**International Journal of Research in Social Sciences**
**Vol. 9 Issue 8, August 2019,**
ISSN: 2249-2496 Impact Factor: 7.081
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

# THE MODEL OF AN AUTOMATIC SPELLING CHECKER, BASED ON AN EDITORIAL METRIC WITH SUBSTITUTION

**Mirzamov Akmal Maxmudjanovich***

## Abstract

In this article, the task of automatic checking the spelling errors in Uzbek texts is modeled on the basis of the fuzzyset theory. In this, an editorial metric with substitution is used to increase the sensitivity of the algorithm to words. On the basis of the proposed model, automatic spelling of texts in solving the problem of error checking in the Uzbek language with high speed, with the use of sensitivity and soft computingis ensured.

**Key Words**: fuzzyset, fuzzy algorithms, editorial metric with substitution, soft computing.

## INTRODUCTION

Currently, automated programs to solve the problem of preventing spelling errors with the help of computers exist and they are widely used. In particular, effective solutions have been proposed in Microsoft Office programs to check and correct spelling of texts in English, Russian and many other languages [1]. As well as this,in the last (2016) samples of Microsoft Office programs,spell-checking packages for the Uzbek language are offered [1]. However, the task of checking spelling errors in Uzbek texts has not yet been fully resolved. Programs designed to check spelling errors for the Uzbek language have been developed by various firms and companies. But since most of them do not give a complete solution to this task, they are also used little in practice. This is due to the fact that in most such programs, incomplete dictionary-based algorithms were used. This can be followed by the fact that their data storageindicatesthe existence of more than 400 000 000 words. However, the number of words and phrases in the Uzbek language does not exceed 85 000 [2]. Hence, more than 400 000 000 words that are in the data storage,are formed with the

* **Namangan State University**

**PhD, Department of Applied Mathematics, Faculty of Mathematics and Physics**

help of adding various prefixes and suffixes to the same word core. Since all the words cannot be covered by such a method, it cannot provide a full-fledged performance of the program. The program cannot assess the correct spelling of the entered word in the data storage due to the absence of a word in some combination, and concludes that the word is incorrect.

Let usexplainthis by the following comparison. For example, in order to formulate natural numbers storage, it is not possible to enter and store all natural numbers in the data storage. Therefore, when developing programs related to natural numbers, this method does not succeed. But, it is possible to generate any natural number in the 10-digit system through a number-based algorithm of generating numbers, and thus, it is possible to cover all natural numbers.

Similarly, it is possible to achieve a complete coverage of all words based on 85 000 words and combinations of words known today, through the algorithm of the rules for forming words in the Uzbek language. With this, while eliminating the above shortcomings, it is possible to achieve a significant increase in the speed of the program, the possibility of saving computer memory and network resources.

It is worth noting that the solution to the task of spelling errors corrections in texts is widely used not only in the process of spelling of users, but also in such matters as automatic recognition of nonverbal (image, optical) texts, recognition of speech sounds, identification of the content of the text, search by content and machine translation.

The fact that there is still a need for new, perfect solutions to the task of automatic detection and correction of spelling errors in the Uzbek language, as well as the practical significance of this task, determines itsactuality and creates the need to continue research and further develop this sphere. Especially, the implementation of fundamental research in solving this problem, in particular the development of new theoretical solutions through the extensive use of mathematical and computer linguistics disciplines, provides a fundamental basis for the improvement of previously developed programs.

In this article, the method of modeling the solution of the above-mentioned problem on the basis of the fuzzyset theory is presented [3]. Of course, the method of modeling this task with the help of classical methods of probability theory was developed. However, the fact that the task is of linguistic nature, that is, close to the word $\omega$, since a similar set of words is considered, the modeling of this task on the basis of the methods of the fuzzysets theory leads to a more efficient functioning of the program [4].

The problem in general is as follows. Let the text $M$in a spoken language $X$be given. Let $A$set consisting of similar words close to the word $y =< y1, y2, y3 \dots yn >$in the $M$ text, be determined.

By solving the problem given, it will be possible to prevent the occurrence of spelling errors in the given text by automatically identifying theentered words with spelling errors and suggesting the user to replace these words with words that are written accurately. In other words, if there is not exactly the same word as $\omega$in the specified $A$set, spelling errors will be detected in the text, and the error will be eliminated by suggesting the $A$set to the user.

