Розглянуто особливості застосування конкуруючих клітинних автоматів до задач з розпізнавання складних captcha-систем. Для цього введено поняття конкуруючих клітинних автоматів та розроблено алгоритми їх функціонування та взаємодії. Описано математичну модель конкуруючих клітинних автоматів на основі теорії множин для опису рухомих клітинних автоматів, що пересуваються по сусідніх станах символів і в такий спосіб реалізують свої правила переходу.На основі математичної моделі розроблено систему розпізнавання captha-зображень, яка реалізована в програмному коді засобами технології JavaFX 2.0, що дозволила досягти кросплатформеності та правильного функціонування на різних операційних системах.

Бібліотеки клітинних автоматів розроблялися для англійської мови. Кожен символ алфавіту представляється у вигляді системи станів, котрій поставлено у відповідність клітинний автомат зі станами, що описують даний символ.

Для розробки було використано мову програмування Java та можливості обробки зображень бібліотекою OpenCV, яка дозволила досягти якісного результату розпізнавання. Розглянуто архітектуру розробленої системи розпізнавання складних captcha-зображень у вигляді діаграм класів головних блоків з детальним описом кожного класу. Проведено комп'ютерні експерименти з різними наборами спотворених символів, що використовуються у реальних captcha-системах та отримано показники якості розпізнавання розробленим програмним забезпеченням.

Показано, що ймовірність отримання правильного результату розпізнавання captcha-зображень перевищує 80 % при ступені деформації символів до 20 %. При ступені деформації символів понад 30 % існує велика ймовірність помилкового розпізнавання символів.

До перевагам методу розпізнавання символів тексту на основі конкуруючих клітинних автоматів слід віднести простоту правил взаємодії, можливість легкого розпаралелювання процесу розпізнавання, можливість розпізнавання спотворених та частково накладених символів, які складають основу сучасних captcha-систем

Ключові слова: конкуруючий клітинний автомат, рухомий клітинний автомат, captcha-системи

UDC 004.93'12; 004.932; 004.932.75
DOI: 10.15587/1729-4061.2018.148307

DEVELOPMENT OF A SYSTEM FOR GRAPHIC CAPTCHA SYSTEMS RECOGNITION USING COMPETING CELLULAR AUTOMATA

I. Myroniv
Postgraduate student*
E-mail: ivan.myroniv@gmail.com
V. Zhebka
PhD, Associate Professor
Department of Software Engineering
State University of Telecommunications
Solomianska str., 7, Kyiv, Ukraine, 03110
E-mail: viktoria_zhebka@ukr.net
S. Ostapov
Doctor of Physical and Mathematical Sciences, Professor, Head of Department*
E-mail: sergey.ostapov@gmail.com
O. Val
PhD, Associate Professor*
E-mail: olexval@bigmir.net
*Department of Computer Systems Software
Yuriy Fedkovych Chernivtsi National University
Kotsiubynskoho str., 2,
Chernivtsi, Ukraine, 58012

1. Introduction

Captcha is most often used to prevent the use of online services by bots, in particular, to prevent automatic mailbox registrations, messaging, file downloads, mass mailings, etc. [1].

The relevance of the use of Captcha can be seen, for example, from the statistics of spam volumes. According to global e-mail service providers, spam volume reaches 97 % of the total number of emails. The use of Captcha protection can complicate the task of registering mailboxes by bots and thus reduce the volume of spam mailings.

2. Literature review and problem statement

The use of cellular automata is very attractive in terms of developing new recognition systems in them. In the paper [2], the author has shown indisputable advantages of using cellular automata (CA) in problems where there is a need of parallel computing that enables the simple implementation of complex image processing algorithms and does not require significant computing resources. Despite these advantages, the cellular automata concept is not so often involved in recognition. The only thorough research in this area is the work [3], the main part of which is devoted to the study of

the characteristics of the CAs in the processes of text recognition. The author uses sequences of different CAs to distinguish the characteristic signs of the text characters: loops, intersections, positions of the ends. The work [4] studies the CAs that makes the recognition of handwritten characters possible. The main drawbacks of such approaches are crockhood and the need for system training.

