\, PTC=\frac{1}{N_h}\sum_{r'\le h} f\left(r'\right)$$

N_h…frequency of all words in the pre-*h*-domain
f(*r'*)…frequency of autosemantic word in the pre-*h*-domain

The variance of *PTC* is defined as:

$$(6.48)\, Var(PTC) = \frac{PTC(1-PTC)}{N_h}$$

For a comparison of two texts we can use the asymptotic *u*-test:

$$(6.49)\, u = \frac{|PTC_1 - PTC_2|}{\sqrt{Var(PTC_1) + Var(PTC_2)}}$$

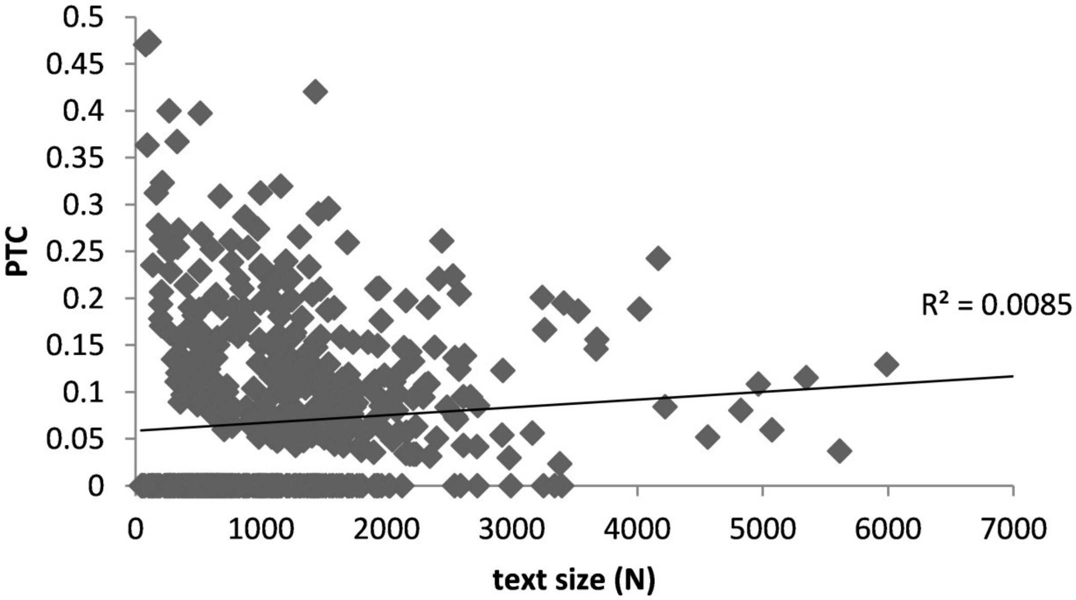


Figure 6.25. Text size impact on Proportional thematic concentration in 658 Czech texts

Example

We used the same texts (Text 3 and Text 4) as in the *TC* and STC examples above. So we do not have to compute the *h*-points again. The problem with choosing thematic words from the word list is also discussed in the chapters above.

Table 6.11
The rank-frequency distribution of Text 3

Text 3, h=5.5

Token	Rank	Average rank	Frequency
to	1	1	8
said	**2**	**2.5**	**7**
i	3	2.5	7
the	4	4.5	6
love	**5**	**4.5**	**6**
we	6	6.5	5
thee	7	6.5	5
of	8	9	4
not	9	9	4
thou	10	9	4
now	11	11.5	3
in	12	11.5	3

Table 6.12
The rank-frequency distribution of Text 4

Text 4, h=6			
Token	Rank	Average rank	Frequency
her	1	1.5	16
the	2	1.5	16
wise	**3**	**3.5**	**8**
thrush	**4**	**3.5**	**8**
she	5	5	7
in	6	6	6
foolish	7	9.5	4
young	8	9.5	4
on	9	9.5	4
were	10	9.5	4
but	11	9.5	4
and	12	9.5	4
a	13	14.5	3
all	14	14.5	3
pretty	15	14.5	3
to	16	14.5	3

$$PTC_{Text3}=\frac{1}{N_h}\sum_{r'\leq h} f\left(r'\right)=\frac{13}{34}=0.38235$$

$$PTC_{Text\,4}=\frac{1}{N_h}\sum_{r^{'}\leq h} f\left(r^{'}\right)=\frac{16}{61}=0.262295$$

$$Var\left(PTC\right)_{Text3}=\frac{PTC\left(1-PTC\right)}{N_h}=\frac{0.38235\left(1-0.38235\right)}{34}=0.0069458$$

$$Var\left(PTC\right)_{Text4}=\frac{PTC\left(1-PTC\right)}{N_h}=\frac{0.262295\left(1-0.262295\right)}{61}=0.003172$$

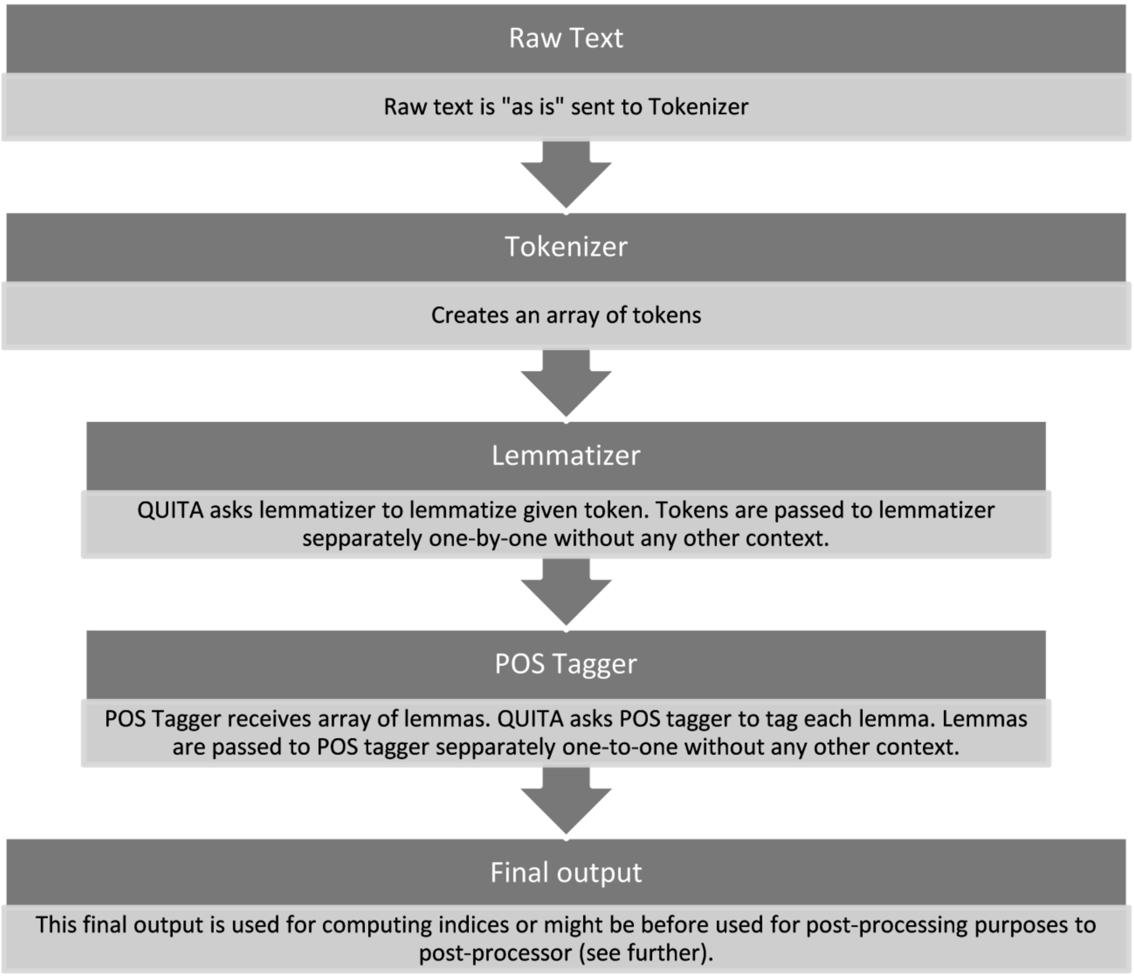
$$u=\frac{\left|PTC_1-PTC_2\right|}{\sqrt{Var\left(PTC_1\right)+Var\left(PTC_2\right)}}=\frac{\left|0.38235-0.262295\right|}{\sqrt{0.0069458+0.003172}}=1.19$$

Since the threshold value is 1.96, u<1.96 means that the difference between the two texts is not significant.

7. Pre-processing: Tokenization, Lemmatization, POS Tagging

QUITA's main task is to reduce the amount of work needed for quantitative analysis of texts. For this reason, QUITA implements the most important tools and interfaces for pre-processing such texts to create their final form capable and suitable for quantitative analysis. The pre-processing procedure starts with the necessary step of tokenization (recognition of single words) and then proceeds through two optional steps: lemmatization and POS (part of speech) tagging. Lemmatization is an optional step as long as the user does not require the use of lemmas or when the user supplies an already lemmatized text as an input to QUITA. POS tagging is also an optional pre-processing step which (internally) supplies part of speech tag for every token.

To clarify the process and the exact order of applying all of the pre-processing tools mentioned above, see the following flow chart:



Raw Text

Raw text is "as is" sent to Tokenizer

Tokenizer

Creates an array of tokens

Lemmatizer

QUITA asks lemmatizer to lemmatize given token. Tokens are passed to lemmatizer sepparately one-by-one without any other context.

POS Tagger

POS Tagger receives array of lemmas. QUITA asks POS tagger to tag each lemma. Lemmas are passed to POS tagger sepparately one-to-one without any other context.

Final output

This final output is used for computing indices or might be before used for post-processing purposes to post-processor (see further).

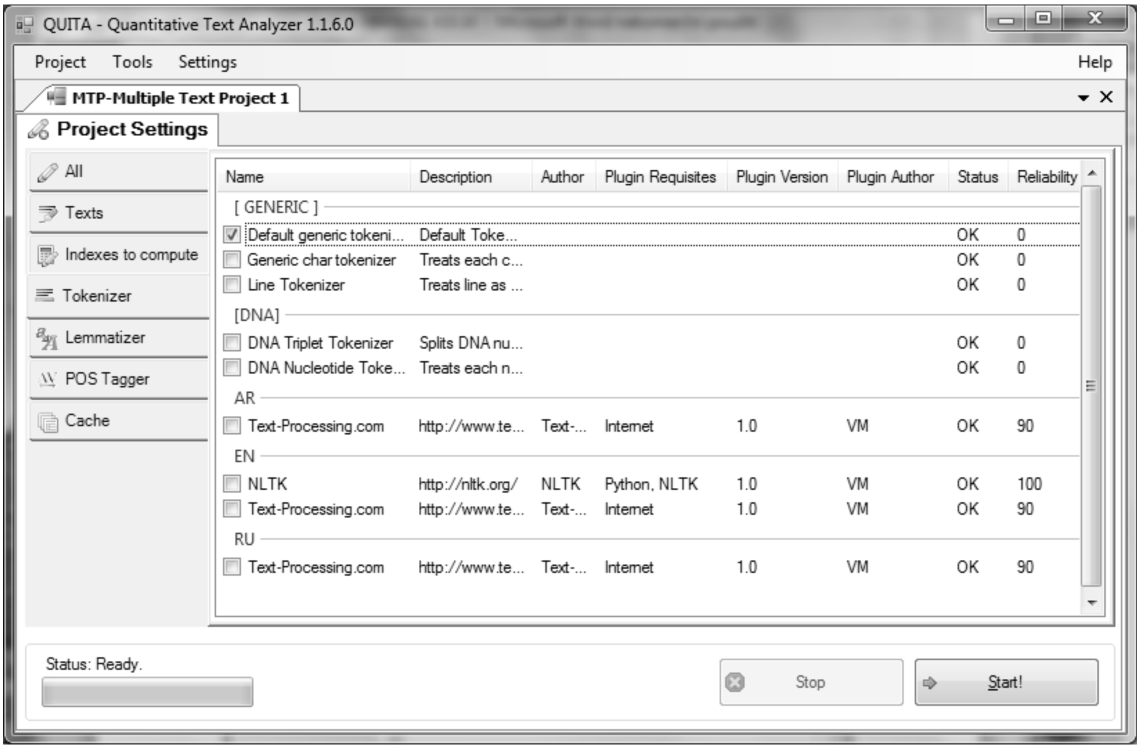
NOTE: The actual model of text pre-processing might be not suitable for efficient and accurate lemmatization or POS Tagging because of passing tokens or lemmas into the Lemmatizer or POS Tagger tools individually (without any context). Lemmatization and POS Tagging tools use contextual information. But you can overcome this problem by inputting already lemmatized text into QUITA in vertical format (lemma-per-line format; for details, see further in this section). Unfortunately, for this first version of QUITA, this problem cannot be overcome for the POS Tagging pre-processing tool.

7.1. QUITA and third party tools

QUITA has the ability to cooperate with programs that allow communication through the system standard input/output (known as "pipe" or STDIN, STDOUT), including tools such as Python scripts, Perl scripts, executable command line tools (EXE files, …) or Internet Web applications with the POST/GET interface. The actually available third party programs were chosen for preview reason only. Adding new tools is not supported in the current version but is planned for the next version of QUITA.

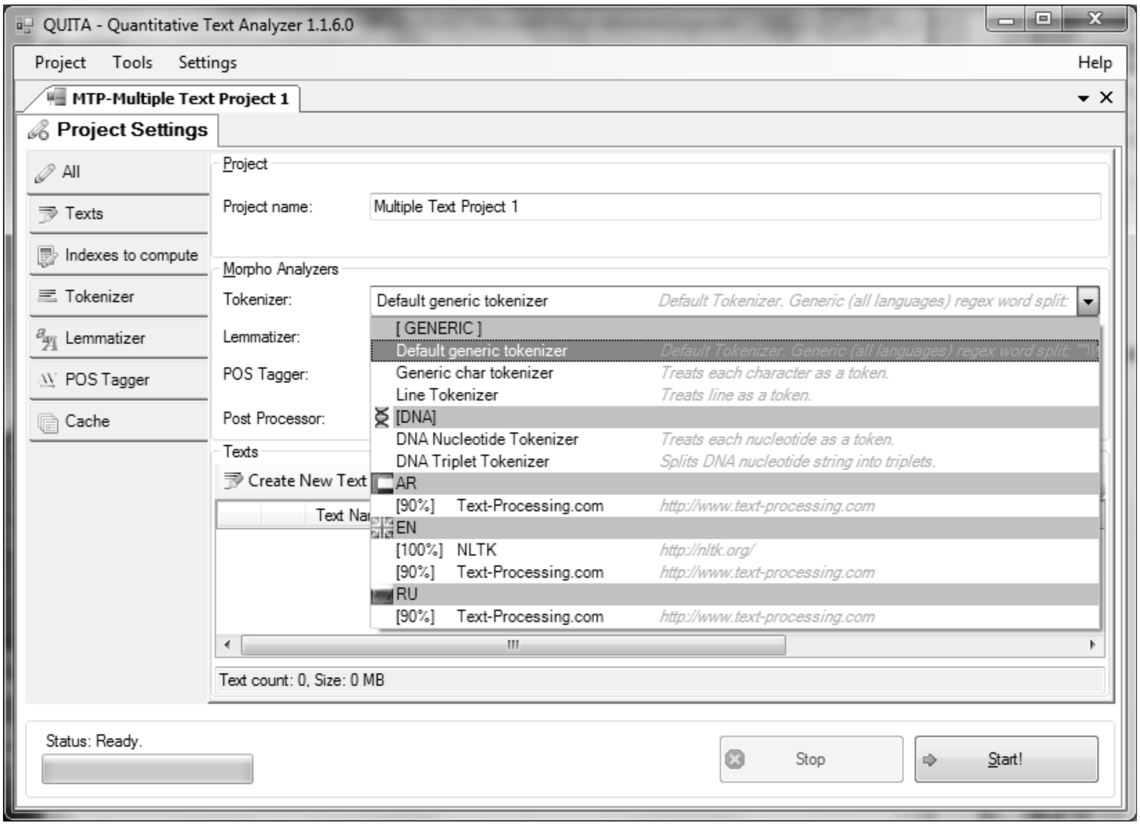
7.2. Tokenizer

Every text inputted into QUITA has to be tokenized before the computation of any indicators can begin. You can find all the available tokenizers in the "Project Settings – Tokenizer" card where you can also tick the one you want to use, or, you can simply set the desired tokenizer in "Project Settings – All (summary)" card.



Name	Description	Author	Plugin Requisites	Plugin Version	Plugin Author	Status	Reliability
[GENERIC]							
☑ Default generic tokeni...	Default Toke...					OK	0
□ Generic char tokenizer	Treats each c...					OK	0
□ Line Tokenizer	Treats line as ...					OK	0
[DNA]							
□ DNA Triplet Tokenizer	Splits DNA nu...					OK	0
□ DNA Nucleotide Toke...	Treats each n...					OK	0
AR							
□ Text-Processing.com	http://www.te...	Text-...	Internet	1.0	VM	OK	90
EN							
□ NLTK	http://nltk.org/	NLTK	Python, NLTK	1.0	VM	OK	100
□ Text-Processing.com	http://www.te...	Text-...	Internet	1.0	VM	OK	90
RU							
□ Text-Processing.com	http://www.te...	Text-...	Internet	1.0	VM	OK	90

Status: Ready.



Project name: Multiple Text Project 1

Tokenizer: Default generic tokenizer — Default Tokenizer. Generic (all languages) regex word split.

Lemmatizer:

POS Tagger:

Post Processor:

[GENERIC]	
Default generic tokenizer	Default Tokenizer. Generic (all languages) regex word split. "\\
Generic char tokenizer	Treats each character as a token.
Line Tokenizer	Treats line as a token.
[DNA]	
DNA Nucleotide Tokenizer	Treats each nucleotide as a token.
DNA Triplet Tokenizer	Splits DNA nucleotide string into triplets.
AR	
[90%] Text-Processing.com	http://www.text-processing.com
EN	
[100%] NLTK	http://nltk.org/
[90%] Text-Processing.com	http://www.text-processing.com
RU	
[90%] Text-Processing.com	http://www.text-processing.com

Text count: 0, Size: 0 MB

Status: Ready.

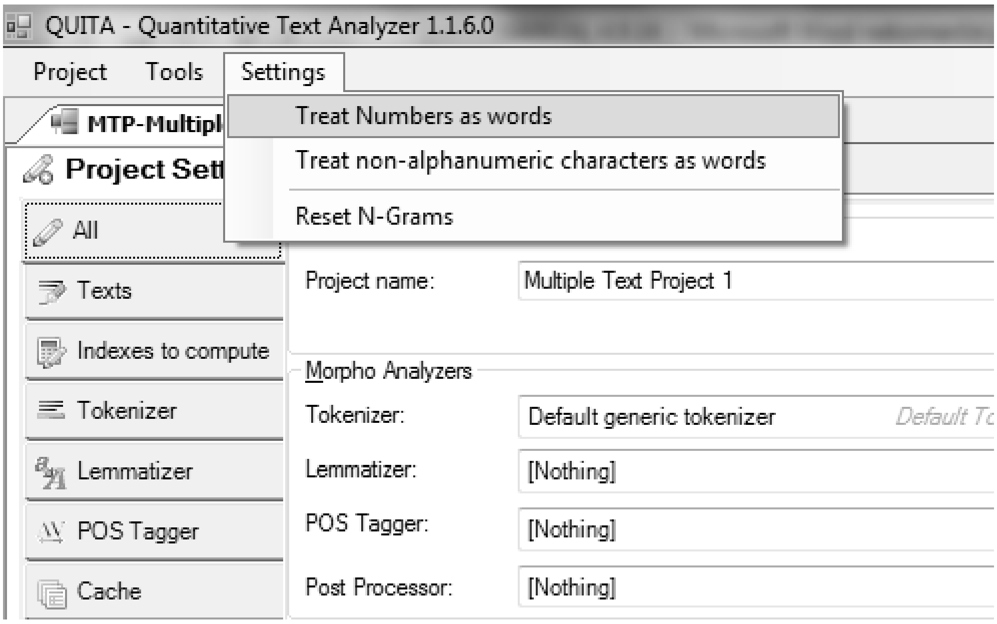
7.2.1. Available tokenizers

QUITA contains several basic built-in tokenizers:

- Default generic tokenizer – this tokenizer uses the regular expression “\W+” to split tokens by “non words characters” (anything except A-Z and 0-9 chars).
- Line Tokenizer – treats each line as a single token – this tokenizer allows users to tokenize text on their own and then supply this fully tokenized text to QUITA in the usual token-per-line format.
- Generic char tokenizer – treats each character as a single token.
- DNA Triplet Tokenizer – treats each three characters as a token. FNA (FASTA) files are fully supported – description lines (starting with “>” character) are ignored while tokenizing.
- DNA Nucleotide Tokenizer – treats each character as a token. FNA (FASTA) files are fully supported – description lines are ignored while tokenizing.

