

Development of generation special short articles for the given topic

Kulik S.D.¹, Belov A.N.², Matveeva K.I.³

1 Professor,

Faculty of Institute of Intelligent Cybernetic Systems, National Research Nuclear University MEPhi (Moscow Engineering Physics Institute) Moscow, Russia

2 Student,

Faculty of Institute of Intelligent Cybernetic Systems, National Research Nuclear University MEPhi (Moscow Engineering Physics Institute) Moscow, Russia

3 Student,

Faculty of Institute of Intelligent Cybernetic Systems, National Research Nuclear University MEPhi (Moscow Engineering Physics Institute) Moscow, Russia

Corresponding author E-mail: on-lain.shop@yandex.ru; sedmik@mail.ru

Abstract

The subject area and reviewed various sources of information relevant to a given topic will be considered. For implementation of algorithm of generation of the texts was chosen the C# language, as it has a large number of advantages compared to other programming languages, and it also has many libraries which it would be convenient to use in solving a given task. The text presents the structure of the generated article and the scheme of the developed algorithm for the application. Also it describes the structures that are used to implement such applications. The implementation of algorithm of generation of short articles on a given topic in English was discussed. Familiarization with the application was conducted and the results of generation were shown. It was shown that the generated articles have a good percentage of the plagiarism by experimental studies. It was shown that the articles would not be repeated even more than 50 generations, as a result of tests. Also was shown the implementation of the program on C# language, it's interface, design generation and design of the English translation.

Keywords: *Intelligent Systems, Translations, Scientific Articles, Generated Text, the Uniqueness of the Text.*

1. Introduction

It is important in our days to write a new unique text, which would be basis for different scientific articles. Studies and software development of generation special short articles for a given topic will be conducted in this work.

The generation of texts is a reproduction of one article into several unique copies and, as a rule, similar topics. The quality of the articles that are made by the reproduction, depends on the generation method that is used in the program. The reproduction is often based on the replacement of words by synonyms, but more often, unique texts are obtained through combinations of various synonyms among themselves.

The main purpose of generating texts on a special topic is obtaining a large amount of unique articles. All other goals derive from this task.

Generation is a simple and convenient way to obtain information on a given topic. It is also useful in announcing the website's articles in the popular services that helps to save time from rewriting or just writing new articles.

Most of the generators are intended to generate texts according to a certain pattern — the number of paragraphs, semantic, or logical constructions. There are programs that use Markov chains in

which text is practically meaningful and coherent. However, to achieve a complete result is not possible, due to the fact that the machine has no intelligence, and only performs texts that were written by a human team.

Currently, there are a number of companies that develop more complex technologies, create a syntactic structure for the parts of speech and members of sentences, words in dictionaries are categorized according to certain semantics, and their programs have the auto completion suggestions.

The studies will be obtained an algorithm of generation of special short articles for a given topic, as well as its software implementation in C# in a console application.

It is necessary to give a short review of the research field. Some important problems were considered in the indicated papers [1, 2, 3, 4, 5]. The paper [1] explores the model for evaluating the effectiveness of search operations. This paper [2] is dealing with the dictionary-based MTS for translating Gujarati bigram idioms to English. This paper [3] and monograph [4] are related to various important problems, for instance: neural network model of artificial intelligence in [3], Neural Networks Theory in [4]. A lot of papers and monographs are concerned with different important problems for example information retrieval [5] and English translation.

2. Development of the Algorithm

2.1. Overview of Automatic Generation of Texts

Currently there are many tools to auto-generate texts. Many of them are based on simple algorithms. We will consider this algorithm on the example of the oscillator "Spring" from Yandex. You must first prepare a template to copy the text, which will be further multiplied. It is easy to create, you just need to know some nuances of generation.

As a result of reproduction, we will have text, which may be random sampling:

- Breeding high-quality unique articles using software.
- Generate high-quality texts for a website by sinonimayzer.
- Generate quality articles with the use of the service.

All these phrases are received after generation, and they will be unique relative to each other, which is what we needed. The greater the number of synonyms and of blocks are used, the more unique the texts will be or the greater their number can be generated.

You should note that not all words in the text should be enclosed in blocks. For example, the dot and the word "quality" is behind the braces, i.e. outside of any block [6]. This means that they will be used in all variants of generation.

In each program you can use other operators, but these two are standard for all services.

2.2. The Advantages of C# Language

Specifically for the platform .NET Microsoft was developed a new programming language C#. C# is a programming language, which syntax is very similar to Java (but not identical). For example, in C# (as in Java) class definition consists of a single file (*.cs), unlike C++, where the class definition is divided into a header (*.h) and implementation (*.cpp). However, to call C# a Java clone would be wrong. C# and Java are based on the syntactic structures of C++. If Java can be called the purified version of C++, in many respects, C# can be described as a clean version of Java.