We will use examples to illustrate the taskprovided, so that the intended purpose is understandable to the reader. For example, in $M$text, they$y =< korsatma >$word is quoted, let the whole set of words of the $X$spoken language consists of 7 words.

$$\text{X} = \{\text{atrof}, \text{borliq}, \text{ko'rsatish}, ko'rsatma, \text{ko'rsichqon}, \quad \text{tantana}, to'plam\}$$

In that case, the $A$set of words, consisting of words that are close or similar to the $y$word, will be as follows.

$$A = \{(\text{atrof}|0.54), (\text{borliq}|0.29), (\text{ko'rsatish}|0.64), (ko'rsatma|0.91), (\text{ko'rsichqon}|0.39), \quad (\text{tantana}|0.33), (to'plam|0.47)\}$$

Here $A$ -is the fuzzy set, $\mu_A$ −the x elements of the set X are the set belonging to $A$ .

$$\mu_A = \{0.54, 0.29, 0.64, 0.91, 0.39, 0.33, 0.47\}$$

If $A$is a normal set, then the error in writing the $y$word is undefined. Otherwise, an error will be detected when writing the $y$word, and in this case the user will be suggested$A$set of words, formed from all the correct set of words in the $X$spokenlanguage. We know that, if the height of $A$set is equal to 1,$A$set is called a normal set, and the height is determined as follows:

$$h(A) = \max(\mu_A(x)), x \in A$$

Of course, from the result obtained, that is, from $A$ fuzzy set, certain actions have to be performed in the program to identify the error and correct it. For example, $A$fuzzyset is sorted in descending order.

$$A = \{ (ko'rsatma|0.91), (\text{ko'rsatish}|0.64), (\text{atrof}|0.54), (to'plam|0.47), (\text{ko'rsichqon}|0.39), (\text{tantana}|0.33), (\text{borliq}|0.29)\}$$

Sorted number of elements of the $A$ set will be equal to the number of elements of the $X$set, and suggesting $A$ in that form will cause inconvenience to the user. Therefore, it is necessary to perform the next action, that is, dephasification.

$$A_{\propto}(x) = \{x \in X | \mu_A(x) \geq \alpha\}$$

In this case, a set is formed, by giving by the following characteristic function.

$$\chi A_{\propto}(x) = \begin{cases} 0, & \mu_A(x) < \alpha \\ 1, & \mu_A(x) \geq \alpha \end{cases}$$

According to the above example, if $\alpha = 0.60$, then the $A_{\propto}$ set consists of the following elements:

$$A_{\propto} = \{ko'rsatma, \text{ko'rsatish}\}$$

In the mathematical language, the solution of the above problem can be expressed as follows:

let $A$–be a fuzzy set, and$A \subseteq X$, that is

$$A = \{(x, \mu_A(x)); x \in X\},$$

In this case$\mu_A: X \to [0,1]$

$$\mu_A(x) = 1 - \frac{d(x,y)}{d(x,y)_{max}}$$

here,$d(x,y)$- is the editing distance to bring the $x$word to the $y$word, $d(x,y)_{max}$ – maximum editing distance to bring the $x$word to the $y$word.

Apparently, one of the main tasks is that,is the calculation of the $d(x,y)$ – editing distance.

Often, as editing distance, Levenshtein or Damerau-Levenshtein metrics are used. However, in order to increase the sensitivity of the Uzbek language dictionaries, we suggest using the editorial metrics with substitution.

In the editorial metric with substitution, additional action of substituting is added to the actions of Damerau-Levenshtein metric and they are given the corresponding values. These actions are listed below:

$\varphi(\varepsilon, b)$ –the value to putting the sign $b$.

$\varphi(a, \varepsilon)$ –the value of deleting character $a$.

$\varphi(a, b)$ –the value of replacingsign $a$ with$b$.

$\varphi(a_{sub}, p) - y_1$replacing $a_{sub}$ the part row of the rowwith$p$ that is, substituting.

We define the values as follows, to the actions of the substitution of editorial distance:

$$\varphi(a, a) = 0$$

$\varphi(\varepsilon, b) = 2$

$\varphi(a, \varepsilon) = 2$

$\varphi(a, b) = 4, a \neq b \text{or} \varphi(a, b) = \varphi(a, \varepsilon) + \varphi(\varepsilon, b) = 2 + 2 = 4$

$\varphi(a_{sub}, p) = 1$

It should be noted that regardless of the $a_{sub}$ length of the line, the value of $\varphi(a_{sub}, p)$ is equal to 1and the substitution and subtraction actions are a private case of the action $\varphi(a_{sub}, p)$, it can be expressed in the substitution action $\varphi(a_{sub}, p)$.

Features of editorial metric with substitution:

1. $d(x, y) = 0 \Leftrightarrow x = y$
2. $d(x, y) \geq \varphi(\varepsilon, b) * \big||x| - |y|\big|$
3. $d(x, y) \leq \varphi(\varepsilon, b) * |x| + \varphi(a, \varepsilon) * |y|$
4. $d(x, y) = \varphi(a, \varepsilon) * D_{xy} + \varphi(\varepsilon, b) * I_{xy} + \varphi(a_{sub}, p) * S_{xy}$

In this case,$D_{xy}$ - $x$is the number of deletions from the line X, $I_{xy}$ - $x$- the number of additions from $y$line to $x$line, $S_{xy}$ - $x$- the number of substitutionsin$x$ Line, $|x|, |y|$- the length of the line.

The calculationof $d(x, y)$ editing distance with substitution [7] is presented in detail in the literature.