All tokenizers in QUITA:

- Ignore non-letter or number (see below) single character tokens. In other words: all tokens with length = 1 are removed when the only char is not a letter or is a number (see further).
- Ignore tokens that are numbers and non-alphanumeric characters. This behaviour can be turned off by ticking “Treat Numbers as words” and “Treat non-alphanumeric characters as words” in the menu Settings (see Figure below).



Project name: Multiple Text Project 1

Tokenizer: Default generic tokenizer

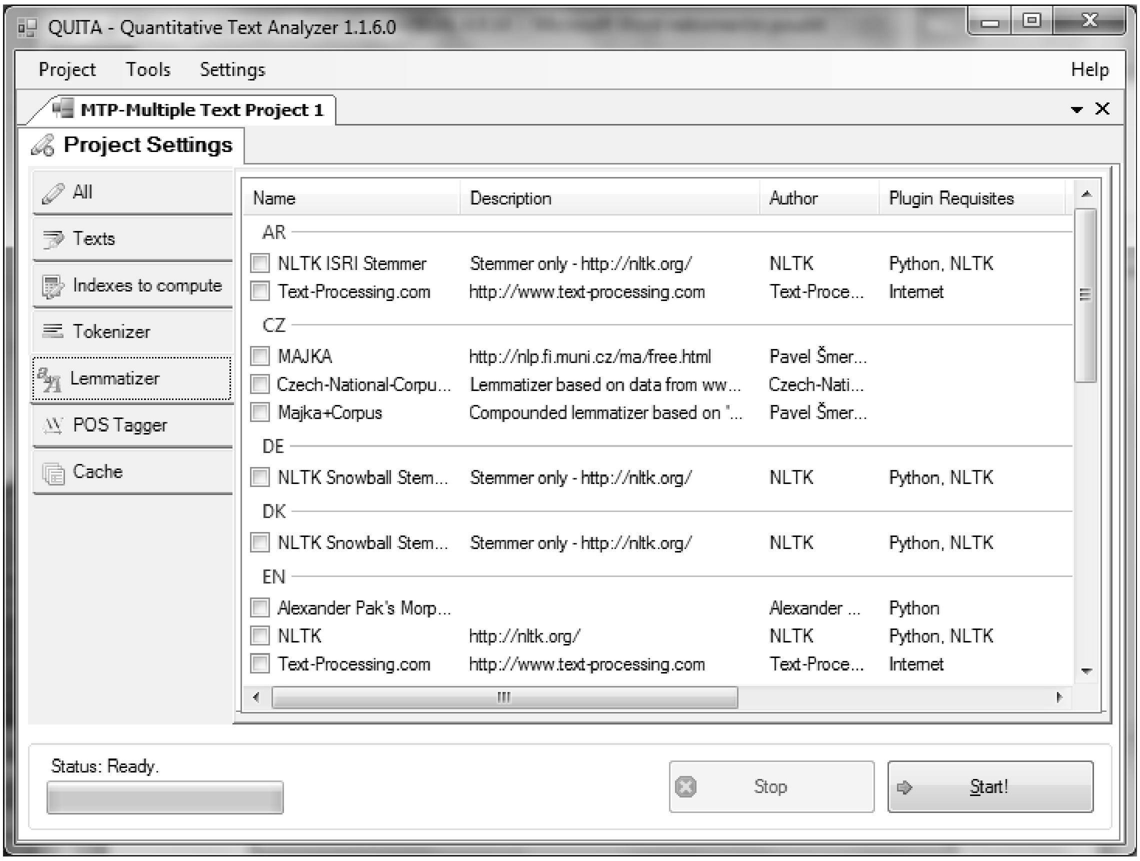
Lemmatizer: [Nothing]

POS Tagger: [Nothing]

Post Processor: [Nothing]

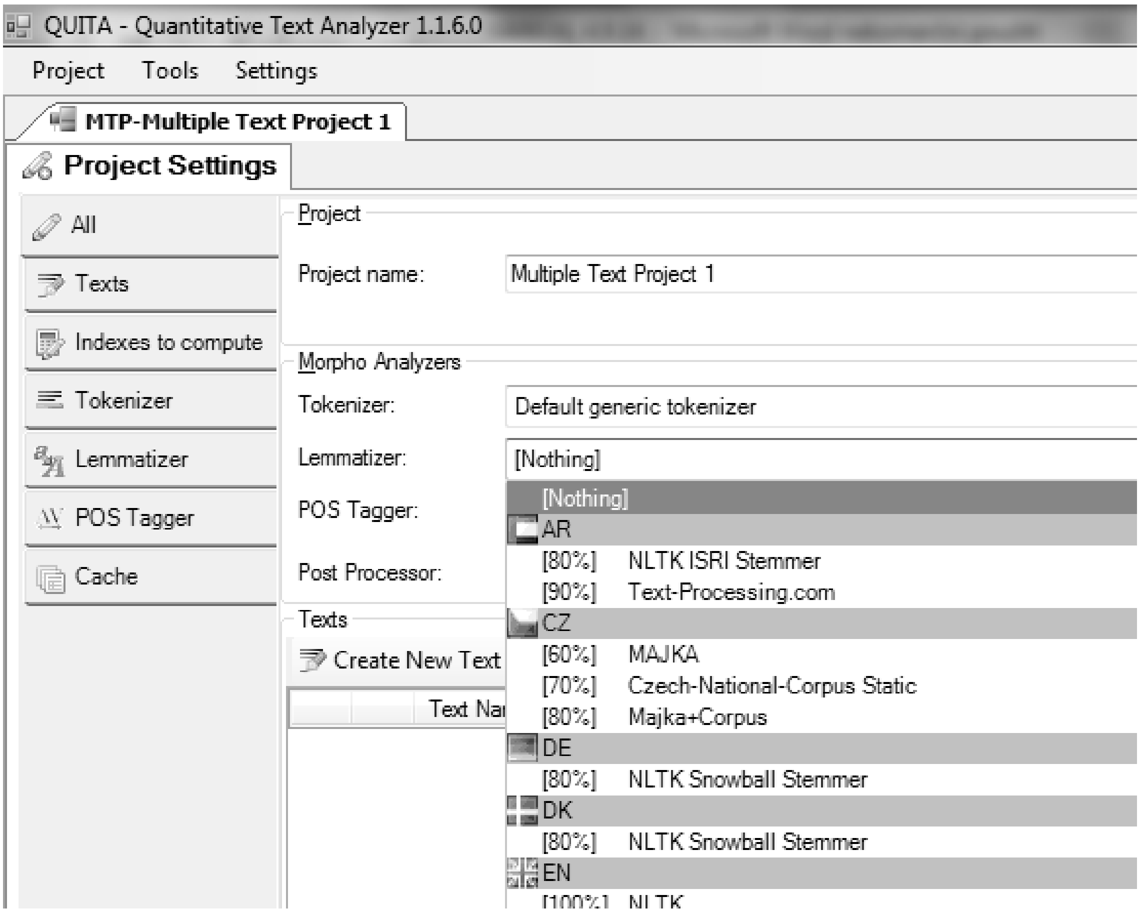
7.3. Lemmatizer

Given that word-forms do not have to be suitable units for some texts (especially those written in inflected languages, e.g. Slavonic languages), QUITA enables users to lemmatize a text. Click on the “Project Settings – Lemmatizer” card, and as with the Tokenizer settings, just tick the one you want; or you can easily set the desired lemmatizer in the “Project Settings – All” card.



Name	Description	Author	Plugin Requisites
AR			
□ NLTK ISRI Stemmer	Stemmer only - http://nltk.org/	NLTK	Python, NLTK
□ Text-Processing.com	http://www.text-processing.com	Text-Proce...	Internet
CZ			
□ MAJKA	http://nlp.fi.muni.cz/ma/free.html	Pavel Šmer...	
□ Czech-National-Corpu...	Lemmatizer based on data from ww...	Czech-Nati...	
□ Majka+Corpus	Compounded lemmatizer based on "...	Pavel Šmer...	
DE			
□ NLTK Snowball Stem...	Stemmer only - http://nltk.org/	NLTK	Python, NLTK
DK			
□ NLTK Snowball Stem...	Stemmer only - http://nltk.org/	NLTK	Python, NLTK
EN			
□ Alexander Pak's Morp...		Alexander ...	Python
□ NLTK	http://nltk.org/	NLTK	Python, NLTK
□ Text-Processing.com	http://www.text-processing.com	Text-Proce...	Internet

Status: Ready.



Project

Project name: Multiple Text Project 1

Morpho Analyzers

Tokenizer: Default generic tokenizer

Lemmatizer: [Nothing]

POS Tagger:

Post Processor:

Texts

[Nothing]	
AR	
[80%]	NLTK ISRI Stemmer
[90%]	Text-Processing.com
CZ	
[60%]	MAJKA
[70%]	Czech-National-Corpus Static
[80%]	Majka+Corpus
DE	
[80%]	NLTK Snowball Stemmer
DK	
[80%]	NLTK Snowball Stemmer
EN	
[100%]	NLTK

As mentioned above, you can supply the already lemmatized and tokenized text in lemma-per-line format to QUITA while setting the tokenizer tool to “Line Tokenizer” and the lemmatizer tool to “[Nothing]”; or you can supply just a lemmatized text in a common format and let QUITA tokenize it with any of the supported tokenizers.

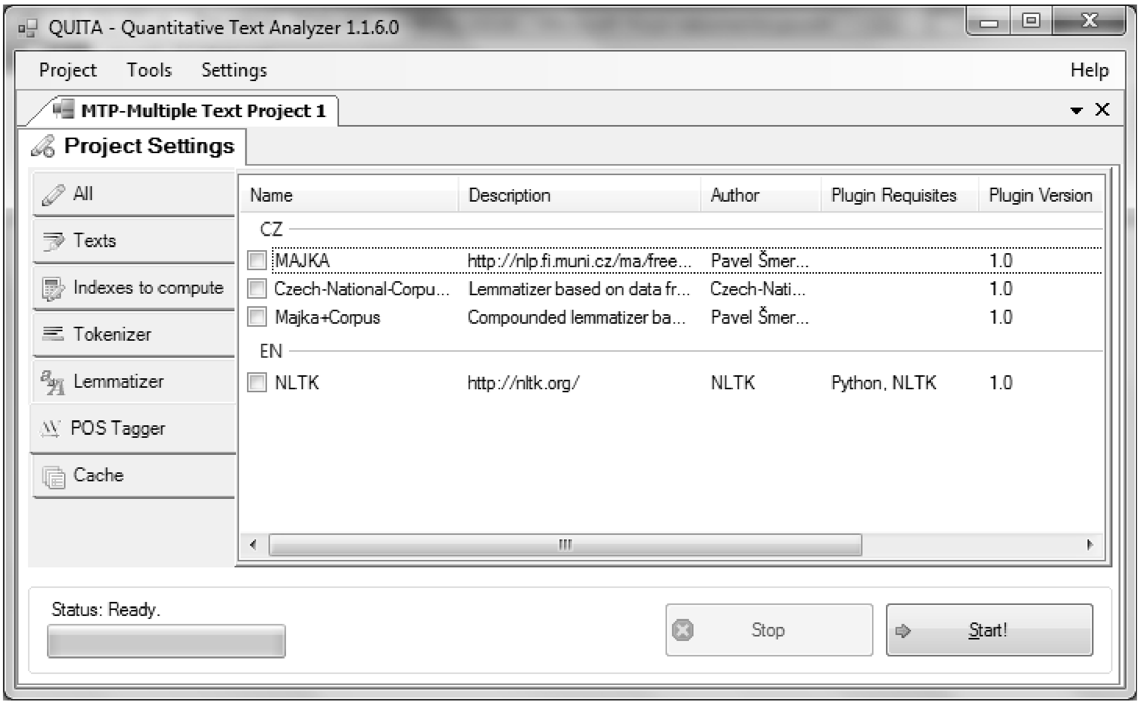
The list of available lemmatizers:

- Arabic (AR),
- Czech (CZ),
- German (DE),
- Danish (DK),
- English (EN),
- Spanish (ES),
- Finnish (FI),
- French (FR),
- Italian (IT),
- Dutch (NL),
- Portuguese (PT),
- Romanian (RO),

- Russian (RU),
- Swedish (SE).

7.4. POS Tagger

The POS Tagger allows users to distinguish parts of speech in a text. You can set the POS Tagger in the “Project Settings – POS Tagger” card or in the “Project Settings – All” card. To active this option, click on “POS Tagger” and tick the one you want. The actually supported POS Taggers were chosen for this preview only. Support for other POS Taggers by plugins will be added in future versions of QUITA.



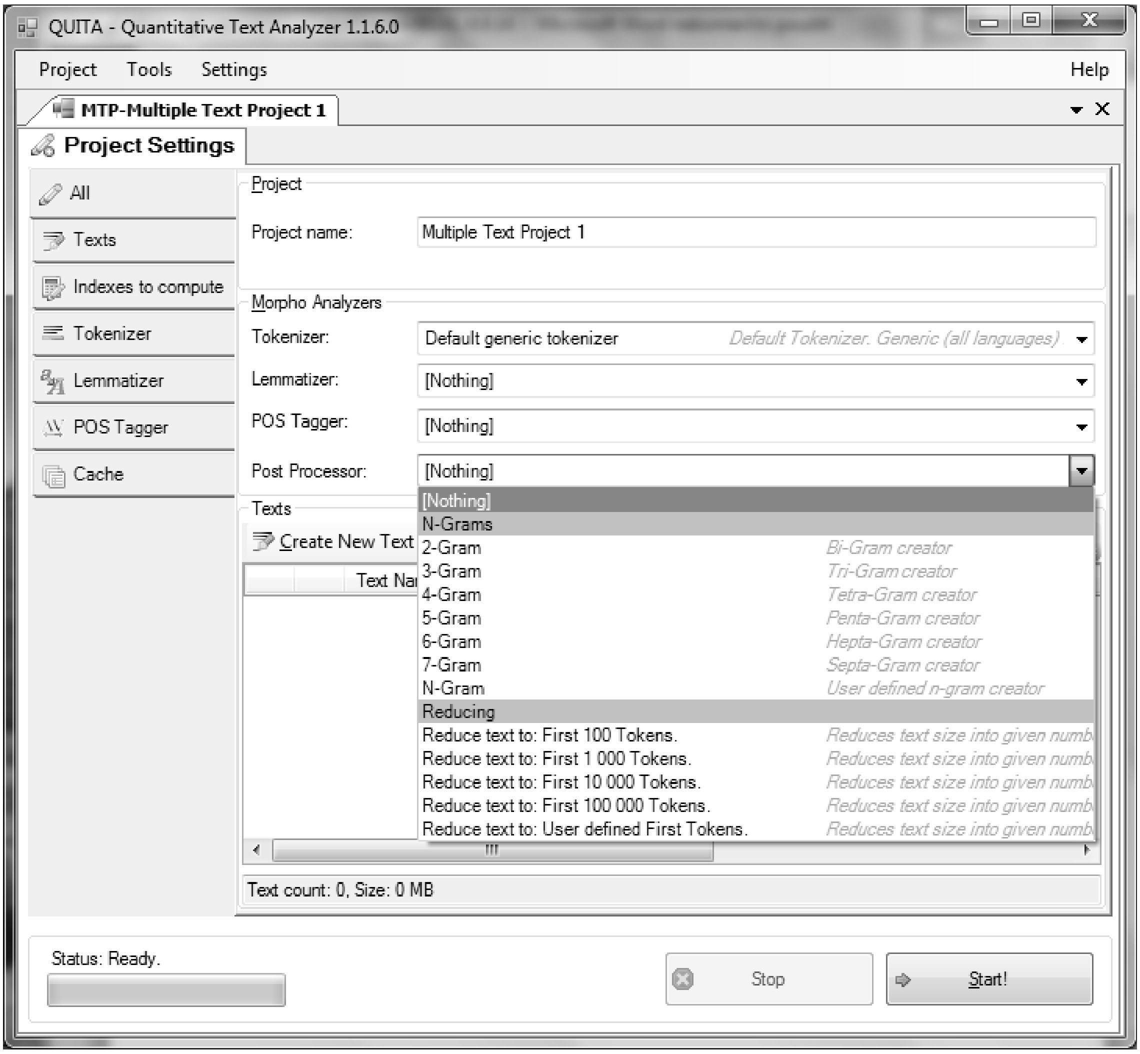
Name	Description	Author	Plugin Requisites	Plugin Version
CZ				
□ MAJKA	http://nlp.fi.muni.cz/ma/free...	Pavel Šmer...		1.0
□ Czech-National-Corpu...	Lemmatizer based on data fr...	Czech-Nati...		1.0
□ Majka+Corpus	Compounded lemmatizer ba...	Pavel Šmer...		1.0
EN				
□ NLTK	http://nltk.org/	NLTK	Python, NLTK	1.0

Status: Ready.

NOTE: The Lemmatizer and POS Tagger overview tables contain a Reliability column which is reserved for future use and meanwhile does not provide any important information.

8. Post-processing

After the pre-processing, the raw text is transformed into its final form which can be edited, in light of all the data obtained from the tokenization, lemmatization and POS tagging. The output of the post-processing procedure is then passed as a final output used for calculations. The setting of the post-processing tool can be found in the "Project settings – All" card (see figure below).



Project

Project name: Multiple Text Project 1

Morpho Analyzers

Tokenizer: Default generic tokenizer — Default Tokenizer. Generic (all languages)

Lemmatizer: [Nothing]

POS Tagger: [Nothing]

Post Processor: [Nothing]

[Nothing]

N-Grams

2-Gram	Bi-Gram creator
3-Gram	Tri-Gram creator
4-Gram	Tetra-Gram creator
5-Gram	Penta-Gram creator
6-Gram	Hepta-Gram creator
7-Gram	Septa-Gram creator
N-Gram	User defined n-gram creator

Reducing

Reduce text to: First 100 Tokens.	Reduces text size into given numb
Reduce text to: First 1 000 Tokens.	Reduces text size into given numb
Reduce text to: First 10 000 Tokens.	Reduces text size into given numb
Reduce text to: First 100 000 Tokens.	Reduces text size into given numb
Reduce text to: User defined First Tokens.	Reduces text size into given numb

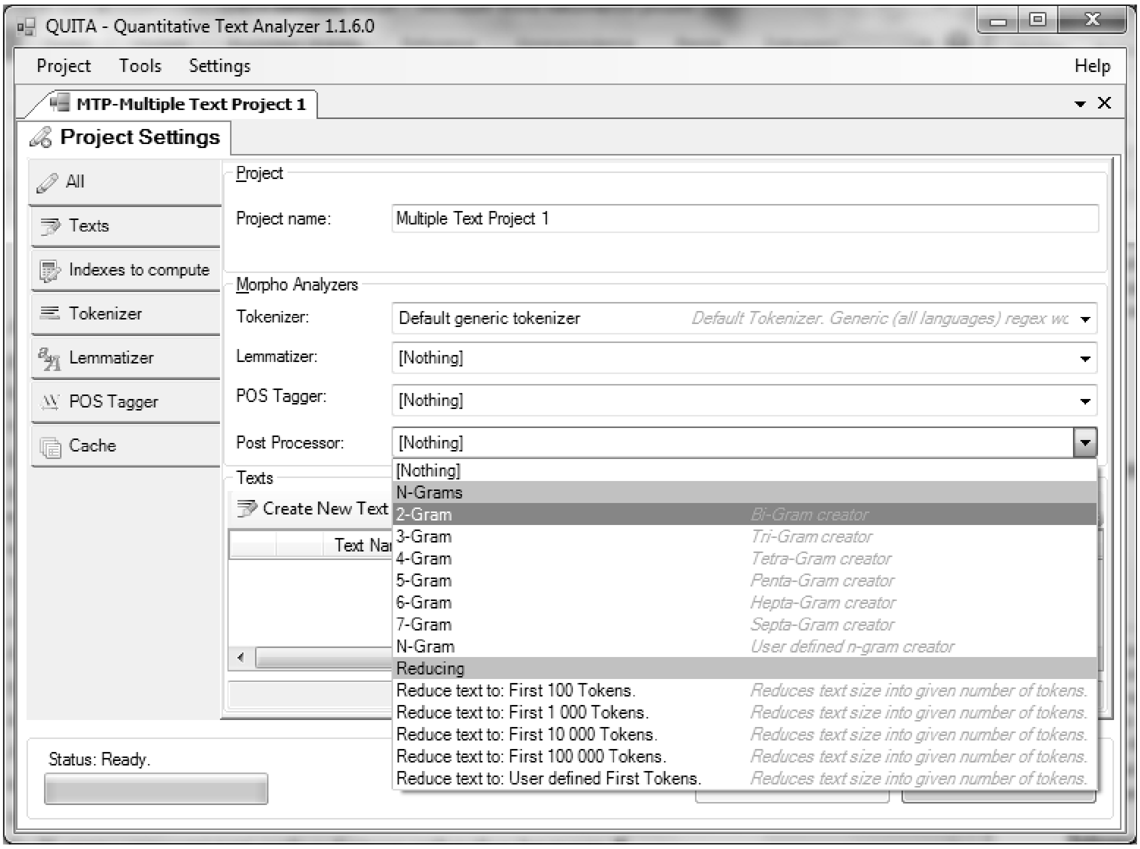
Texts

Text count: 0, Size: 0 MB

Status: Ready.

