UDC 004.4

RESEARCH OF THE SOFTWARE TOOL FOR TRANSLATING NATURAL LANGUAGE BUSINESS RULES INTO SMART CONTRACTS

***Shynkarenko D.V.*[1], *Kopp A.M.*[2]**

[1] Master's Student of the SE & MIT Department, NTU «KhPI», Kharkiv, Ukraine
[2] Associate Professor of the SE & MIT Department, Ph.D., NTU «KhPI», Kharkiv, Ukraine
shinkarenko.cmds@gmail.com

In modern information technology, various intelligent technologies have become the most relevant area. Systems based on artificial intelligence have found wide application in a large number of systems designed to perform a wide range of tasks. One type of such intelligent system is an intelligent information system (IIS) designed to support human activity and information retrieval. However, the work of IIS faced the problem of processing large amounts of unstructured data that were not suitable for computer understanding. An appropriate solution to this problem was the transformation of data using smart contracts. The problem of the development of a software tool for smart-contract generation using natural language processing (NLP) tools becomes relevant because, with the bigger globalization of the world, more and more issues appear due to the inability to formulate business rules concisely and clearly and then record them to be accessible by all parties involved.

This work set out to research and develop software solutions to generate smart contracts based on business rules using NLP tools. The main purpose of the work is to simplify the process of developing software components for decentralized systems by generating smart contracts based on business rules.

In the research work, analyzed existing solutions and publications. During this analysis, several blockchain platforms and existing publications were researched [1–2]. Based on that, formulated functional and non-functional requirements for a developed product [3].

Researched existing business processes for creating smart contracts and purposed improvements to the flow of business processes regarding smart contract creation with the use of developed software. Created a mathematical model of the problem of creating smart contracts based on business rules using NLP tools. Conducted comparison analysis of different NLP algorithms for sentence tokenization and based on this designed an algorithm to process business rules input in natural language and generate smart contracts based on them.

Analyzed different software development architectures and selected the most fitting one according to the requirements specified. Next, the appropriate database management system was chosen and the database was modeled according to the software design. Developed software to fit the requirements of the work using previously chosen architecture and database.

Finally, the developed software components were tested and the results were analyzed. The results of the testing algorithm with different business rules can be seen in Table 1 and 2.

Table 1 – Algorithm testing results (part 1)

Business rule	Result
Lets create token Gregt that has symbol GGT and supply 5560	Pass
Let's produce the Gregt token, with the symbol GGT, and distribute supply of 5560.	Pass
Create the Gregt token with the symbol GGT and distribute 5560.	Pass
Let's make Gregt a token with the symbol GGT and issue 5560.	Pass
Let's make the Gregt token, with the sign GGT, and distribute 5560.	Error
Create the Gregt token with the identifier GGT and distribute 5560.	Error

Table 2 – Algorithm testing results (part 2)

Business rule	Result
For our company GTechStartup, lets create token Gregt that has symbol GGT and supply 5560	Error
Lets create token Gregt that has symbol GGT and supply 5560 for our startup THK	Pass
Let's make Gregt a token with the symbol GGT and issue 5560, which we will distribute among 100 employees.	Defect

According to those results, the following metrics were determined:

$$ErrorRate = \frac{Error}{Defect + Error + Success} = \frac{3}{1+3+5} = 0.33,$$
$$DefectRate = \frac{Defect}{Defect + Error + Success} = \frac{1}{1+3+5} = 0.11, \qquad (1)$$
$$SuccessRate = \frac{Success}{Defect + Error + Success} = \frac{5}{1+3+5} = 0.56,$$

where *ErrorRate* – portion of business rule processing errors *Error*; *DefectRate* – portion of incorrect business rule processing cases *Defect*; *SuccessRate* – portion of successful business rule processing cases *Success*.

According to the results in (1), there is room for improvement in the algorithm to drive error (33%) and defect (11%) rates down.

Software components were also tested for load handling with multiple concurrent users and starting from 250, there start to appear connection errors (fig 1).

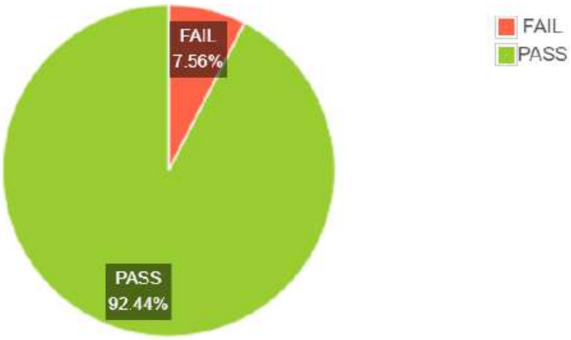

Figure 1 – Pie chart of request results with 250 concurrent users

Currently, these software components still have room for improvement, for example, to have a built-in system to interact with generated smart contracts. Also, there is progress to be made to lower error rates both in the algorithm and in load handling.

References:

1. *Tateishi, T.* Automatic smart contract generation using controlled natural language and template / *T. Tateishi, S. Yoshihama, N. Sato, S. Saito* // IBM Journal of Research and Development. – 2019. – Vol. 63 – P. 1 – 6.

2. *Tong, Y.* Smart Contract Generation Assisted by AI-Based Word Segmentation / Y. *Tong, W. Tan, J. Guo, B. Shen, P. Qin, S. Zhuo* // Applied Sciences. – 2020. – Vol. 12, No. 9. – P. 4773.

3. *Kopp, A. M.* Smart contract code generation based on natural language business rules for cryptocurrency tokens creation / *A. M. Kopp, D. V. Shynkarenko* // Computer Games & Multimedia. – Odesa, ONTU Publishing house, 2022. –P. 89 – 85.