Perhaps the most important thing that should be said about the C# language is that it generates code intended to run only at runtime .NET [7]. For example, you cannot use C# to create a classic COM server. According to Microsoft code meant to work at runtime .NET is a managed code (managed code). Binary file that contains the managed file is referred to as an Assembly (assembly).

Syntax of C# inherited not only from C++, but from Visual Basic too. For example, in C# as in Visual Basic, is used the properties of classes. C# allows operator overloading when you create your own types, like C++ (Java does not support neither one nor the other). C# is actually a hybrid of different languages. While C# is syntactically at least friendly than Java, as simple as Visual Basic, and has almost the same power and flexibility as C++. Summing up, again, you can highlight the main advantages of the C# language:

- Pointers are no longer needed! In C#, as a rule, there is no need to work with them (however, if you need it, please, — opportunities to work with pointers at your disposal).
- Memory management is done automatically.
- C# provides built-in syntax for enumerations, structures and properties of classes.
- It would be possible to overload the operators inherited from C++. A substantial portion of the resulting difficulties eliminated.
- Fully support of the use of software interfaces. However, in contrast to the classical use of COM interfaces is not the only way of working with types using the various binary modules. .NET allows you to pass objects (as references or as values) across the boundaries of program modules.
- Also C# provides full support for aspect-oriented software technologies (such as attributes). This allows you to specify types of characteristics (which is much like COM IDL) to describe future behaviour of the entity.
- C# pretends to be an authentic object-orientation Initial orientation for a security code.
- Language is not simply the sum of the advantages of modern programming languages (Java, C++, Visual Basic), and programming language of a new generation.

2.3. Structure of an Article

We determined the structure of the generated article. It is represented in Figure 1.

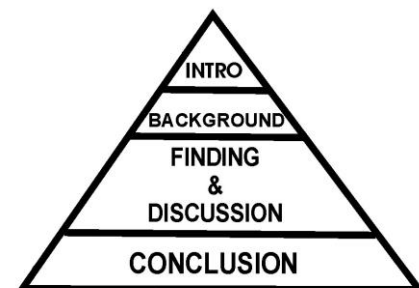


Fig. 1: The structure of the generated article

Each generated article will have 4 paragraphs:

- introduction
- background,
- research, calculations and reasoning on the subject,
- conclusion.

2.4. Scheme of the Developed Algorithm

The algorithm that was developed to create a console application is shown in Figure 2 and Figure 3.

The flowchart shows the algorithm of a software system. The app displays interface with a single button, after logging in, clicking

on which will be generating the article. The resulting article will appear in the box to the right. Tap on the “Next” button will generate a new article.

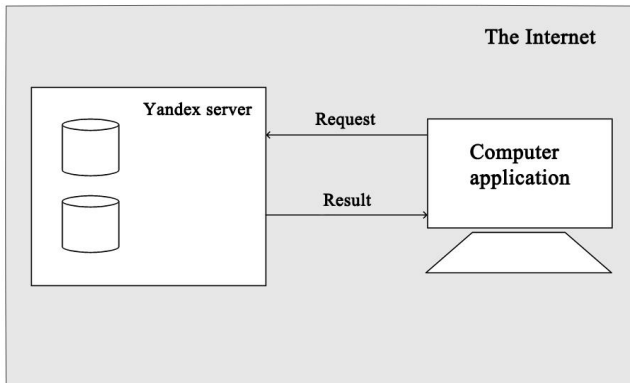


Fig. 2: The scheme of the algorithm of application

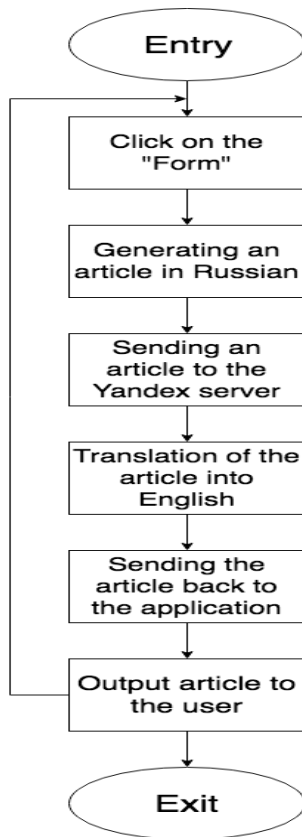


Fig. 3: The block diagram of the developed algorithm

2.5. Used Construction

Whichever program you decide to propagate the text, all they support two standard operators:

{ } — open and closing curly bracket are used to create blocks of synonyms.

| — vertical line (not an uppercase “i” or a lowercase “l” that look also), this symbol is used to separate the options of synonyms among themselves.

As a result, the design we get the following:

{version 1.1|version 1.2|...|option 1.n} {version 2.1|version 2.2|option 2.n} ... {variant m.1|m.2|...|m.n}

Thus, we've got a pattern consisting of m-number of blocks. Each block contains the n-th number of synonyms.