8.1. N-grams

An N-gram is a continuous sequence consisting of *n* units in a given text. The N-grams in a text can then be counted. Characters and words are mostly regarded as units but one can consider phonemes, syllables, sentences, etc. as units too. Click on the "Post Processor" arrow to open a menu where you can choose from bigrams, tri-grams, tetra-grams or you can select whatever n-grams you want. N-gram settings can be removed by clicking on “Reset N-Grams” in “Settings”.


QUITA - Quantitative Text Analyzer 1.1.6.0
Project
Tools
Settings
Help
MTP-Multiple Text Project 1
Project Settings
All
Texts
Indexes to compute
Tokenizer
Lemmatizer
POS Tagger
Cache
Project
Morpho Analyzers
Texts
Create New Text
Text Na

Project name: Multiple Text Project 1

Tokenizer: Default generic tokenizer — Default Tokenizer. Generic (all languages) regex wc

Lemmatizer: [Nothing]

POS Tagger: [Nothing]

Post Processor: [Nothing]

[Nothing]
N-Grams
2-Gram Bi-Gram creator
3-Gram Tri-Gram creator
4-Gram Tetra-Gram creator
5-Gram Penta-Gram creator
6-Gram Hepta-Gram creator
7-Gram Septa-Gram creator
N-Gram User defined n-gram creator
Reducing
Reduce text to: First 100 Tokens. Reduces text size into given number of tokens.
Reduce text to: First 1 000 Tokens. Reduces text size into given number of tokens.
Reduce text to: First 10 000 Tokens. Reduces text size into given number of tokens.
Reduce text to: First 100 000 Tokens. Reduces text size into given number of tokens.
Reduce text to: User defined First Tokens. Reduces text size into given number of tokens.
Status: Ready.

N-grams are units which can be used in several fields such as probability, communication theory, natural language processing, computational biology or data compression. In quantitative linguistics, n-grams are usually used in stylometry. N-grams even seem to be the most powerful tool in authorship attribution.

Although studies based on n-grams yield very good results, it is important to mention that there are also considerable disadvantages. These units are not used in traditional linguistics because they are not connected to any linguistic theory. So N-grams represent text surface features rather than language.

8.2. Text Length Reduction

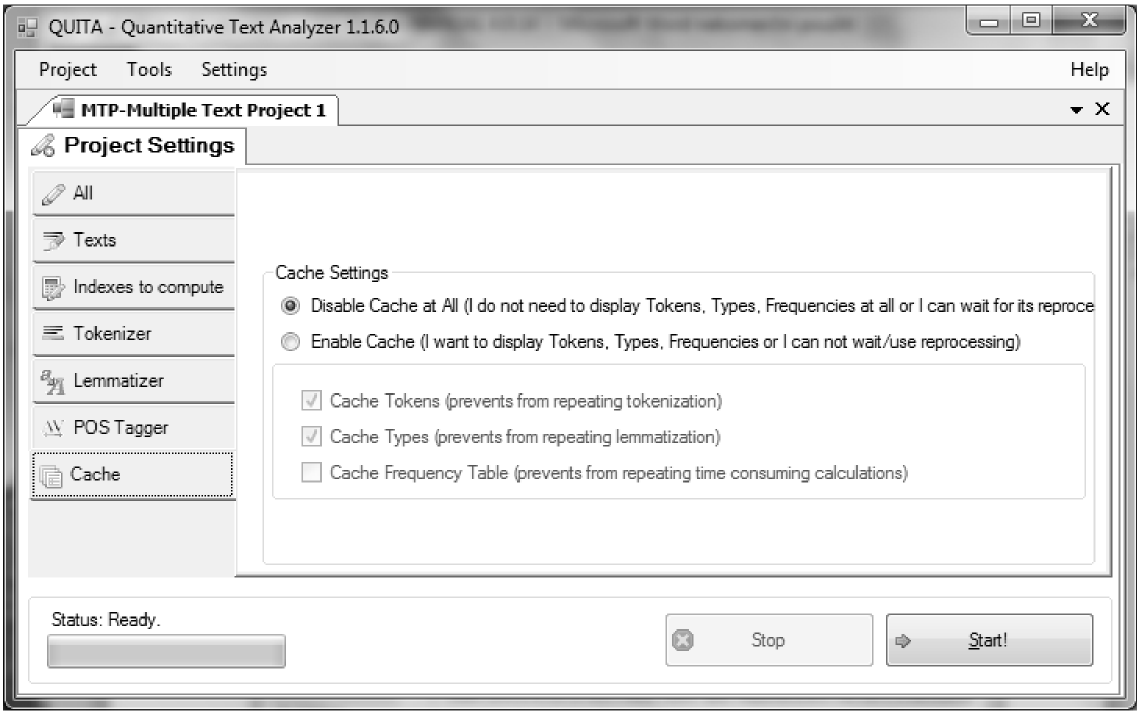
Since most indicators of frequency structure are influenced by text length, comparing resulting values between texts of different sizes can be misleading. The easiest way to avoid this problem is to reduce texts to the first *n* tokens. Click on the “Post Processor” arrow to open a menu where you can reduce texts to the first *n* tokens.

Although text reduction is a simple solution, it is very problematic from a linguistics point of view. Linguists consider each text as one cohesive unit. Thus, only complete texts can be analysed without any misleading results. So each researcher has to decide which solution is better in his or her analysis. It depends on many aspects of the research and it is therefore impossible to say which solution is better in general. That is why each indicator in this manual is accompanied by a graph which shows the results in 658 texts of various lengths – everyone can see how much each indicator is influenced by the text size.

9. Cache Settings

Using tokenizers, lemmatizers and POS taggers may require a lot of time. If a user wants to show only the word list, the data are loaded from the cache memory (the text is not tokenized, lemmatized or tagged again). Thus, the process is much faster. On the other hand, if you work with many texts, the cache memory may be overloaded. On the "Cache" card, you can adjust the memory settings:

- Disable Cache
- Enable Cache
  - Cache Tokens (prevents repeating tokenization)
  - Cache Types (prevents repeating lemmatization)
  - Cache Frequency Table (prevents repeating time consuming calculations)



Cache Settings

● Disable Cache at All (I do not need to display Tokens, Types, Frequencies at all or I can wait for its reproce

○ Enable Cache (I want to display Tokens, Types, Frequencies or I can not wait/use reprocessing)

☑ Cache Tokens (prevents from repeating tokenization)

☑ Cache Types (prevents from repeating lemmatization)

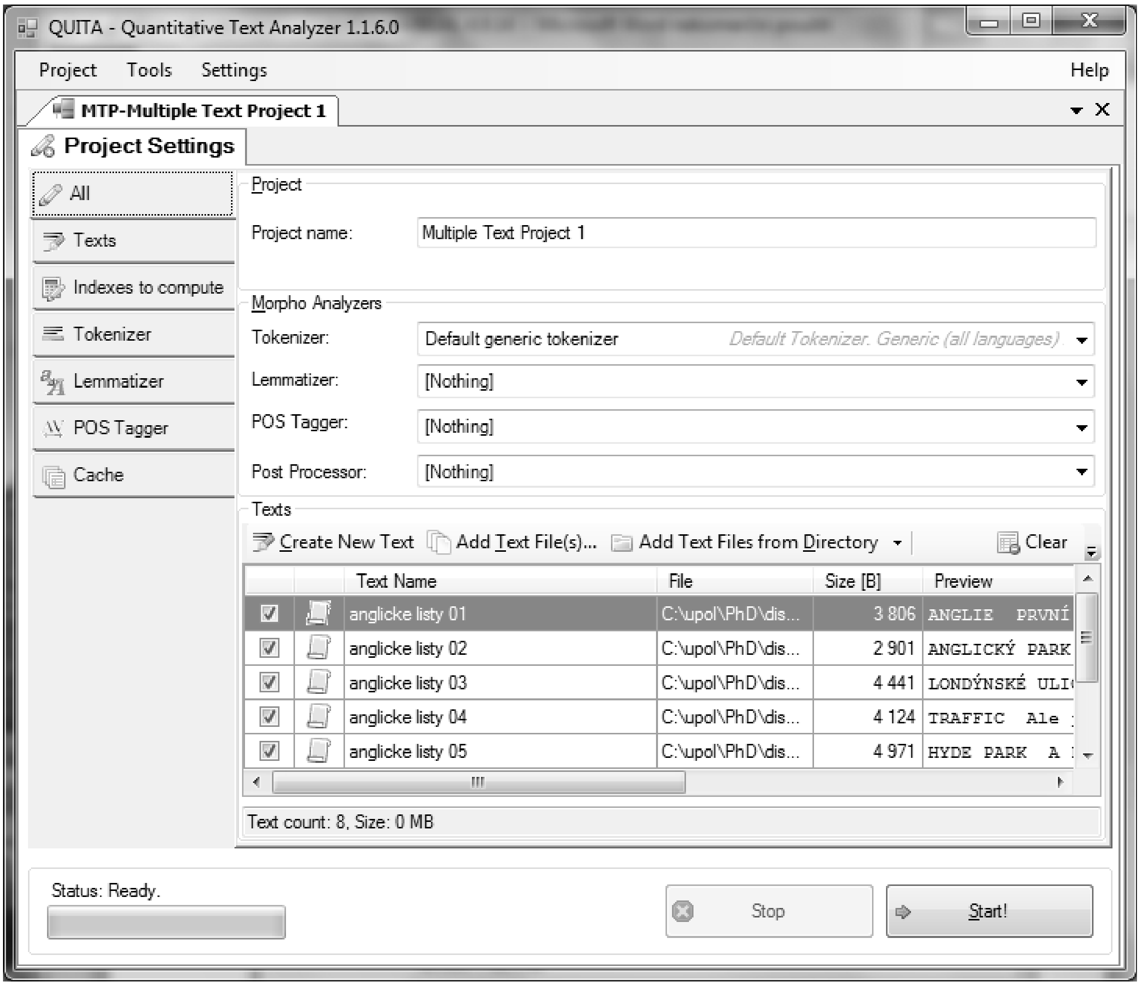
□ Cache Frequency Table (prevents from repeating time consuming calculations)

Status: Ready.

Every user must decide which cache settings are appropriate for the current study. Generally, it is recommended to use cache memory only when you work with few texts.

10. Starting Calculations

After selecting all the desired options, the computing process is ready to begin. Just click on the “Start!” button at the bottom right and wait until a new window with results opens.



Project name: Multiple Text Project 1

Tokenizer: Default generic tokenizer — Default Tokenizer. Generic (all languages)

Lemmatizer: [Nothing]

POS Tagger: [Nothing]

Post Processor: [Nothing]

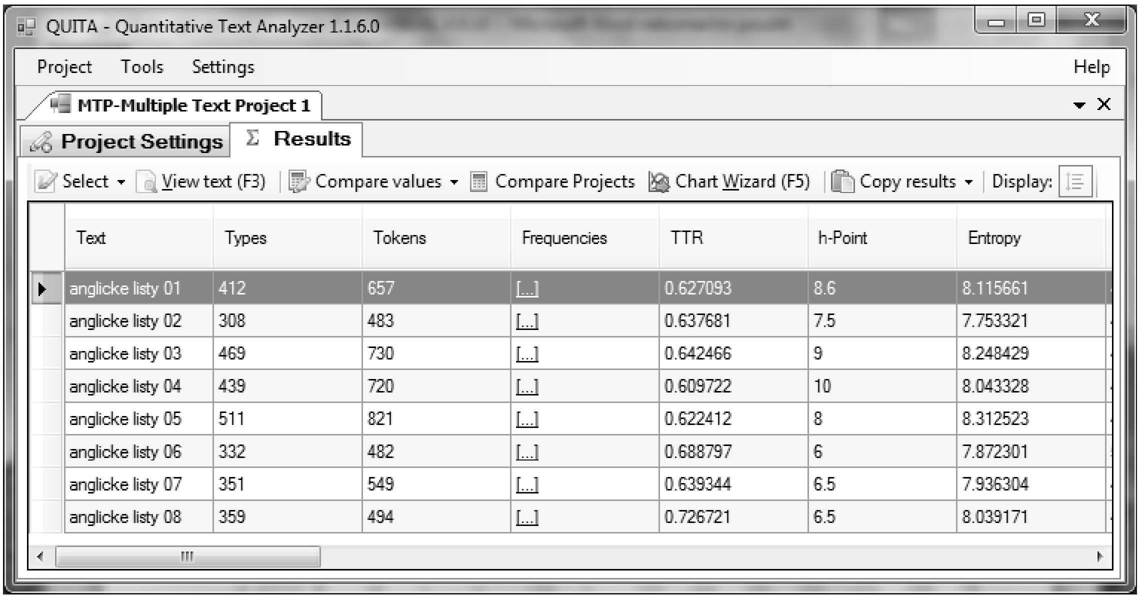
		Text Name	File	Size [B]	Preview
☑		anglicke listy 01	C:\upol\PhD\dis...	3 806	ANGLIE PRVNÍ
☑		anglicke listy 02	C:\upol\PhD\dis...	2 901	ANGLICKÝ PARK
☑		anglicke listy 03	C:\upol\PhD\dis...	4 441	LONDÝNSKÉ ULI
☑		anglicke listy 04	C:\upol\PhD\dis...	4 124	TRAFFIC Ale
☑		anglicke listy 05	C:\upol\PhD\dis...	4 971	HYDE PARK A

Text count: 8, Size: 0 MB

Status: Ready.

11. Results

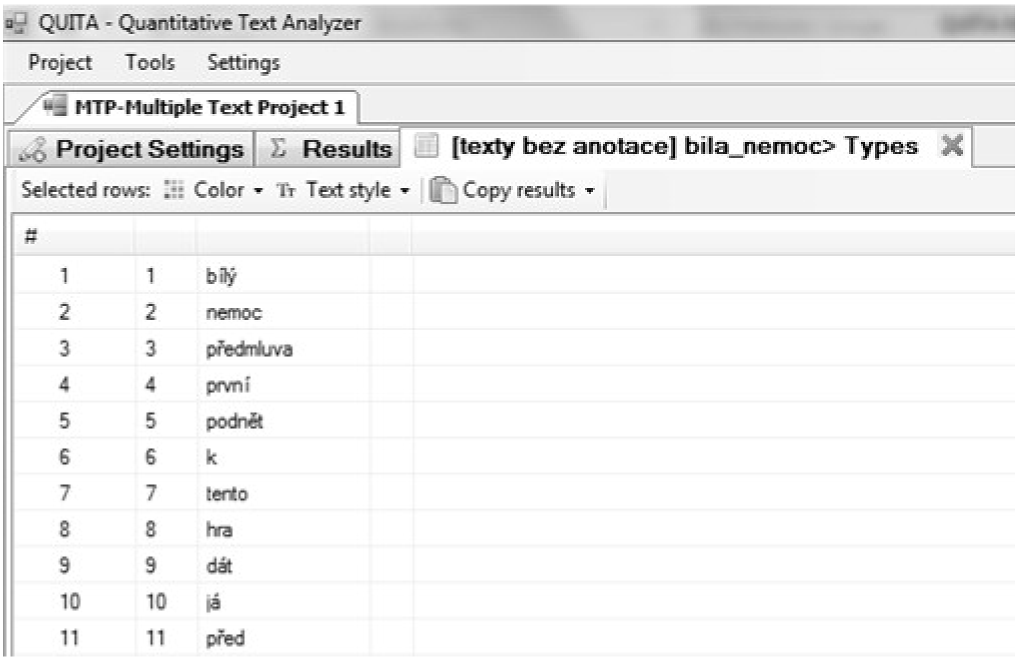
Once the computing process has finished, a window with the results will open. All the results are displayed in a table.



Text	Types	Tokens	Frequencies	TTR	h-Point	Entropy
anglicke listy 01	412	657	[...]	0.627093	8.6	8.115661
anglicke listy 02	308	483	[...]	0.637681	7.5	7.753321
anglicke listy 03	469	730	[...]	0.642466	9	8.248429
anglicke listy 04	439	720	[...]	0.609722	10	8.043328
anglicke listy 05	511	821	[...]	0.622412	8	8.312523
anglicke listy 06	332	482	[...]	0.688797	6	7.872301
anglicke listy 07	351	549	[...]	0.639344	6.5	7.936304
anglicke listy 08	359	494	[...]	0.726721	6.5	8.039171

11.1. Underlined Results alias "Alternative data"

The underlined results (also called"Alternative data") are data obtained from the calculation of given indices which might be interesting for the user to see. For example, the calculation of the "Types" index produces alternative data: a list of all types in given text. This list is then accessible by double-clicking on the underlined result of the Type index and is then displayed in a new tab page, as can be seen below:


QUITA - Quantitative Text Analyzer
Project
Tools
Settings
MTP-Multiple Text Project 1
Project Settings
Results
[texty bez anotace] bila_nemoc> Types
Selected rows:
Color
Text style
Copy results

#		
1	1	bílý
2	2	nemoc
3	3	předmluva
4	4	první
5	5	podnět
6	6	k
7	7	tento
8	8	hra
9	9	dát
10	10	já
11	11	před

While displaying alternative data, you can alternate text style ("Text style" menu) and background color ("Color" menu) of each line to mark interesting lines. You can also use the "Copy results" menu copy the whole table to the clipboard or export it to a CSV file and then import it to any table processor, such as Excel or OpenOffice Calc.

NOTES:

- The TAB character is used as delimiter for CSV files. The file encoding is set to UTF-8.
- For importing data to Excel, it is better to use the "Copy to Clipboard" option (and then just paste it by CTRL+V keyboard shortcut directly into an Excel table) rather than importing CSV files, which is problematic for some reason.

11.1.1. Types

The Alternative data of Types (count) index (as has been already mentioned) is a list of all types appearing in given text. The order of the displayed types and its indexation is based on the first appearance of given type: While the first word will be always displayed as the first type in this list, its next appearance is not listed again. Thus: Any list of types will likely follow the original order of words in the text while ignoring next occurrences of already listed types.

For better clarity, let's look at this sentence: "Fox jumps over the lazy dog. Fox is great." The Alternative data of Types index is a list of types:

1. Fox,
2. jumps,
3. over,
4. the,
5. lazy,
6. dog,
7. is,
8. great.

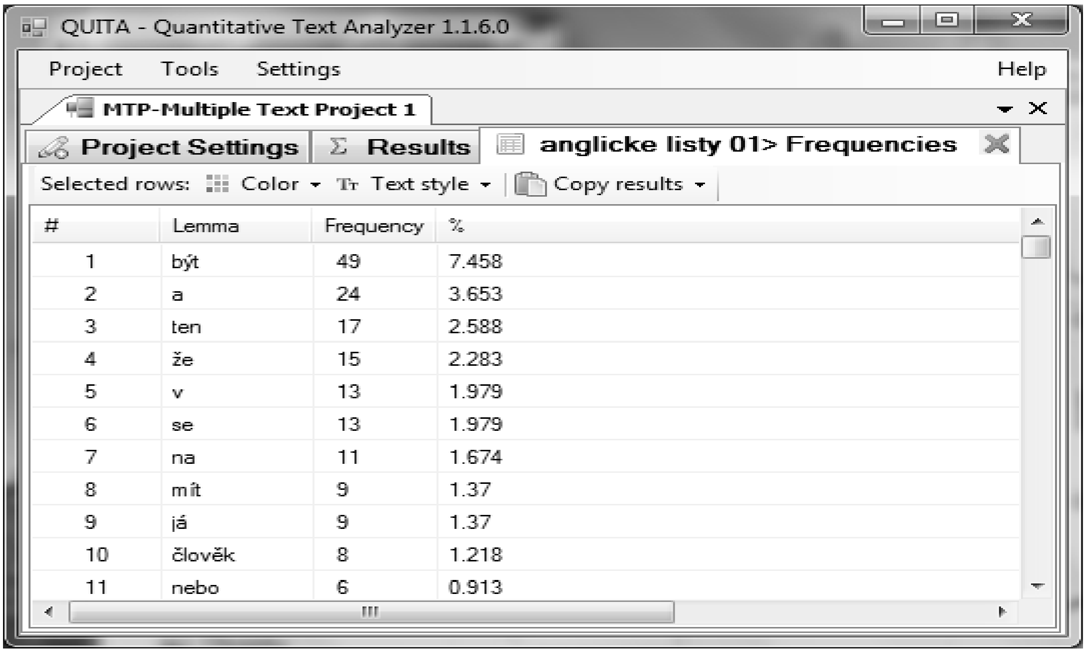
The type “Fox” has already been mentioned in the types list and is not listed again.

