Building Text's Fingerprint

Professor, PhD. Ion IVAN, stud. Mihai AMITROAIE
Economy Informatics Department, A.S.E. Bucharest

Using the computer to analyze a text implies defining aggregate indicators, generally known as „text's fingerprint". This paper tries to define the text's fingerprint main issues, and to present a software product designed especially for measuring the level of defined indicators and characteristics. A new software product will be created, that will help us building text's fingerprints. All the calculated characteristics will be used in finding the resembling degree of two texts. Also, those signatures will be used to determine and identify the clones made upon a specific text.

Keywords: *text's fingerprint, text analyze, text clones.*

1. Text characteristics

Considering the following alphabet A={a_1,a_2, ...a_n} where: n – number of alphabet A symbols; a_i – the i positioned symbol in the A alphabet and B={b_1,b_2, ... b_m} where: n – number of alphabet B symbols; b_j – the j positioned symbol in the B alphabet, the following alphabet can be built: C=$A \bigcup B$= {a_1,a_2,...a_n,b_1,b_2,...b_m} called the extended alphabet.

For example, the alphabet A is made up of small letters a,b,c,...,x,y,z and the alphabet B contains the symbols: blank, coma and dot. Then the alphabet C contains 30 symbols that are used to elaborate texts. The words are made with the symbols from alphabet A, and the symbols from alphabet B are used as separators. A text is produced by using words, blanks, comas and dots. We enforce the convention that the text doesn't contain sequences of separators. A sequence is formed by successively repeating the same symbol.

Considering this, „aaaa" is a sequence obtained by repeating four times the symbol 'a'. According with our convention, the sequence "….." that represents five symbols '.' arranged one after the other is an incorrect construction. On the cases where those kinds of incorrect sequences will appear, we will process them by normalizing the texts. This means that any sequence of blanks, comas or dot is replaced with a single symbol.

For example the text: T={abc? ? ? ? ij,,,xxx…}, where ? =blank, becomes, after the normalizing process: T'={abc? ij'xxx.}

Text Length L_t represents the number of symbols that are used for building the text. For example, the text: T_1={abcdaaaayxzabb } has Lt=14 symbols. And the text: T_2={a? b? c? d? x.}has L_t=10 symbols. The text: T_3={aa…bb,,,c? ? dd }has Lt=15 and after normalizing it: T_3'={aa.bb,c? dd} it has L_t=10 symbols.

Vocabulary is formed by all of the words which are used in the text. The text:

T_4={casa? masa? si? la? casa? la? si? si? si
? casa? casa? casele ?
masa? ma,sa,casa,mas? a? la? masa}

Has the vocabulary V(T_4) defined as:

{casa, masa, si, la, casele, ma, sa, mas, a}.

The text:

T_5={copac,copac,? copac? opac,ac,ac,copac, copac }

is formed as: V(T_5)={copac,opac,ac }.

The L_V vocabulary length represents the number of words which are contained into the vocabulary. For example, the text:

T_6={ aaa, bbb, ccc, ddd, aaa, aaa, aaa, ddd, ddd, aaa }

is made of the words from the vocabuulary:

V(T_6)={ aaaa, bbb, ccc, ddd }

Where: L_t=39 symbols; L_V=4 words.

Frequency f_i of a word represents the usage of that word, and it is equal to the number of a word uses in order to construct a text. If considered the text:

T_7 =
{a? aaa? a? aa? aa? aaa? a? aaa? aaa? aaa? a
}
that was built using the vocabulary $V(T_7)$ = { a,aa,aaa} we will have the frequency structure of this text as it is shown in the table 1.

Table 1 - Word's frequencies

Word	Frequency
A	4
Aa	2
Aaa	5

Frequency g_j represents the usage of the word that has the j position into the extended C alphabet. For the T_8 text:
T_8={aa,a,aaa,ab,ac,acc,cccccc,b,bbaa.,abc}
the extended C alphabet is made of the following symbols: {a b c **, .** }and the vocabulary usage is shown by the frequencies from table 2.

Table 2 - Symbol's frequencies

Symbol	Frequency
a	12
b	5
c	10
,	9
.	1

Consider N_i the number of interchanges between the needed vocabulary elements for arranging descendingly upon the frequencies. The text:
T_9={a? aa? aaa? aa? aa? a? a? a? aa? aa? aa aa,aa}
has the vocabulary $V(T_9)$ formed with the following set of words {a, aa, aaa, aaaa} and the frequencies are shown in table 3.