In addition, in another work [5], the author proposed a new algorithm for the recognition of JPEG watermark images based on cellular automata. One more paper [6] represents specialized cellular automaton structures for the analysis of contour image [5, 6].

In other papers [7, 8], the authors proposed an approach to segmentation of bound symbols in a text CAPTCHA and obtained the result of the recognition of characters that are not separated. The author also [9] evaluates the latest research on the recognition of Captcha systems.

The world-famous service for online recognition [10] of Captcha systems, which actually is a plugin for popular browsers Chrome and Firefox, successfully recognizes characters, the structure of which is not modified by deformation, and the overlapping characters are generally ignored. The average detection time of one captcha is 8 seconds.

However, there is a real way to use the new type of CA in the process of character recognition suggested by the authors [11]. This approach is based on movable CAs, which must realize all their states on the corresponding symbol of the text. The variety of interpretations of characters, which arises in this case, is compensated by the developed mechanism of competition, when the CA with a maximum number of viable states "wins". This CA is the most correct reflection of the symbol under recognition.

At the same time, all the examined methods and systems of recognition work ineffectively on typed and partially distorted characters, and partially overlapped characters, which form the basis of modern Captcha systems, as shown in the following Fig. 1.

Fig. 1. CAPTCHA systems used by Google

Fig. 1 shows the variant of Captcha systems used by such Internet giant as Google. These Captcha systems are characterized by non-linear distortions of the text, shifting symbols one by one, close symbols location, and different fonts. Noises are not applied, but characters are not always merged without spaces, which complicates the recognition process itself.

3. The aim and objectives of the study

The aim of the work is to develop a system for recognition of deformed and partially overlapped characters based on competing cellular automata. Similar to Google Captcha engines, which, however, are unable to recognize the existing systems.

To achieve this aim, it is necessary to accomplish the following objectives:

– to develop a mathematical model of movable cellular machines suitable for use in recognition tasks;

– to develop a mechanism of competition of cellular automata for increasing the efficiency of recognition;

– to develop the architecture and interface of the recognition system based on competing CA;

– to study the effectiveness of the developed software.

4. Mathematical model of movable competing cellular automata

The number of cellular automata used in this work can be written as follows:

$$U = \{\sigma_i\}_{i=1}^{N}, \tag{1}$$

where N is the number of symbols in the alphabet.

Each machine has its own set of states U, label ξ (color, position in the alphabet, etc.), and depends on the discrete time:

$$\sigma_i = \sigma_i(U,t,\xi), \tag{2}$$

$U = \{u_k\}_{k=1}^{K}$; K is the number of states of the current automaton. The shift of the automaton from the current state k to the next k+1, which can be described as follows:

$$\sigma_i(u_k,t,\xi) \overset{t\to t+1}{\rightarrow} \sigma_i(u_{k+1},t+1,\xi). \tag{3}$$

The transition of the CA to the new state is controlled by the transition function φ, so we can do the following:

$$\sigma_i\left(u_k,t,\xi\right) = \varphi(\sigma_i\left(u_k,t,\xi\right)). \tag{4}$$

The movable CA moves from the current state to the next, using the rules generated by the transition function.

Let us assume that the image of the characters to be recognized is presented in the form of a set of states similar to the states of the CA. Then $\Omega = \{\omega_p\}_{p=1}^{P}$ is the set of these states. The number of states P of the character is unknown.

When getting on the character, the automaton will move through its states ω_p the number of which P, generally speaking, is not equal to the number of states of the current automaton K, $P \neq K$.

The transitions will be executed by the CA according to the states of the character, that is:

$$\sigma_i(\omega_p,t,\xi) \overset{t\to t+1}{\rightarrow} \sigma_i(\omega_{p+1},t+1,\xi), \tag{5}$$

when and only when $\omega_p = u_k \in U$ and $\omega_{p+1} = u_{k+1} \in U$ that is, when the states of the character coincide with similar states of the CA. If $\omega_p, \omega_{p+1} \neq u_k, u_{k+1} \in U$ such CA has no allowed transitions and is removed from the cellular-automatic field.