11.1.2. Tokens

The Alternative data of Tokens (count) index contain a list of tokens obtained from the given text. The order and indexation in the displayed list has the same order as the original text. The order can be mismatched, however, in the case when QUITA detects invalid tokens (e.g. numbers when the option “Treat numbers as words” is unticked; see 7.2.1 Available tokenizers for more details).

11.1.3. Frequencies

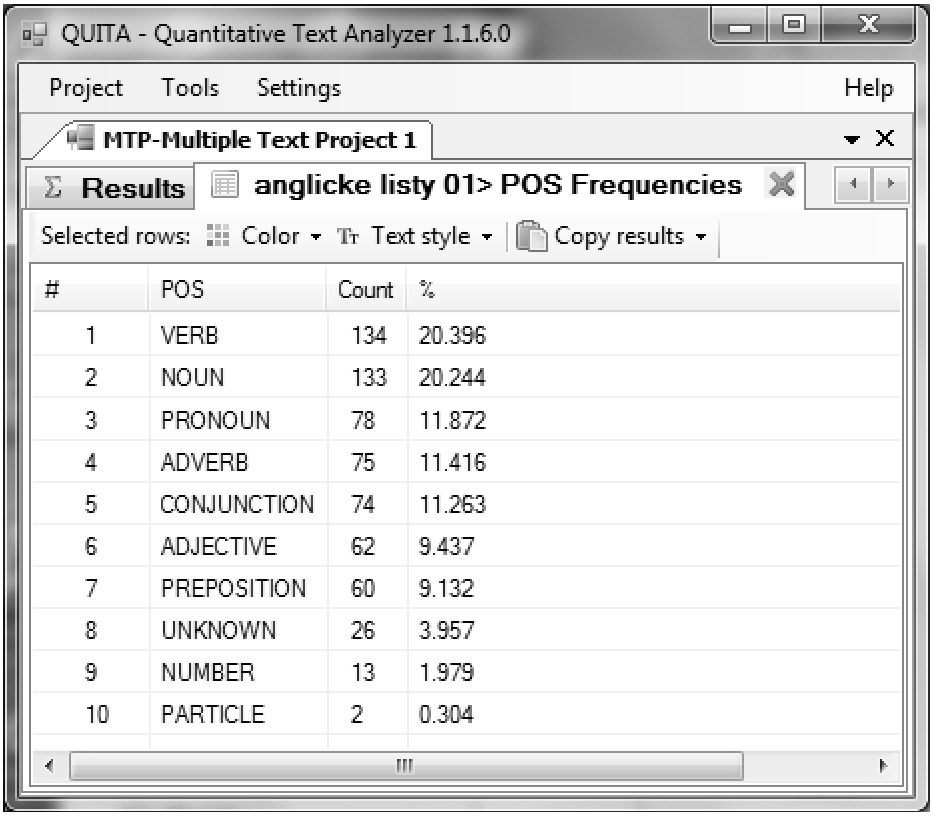
There can be displayed a list of types or lemmas in a text with rank, frequency and percentage of their occurrence (counted for the whole text).


QUITA - Quantitative Text Analyzer 1.1.6.0
Project
Tools
Settings
Help
MTP-Multiple Text Project 1
Project Settings
Σ Results
anglicke listy 01> Frequencies
Selected rows:
Color
Text style
Copy results

#	Lemma	Frequency	%
1	být	49	7.458
2	a	24	3.653
3	ten	17	2.588
4	že	15	2.283
5	v	13	1.979
6	se	13	1.979
7	na	11	1.674
8	mít	9	1.37
9	já	9	1.37
10	člověk	8	1.218
11	nebo	6	0.913

11.1.4. POS Frequencies

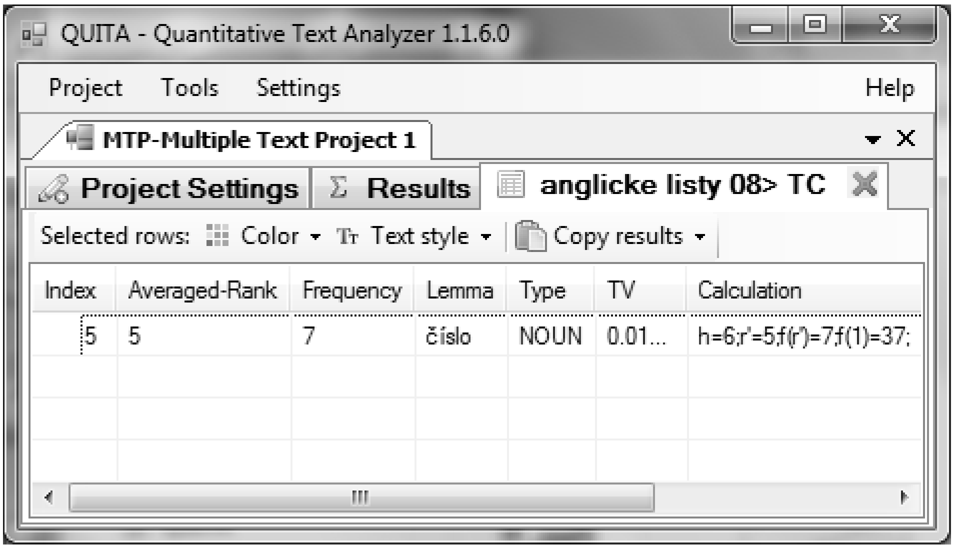
Similarly to Frequencies, POS Frequencies display a list of the parts of speech in a text with their frequency and percentage (counted for the whole text).


QUITA - Quantitative Text Analyzer 1.1.6.0
Project
Tools
Settings
Help
MTP-Multiple Text Project 1
Σ Results
anglicke listy 01> POS Frequencies
Selected rows:
Color
Text style
Copy results

#	POS	Count	%
1	VERB	134	20.396
2	NOUN	133	20.244
3	PRONOUN	78	11.872
4	ADVERB	75	11.416
5	CONJUNCTION	74	11.263
6	ADJECTIVE	62	9.437
7	PREPOSITION	60	9.132
8	UNKNOWN	26	3.957
9	NUMBER	13	1.979
10	PARTICLE	2	0.304

11.1.5. Thematic concentration

There is a list of so-called thematic words with their rank, average rank, frequency, part of speech, thematic weigh (*TW*), *h*, *r'*, *f*(*r'*) and *f*(1).



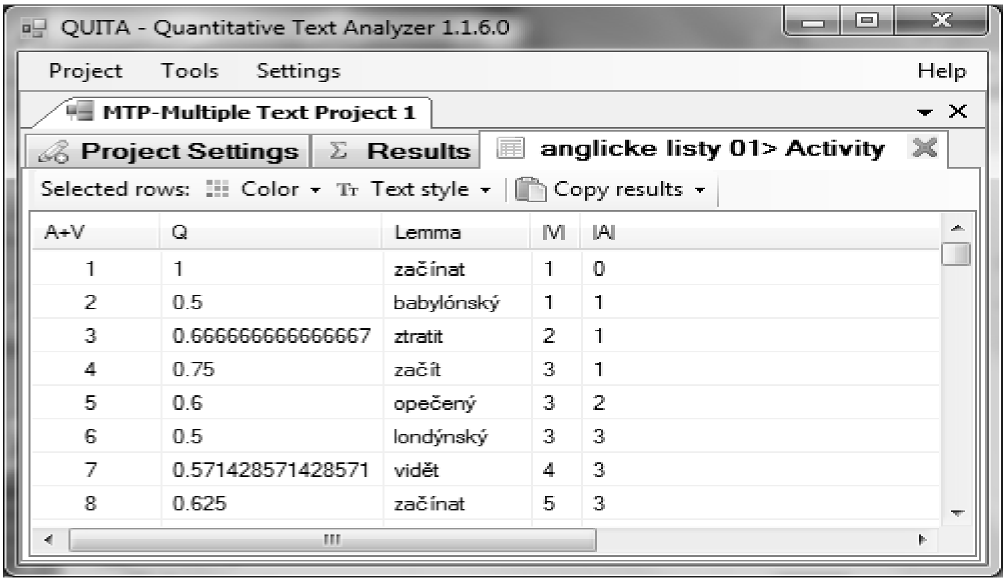
Index	Averaged-Rank	Frequency	Lemma	Type	TV	Calculation
5	5	7	číslo	NOUN	0.01...	h=6;r'=5;f(r')=7;f(1)=37;

11.1.6. Secondary Thematic Concentration

There are the same columns as in the “Thematic Concentration”. The secondary concentration differs only in the *h*-point (*h*) (see 6.1.2 h-point) which is multiplied by 2. Thus there are many more thematic words.

11.1.7. Activity

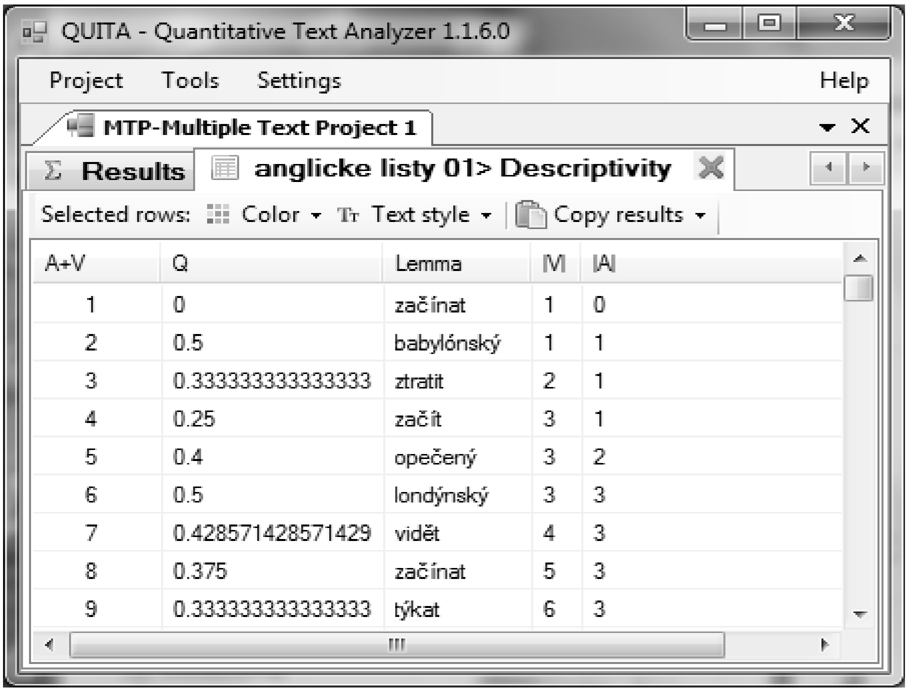
There is a list of verbs and adjectives in a text with the cumulative sums for Q, A+*V*, *V*, *A*.


QUITA - Quantitative Text Analyzer 1.1.6.0
Project
Tools
Settings
Help
MTP-Multiple Text Project 1
Project Settings
Σ Results
anglicke listy 01> Activity
Selected rows:
Color
Text style
Copy results

A+V	Q	Lemma	|V|	|A|
1	1	začínat	1	0
2	0.5	babylónský	1	1
3	0.666666666666667	ztratit	2	1
4	0.75	začít	3	1
5	0.6	opečený	3	2
6	0.5	londýnský	3	3
7	0.571428571428571	vidět	4	3
8	0.625	začínat	5	3

11.1.8. Descriptivity

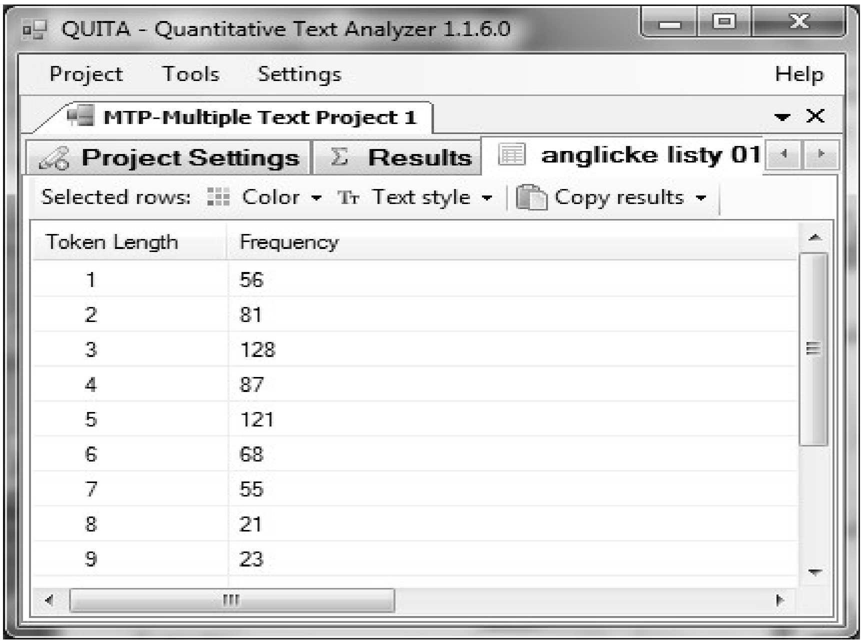
There is a list of verbs and adjectives in a text with the cumulative sums for *D*, A+*V*, *V*, *A*.


QUITA - Quantitative Text Analyzer 1.1.6.0
Project
Tools
Settings
Help
MTP-Multiple Text Project 1
Results
anglicke listy 01> Descriptivity
Selected rows:
Color
Text style
Copy results

A+V	Q	Lemma	|V|	|A|
1	0	začínat	1	0
2	0.5	babylónský	1	1
3	0.333333333333333	ztratit	2	1
4	0.25	začít	3	1
5	0.4	opečený	3	2
6	0.5	londýnský	3	3
7	0.428571428571429	vidět	4	3
8	0.375	začínat	5	3
9	0.333333333333333	týkat	6	3

11.1.9. Token Length Frequency Spectrum

There is a list of all token lengths in a text with their frequency.

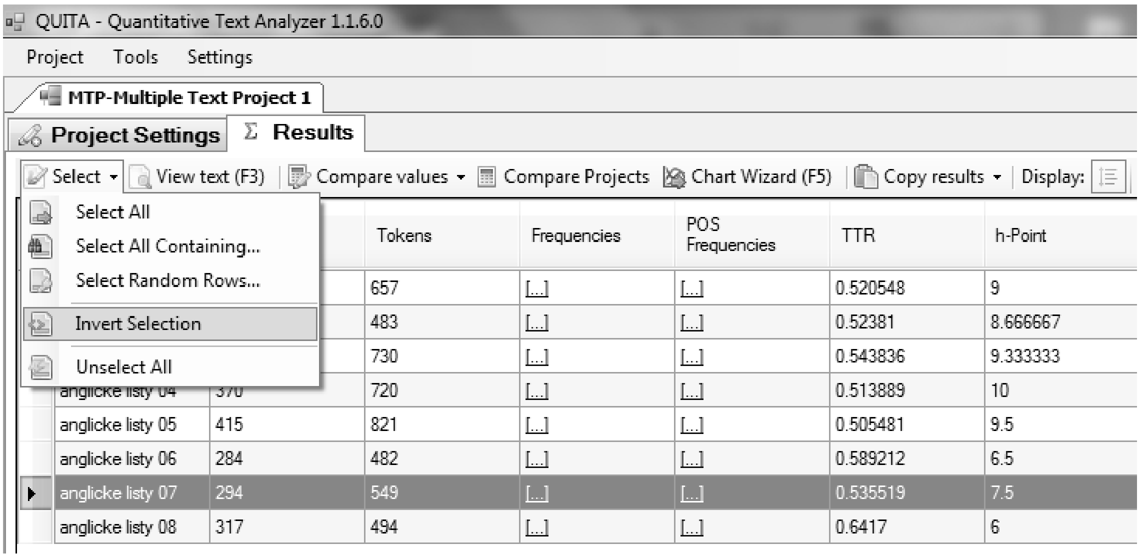


Token Length	Frequency
1	56
2	81
3	128
4	87
5	121
6	68
7	55
8	21
9	23

11.2. Results tools

11.2.1. Text Selection

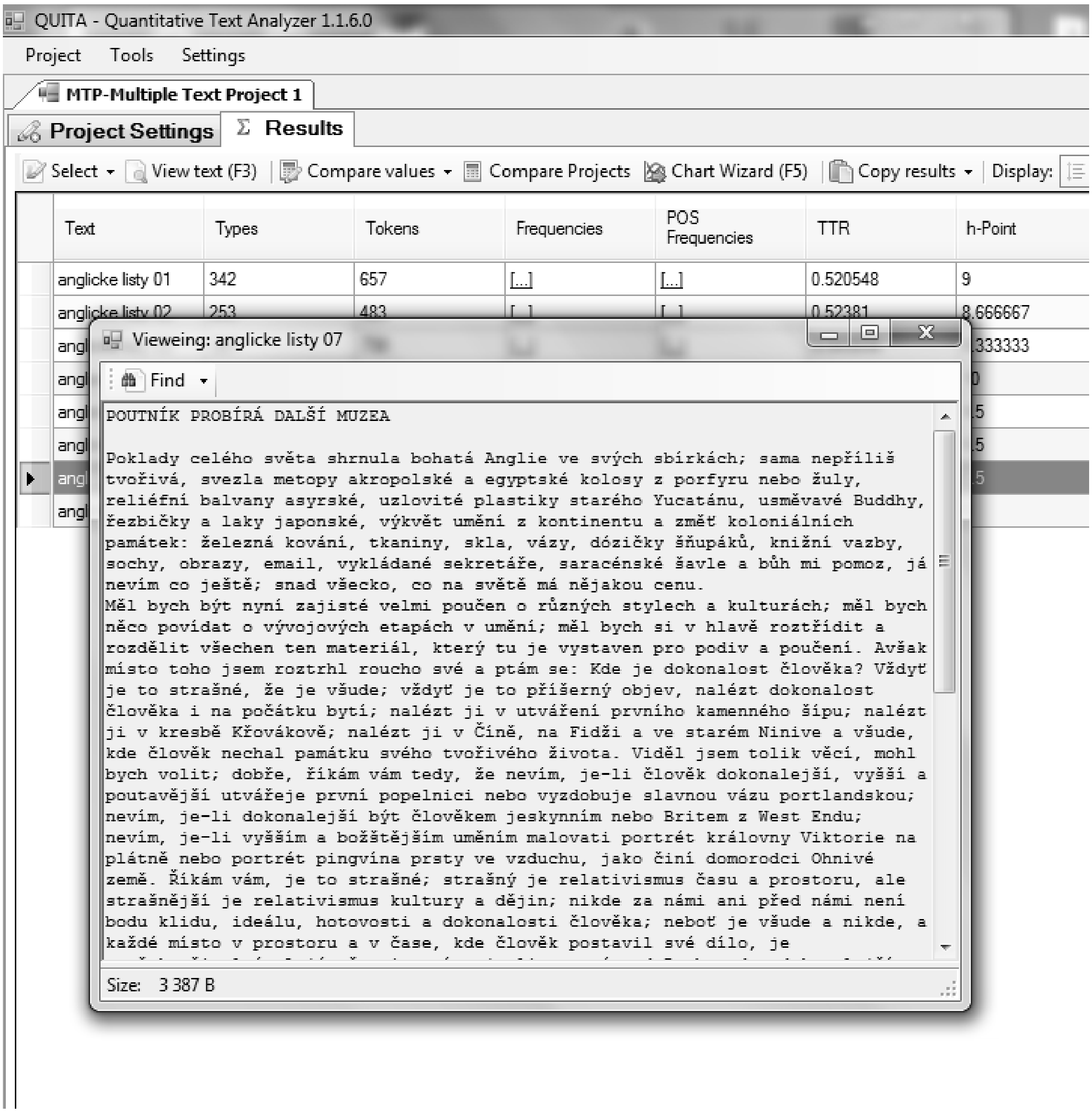
To use some functions in the toolbar, it is necessary to select the rows first. You can just click on the required row and use the mouse click while holding CTRL to select more rows, or use the common "CTRL+A" keyboard shortcut to select all rows. The next option is to use any selection tool from the "Select" menu in the results toolbar, where you can use more advanced selections like "Select All Containing…" or "Select Random Rows…" which selects random rows to a given count specified by the user.