2.6. Interface Design and Main Elements of the Program

For the implementation of the algorithm presented in Chapter 2, we developed a console application for Windows, running on Windows 7.

For the creating of application was used the programming language C#.

This application was developed by using Microsoft Visual Studio 2015.

We will consider the development of the interface and basic elements of the program.

Figure 4 shows us the graphical interface of this application, that appears when you open this app.



Fig. 4: GUI when opening the app

For the next, the user should click "form" to generate an article on a special topic in English. In Figure 5 is shown the generated article in English. To get the other article, please click on the "next" button.



Fig. 5: GUI when opening the app

2.7. Debugging and Testing of the Program

At the initial stage debugging the application was checked:

- correct display of the graphical interface,
- correct display generated articles in the window.

The next phase of debugging — checking for the generation of articles:

- when you click the button "form"- should be printed the generated article in the box on the right,
- when you click on the "next" button should be generated the following article, different from the previous one.

When you click the "next" button more than 50 times, and each generated article was different from all the other articles, the check for plagiarism was more than 70%. In Figure 6 is shown the screenshot of checking for plagiarism of one of the generated articles.

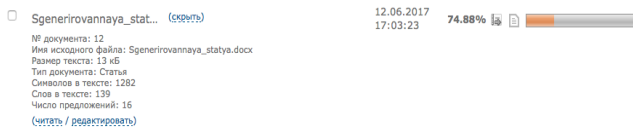


Fig. 6: Check the article for plagiarism

2.8. Experimental Investigations of the Developed Program

There were taken 10 generated articles that have been checked for plagiarism as experimental studies. The results of the study were listed in Table 1.

Table 1: Check the generated articles for plagiarism

The number of the generated articles	The percentage of anti-plagiarism
1	75
2	72
3	81
4	76
5	72
6	75
7	71
8	81
9	76
10	72

The generation of the article takes not much time (less than 1 second). It takes about 10-15 minutes for a human to write this article. This application will allow you to save time and to be sure in passing the anti-plagiarism of their work.

Compared to generator articles, such as "Spring" from Yandex, this app generates articles in English language and has a certain structure of 4 paragraphs.

This app combines the functions of oscillator "Spring" and "SeoGenerator" that makes it more suitable for use when writing short articles on given topic.

3. Conclusion

We have investigated the current automatic translation tools in this work and was made a review of tools for automatic generation of texts.

C# language was selected because of its advantages over other languages and settings.

It was further submitted that the term "translation studies" and the major transformation in the translation. Was the structure of the algorithm of generation, its diagram and description. Utilized design and support elements.

The debug and testing program generated articles were reviewed. The results of testing were shown. After that, was given an experimental study that was developed by the program.

The comparing the resulting research programme and its analogues in the Internet was performed. By the details, that were researched, we can say that the developed application will be more efficient (preferable).

Future steps include getting the answers to the following important questions:

How do parameters of the translation algorithm influence information retrieval?

Which structure of neural network is most suitable for translator in English?

What activation function is most suitable for recognizing English or Russian text?

Could one recognize English or Russian text automatically?

Which structure of data base is most suitable for translator in English language?

Acknowledgement

This work was supported by Competitiveness Growth Program of the Federal Autonomous Educational Institution of Higher Education National Research Nuclear University MEPhI (Moscow Engineering Physics Institute). We would like to express our gratitude students of NRNU MEPhI for their valuable discussion and suggestions.

References

- [1] S.D. Kulik, Model for evaluating the effectiveness of search operations, Journal of ICT Research and Applications (ITB Journal of Information and Communication Technology), Vol.9, No.2, (2015) 177-196.
- [2] J. R. Saini, J. C. Modh, GIdTra: A dictionary-based MTS for translating Gujarati bigram idioms to English, Conference Paper published 2016 in 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC) Wagnaghat, India 22-24 December 2016, Department of CSE & IT Jaypee University of Information Technology (2016) 191-196.
- [3] S.D. Kulik, Neural network model of artificial intelligence for handwriting recognition, Journal of Theoretical and Applied Information Technology, Vol.73, No.2, (2015) 202-211.
- [4] A. I. Galushkin, Neural Networks Theory, Springer Berlin Heidelberg, 2007.
- [5] S. Kulik, Factographic information retrieval for communication in multicultural society, Procedia - Social and Behavioral Sciences (International Conference on Communication in Multicultural Society, CMSC 2015, 6-8 December 2015, Moscow, Russian Federation), Vol. 236, (2016) 29-33.
- [6] <https://yandex.ru/referats/> [Accessed on 12.04.2017].
- [7] Shildt G., C# 7.0 Full guide. — London, Atlas Press, 2015.
- [8] <https://www.antiplagiat.ru/> [Accessed on 20.04.2017].
- [9] Archer T., C# basics. — Manchester, Carcanet Press, 2015.
- [10] Watson C., C# Language. — Amsterdam, Holland Park Press, 2016.
- [11] Troelsen A., Program Library. — London, Longman, 2016