If the CA can realize all its states, specified by the transition function on the current character, that is, if $\forall u_k \in U \exists \omega_p \in$ Ù we will assume that this CA "successfully" describes the current character.

If $\forall u_k \in U \nexists \omega_p \in$ Ù that is, for at least one state of the CA there is no analogous state of the character, such automaton is removed from the CA field.

There may be several CAs that implement all their possible states on the current character. In order to choose from

them the one that exactly matches this character, the competition mechanism is used.

Let 3 CAs move on a single character:

$$\sigma_i \to U = \{u_k\}_{k=1}^{K},$$

$$\sigma_j \to V = \{v_l\}_{l=1}^{L},$$

$$\sigma_h \to W = \{w_s\}_{s=1}^{S}. \quad (6)$$

Each of them realizes all its states on it. We will assume that the competition is "won" by the CA, the number of states of which is the largest. So, it is necessary to find

$$\max(K, L, S). \quad (7)$$

Let $\max(K, L, S)=S$. Then the CA $\sigma_h = \sigma_h(S,t,\eta)$ will be considered the one that describes the current character most "successfully". Reading its label η, we find out which character has been recognized.

The algorithm of recognition itself is based on the construction of the CA and its graph of transitions, the states of which the given automaton moves through.

5. Architecture of the recognition system

The proposed recognition system has been implemented as a software product using JavaFX 2.0 technology. The libraries of cellular automata were developed for the English language. We used Java programming language for development and OpenCV library [12] for the ability to handle images, which allowed us to achieve high-quality recognition results.

The diagram of the main classes [13], which is responsible for working with the device camera, is shown in Fig. 2.

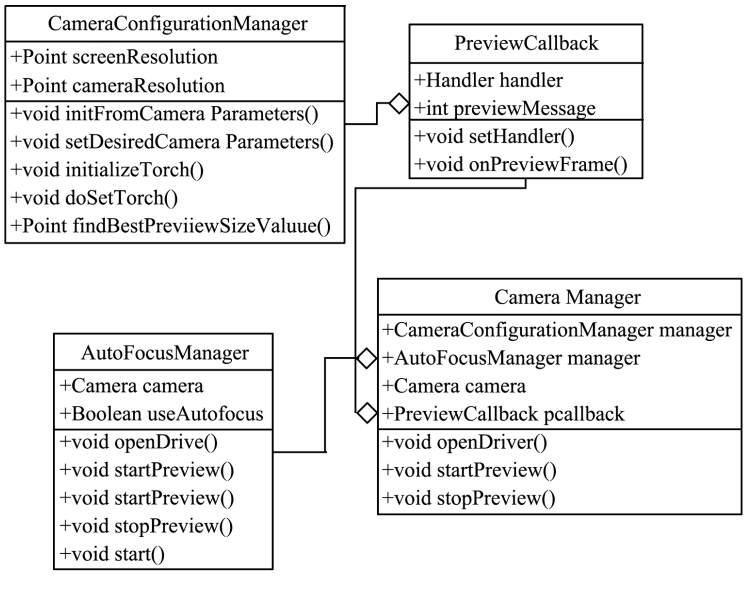

Fig. 2. Diagram of the main classes for working with the camera

The description of these basic classes of the block of working with the camera is provided in Table 1.

The unit of working with the camera consists of one main class: CameraManager, which describes the work of the camera in the mode of taking images for OCR, and three auxiliary classes, which describe the logic of camera settings and operating.

Table 1

Description of classes of the unit of working with the camera

Class	Description
CameraManager	Basic class that describes the operation of the camera in the image capture mode for recognition
AutoFocusManager	Class describing camera operation in autofocus mode
CameraConfiguratonManager	Class that contains all the logic of settings for the camera
PreviewCallback	Class that is responsible for receiving the image

The main classes diagram, which is responsible for pre-processing the image received from the camera, or from the scanning device is shown in Fig. 3.