		Tokens	Frequencies	POS Frequencies	TTR	h-Point
		657	[...]	[...]	0.520548	9
		483	[...]	[...]	0.52381	8.666667
		730	[...]	[...]	0.543836	9.333333
anglicke listy 04	370	720	[...]	[...]	0.513889	10
anglicke listy 05	415	821	[...]	[...]	0.505481	9.5
anglicke listy 06	284	482	[...]	[...]	0.589212	6.5
anglicke listy 07	294	549	[...]	[...]	0.535519	7.5
anglicke listy 08	317	494	[...]	[...]	0.6417	6

11.2.2. View Text

Click on "View Text" or just press F3 to display the original content of the selected text or, in the case that the text has been post-processed (eg. by n-grammization), displays the post-processed text.



Text	Types	Tokens	Frequencies	POS Frequencies	TTR	h-Point
anglicke listy 01	342	657	[...]	[...]	0.520548	9
anglicke listy 02	253	483	[...]	[...]	0.52381	8.666667
angl						.333333
angl						0
angl						.5
angl						.5
ang						.5
angl						

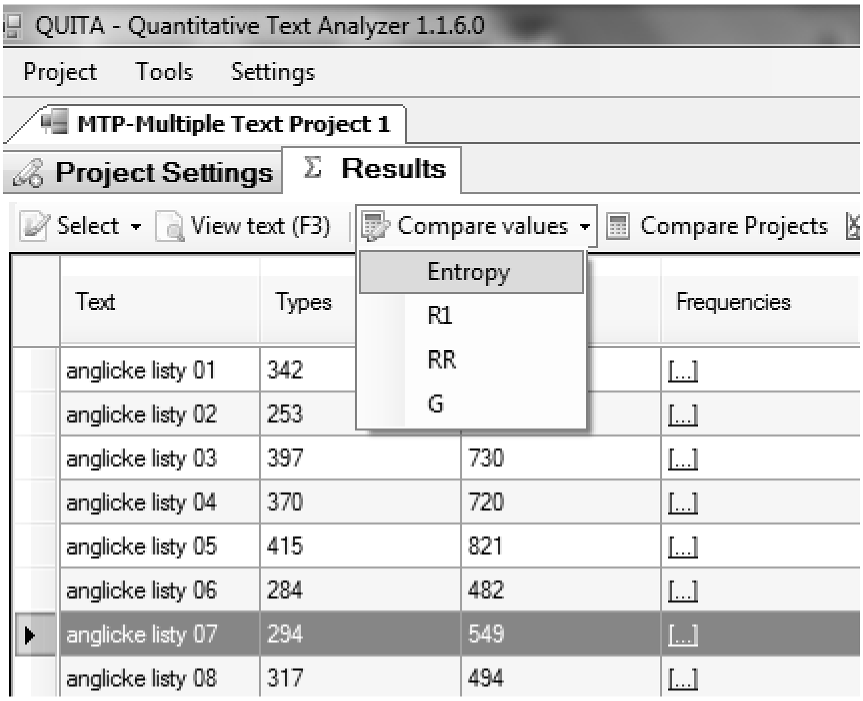
```
POUTNÍK PROBÍRÁ DALŠÍ MUZEA

Poklady celého světa shrnula bohatá Anglie ve svých sbírkách; sama nepříliš
tvořivá, svezla metopy akropolské a egyptské kolosy z porfyru nebo žuly,
reliéfní balvany asyrské, uzlovité plastiky starého Yucatánu, usměvavé Buddhy,
řezbičky a laky japonské, výkvět umění z kontinentu a změť koloniálních
památek: železná kování, tkaniny, skla, vázy, dózičky šňupáků, knižní vazby,
sochy, obrazy, email, vykládané sekretáře, saracénské šavle a bůh mi pomoz, já
nevím co ještě; snad všecko, co na světě má nějakou cenu.
Měl bych být nyní zajisté velmi poučen o různých stylech a kulturách; měl bych
něco povídat o vývojových etapách v umění; měl bych si v hlavě roztřídit a
rozdělit všechen ten materiál, který tu je vystaven pro podiv a poučení. Avšak
místo toho jsem roztrhl roucho své a ptám se: Kde je dokonalost člověka? Vždyť
je to strašné, že je všude; vždyť je to příšerný objev, nalézt dokonalost
člověka i na počátku bytí; nalézt ji v utváření prvního kamenného šípu; nalézt
ji v kresbě Křovákově; nalézt ji v Číně, na Fidži a ve starém Ninive a všude,
kde člověk nechal památku svého tvořivého života. Viděl jsem tolik věcí, mohl
bych volit; dobře, říkám vám tedy, že nevím, je-li člověk dokonalejší, vyšší a
poutavější utvářeje první popelnici nebo vyzdobuje slavnou vázu portlandskou;
nevím, je-li dokonalejší být člověkem jeskynním nebo Britem z West Endu;
nevím, je-li vyšším a božštějším uměním malovati portrét královny Viktorie na
plátně nebo portrét pingvína prsty ve vzduchu, jako činí domorodci Ohnivé
země. Říkám vám, je to strašné; strašný je relativismus času a prostoru, ale
strašnější je relativismus kultury a dějin; nikde za námi ani před námi není
bodu klidu, ideálu, hotovosti a dokonalosti člověka; neboť je všude a nikde, a
každé místo v prostoru a v čase, kde člověk postavil své dílo, je
```

Size: 3 387 B

11.2.3. Comparison of Results

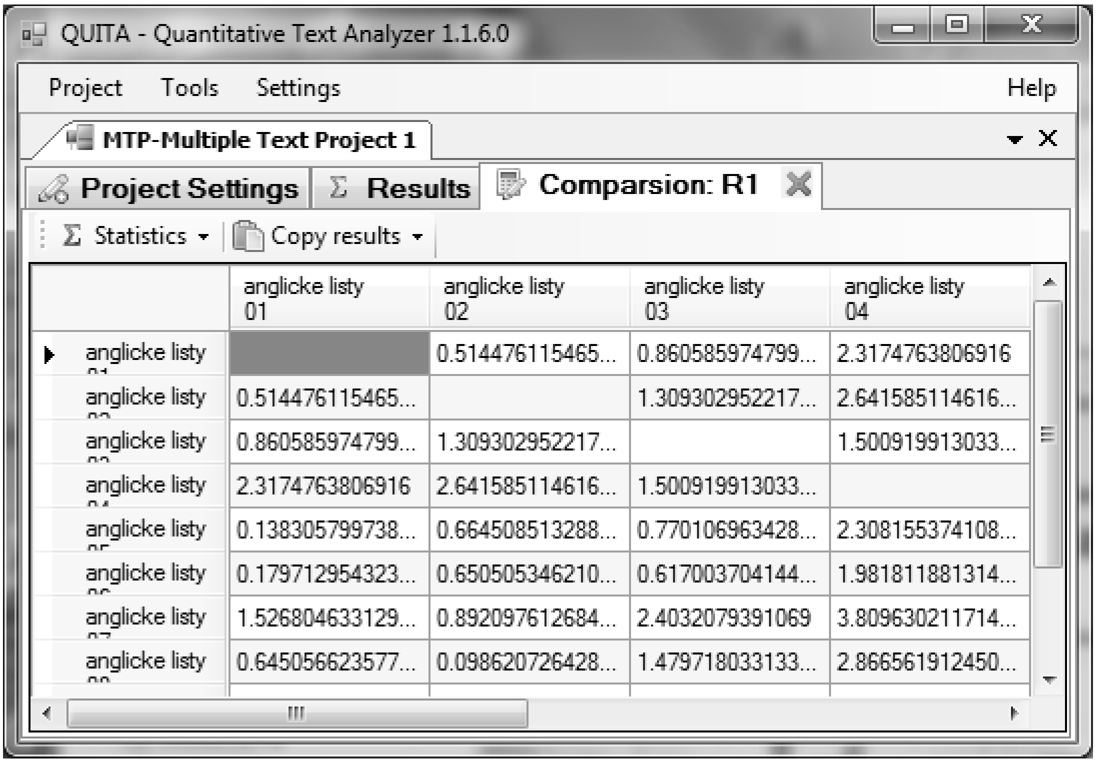
QUITA provides an option to compare the resulting values between two texts. Comparison is based on statistical test with the default significance level 0.05. The following indicators can be statistically tested: Entropy, R_1, *RR*, *TC*, STC, PTC, Activity, Descriptivity, *G*.

Click on "Compare values" and choose a required indicator.



	Text	Types		Frequencies
	anglicke listy 01	342		[...]
	anglicke listy 02	253		[...]
	anglicke listy 03	397	730	[...]
	anglicke listy 04	370	720	[...]
	anglicke listy 05	415	821	[...]
	anglicke listy 06	284	482	[...]
▶	anglicke listy 07	294	549	[...]
	anglicke listy 08	317	494	[...]

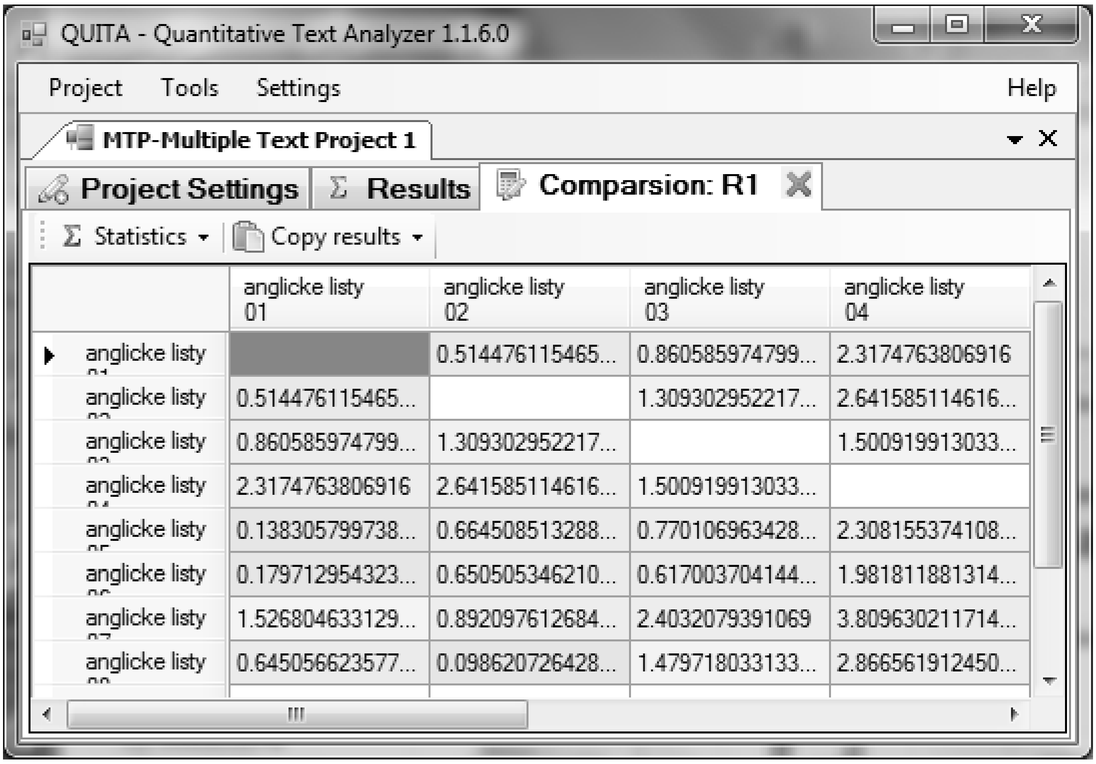
A table with results of the test will open. In the table, you will also find average values and values of the standard deviation.



	anglicke listy 01	anglicke listy 02	anglicke listy 03	anglicke listy 04
▶ anglicke listy		0.514476115465...	0.860585974799...	2.3174763806916
anglicke listy	0.514476115465...		1.309302952217...	2.641585114616...
anglicke listy	0.860585974799...	1.309302952217...		1.500919913033...
anglicke listy	2.3174763806916	2.641585114616...	1.500919913033...	
anglicke listy	0.138305799738...	0.664508513288...	0.770106963428...	2.308155374108...
anglicke listy	0.179712954323...	0.650505346210...	0.617003704144...	1.981811881314...
anglicke listy	1.526804633129...	0.892097612684...	2.4032079391069	3.809630211714...
anglicke listy	0.645056623577...	0.098620726428...	1.479718033133...	2.866561912450...

For better clarity, data can be coloured. Click on the “Statistics” menu and “Colorize” and the obtained values will be coloured by three colours:

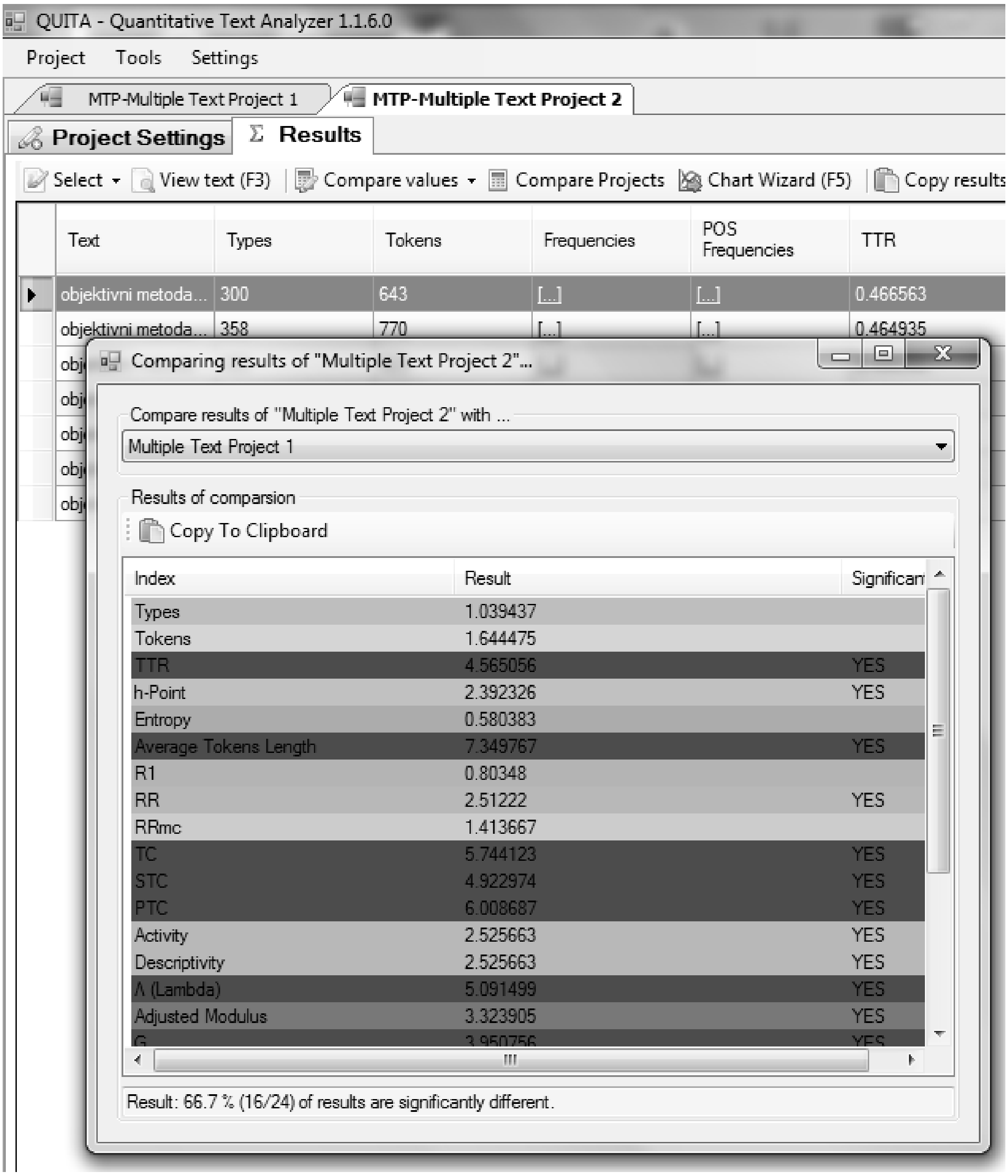
- Green = there is no significant difference between two texts.
- Orange = there is a significant difference between two texts.
- Yellow = the resulting value is on the border between significant and non-significant difference between two texts.



	anglicke listy 01	anglicke listy 02	anglicke listy 03	anglicke listy 04
anglicke listy		0.514476115465...	0.860585974799...	2.3174763806916
anglicke listy	0.514476115465...		1.309302952217...	2.641585114616...
anglicke listy	0.860585974799...	1.309302952217...		1.500919913033...
anglicke listy	2.3174763806916	2.641585114616...	1.500919913033...	
anglicke listy	0.138305799738...	0.664508513288...	0.770106963428...	2.308155374108...
anglicke listy	0.179712954323...	0.650505346210...	0.617003704144...	1.981811881314...
anglicke listy	1.526804633129...	0.892097612684...	2.4032079391069	3.809630211714...
anglicke listy	0.645056623577...	0.098620726428...	1.479718033133...	2.866561912450...

11.2.4. Comparison of Projects

QUITA can also test pairs of projects (i.e. two groups of texts). For example, you may want to compare the styles of two authors. The first author wrote five novels and the second one wrote eight novels. You need to create two projects (one for each of them) and let QUITA create its results. When you have the results of the required indicators, it is possible to compare the two projects. Click on “Compare Projects” and you will get a colorized table with the resulting values of the statistical test.



Text	Types	Tokens	Frequencies	POS Frequencies	TTR
objektivni metoda...	300	643	[...]	[...]	0.466563
objektivni metoda...	358	770	[...]	[...]	0.464935
obj					
obj					
obj					
obj					
obj					

Comparing results of "Multiple Text Project 2"...

Compare results of "Multiple Text Project 2" with ...

Multiple Text Project 1

Results of comparsion

Index	Result	Significan
Types	1.039437	
Tokens	1.644475	
TTR	4.565056	YES
h-Point	2.392326	YES
Entropy	0.580383	
Average Tokens Length	7.349767	YES
R1	0.80348	
RR	2.51222	YES
RRmc	1.413667	
TC	5.744123	YES
STC	4.922974	YES
PTC	6.008687	YES
Activity	2.525663	YES
Descriptivity	2.525663	YES
Λ (Lambda)	5.091499	YES
Adjusted Modulus	3.323905	YES
G	3.950756	YES

Result: 66.7 % (16/24) of results are significantly different.

The formula of *u*-test is defined as:

$$u=\frac{\left|\acute{X}_1-\acute{X}_2\right|}{\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}}$$

$\acute{X}_1$, $\acute{X}_2$...arithmetic mean of results in each group
S_1, S_2…standard deviation
n_1, n_2…number of results in each group

Example

We would like to compare the style of two collections of poems written by two different authors (namely the Czech poets František Gellner and Jan Skácel) in terms of vocabulary richness indicator R_1. So we want to discover differences between whole collections. We cannot therefore use a statistical test for a pair of texts because we need to compare two groups with different numbers of texts. For this purpose, we must use the average values over the individual texts.

In our case, Gellner wrote 19 poems and Skácel wrote 20 poems.

This comparison can be performed by QUITA using two projects. The first project contains 19 Gellner's poems and the second one contains 20 Skácel's poems. Then we can find out whether the difference between two collections of poems is significant by clicking on "Compare Projects".

The calculation by hand is as follows:

We must firstly compute the R_1 results of individual poems, average values and standard deviation values.