Table 3 - Word's frequencies

Word	Frequency
a	4
aa	5
aaa	2
aaaa	1

The number of interchanges is $N_i = 1$, because we need to change the position of word „a" with the position of word „aa" to obtain a words list arranged by frequency. The result of the sorting process is shown in table 4.

Table 4 - Words ordered by frequency

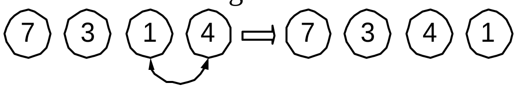
Word	Frequency
aa	5
a	4
aaa	2
aaaa	1

The number of interchanges depends on the sort algorithm that is implemented. It should be established a default algorithm that will be used to guarantee the compatibility between two or more applications. To calculate the number of the interchanges N_S needed to order the words by frequency we will have to roll through a number of steps.
Consider the text T_{10} that is defined like: { a? b,aaaa,bb,aa,c,dddd }. The initial image of the vocabulary is shown in table 5.

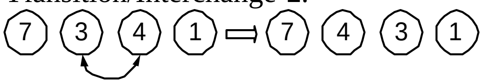
Table 5 - Initial frequencies

Word	Frequency
a	7
b c	3 1
d	4

Using the sorting algorithm that uses the text semantics, we'll proceed upon the following transitions:
Transition/Interchange 1:



Transition/Interchange 2:



The total number of interchanges is $N_S = 2$. The total number of interchanges is equal to zero when the frequencies are already sorted descendingly. The maximum number for a text with n components is: $\frac{n(n-1)}{2}$. The interval for N_S is: $0 \le N_S \le \frac{n(n-1)}{2}$. Figure 1 presents the most unfavorable decision. For a set o 5 symbols the maximum number of interchanges is f(5) = 4 + 3 + 2 + 1= 10.

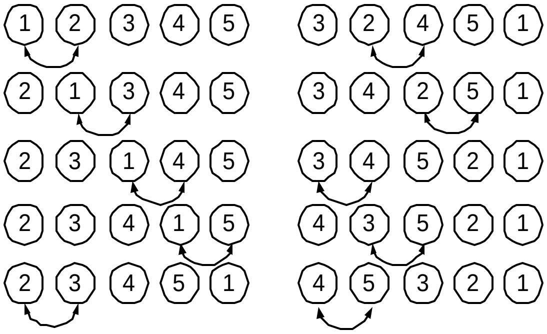


Fig.1. Unfavorable situation

The set of numbers that defines the distances between two consecutives apparitions of a character shows the particular aspects of a text. For example, the text: T_{11}={abbacdabcdab}.

has the particular structure:

a	2, 2, 3
b	0, 4, 3
c	3
d	3

The matrix of precedents is formed from as many lines and columns as the number of symbols that are used. The cell that is located at line i and column j contains the value h_{ij} that represents the number of apparitions of word c_j after the word c_i in the text.

Parsing the following text:

T_{12}={aa? bb? cc? aa? aa? bb? bb? bb}

we will obtain the vocabulary {aa, bb, cc}and the matrix of precedency:

	aa	bb	cc
aa	1	2	0
bb	0	2	1
cc	1	0	0

Another property of a text is the set of numbers that represents the length of the words contained into the vocabulary. Considering the text: T_{13}={a,aa,aaa,bb,cccc,a} the size of the words is shown into the table.6.

Table 6 - Word's lengths

Word	Length
a	1
aa	2
aaa	3
bb	2
cccc	4

2. Reporting the text to the vocabulary

Considering the C alphabet and the V vocabulary which is constructed with symbols from C, they will be used in analyzing the following texts: C={x, y, z}, V={x, xy, yz, zzz, xyx}. Any constructed text will contain only words from the V vocabulary and the separator used will be **","** (coma). In this way vocabularies of the analyzed texts will be included into the reference vocabulary V and their alphabets will be included into the reference alphabet C,too.

Considering the text: T_{14}={x,xy, xyx, xyx, x, x,xy}. Vocabulary length for this text is L_V = 3.

We have the length of the reference vocabulary V, L_V=5 , so it results that in those circumstances: L_{ref}>=$L_{V.}$ The reference vocabulary is build by using the alphabet C, and the length of this vocabulary reporting to the number of symbols is L_C=4 (including coma). The length of the alphabet for the text T_{14} is L_C=3. The words from the vocabulary of this text are constructed by using only the symbols x and y and are separated by coma.