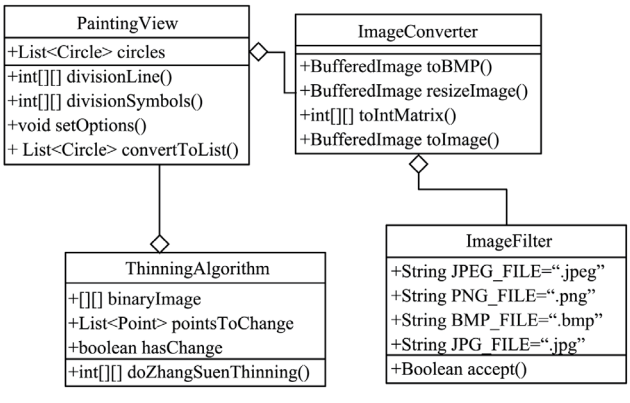

Fig. 3. Diagram of the main classes for working with the image

The description of the basic classes of the image pre-processing unit is provided in Table 2.

Table 2

Description of the classes of the image pre-processing unit

Class	Description
ThinningAlgorithm	Basic class class that describes the work of algorithms for image processing
ImageFilter	Class that describes the choice of an appropriate image processing algorithm in accordance with its format
ImageConverter	Class that contains the logic of converting an image into a corresponding pixel matrix model
PaintingView	Class that is responsible for the image division into lines and characters

The image preprocessor consists of one main class: ThinningAlgorithm, which describes the basic algorithms for image processing. Three auxiliary classes that are responsible for choosing the appropriate image processing algorithm according to its format, for obtaining the matrix

of pixels of the input image and for splitting it into lines and symbols.

A simplified diagram of classes of the developed software is shown in Fig. 4. It only shows the interaction of classes that describe cellular automata, that is, only that which is important for the recognition process.

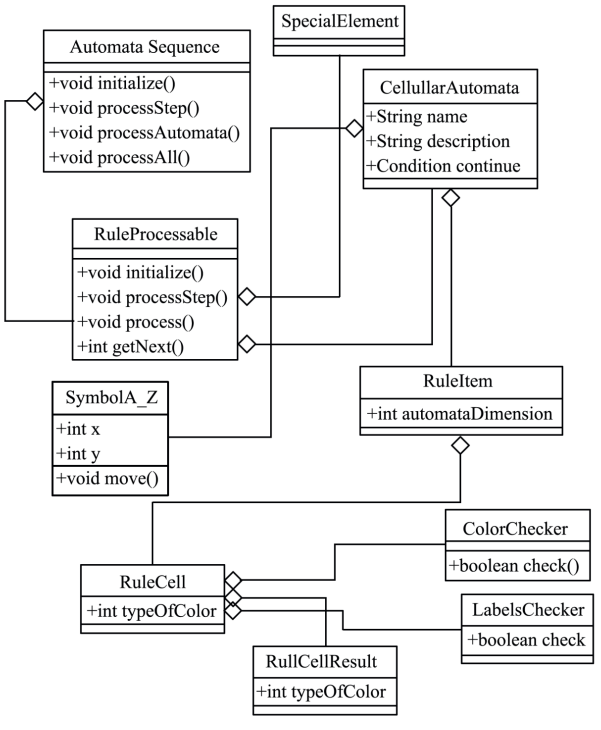

Fig. 4. Part of the class diagram of the developed software, which implements the work with cellular automata

Table 3 describes the classes of the unit of cellular automata.

Table 3

Description of classes of the unit of cellular automata

Class	Description
CellularAutomata	Basic class describing the work of the CA
RuleItem	Class that describes the rules of transition of the CA
RuleCell	Class describing the condition imposed on the cell under the CA rule
ColorChecker	Abstract class that checks the color of the cell
LabelsChecker	Abstract class that checks the cell labels in the transition graph
RullCellResult	Class that describes the events determining color and cell labels within the CA rule
AutomataSequence	Basic class that describes the work of the sequence of CA and the check of its competition
RuleProcesable	Abstract class that describes the work of a sequence item
SpecialElement	Abstract class that describes the work of a special item of the sequence
SymbolA_Z	Class that describes the characters of the Latin alphabet with the help of CAs