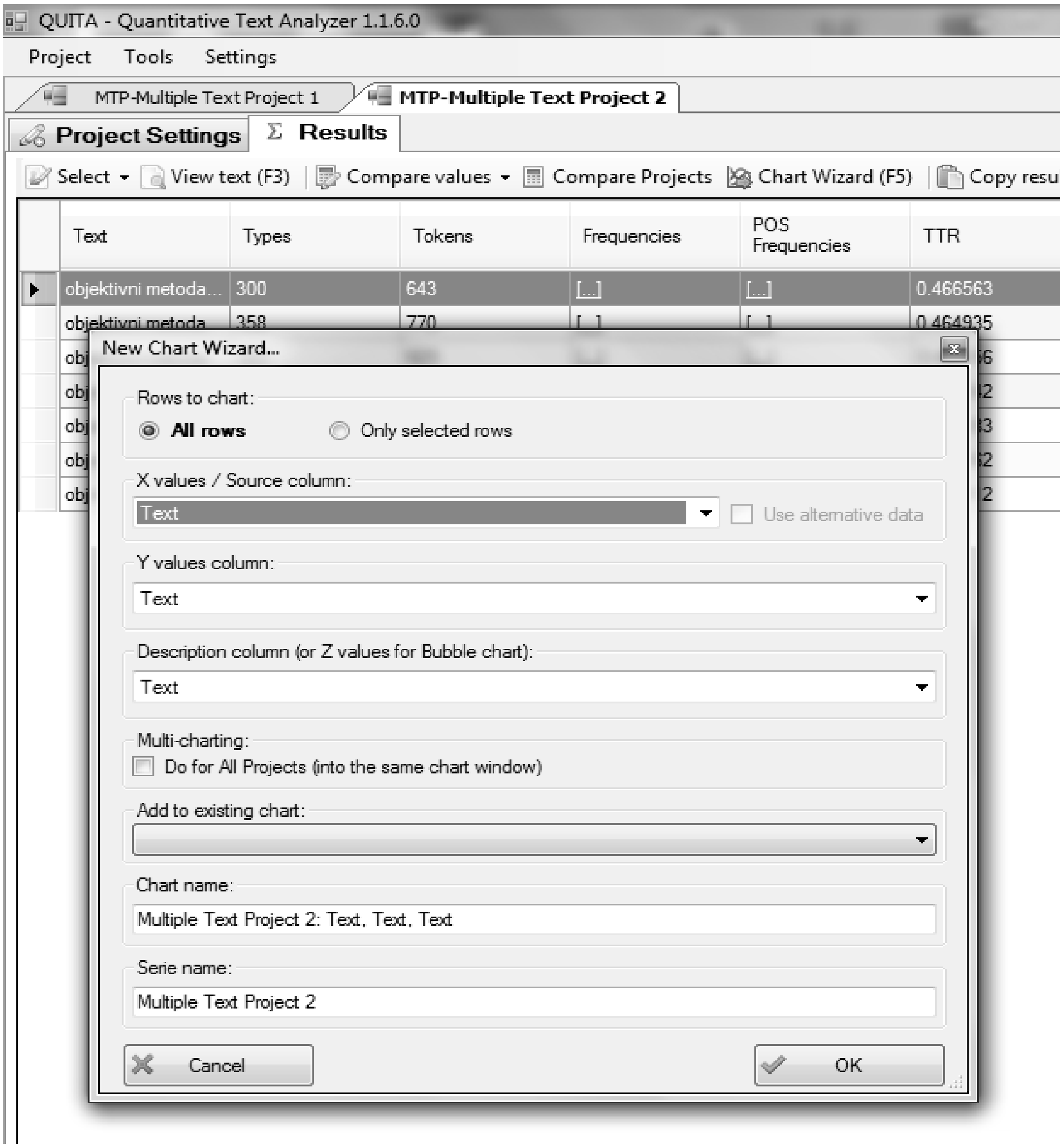
	Gellner	Skácel
1	0.886875	0.946078
2	0.945545	0.945755
3	0.877273	0.963636
4	0.937500	0.900510
5	0.949219	0.889831
6	0.912568	0.875000
7	0.896126	0.936475
8	0.950431	0.875000
9	0.910928	0.941327
10	0.895105	0.952161
11	0.894353	0.892405
12	0.927185	0.960000
13	0.903378	0.950000
14	0.760870	0.893617
15	0.911765	0.892157
16	0.889610	0.890805
17	0.818824	0.962264
18	0.824675	0.890046
19	0.915441	0.945455
20		0.942308
$\boldsymbol{X}$	**0.89514059**	**0.92224144**
$\boldsymbol{s}$	**0.04695488**	**0.03135431**

Then we can compare two groups of texts by the *u*-test.

$$u=\frac{|\acute{X}_1-\acute{X}_2|}{\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}}=\frac{|0.89514059-0.92224144|}{\sqrt{\frac{0.04695488^2}{19}+\frac{0.03135431^2}{20}}}=2.06$$

11.2.5. Chart Wizard

QUITA includes a tool for creating charts from the results. Click on the “Chart Wizard” or press F5 to create a new chart.



Text	Types	Tokens	Frequencies	POS Frequencies	TTR
objektivni metoda...	300	643	[...]	[...]	0.466563
objektivni metoda	358	770	[...]	[...]	0.464935
obj					56
obj					12
obj					13
obj					52
obj					12

New Chart Wizard...

Rows to chart: ● All rows ○ Only selected rows

X values / Source column: Text ☐ Use alternative data

Y values column: Text

Description column (or Z values for Bubble chart): Text

Multi-charting: ☐ Do for All Projects (into the same chart window)

Add to existing chart:

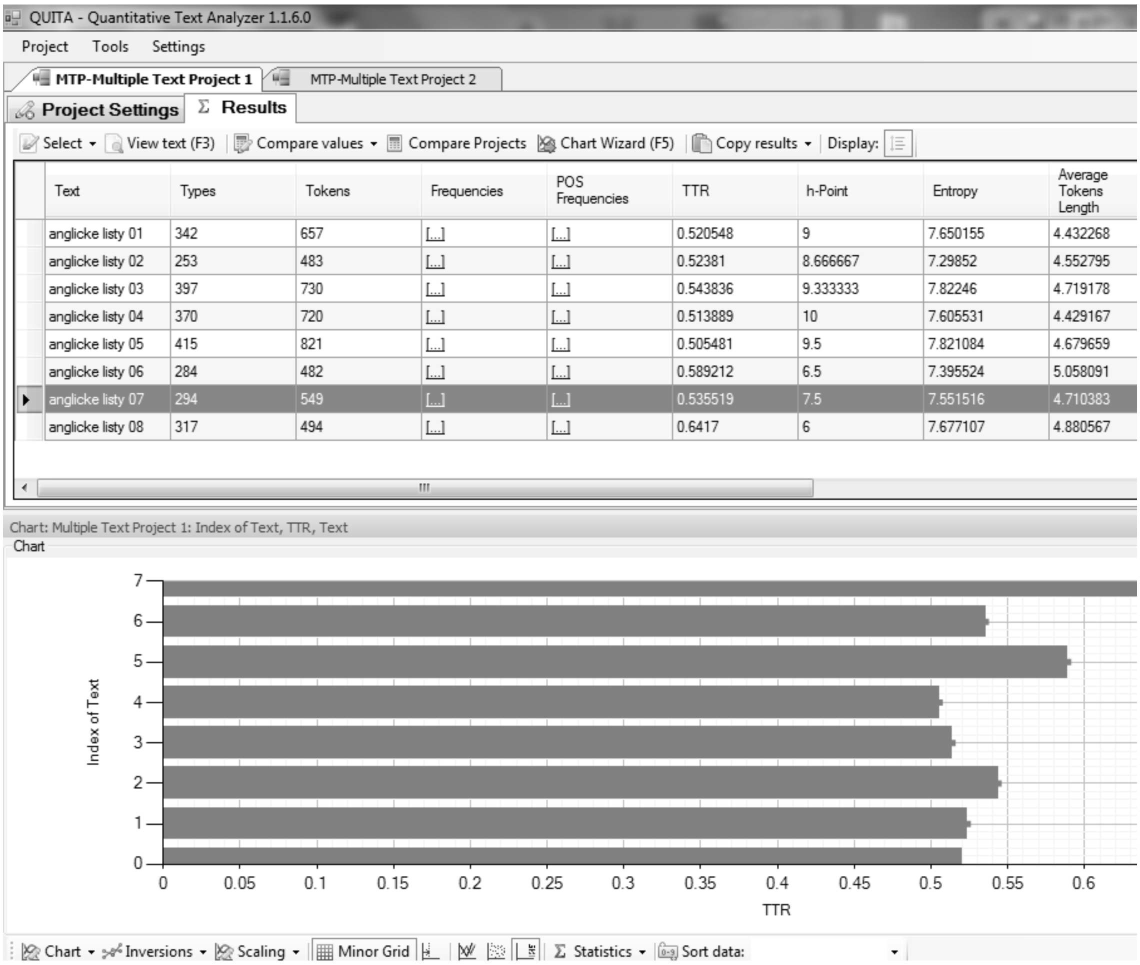
Chart name: Multiple Text Project 2: Text, Text, Text

Serie name: Multiple Text Project 2

In the dialog box, you can select which data to display in the graph. You can:

- Select which rows to use.
- Select data for the x, y and z axes.
- Select “Use alternative data”, to use data which are displayed when you double click on underlined values in the results table.
- Tick “Multi-charting” to merge data from all the projects into one graph.
- Add a new chart to one of the already existing graphs.
- Enter the chart name in the penultimate box.
- Enter the series name in the last box.

When all the required options are selected, press “OK” to create the graph.

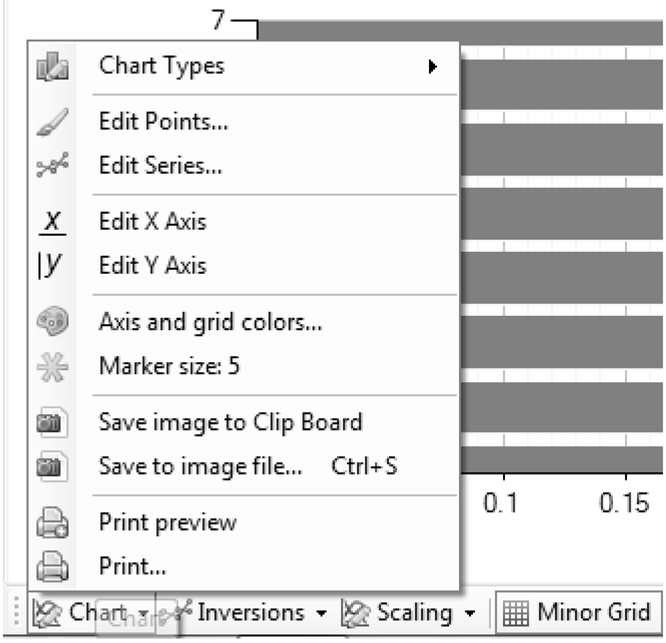


Text	Types	Tokens	Frequencies	POS Frequencies	TTR	h-Point	Entropy	Average Tokens Length
anglicke listy 01	342	657	[...]	[...]	0.520548	9	7.650155	4.432268
anglicke listy 02	253	483	[...]	[...]	0.52381	8.666667	7.29852	4.552795
anglicke listy 03	397	730	[...]	[...]	0.543836	9.333333	7.82246	4.719178
anglicke listy 04	370	720	[...]	[...]	0.513889	10	7.605531	4.429167
anglicke listy 05	415	821	[...]	[...]	0.505481	9.5	7.821084	4.679659
anglicke listy 06	284	482	[...]	[...]	0.589212	6.5	7.395524	5.058091
anglicke listy 07	294	549	[...]	[...]	0.535519	7.5	7.551516	4.710383
anglicke listy 08	317	494	[...]	[...]	0.6417	6	7.677107	4.880567

Chart: Multiple Text Project 1: Index of Text, TTR, Text

The chart wizard provides plenty of options for displaying results. There are options in the toolbar at the bottom for what to do with the chart:

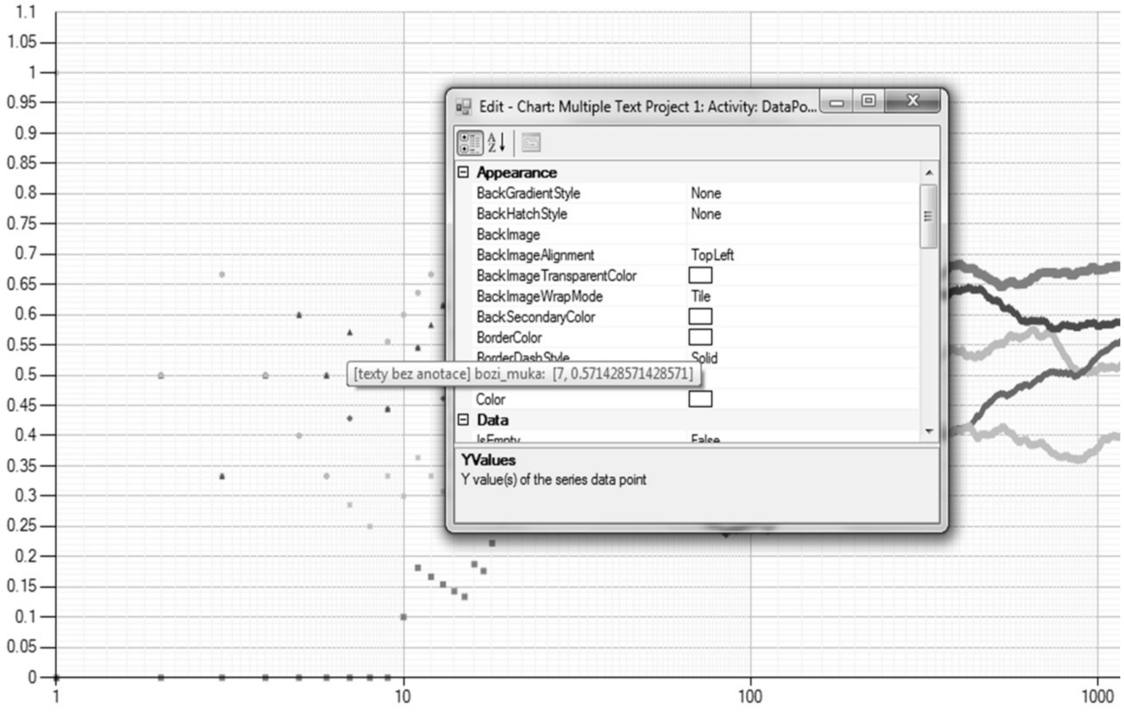
Click on “Chart” to adjust basic features:



In the resulting menu, you can:

- Choose the type of chart
- Edit points
- Edit series
- Edit the X axis
- Edit the Y axis
- Adjust the axis and grid colours
- Select the marker size
- Save the image to the clipboard
- Save image to your PC as an EMF, PNG or GIF file (EMF is vector file format, PNG and GIF are not vector file formats)
- See the print preview
- Print.

NOTE: You can also edit any desired point by simple double clicking on it directly on the graph.



Edit - Chart: Multiple Text Project 1: Activity: DataPo...

Appearance

BackGradientStyle	None
BackHatchStyle	None
BackImage	
BackImageAlignment	TopLeft
BackImageTransparentColor	□
BackImageWrapMode	Tile
BackSecondaryColor	□
BorderColor	□
BorderDashStyle	Solid
Color	□

Data

IsEmpty	False

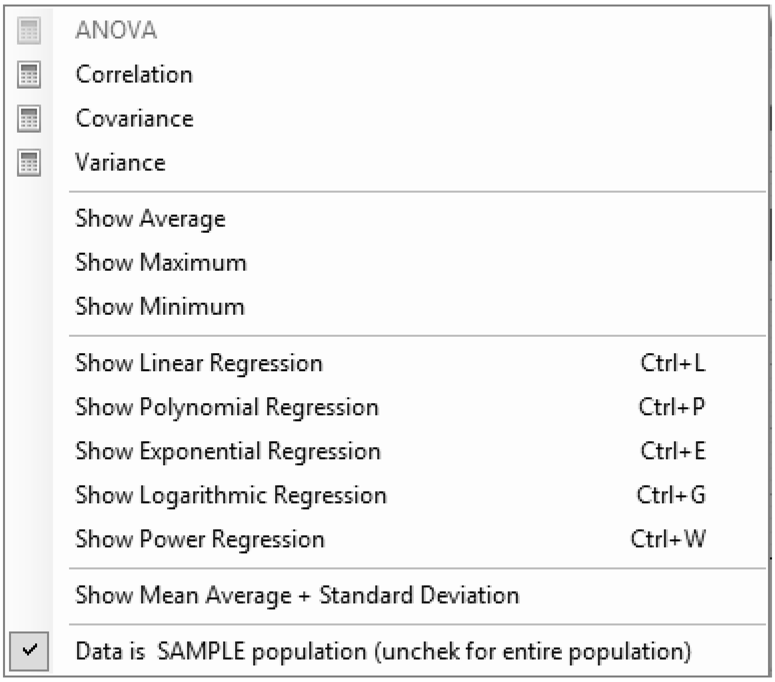
YValues

Y value(s) of the series data point

[texty bez anotace] bozi_muka: [7, 0.571428571428571]

The toolbar at the bottom of the chart window provides the most widely used settings:

- Click on “Inversions” to reverse or invert axes.
- Click on “Scaling” to logarithm axis or fit maximum and minimum of the chart.
- Click on “Minor Grid” to show or hide minor grid.
- Click on to show margin.
- Click on to choose line chart.
- Click on to choose point chart.
- Click on to display legend.
- Click on “Statistics” to open a menu with the following statistical functions:


ANOVA
Correlation
Covariance
Variance
Show Average
Show Maximum
Show Minimum
Show Linear Regression Ctrl+L
Show Polynomial Regression Ctrl+P
Show Exponential Regression Ctrl+E
Show Logarithmic Regression Ctrl+G
Show Power Regression Ctrl+W
Show Mean Average + Standard Deviation
✓ Data is SAMPLE population (unchek for entire population)

- Click on "Sort data" to sort data in descending or ascending order.

11.2.6. Copy Results

There are two ways to copy the results. Click on "Copy results" and then choose: "Copy Grid to Clipboard" or "Export Grid CSV File".

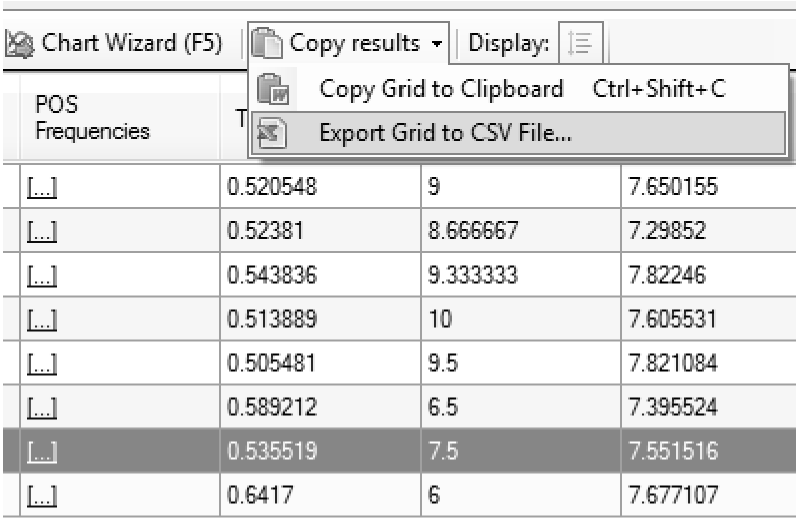


Chart Wizard (F5)
Copy results
Display:
Copy Grid to Clipboard Ctrl+Shift+C
Export Grid to CSV File...

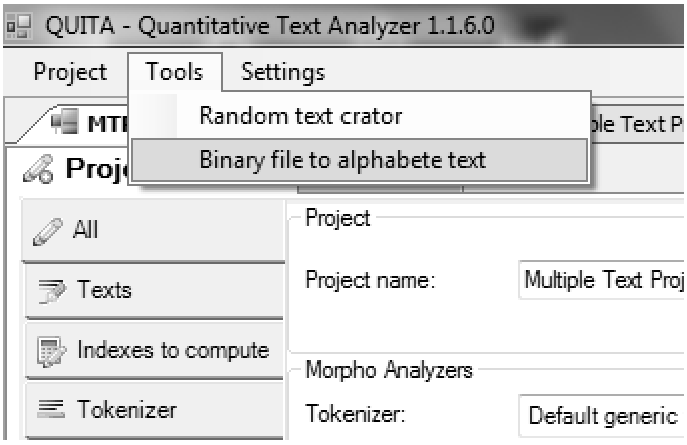
POS Frequencies	T		
[...]	0.520548	9	7.650155
[...]	0.52381	8.666667	7.29852
[...]	0.543836	9.333333	7.82246
[...]	0.513889	10	7.605531
[...]	0.505481	9.5	7.821084
[...]	0.589212	6.5	7.395524
[...]	0.535519	7.5	7.551516
[...]	0.6417	6	7.677107

NOTES:

- The TAB character is used as delimiter for CSV files. File encoding is set to UTF-8.
- For importing data to Excel, it is better to use the “Copy to Clipboard” option (and then just paste it by the CTRL+V keyboard shortcut directly to an Excel table) rather than importing CSV files, which is problematic for some reason.

12. Tools

QUITA actually provides two advanced tools: “Random text creator” and “Binary file to alphabetic text”.