When there are built texts with words from the reference vocabulary V we can meet situations when, accidentally, some series of sequences:

- are not words from the reference vocabulary although all the symbols are from the reference alphabet
- include some others symbols beside the ones used to build the reference vocabulary

In both cases we have to distinguish the main properties of the words that were accidentally generated and used in the text. Considering the alphabet: C_{13} = { r, s, t, p }. We will use it to build the following vocabulary:

V_{13}={rr, rs, sp, stp, rtp, srrs, sss, ppp, tt }

Generating the text T_{15} we will use some words from the vocabulary V and a set of new words that are accidentally created:

- words that are not contained in vocabulary V but the symbols are from C alphabet

sspp
r
trs
tps

- words that are not contained in vocabulary V and the symbols are not from alphabet C

tppp
tt*
rtp-
rr/
rpa

The symbols +, *, -, a does not appear in C_{13} alphabet.

The analyzed text is: T_{15}={rr, rs, ppp, +ppp, tt,sss, tt*, tt*, -ppp, +ppp, rtp, rtp-, rs, rr/, rtp, spa, sspp, rs,trs, sss, ppp, r, tps, rr, rtp-, tt, stp}. It is studied by using it's particular vocabulary and alphabet.

It is interesting to analyze the text T_{15} referring the reference vocabulary and the reference alphabet and keeping the words that were accidentally generated. Those words will be grouped in two sets of words. The first set, A, contains words that do not exist in the reference vocabulary V but the symbols are from the alphabet C. The second set, B, contains words that are not included into the vocabulary V and the symbols are from the alphabet C. The text T_{15} contains A and B sets of words with the characteristics shown in table 7.

Table 7 - The sets of words that are not included into the vocabulary..

Set	Word	Frequency	Length	Positions
A	tps	1	3	23
B	+ppp	2	4	4, 10
	tt*	2	3	7, 8
	rtp-	2	4	12, 25
	rr/	1	3	14
	spa	1	3	16

We consider V' to be the extended vocabulary that includes, besides the words from the reference vocabulary, words that are accidentally generated. This vocabulary will be used to proceed with the study. The study will calculate some special indicators that will be used in defining text's fingerprint. The extended alphabet C' is obtained by supplementing the alphabet C with symbols that were accidentally used. For an exhaustive analyze we should use both alphabets and both vocabularies. The study should treat different the results obtained. We should consider what is defined and used under normal condition and report it to the particular cases when accidental situations occur.

This issue is observed when we are handling large sets of papers or when the database that is used is too big. If we want to evitate those kinds of situations we have to define some rules and steps to be followed if the process of acquiring information:

e1 - primary data is collected from the texts contained in documents. All the documents have an unique key;

e2 - the documents are divided into sets of documents;

e3 - every operator is working with a set of documents;

e4 - the operator records the document's data and saved it into a file;

e5 - the files are concatenated and sorted after a unique key, the result is the file F1;

e6 - the subsets of documents are randomly distributed to the operators;

e7 - the operators records the document's data and saved it into files;

e8 - the files are concatenated and sorted after a unique key, the result is the file F2;

The files F_1 and F_2 do contain the texts T_1 and T_2. If the following criteria is accomplished:

- the lengths of the texts T_1 and T_2 are equal;
- the lengths of the vocabularies that are obtained are the same;
- the vocabularies are the same;
- the word's frequencies are equal for the texts T_1 and T_2;
- the number of interchanges between the elements of the vocabulary, that are needed for arranging descending upon the frequencies, are equal for T_1 and T_2;

- the precedence matrix of text T_1 is the same with the precedence matrix of text T_2;
- the series containing the positions of the words from vocabulary are the same for the texts T_1 and T_2;

we can conclude that the process of collecting the data was normally accomplished..

If we compare the vocabulary with the text content we don't observe inadequate situations, we can conclude that the F_1 and F_2 files do contain complete and correct data about the documents. But the facts are quite different; the discrepancies between F_1 and F_1 can appear mostly because: some documents are not recorded; some documents are recorded more than once.