The system consists of two basic classes: CellularAutomata and AutomataSequence, which describe the work of the CAs and their sequences to launch the competition mechanism. The cellular automaton interactivity includes such elements as separating the image into separate characters, checking the conditions in the transition graph, etc. The states of each automaton that corresponds to a certain letter of the alphabet and the rule of transition through the graph are implemented in the RuleItem, RuleCell and SymbolA_Z classes. To simplify the diagram, descriptions of all letters are shown in one class, although they are actually implemented separately. These classes are responsible for implementation of movement of the CAs and describe the transition graph. An index of the type of automaton that allows identifying it unambiguously is a color label. By reading this label, we can determine the recognized letter. The RuleCellResult, LabelsChecker, ColorChecker classes correspond to this process. The result of this work will be a text recognized by competing cellular automata.

Interaction with the hardware and the output of the recognized text is performed using the standard functions of the Windows API. Image processing was performed using OpenCV open source libraries.

6. Description of the interface of the recognition system

The developed software has a very simple interface, since it is designed for testing the developed algorithms and methods of recognition only, and not for commercial use. The software consists of one window, which is divided into two blocks. The upper block loads the captcha image obtained from Captcha generators or simply saving the image from the browser. The lower block displays the result.

The main window of the program in the recognition mode is shown in Fig. 5.

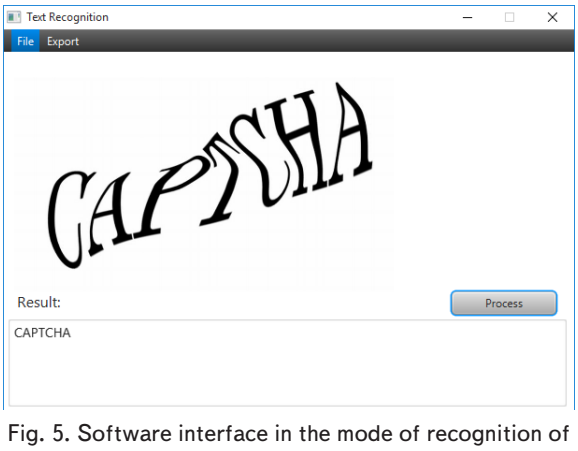

Fig. 5. Software interface in the mode of recognition of Captcha systems

The system is designed to recognize the displaced objects and close arrangement of characters, drawn in different fonts, and fuzzy images, that consist of deformed characters, united by several groups.

7. Discussion of captcha image recognition research results

The quality of the work of the developed system was evaluated by means of captcha image recognition on the personal computer of the following configuration:

1. Processor – Intel(R) Core(TM) i7-3612QM CPU @ 2.10 GHz.

2. RAM – 8 GB.

3. Video adapter – ATI AMD Radeon HD 7600M (1024 MB).

4. HDD – Seagate ST1000LM (931GB, SATA II).

5. DVD-RW – LG DVD+-RW.

This computer is running Windows 10 Professional.

Captcha generators [14] have been used to generate the incoming images. The results of the research are shown in Fig. 6 in the form of a graph of the dependence of recognition quality on the degree of deformation.

Recognition quality is the averaging of the data obtained from ten independent experiments.

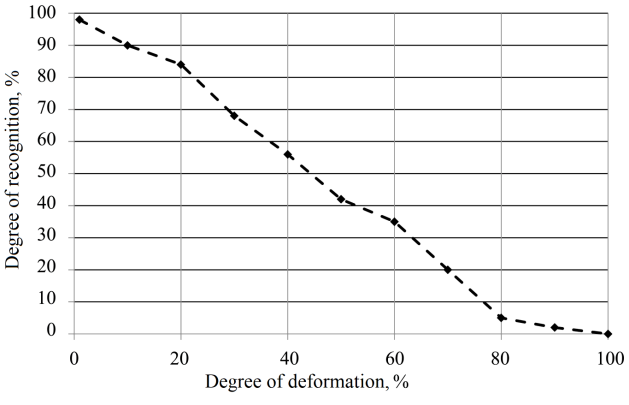

Fig. 6. Characteristics of Captcha Characters Recognition by the Developed Software Depending on the Degree of Deformation