If the number of elements of the $X$set is equal to $n$and the number of words in the $M$text is equal to m, then $d(x, y)$the editingdistance is calculated in $n * m$times. For example, the number of all words in the Uzbek language is $n = 85\,000$, the number of words in the given $M$ text is $m = 1\,000$, then$n * m = 85\,000\,000$. From the figures it can be seen that the number of actions is large enough, and this in turn causes a decrease in the speed of the algorithm. The speed of the algorithm is one of its important quality indicators. Therefore, in addition to the effectiveness of the algorithm, its speed should also be taken into account.

In our case, the presence of processes of phasification and dephasification, that is, the use of the fuzzy sets theory, allows the use of soft computing, this in turn contributes to a significant increase in the speed of the algorithm.

Soft computing was introduced by LotfiZade in 1994 and is organized in different ways in different tasks [5]. Soft computing means approximate, closecalculation, whichmeans, it should be understood in the sense of calculating it with sufficient accuracy. In particular, in most cases, the intermediate calculations that occur incalculating the result calculations are substituted with simple functions that calculate functions with sufficient accuracy. In this, simple functions that calculate with sufficient accuracy must guarantee the accuracy of the final result.

Algorithms based on soft computing are called fuzzy algorithms and simple functions in them are called fuzzy functions [3].

In particular, in our task, in the process of calculating $A$ fuzzy set, all $\mu_A(x)$functions are replaced by fuzzy functions, i.e:

$$\mu_A(x)_{fuzzy} = 1 - \frac{d(x,y)_{fuzzy}}{d(x,y)_{max}}$$

In this process, it is possible to apply soft computing several times. In particular, according to the characteristics of the editorial metric with substitution,the $d(x,y)_{fuzzy}$ fuzzy function is initially calculated through the function$d_1(x,y)_{fuzzy} = \varphi(a,\varepsilon) * \big||x| - |y|\big|$, after which is dephasified and $A'_{\propto}(x)$is generated. $A'_{\propto}(x)$relative to the set

$d_2(x,y)_{fuzzy} = \varphi(a,\varepsilon) * (|x| + |y| - \varphi(\varepsilon,y) * |X_T \cap Y_T|)$, $(X_T$ $and$ $Y_T$ −sets of strings consisting of characters) are calculated and dephasified by the function. As a result, $A''_{\propto}(x)$ is formed . $d(x,y)$function is applied to $A''_{\propto}(x)$and dephasified. As a result, we get the required $A_{\propto}(x)$set .

The instructions expressed through the above fuzzy functions are called the fuzzy algorithms in general [6].

Soft computing applied and unapplied algorithms are tested at specific, precise times. In the test, the initial sample was conducted in an electronic dictionary program, where the length of the word is $|x| = 17$, and the number of words in the data storage consists of 19 136 units. The results of the tests are presented in the form of a diagram in Figure 1.

In Figure 1, a diagram of the line length and time of the search algorithms developed on the basis of Levenshtein,Damerau-Levenshtein, OTM and soft computing is expressed. From the diagram it can be seen that, in relation with a word or word storage, algorithms where a soft computing is applied are more stable and of course the speed is higher.

For the task of automatic checking the spelling errors in texts in the proposed model, the sensitivity of the algorithm was increased with the help of an editorial metric with substitution. Additionally, using the theory of fuzzy sets, high speed was achieved with the use of soft computing, flexibilityand high accuracy of the result.

The developed model, along with the advantages listed above, doesn't have the shortcomings that were mentionedin the beginning of the article, and is different from other models. It can achieve even higher results if applied to the automatic checking

the spelling errors in texts, search engines, optical text and speech recognition with sound.

## List of literature


1. “Language Accessory Pack forMS Office”, https://support.office.com/ru-ru/article/Language-Accessory-Pack-для-Office.
2. E.A.Begmatov, A.P.Madvaliev “Spelling dictionary of the Uzbek language” y.2013.
3. G.E.Yaxyaeva “Nechetkiemnojestvaineuronnieseti”. Moscow-2006, p.316.
4. “Theoriya veroyatnostey i theoriya nechetkix mnojestv L. Zade: razlichiya i sxodstvo”, M.I.Aliev, E.A.Isaeva, I.M.Aliev. https://www.researchgate.net/publication
5. Zade L.A. Rol myagkix vichisleniy i nechetkoy logiki v ponimanii, konstruirovanii i razvitii informasionnix intellektualnix sistem. // Novosti iskusstvennogo intellekta, №2-3, 2001, (44-45), p.7-15
6. Mirzamov A.MFuzzy algorithms in search engines / / Informatics and energy problems. - Tashkent, 2010. - № 6. p.37 to 42.
7. Mirzamov A.M Redaktsionnae rasstoyanie s podstanovkoy / / Problemi optimizasii slojnix system: Sedmaya mejdunarodnaya aziatskaya shkola-seminar. - Tashkent: 2011. p. 204-209.