Usually there are differences between the files when we compare the lengths of the vocabularies or the frequencies of the words. Every improper situation has a rational explication. So, by identifying it and trying to establish some basic rules, we can avoid those kinds of situations that can generate unwanted results. If between the files F_1 and F_2 are some differences that are discovered analyzing the text, we can proceed on correcting the files. The result will be a set of files: ($F_1^{(1)}$, $F_2^{(1)}$), ($F_1^{(2)}$, $F_2^{(2)}$), ... , ($F_1^{(k)}$, $F_2^{(k)}$).

The number of steps, k, must be a reasonable number, so we can conclude in the necessary time that the files do contain complete and correct data.

3. Software for building text's fingerprint

For the entire range of indicators used for measuring the structure of a text (I_1, I_2, ..., I_h), there are built evaluation procedures incorporated in a software product which has the following functions:

- reading a text memorized in a file
- building the set of symbols
- building the set of separators
- building the set of words in the vocabulary
- counting the frequency of words in the text
- counting the frequency of words from the vocabulary
- establishing the length of the text
- establishing the length of the words in the vocabulary
- building the array with the positions of the words from the text
- building the precedence matrix
- calculate the number of interchanges in order to sort on frequencies the words from the vocabulary
- calculate the number of interchanges in order to sort on the length criteria the words from the vocabulary
- defining a reference vocabulary
- defining a reference alphabet
- build vocabulary A and vocabulary B
- printing the results
- create database for text's fingerprints, containing the name of the text file and the values of the calculated indicators
- create the symmetrical matrix for compared analysis of the texts in the database in order to determine if the texts differ from each other or not

The clones are texts with identical indicators. Establishing that text T_i is the clone of T_j supposes the following steps:

1. build the fingerprints of those two texts;
2. compare all the calculated indicators; if they have identical levels, proceed with the third step, otherwise the texts are not clones or they have common elements;
3. analyze the subsets A and B;
4. if the levels of the indicators are identical it is assumed that one text is the other's clone, otherwise proceed with the next step;
5. compare the texts in order to see the identical substrings and to determine their weights;

The following indicators will be calculated for the frequencies: the median, the spread and the factor of variation. Also the following indicator can be calculated:

$$S = \frac{\sum_{h=1}^{H} L_h}{\max\{L_i, L_j)}$$

where: H – number of common substrings; L_i – length of text T_i; L_j - length of text T_j; L_h – length of the common substring h from the two texts.

Conclusions

The software product was build by using Java, the program can be found at www.aprenta.ase.ro and it was tested on several texts.

This product is suited in analyzing any kind of projects, including papers, articles, books of projects that are submitted to obtain a financial support. It can be seen as a start point to a complete software product, with database support and other kind of facilities,

Bibliography

[IVAN03] Ion IVAN, Adrian POCOVNICU - *Text's fingerprint building, using sampling, based on generation of pseudo-random numbers*, accepted for 'Sesiunea stiintifica Invatamantul de Informatica Economica', Timisoara, may 2003

[IVAN02a] Ivan Ion, Niculescu Silviu, Catalin Boja – *Clonarea bazelor de date*, Revista Româna de Informatica si Automatica, Bucharest, vol. 12, no. 4, 2002, pg. 46 – 53

[IVAN02b] Ivan Ion, Pocatilu Paul, Popa Marius, Sacala Mihai, Ungureanu Doru – *Information Cloning*, Probleme regionale în contextul procesului de globalizare – Simpozion International, Chisinau, Moldavian Republic , October 9th, 2002, pg. 371 - 375

[IVAN02c] Ivan Ion, Popa Marius, Sacala Mihai – *Ortogonalitatea datelor*, Revista Româna de Statistica, Bucharest, no. 4, 2002

[IVAN02] Ivan Ion, Popa Marius, Sacala Mihai - *Ortogonalitatea datelor*, Revista Româna de Statistica – National Statistics Institute, Bucharest, no.4, 2002

[SMEU01] Smeureanu Ion, Dârdala Marian – *Programarea în limbajul C/C++*, CISON, Bucharest, 2001

[PORO93] Porojan Dumitru – *Statistica si teoria sondajului*, Casa de editura si presa "Sansa" SRL. Bucharest, 1993

[KAFU81] Kafura, D. ,Henry, S. - *Software Quality metrics based on interconnectivity*, Journal of System and Software no.2, 1981, pg. 121-131

[HALS77] Halstead, M.H. - Elements of Software Science Elsevier-North Holland, Amsterdam, 1977

[****70] I.B.M. - *Application Program, Fifth Edition*, August 1970