Analysis of Fig. 6 shows that the dependence of the degree of recognition of CAPTCHA characters by the developed software on the degree of deformation is almost linear and consists of three sections. The first section – from 0 to 20 % deformation, where the system recognizes more than 80 % of the provided characters, that is, shows rather good results. The second section – from 20 % to 80 % of deformation, where the slope of the graph increases, shows a gradual decrease in the probability of recognition, and at 45 % deformation it recognizes only 50 % of the characters. Further, the decrease continues and the system almost ceases to correctly recognize Captcha with 80 % deformation (showing only 5 % of correctly recognized characters). With a further increase in the degree of deformation, the probability of correct recognition decreases to zero.

Thus, one can argue that the developed method can be successfully (up to 70 % probability of correct recognition) used for Captcha recognition, the characters of which are deformed not more than for 30 %. It should be noted, however, that the existing Captcha recognition systems cannot work with deformed characters at all, which is why the deformation has been introduced. The fact that the system developed in this paper can handle partially deformed and superimposed characters can be considered as a significant advantage.

The disadvantages include the fact that the degree of confident recognition is limited to 30 % of character deformations. Of course, this is not enough. Further improvement of the mechanism of competing CAs should increase this area; however, it is clear that successful recognition of completely deformed characters (or those having common lines) is impossible without the involvement of Machine Learning, which is not the subject of this work. In future, the combination of the theory of CAs with the means of Machine Learning can lead to a significant breakthrough in recognition systems, working in extreme conditions with low-quality recognition objects.

7. Conclusions

1. A new class of movable CAs has been introduced. The motion of the CAs is described by the transition rules:

$$\sigma_i(\omega_p, t, \xi) \overset{t \to t+1}{\rightarrow} \sigma_i(\omega_{p+1}, t+1, \xi),$$

which makes it possible to compare the trajectory of motion with the states of the character described by the CA.

2. A mechanism of competition of the CAs is developed, which consists in the fact that the automaton with a maximum number of the implemented states on a particular character "wins" the competition among all that simultaneously move through the character and is recognized as the most correctly describing the current character.

3. The architecture and streamlined single-window interface system for Captcha characters recognition based on competing CAs were developed for the study of the adequacy of the model and quality of recognition.

4. It is shown that the developed system demonstrates a high probability of correct recognition of Captcha characters (up to 70 %) at low degrees of deformation (up to 30 %). At higher degrees of deformation, the probability of correct recognition significantly decreases.

References

1. T'yuring A. M. Vychislitel'nye mashiny i razum. Samara: Bahrah-M, 2003. 128 p.
2. Wolfram S. A. New Kind of Science. Wolfram Media. Inc., 2002. 1197 p.
3. Oliveira C. C., de Oliveira P. P. B. An Approach to Searching for Two-Dimensional Cellular Automata for Recognition of Handwritten Digits // Lecture Notes in Computer Science. 2008. P. 462–471. doi: https://doi.org/10.1007/978-3-540-88636-5_44
4. Suyasov D. I. Retrieving structural features from symbol images based on the cellular automata with labels // Informacionno-upravlyayushchie sistemy. 2010. Issue 4. P. 39–45.
5. A new JPEG Image Watermarking Algorithm Based on Cellular Automata / Wu H., Zhou J., Gong X., Wen Y., Li B. // Journal of Information & Computational Science. 2011. Vol. 8, Issue 12. P. 2431–2439.
6. Belan S. N. Specialized cellular structures for image contour analysis // Cybernetics and Systems Analysis. 2011. Vol. 47, Issue 5. P. 695–704. doi: https://doi.org/10.1007/s10559-011-9349-8
7. Hussain R., Gao H., Shaikh R. A. Segmentation of connected characters in text-based CAPTCHAs for intelligent character recognition // Multimedia Tools and Applications. 2016. Vol. 76, Issue 24. P. 25547–25561. doi: https://doi.org/10.1007/s11042-016-4151-2