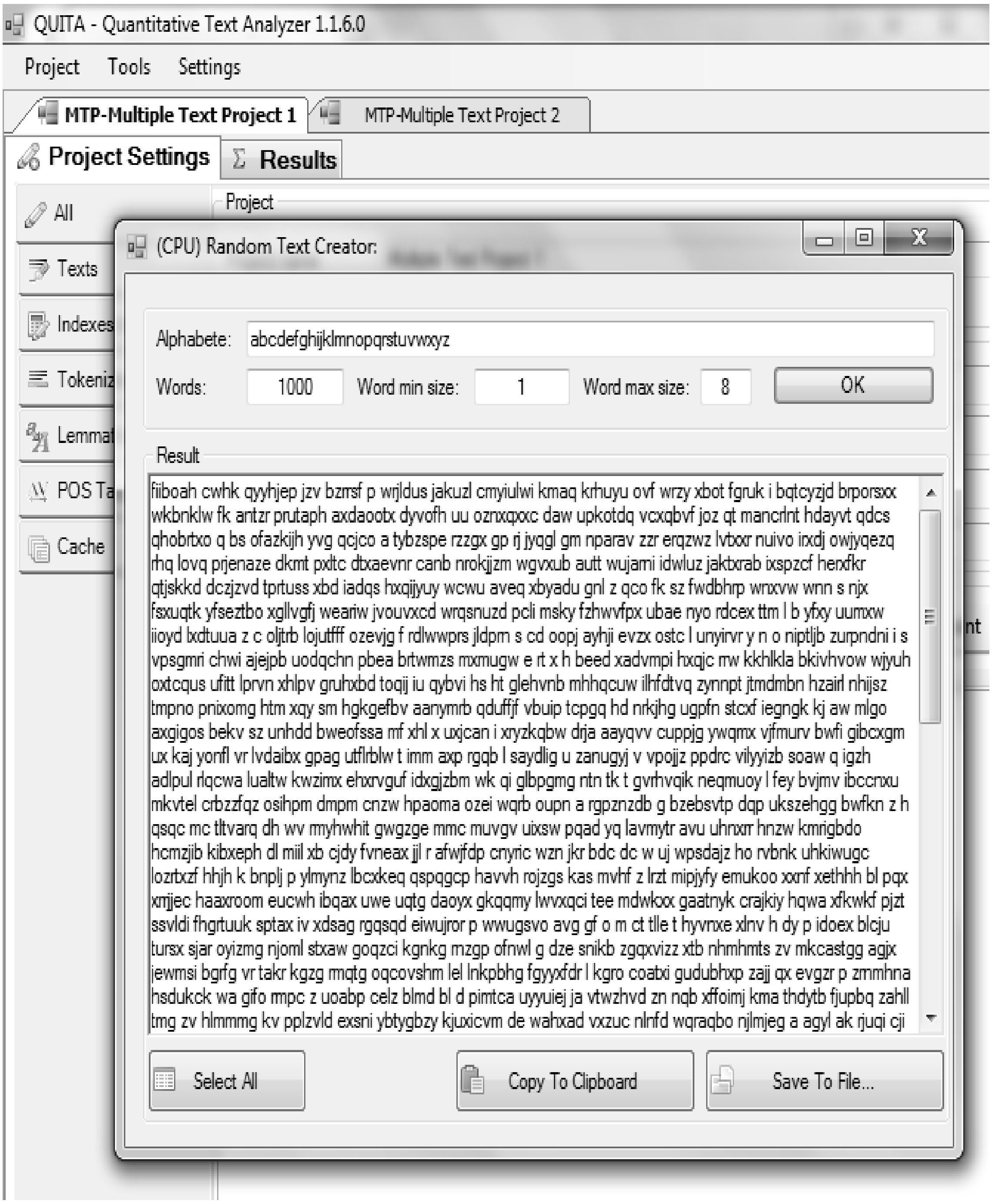
Project name: Multiple Text Proj

Tokenizer: Default generic

12.1. Random Text Creator

In linguistics, it is often very useful to compare the results of a real text with the results of a random text. For this purpose, QUITA includes a tool for creating random text. You can select characters, text length, and the minimal and maximal word size. The created text can be copied or directly saved as a .txt file to your PC.

NOTE: The randomization function is provided by the .NET Framework 3.5: System.Random (class) Next (method). Each call of this function generates a random number used to pick an alphabetic character.



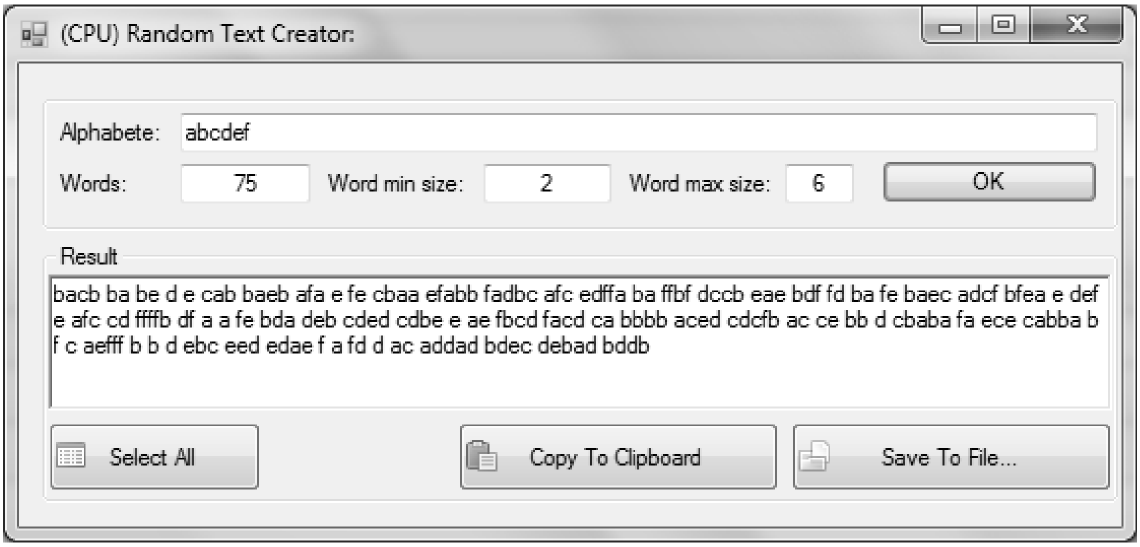
Alphabete: abcdefghijklmnopqrstuvwxyz

Words: 1000 Word min size: 1 Word max size: 8

```
fiiboah cwhk qyyhjep jzv bzrrsf p wrjldus jakuzl cmyiulwi kmaq krhuyu ovf wrzy xbot fgruk i bqtcyzjd brporsxx
wkbnklw fk antzr prutaph axdaootx dyvofh uu oznxqxc daw upkotdq vcxqbvf joz qt mancrlnt hdayvt qdcs
qhobrtxo q bs ofazkijh yvg qcjco a tybzspe rzzgx gp rj jyqgl gm nparav zzr erqzwz lvbxr nuivo irxdj owjyqezq
rhq lovq prjenaze dkmt pxltc dtxaevnr canb nrokjjzm wgvxub autt wujami idwluz jaktxrab ixspzcf herxfkr
qtjskkd dczjzvd tprtuss xbd iadqs hxqijyuy wcwu aveq xbyadu gnl z qco fk sz fwdbhrp wrxvw wnn s njx
fsxuqtk yfseztbo xgllvgfj weariw jvouvxcd wrqsnuzd pcli msky fzhwvfpx ubae nyo rdcex ttm l b yfxy uumxw
iioyd lxdtuua z c oljtrb lojutfff ozevjg f rdlwwprs jldpm s cd oopj ayhji evzx ostc l unyirvr y n o niptljb zurpndni i s
vpsgmri chwi ajejpb uodqchn pbea brtwmzs mxmugw e rt x h beed xadvmpi hxqjc rrw kkhlkla bkivhvow wjyuh
oxtcqus ufitt lprvn xhlpv gruhxbd toqij iu qybvi hs ht glehvnb mhhqcuw ilhfdtvq zynnpt jtmdmbn hzairl nhijsz
tmpno pnixomg htm xqy sm hgkgefbv aanymrb qduffjf vbuip tcpgq hd nrkjhg ugpfn stcxf iegngk kj aw mlgo
axgigos bekv sz unhdd bweofssa mf xhl x uxjcan i xryzkqbw drja aayqvv cuppjg ywqmx vjfmurv bwfi gibcxgm
ux kaj yonfl vr lvdaibx gpag utflrblw t imm axp rgqb l saydlig u zanugyj v vpojjz ppdrc vilyyizb soaw q igzh
adlpul rlqcwa lualtw kwzimx ehxrvguf idxgjzbm wk qi glbpgmg ntn tk t gvrhvqik neqmuoy l fey bvjmv ibccnxu
mkvtel crbzzfqz osihpm dmpm cnzw hpaoma ozei wqrb oupn a rgpznzdb g bzebsvtp dqp ukszehgg bwfkn z h
qsqc mc tltvarq dh wv rnyhwhit gwgzge mmc muvgv uixsw pqad yq lavmytr avu uhrxrr hnzw kmrigbdo
hcmzjib kibxeph dl miil xb cjdy fvneax jjl r afwjfdp cnyric wzn jkr bdc dc w uj wpsdajz ho rvbnk uhkiwugc
lozrtxzf hhjh k bnplj p ylmynz lbcxkeq qspqgcp havvh rojzgs kas mvhf z lrzt mipjyfy emukoo xxnf xethhh bl pqx
xrnjjec haaxroom eucwh ibqax uwe uqtg daoyx gkqqmy lwvxqci tee mdwkxx gaatnyk crajkiy hqwa xfkwkf pjzt
ssvldi fhgrtuuk sptax iv xdsag rgqsqd eiwujror p wwugsvo avg gf o m ct tlle t hyvrxe xlnv h dy p idoex blcju
tursx sjar oyizmg njoml stxaw goqzci kgnkg mzgp ofnwl g dze snikb zgqxvizz xtb nhmhmts zv mkcastgg agjx
jewmsi bgrfg vr takr kgzg rnqtg oqcovshm lel lnkpbhg fgyyxfdr l kgro coatxi gudubhxp zajj qx evgzr p zmmhna
hsdukck wa gifo rnpc z uoabp celz blmd bl d pimtca uyyuiej ja vtwzhvd zn nqb xffoimj kma thdytb fjupbq zahll
tmg zv hlmmmg kv pplzvld exsni ybtygbzy kjuxicvm de wahxad vxzuc nlnfd wqraqbo njlmjeg a agyl ak rjuqi cji
```

Example

We would like to create a random text which consists of the first six letters of the alphabet (abcdef). The text must be 75 words long and the word length must vary from 2 to 6. First, type the letters you want to use. Then, choose the number of words (text size) and the minimal and maximal word length. Finally, click on the “OK” button to create a text.



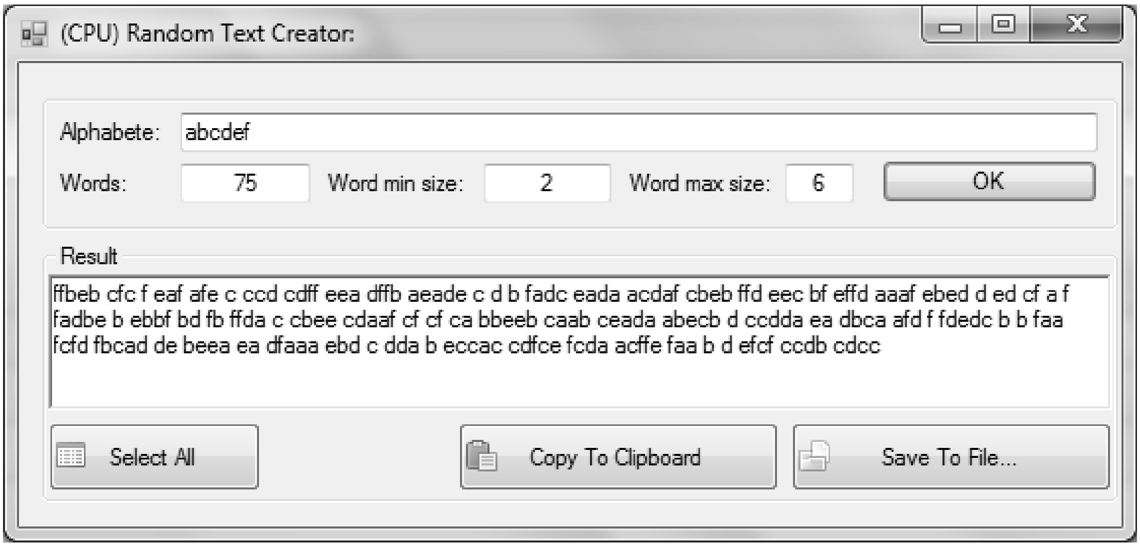
Alphabete: abcdef

Words: 75 Word min size: 2 Word max size: 6

Result

```
bacb ba be d e cab baeb afa e fe cbaa efabb fadbc afc edffa ba ffbf dccb eae bdf fd ba fe baec adcf bfea e def
e afc cd ffffb df a a fe bda deb cded cdbe e ae fbcd facd ca bbbb aced cdcfb ac ce bb d cbaba fa ece cabba b
f c aefff b b d ebc eed edae f a fd d ac addad bdec debad bddb
```

If we want to create another text with the same settings, we can simply click on the “OK” button again. This step can be repeated as many times as we want.



Alphabete: abcdef

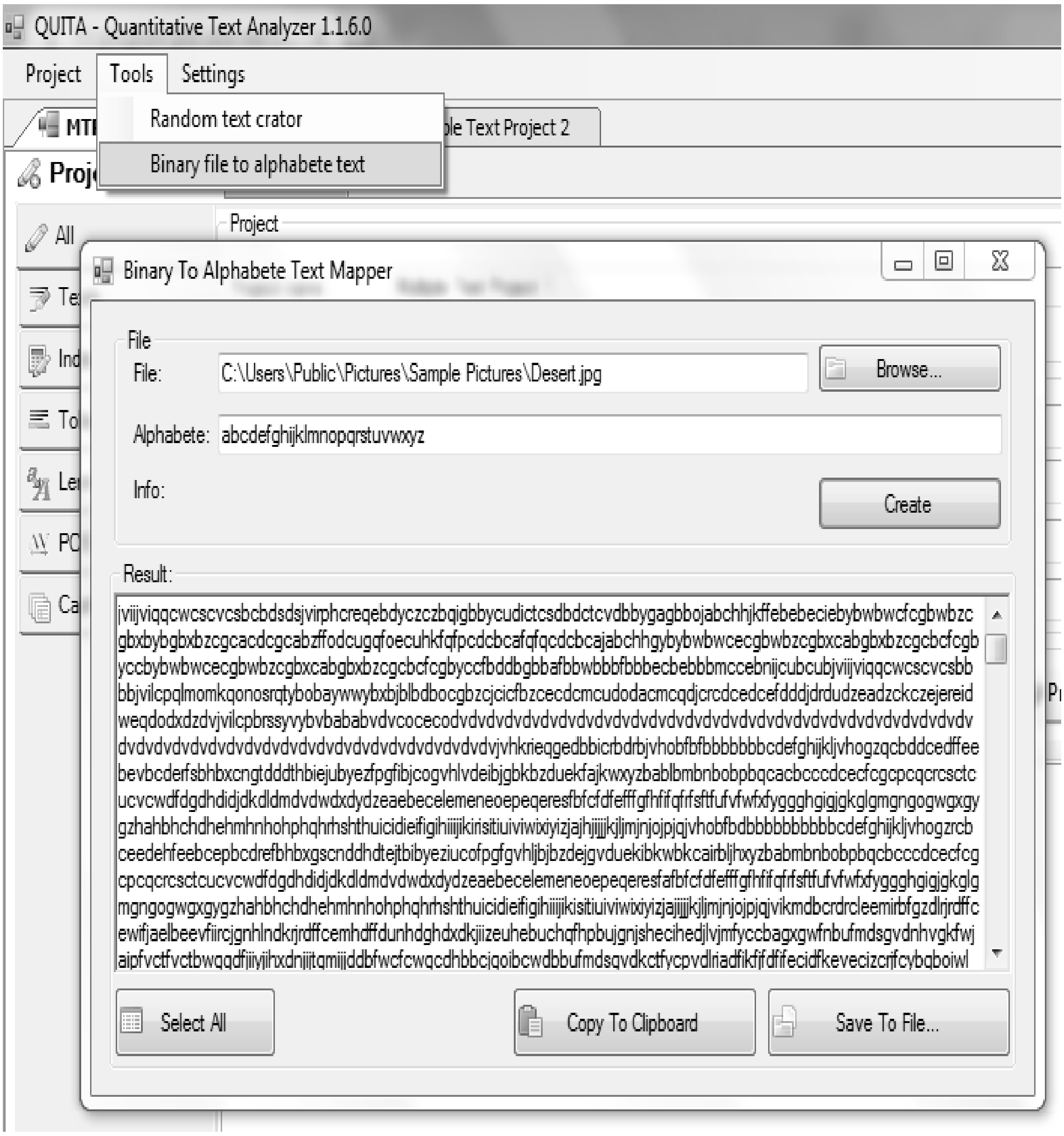
Words: 75 Word min size: 2 Word max size: 6

Result

```
ffbeb cfc f eaf afe c ccd cdff eea dffb aeade c d b fadc eada acdaf cbeb ffd eec bf effd aaaf ebed d ed cf a f
fadbe b ebbf bd fb ffda c cbee cdaaf cf cf ca bbeeb caab ceada abecb d ccdda ea dbca afd f fdedc b b faa
fcfd fbcad de beea ea dfaaa ebd c dda b eccac cdfce fcda acffe faa b d efcf ccdb cdcc
```

12.2. Binary File Translator

This tool transforms any binary file (executable files, pictures, compressed files, sounds, …) to a text by coding its bytes into a user defined alphabet by the same manner as common numeric base conversion is done. E.g., if we specify the alphabet as “0123456789ABCDEF”, the generated text is the same as that displayed in any hexadecimal file editor. Thus, any coded output of this tool is revertible to its original without losing any information. If we specify the alphabet as “ACGT” (nucleotide letters), you can examine the differences between the executable file from your system and live form DNA. The created text can be copied or directly saved as a .txt file to your PC.