8. Recognition based segmentation of connected characters in text based CAPTCHAs / Hussain R., Gao H., Shaikh R. A., Soomro S. P. // 2016 8th IEEE International Conference on Communication Software and Networks (ICCSN). 2016. doi: https://doi.org/10.1109/iccsn.2016.7586608
9. Abdullah Hasan W. K. A Survey of Current Research on CAPTCHA // International Journal of Computer Science & Engineering Survey. 2016. Vol. 7, Issue 3. P. 1–21. doi: https://doi.org/10.5121/ijcses.2016.7301
10. Anti-captcha. URL: https://anti-captcha.com/mainpage/
11. Myroniv I. Development of the character recognition software on the base cellular authomata // VI-th International Conference of Students, PhD-Students and Young Scientists "Engineer of XXI Century". 2016. P. 229–240.
12. OpenCV library. URL: https://opencv.org/
13. Leonenkov A. V. Samouchitel' UML. Sankt-Peterburg: BHV Peterburg, 2004. 576 p.
14. Fake Captcha is the #1 free fake captcha maker! URL: https://fakecaptcha.com/

Показано, що словники предметних областей широко використовуються на різних етапах створення і експлуатації програмних продуктів. Процес створення словника, особливо виділення термінів, досить трудомісткий та вимагає високої кваліфікації експерта. Проведено дослідження по виявленню найбільш важливих характеристик багатослівних термінів, таких як: ймовірності присутності в документі термінів, що містять різну кількість слів; розташування іменників в багатослівних термінах; можливу кількість іменників в багатослівних термінах. Проаналізовано контекст використання термінів та визначено можливі межі термінів в тексті. Запропоновано процедуру попереднього групування документів, що дозволяє уникнути «втрати» термінів, що входять в короткі документи. Визначено залежність помилок при виділенні термінів від розміру аналізованого документа.

Запропоновано математичну модель представлення терміна, що заснована на визначенні безлічі ланцюжків слів, згрупованих близько опорного слова – іменника. Фільтрація ланцюжків виробляється в залежності від частоти їх входження в текст на основі зіставлення нормалізованих уявлень багатослівних термінів.

Розроблено механізми заповнення словника предметної області новими записами і коригування існуючих у міру аналізу вхідного документа. Запропоновано рішення щодо коригування частоти появи термінів на основі виявлення міжфразових зв'язків. Всі процеси і моделі об'єднані в єдину інформаційну технологію створення словника предметної області. Проблема визначення тлумачень термінів в даній роботі не розглядається, оскільки вимагає окремого рішення. Розроблено програмний продукт, що дозволяє в значній мірі автоматизувати процес виділення термінів з текстових документів. Результати апробації запропонованих рішень показали відсутність «загублених термінів» і, як результат, скорочення часу виділення термінів з текстів обсягом в 10000 слів на 1.5 години за рахунок звільнення експерта від аналізу вихідного документа. Результати дослідження можуть бути використані на різних етапах створення і експлуатації програмних продуктів

Ключові слова: словник предметної області, багатослівний термін, морфологічний розбір, математична модель терміна, текстовий документ

UDC 004.4'413
DOI: 10.15587/1729-4061.2018.147978

DEVELOPMENT OF INFORMATION TECHNOLOGY OF TERM EXTRACTION FROM DOCUMENTS IN NATURAL LANGUAGE

O. Kungurtsev
PhD, Associate Professor*
E-mail: abkun@te.net.ua
S. Zinovatnaya
PhD, Associate Professor*
E-mail: zinovatnaya.svetlana@opu.ua
Ia. Potochniak
Postgraduate student*
E-mail: yana.onpu@gmail.com
M. Kutasevych*
E-mail: masteryoda290@gmail.com
*Department of System Software
Odessa National
Polytechnic University
Shevchenka ave., 1,
Odessa, Ukraine, 65044

1. Introduction

Domain dictionaries (DD) are widely used in software design [1]. In particular, when determining the roles of members of the development team [2]; when constructing data dictionaries [3, 4]; in the problems of selection and clustering of materialized database representations [5, 6]. Based on DD, job descriptions and many other documents