File

File: C:\Users\Public\Pictures\Sample Pictures\Desert.jpg

Alphabete: abcdefghijklmnopqrstuvwxyz

Info:

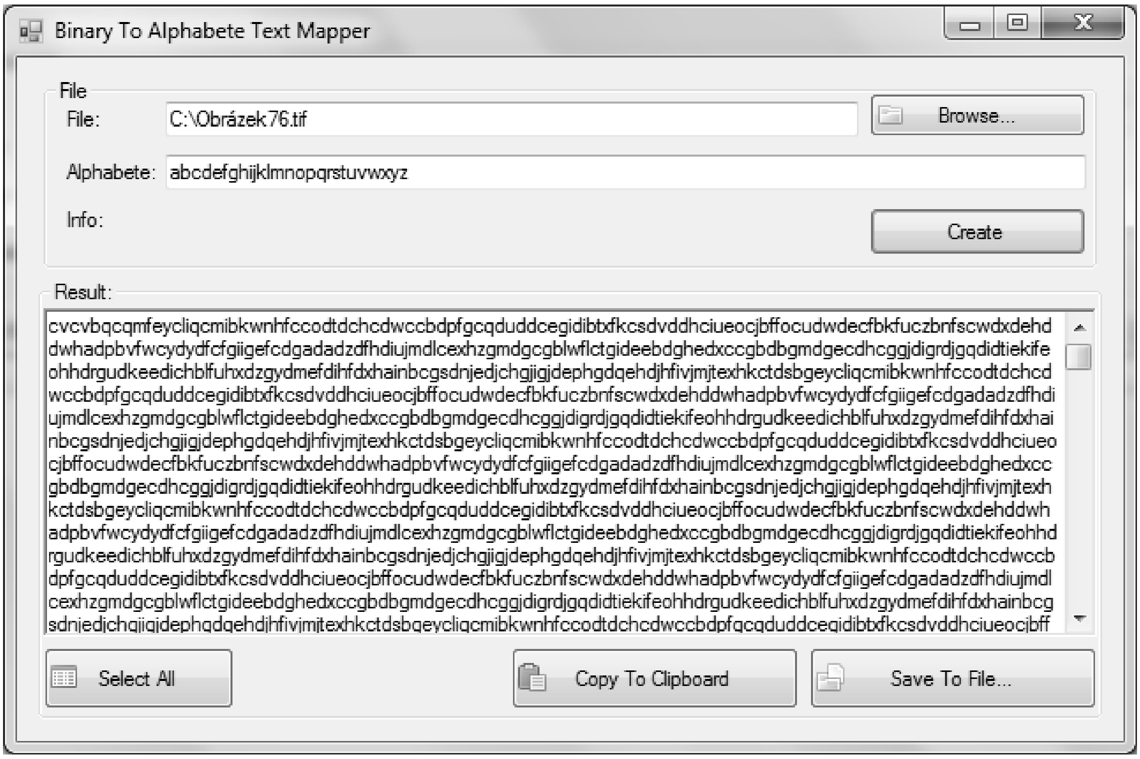
Result:

```
jviijviqqcwcscvcsbcbdsdsjvirphcreqebdyczczbqigbbycudictcsdbdctcvdbbygagbbojabchhjkffebebeciebybwbwcfcgbwbzc
gbxbybgbxbzcgcacdcgcabzffodcugqfoecuhkfqfpcdcbcafqfqcdcbcajabchhgybybwbwcecgbwbzcgbxcabgbxbzcgcbcfcgb
yccbybwbwcecgbwbzcgbxcabgbxbzcgcbcfcgbyccfbddbgbbafbbwbbbfbbbecbebbbmccebnijcubcubjviijviqqcwcscvcsbb
bbjvilcpqlmomkqonosrqtybobaywwybxbjblbdbocgbzcjcicfbzcecdcmcudodacmcqdjcrcdcedcefdddjdrdudzeadzckczejereid
weqdodxdzdvjvilcpbrssyvybvbababvdvcocecodvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdv
dvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvdvjvhkrieqgedbbicrbdrbjvhobfbfbbbbbbbcdefghijkljvhogzqcbddcedffee
bevbcderfsbhbxcngtdddthbiejubyezfpgfibjcogvhlvdeibjgbkbzduekfajkwxyzbablbmbnbobpbqcacbcccdcecfcgcpcqcrcsctc
ucvcwdfdgdhdidjdkdldmdvdwdxdydzeaebecelemeneoepeqeresfbfcfdfefffgfhfifqfrfsftfufvfwfxfygggghgigjgkglgmgngogwgxgy
gzhahbhchdhehmhnhohphqhrhshthuicidieifigihiiijikirisitiuiviwixiyizjajhjjjjkjljmjnjojpjqjvhobfbdbbbbbbbbbbcdefghijkljvhogzrcb
ceedehfeebcepbcdrefbhbxgscnddhdtejtbibyeziucofpgfgvhljbjbzdejgvduekibkwbkcairbljhxyzbabmbnbobpbqcbcccdcecfcg
cpcqcrcsctcucvcwdfdgdhdidjdkdldmdvdwdxdydzeaebecelemeneoepeqeresfafbfcfdfefffgfhfifqfrfsftfufvfwfxfygggghgigjgkglg
mgngogwgxgygzhahbhchdhehmhnhohphqhrhshthuicidieifigihiiijikisitiuiviwixiyizjajjjjjkjljmjnjojpjqjvikmdbcrdrcleemirbfgzdlrjrdffc
ewifjaelbeevfiircjgnhlndkrjrdffcemhdffdunhdghdxdkjiizeuhebuchqfhpbujgnjshecihedjlvjmfyccbagxgwfnbufmdsgvdnhvgkfwj
aipfvctfvctbwqqdfjiyjihxdnjijtqmijjddbfwcfcwqcdhbbcjqoibcwdbbufmdsqvdkctfycpvdlriadfikfjfdfifecidfkevecizcrfcybqboiwl
```

Example

We would like to transform a picture in jpeg format (see below) to a text consisting of letters from the English alphabet. The steps to do this are:

1. Open the binary file translator in “TOOLS”.
2. Choose the file using the “BROWSE” button.
3. Type the letters for the alphabet.
4. Click on the “CREATE” button to complete the process.



Binary To Alphabete Text Mapper

File

File: C:\Obrázek76.tif

Alphabete: abcdefghijklmnopqrstuvwxyz

Info:

Result:

```
cvcvbqcqmfeycliqcmibkwnhfccodtdchcdwccbdpfgcqduddcegidibtxfkcsdvddhciueocjbffocudwdecfbkfuczbnfscwdxdehd
dwhadpbvfwcydydfcfgiigefcdgadadzdfhdiujmdlcexhzgmdgcgblwflctgideebdghedxccgbdbgmdgecdhcggjdigrdjgqdidtiekife
ohhdrgudkeedichblfuhxdzgydmefdihfdxhainbcgsdnjedjchgjigjdephgdqehdjhfivjmjtexhkctdsbgeycliqcmibkwnhfccodtdchcd
wccbdpfgcqduddcegidibtxfkcsdvddhciueocjbffocudwdecfbkfuczbnfscwdxdehddwhadpbvfwcydydfcfgiigefcdgadadzdfhdi
ujmdlcexhzgmdgcgblwflctgideebdghedxccgbdbgmdgecdhcggjdigrdjgqdidtiekifeohhdrgudkeedichblfuhxdzgydmefdihfdxhai
nbcgsdnjedjchgjigjdephgdqehdjhfivjmjtexhkctdsbgeycliqcmibkwnhfccodtdchcdwccbdpfgcqduddcegidibtxfkcsdvddhciueo
cjbffocudwdecfbkfuczbnfscwdxdehddwhadpbvfwcydydfcfgiigefcdgadadzdfhdiujmdlcexhzgmdgcgblwflctgideebdghedxcc
gbdbgmdgecdhcggjdigrdjgqdidtiekifeohhdrgudkeedichblfuhxdzgydmefdihfdxhainbcgsdnjedjchgjigjdephgdqehdjhfivjmjtexh
kctdsbgeycliqcmibkwnhfccodtdchcdwccbdpfgcqduddcegidibtxfkcsdvddhciueocjbffocudwdecfbkfuczbnfscwdxdehddwh
adpbvfwcydydfcfgiigefcdgadadzdfhdiujmdlcexhzgmdgcgblwflctgideebdghedxccgbdbgmdgecdhcggjdigrdjgqdidtiekifeohhd
rgudkeedichblfuhxdzgydmefdihfdxhainbcgsdnjedjchgjigjdephgdqehdjhfivjmjtexhkctdsbgeycliqcmibkwnhfccodtdchcdwccb
dpfgcqduddcegidibtxfkcsdvddhciueocjbffocudwdecfbkfuczbnfscwdxdehddwhadpbvfwcydydfcfgiigefcdgadadzdfhdiujmdl
cexhzgmdgcgblwflctgideebdghedxccgbdbgmdgecdhcggjdigrdjgqdidtiekifeohhdrgudkeedichblfuhxdzgydmefdihfdxhainbcg
sdnjedjchgjigjdephgdqehdjhfivjmjtexhkctdsbgeycliqcmibkwnhfccodtdchcdwccbdpfgcqduddcegidibtxfkcsdvddhciueocjbff
```

The resulting sequence of letters of the picture can be saved and analysed by all the indicators in QUITA, as any other text.

13. Additional information

QUITA, although it has been tested, may still contain bugs and flaws. If you find any troubles, difficulties, mismatches or you have any suggestions on how to improve QUITA, do not hesitate to contact us at the project homepage:

http://oltk.upol.com/software

14. References

Altmann, G., Wimmer, G. (1999). Review Article: On Vocabulary Richness. *Journal of Quantitative Linguistics 6(2), 1–9.*

Bennett, W. R. (1976). *Scientific and engineering problem-solving with the computer.* Englewood Cliffs, N.J.: Prentice Hall.

Čech, R. (2011). Frequency structure of New Year's presidential speeches in Czech. The authorship analysis. In: Kelih et al. (eds.) *Issues in Quantitative Linguistics 2.* Lüdenscheid: RAM-Verlag, 82–94.

Čech, R. (2013). Language and ideology: Quantitative thematic analysis of New Year speeches given by Czechoslovak and Czech presidents (1949–2011). *Quality & Quantity 48(2), 899–910.*

Čech, R. (2014). Text length and the lambda frequency structure of the text. In: *Sequences in language and text.* (accepted).

Čech, R., Garabík, R., Altmann, G. (2014). Some new indicators of thematic concentration (in press).

Čech, R., Popescu, I. I., Altmann, G. (2013): Methods of analysis of a thematic concentration of the text. *Czech and Slovak Linguistic Review* (in press).

Čech, R., Popescu, I. I., Altmann, G. (2014). *Metody kvantitativní analýzy (nejen) básnických textů.* Olomouc: Univerzita Palackého v Olomouci.

Covington, M. A., McFall J. D. (2010). Cutting the Gordian Knot: The Moving-Average Type-Token Ratio (MATTR). *Journal of Quantitative Linguistics 17(2), 94–100.*

David, J., Čech, R., Radková, L., Davidová Glogarová, J., Šústková, H. (2013). Slovo a text v historickém kontextu - perspektivy historicko-sémantické analýzy jazyka. Brno: Host.

Davidová Glogarová, J., Čech, R. (2013). Tematická koncentrace textu – některé aspekty autorského stylu Ladislava Jehličky. *Naše řeč 96, 234–245.*

Davidová Glogarová, J., David, J., Čech, R. (2013). Analýza tematické koncentrace textu – komparace publicistiky Ladislava Jehličky a Karla Čapka. *Slovo a slovesnost, 74, 41–54.*

Doležel, L. (1963). Předběžný odhad entropie a redundance psané češtiny. *Slovo a slovesnost 24, 165–174.*

Esteban, M. D., Morales, D. (1995). A summary of entropy statistics. *Kybernetica 31(4), 337–346.*

Hirsch, J. E. (2005). An indicator to quantify an individual's research output. *Proceedings of the National Academy of Sciences of the USA 102 (46), 16569–16572.*

Kjell, B. (1993). Woods, W. Addison, & Frieder, Ophir. Discrimination of authorship using visualization. *Information Processing & Management 30(1), 141–150.*

Kjell, B. (1994). Authorship determination using letter pair frequency features with neural network classifiers. *Literary and Linguistic Computing, 9(2), 119–124.*

Kubát, M., Milička, J. (2013). Vocabulary Richness Measure in Genres. *Journal of Quantitative Linguistics 20(4), 339–349.*

Livio, M. (2002). *The Golden Ratio: The Story of Phi, The World's Most Astonishing Number.* New York: Broadway Books.

Markov, A. A. (1913). An Example of Statistical Analysis of the Text of "Evgenii Onegin" Illustrating the Linking of Events into a Chain. *Bulletin de l, Acadamie Imperiale des Sciences de St. Petersburg, 6(7), 153–162.*

McIntosh, R. P. (1967). An indicator of diversity and the relation of certain concepts to diversity. *Ecology 48, 392–404.*

Mikros, G., Perifanos, K. (2013). Authorship attribution in Greek tweets using multilevel author's n-gram profiles. In: E. Hovy, V. Markman, C. H. Martell & D. Uthus (Eds.), *Papers from the 2013 AAAI Spring Symposium "Analyzing Microtext", 25–27 March 2013, Stanford, California (pp. 17–23)*. Palo Alto, California: AAAI Press.

Popescu, I. I., Altmann, G. (2007). Writer′s view of text generation. *Glottometrics 15: 71–81.*

Popescu, I. I., Altmann, G. (2011). Thematic concentration in texts. In: *Issues in Quantitative Linguistics 2,* ed. by Kelih, E., Levickij, V., Matskulyak, Y. Lüdenscheid: RAM, 110–116.

Popescu, I. I., Altmann, G., Grzybek, P., Jayaram, B. D., Köhler, R., Krupa, V., Mačutek, J., Pustet, R., Uhlířová, L., Vidya, M. N. (2009). *Word frequency studies.* Berlin-New York: Mouton de Gruyter.

Popescu, I. I., Čech, R., Altmann, G. (2011). *The lambda-structure of texts.* Lüdenscheid: RAM.

Popescu, I. I., Čech, R., Altmann, G. (2012). Some Geometric Properties of Slovak Poetry. *Journal of Quantitative Linguistics 19 (2): 121–131.*

Popescu, I. I., Mačutek, J., Altmann, G. (2009). *Aspects of word frequencies*. Lüdenscheid: RAM.

Popescu, I. I., Mačutek, J., Kelih, E., Čech, R., Best, K. H., Altmann, G. (2010) *Vectors and codes of text.* Lüdenscheid: RAM-Verlag.

Sanada, H. (2013). Thematic concentration in Japanese prose. In Obradovic, I., Kelih, E., Köhler, R. (eds.), *Methods and Applications of Quantitative Linguistics. Selected papers of the 8th International Conference on Quantitative Linguistics (QUALICO), Belgrade, Serbia, April 26–29, 2012.* Belgrade: University of Belgrade, 130-140.

Těšitelová, M. (1974). *Otázky lexikální statistiky*. Praha: Academia.

Těšitelová, M. (1987). *Kvantitativní lingvistika*. Praha: SPN.

Tuzzi, A., Popescu, I. I., Altmann, G. (2010a). The golden section in texts. *ETC – Empirical Text and Culture Research 4, 30–41.*

Tuzzi, A., Popescu, I. I., Altmann, G. (2010b). *Quantitative Analysis of Italian texts*. Lüdenscheid: RAM.

Wilson, A. (2009). Vocabulary richness and thematic concentration in internet fetish fantasies and literary short stories. *Glottotheory 2 (2), 97–107.*

15. Appendix

15.1. Text 1

The second paragraph of the book *Nineteen Eighty-Four* by George Orwell

The hallway smelt of boiled cabbage and old rag mats. At one end of it a coloured poster, too large for indoor display, had been tacked to the wall. It depicted simply an enormous face, more than a metre wide: the face of a man of about forty-five, with a heavy black moustache and ruggedly handsome features. Winston made for the stairs. It was no use trying the lift. Even at the best of times it was seldom working, and at present the electric current was cut off during daylight hours. It was part of the economy drive in preparation for Hate Week. The flat was seven flights up, and Winston, who was thirty-nine and had a varicose ulcer above his right ankle, went slowly, resting several times on the way. On each landing, opposite the lift-shaft, the poster with the enormous face gazed from the wall. It was one of those pictures which are so contrived that the eyes follow you about when you move. BIG BROTHER IS WATCHING YOU, the caption beneath it ran.

15.2. Text 2

The first paragraphs of the novel *Animal Farm* by George Orwell

Mr. Jones, of the Manor Farm, had locked the hen-houses for the night, but was too drunk to remember to shut the pop-holes. With the ring of light from his lantern dancing from side to side, he lurched across the yard, kicked off his boots at the back door, drew himself a last glass of beer from the barrel in the scullery, and made his way up to bed, where Mrs. Jones was already snoring.

As soon as the light in the bedroom went out there was a stirring and a fluttering all through the farm buildings. Word had gone round during the day that old Major, the prize Middle White boar, had had a strange dream on the previous night and wished to communicate it to the other animals. It had been agreed that they should all meet in the big barn as soon as Mr. Jones was safely out of the way. Old Major (so he was always called, though the name under which he had been exhibited was Willingdon Beauty) was so highly regarded on the farm that everyone was quite ready to lose an hour's sleep in order to hear what he had to say.

15.3. Word list of Text 1 and Text 2

Nineteen Eighty-Four		
RANK	WORD	FREQUENCY
1	the	16
2	it	7
3	was	7
4	of	7
5	and	5
6	a	5
7	for	3
8	you	3
9	at	3
10	face	3
11	times	2
12	had	2
13	enormous	2
14	on	2
15	with	2
16	wall	2
17	winston	2
18	about	2
19	lift	2
20	poster	2
21	one	2
22	ankle	1
23	hate	1
24	preparation	1
25	his	1
26	right	1
27	drive	1
28	economy	1
29	resting	1
30	in	1
31	went	1
32	slowly	1
33	thirty	1
34	nine	1
35	flights	1
36	up	1
37	who	1
38	seven	1
39	flat	1

40	week	1
41	above	1
42	varicose	1
43	ulcer	1
44	follow	1
45	when	1
46	move	1
47	contrived	1
48	that	1
49	eyes	1
50	big	1
51	caption	1
52	beneath	1
53	ran	1
54	brother	1
55	is	1
56	watching	1
57	landing	1
58	opposite	1
59	shaft	1
60	several	1
61	way	1
62	each	1
63	gazed	1
64	which	1
65	are	1
66	so	1
67	from	1
68	those	1
69	pictures	1
70	depicted	1
71	simply	1
72	an	1
73	been	1
74	tacked	1
75	to	1
76	wide	1
77	man	1
78	forty	1
79	more	1
80	than	1
81	metre	1
82	display	1

83	cabbage	1
84	old	1
85	rag	1
86	hallway	1
87	smelt	1
88	boiled	1
89	too	1
90	large	1
91	indoor	1
92	mats	1
93	end	1
94	coloured	1
95	present	1
96	electric	1
97	current	1
98	best	1
99	seldom	1
100	working	1
101	daylight	1
102	hours	1
103	part	1
104	cut	1
105	off	1
106	during	1
107	even	1
108	moustache	1
109	ruggedly	1
110	handsome	1
111	five	1
112	heavy	1
113	black	1
114	no	1
115	use	1
116	trying	1
117	features	1
118	made	1
119	stairs	1

Animal Farm		
RANK	WORD	FREQUENCY
1	the	20

2	to	9
3	was	8
4	had	7
5	he	4
6	as	4
7	a	4
8	in	4
9	of	4
10	jones	3
11	and	3
12	from	3
13	his	3
14	farm	3
15	that	3
16	been	2
17	light	2
18	it	2
19	so	2
20	side	2
21	way	2
22	on	2
23	night	2
24	all	2
25	major	2
26	mr	2
27	old	2
28	soon	2
29	out	2
30	dream	1
31	strange	1
32	prize	1
33	they	1
34	agreed	1
35	day	1
36	animals	1
37	wished	1
38	middle	1
39	white	1
40	communicate	1
41	previous	1
42	other	1
43	boar	1
44	should	1

45	ready	1
46	lose	1
47	an	1
48	regarded	1
49	everyone	1
50	quite	1
51	hour	1
52	hear	1
53	what	1
54	say	1
55	s	1
56	sleep	1
57	order	1
58	highly	1
59	safely	1
60	always	1
61	called	1
62	meet	1
63	big	1
64	barn	1
65	though	1
66	exhibited	1
67	willingdon	1
68	beauty	1
69	name	1
70	under	1
71	which	1
72	during	1
73	dancing	1
74	lurched	1
75	across	1
76	with	1
77	ring	1
78	lantern	1
79	boots	1
80	at	1
81	back	1
82	yard	1
83	kicked	1
84	off	1
85	houses	1
86	for	1
87	but	1

88	manor	1
89	locked	1
90	hen	1
91	shut	1
92	pop	1
93	holes	1
94	too	1
95	drunk	1
96	remember	1
97	door	1
98	went	1
99	there	1
100	stirring	1
101	already	1
102	snoring	1
103	bedroom	1
104	word	1
105	gone	1
106	round	1
107	fluttering	1
108	through	1
109	buildings	1
110	glass	1
111	beer	1
112	barrel	1
113	drew	1
114	himself	1
115	last	1
116	bed	1
117	where	1
118	mrs	1
119	scullery	1
120	Made	1
121	up	1

15.4. Text 3

The poem *I Said to Love* by Thomas Hardy.

I said to Love,
"It is not now as in old days
When men adored thee and thy ways

All else above;
Named thee the Boy, the Bright, the One
Who spread a heaven beneath the sun,"
I said to Love.

I said to him,
"We now know more of thee than then;
We were but weak in judgment when,
With hearts abrim,
We clamoured thee that thou would'st please
Inflict on us thine agonies,"
I said to him.

I said to Love,
"Thou art not young, thou art not fair,
No faery darts, no cherub air,
Nor swan, nor dove
Are thine; but features pitiless,
And iron daggers of distress,"
I said to Love.

"Depart then, Love! . . .
- Man's race shall end, dost threaten thou?
The age to come the man of now
Know nothing of? -
We fear not such a threat from thee;
We are too old in apathy!
Mankind shall cease.--So let it be,"
I said to Love.

15.5. Text 4

The poem *The Two Nests* by Dora Sigerson.

The wise thrush, the wise thrush, she choseth well her tree,
Made her nest in the laurel's leafy shade.
But the foolish young girl, all laughing in her glee,
She built on a reed that all winds swayed,
She built on a reed that swung and swayed.

The wise thrush, the wise thrush, she crouchèd on her nest,
When the hawk in the clouds hunted nigh,
But the foolish young maid did sing in soft request

He pass not unpraised her nestlings by,
Her gentle hopes and pretty dreaming by.

The wise thrush, the wise thrush, she lingered and she spied
A safe flight her fledgelings to gain,
But the foolish young girl, all careless in her pride,
Found her pretty ones were scattered and were slain,
In her ravished heart her pretty ones were slain.

The wise thrush, the wise thrush, she drowsèd at her ease
While her nestlings did pipe on the tree.
But the foolish young maid could not her grief appease,
For her dying hopes were pitiful to see,
Oh, pitiful her perished dreams to see.