






РАЗДЕЛ III. ИСПОЛЬЗОВАНИЕ НЕЙРОННЫХ СЕТЕЙ
В ОБРАБОТКЕ ЕСТЕСТВЕННОГО ЯЗЫКА
SECTION III. NEURAL NETWORKS IN NATURAL LANGUAGE PROCESSING

UDC 004.5

DOI: 10.18413/2313-8912-2023-9-1-1-1

Alexander G. Sboev¹ 
Artem V. Gryaznov² 
Roman B. Rybka³ 
Maxim S. Skorokhodov⁴ 
Ivan A. Moloshnikov⁵ 

A deep learning method based on language models
for processing natural language Russian commands
in human robot interaction

¹ Kurchatov Institute National Research Center
1 Academician Kurchatov Sq., Moscow, 123098, Russia
E-mail: Sboev_AG@nrcki.ru

² Kurchatov Institute National Research Center
1 Academician Kurchatov Sq., Moscow, 123098, Russia
E-mail: Gryaznov_AV@nrcki.ru

³ Kurchatov Institute National Research Center
1 Academician Kurchatov Sq., Moscow, 123098, Russia
E-mail: Rybka_RB@nrcki.ru

⁴ Kurchatov Institute National Research Center
1 Academician Kurchatov Sq., Moscow, 123098, Russia
E-mail: Skorokhodov_MS@nrcki.ru

⁵ Kurchatov Institute National Research Center
1 Academician Kurchatov Sq., Moscow, 123098, Russia
E-mail: Moloshnikov_IA@nrcki.ru

Received 23 January 2023; accepted 09 March 2023; published 30 March 2023

Abstract. The development of high performance human-machine interface systems for controlling robotic platforms by natural language is a relevant task in interdisciplinary field «Human-Robot Interaction». In particular, it is in demand, when the robotic platform is controlled by an operator without any skills necessary to use specialized control tools. The paper describes a complex Russian language commands processing into a formalized RDF graph format to control a robotic platform. In this processing, neural network models are consistently used to search and replace pronouns in commands, restore missing verbs-actions, decompose a complex command with several actions into simple commands with only one action and classify simple command attribute. State-of-the-art solutions are applied as neural network models in this work. It is language models based on deep neural networks transformer architecture. The previous our papers show synthetic datasets






based on developed generator of Russian language text commands, data based on crowdsourcing technologies and data from open sources for each of the described stages of processing. These datasets were used to fine-tune the language models of the neural networks. In this work, the resulting fine-tuned language models are implemented into the interface. The impact of the stage of searching and replacing pronouns on the efficiency of command conversion are evaluated. Using the virtual three-dimensional robotic platform simulator created at the National Research Center «Kurchatov Institute», the high efficiency of complex Russian language commands processing as part of a human-machine interface system is demonstrated.

Keywords: Human-robot interaction; Natural language processing; Deep learning; Artificial intelligence; Human-robot interface

How to cite: Sboev, A. G., Gryaznov, A. V., Rybka, R. B., Skorokhodov, M. S. and Moloshnikov, I. A. (2023). A deep learning method based on language models for processing natural language Russian commands in human robot interaction, *Research Result. Theoretical and Applied Linguistics*, 9 (1), 174-191. DOI: 10.18413/2313-8912-2023-9-1-1-1

УДК 004.5

DOI: 10.18413/2313-8912-2023-9-1-1-1

Сбоев А. Г.¹ 
Грязнов А. В.² 
Рыбка Р. Б.³ 
Скороходов М. С.⁴ 
Молошников И. А.⁵ 

Метод глубокого обучения на основе языковых моделей для обработки русскоязычных команд естественного языка при взаимодействии человека и робота

¹ Национальный исследовательский центр «Курчатовский институт»
пл. Академика Курчатова, 1, Москва, 123098, Россия
E-mail: Sboev_AG@nrcki.ru

² Национальный исследовательский центр «Курчатовский институт»
пл. Академика Курчатова, 1, Москва, 123098, Россия
E-mail: Gryaznov_AV@nrcki.ru

³ Национальный исследовательский центр «Курчатовский институт»
пл. Академика Курчатова, 1, Москва, 123098, Россия
E-mail: Rybka_RB@nrcki.ru

⁴ Национальный исследовательский центр «Курчатовский институт»
пл. Академика Курчатова, 1, Москва, 123098, Россия
E-mail: Skorokhodov_MS@nrcki.ru

⁵ Национальный исследовательский центр «Курчатовский институт»
пл. Академика Курчатова, 1, Москва, 123098, Россия
E-mail: Moloshnikov_IA@nrcki.ru

*Статья поступила 23 января 2023 г.; принята 09 марта 2023 г.;
опубликована 30 марта 2023 г.*

Аннотация. Реализация высокопроизводительных человеко-машинных интерфейсов для управления робототехническими платформами с помощью естественного языка представляется современной задачей междисциплинарной области взаимодействия человека и робота. В частности, это востребовано в том случае, когда управление платформой осуществляется оператором, который не обладает навыками, необходимыми для использования специализированных инструментов управления. В данной работе описан процесс преобразования сложных русскоязычных команд естественного языка в формализованный графовый RDF формат для взаимодействия с робототехнической платформой. В этом процессе последовательно задействованы модели нейронных сетей для поиска и замены местоимений в командах, восстановления пропущенных глаголов-действий, декомпозиции сложных команд с несколькими действиями на простые команды с одним действием, классификации атрибутов простых команд. В качестве моделей нейронных сетей используются современные решения – языковые модели, основанные на архитектуре глубоких нейронных сетей «трансформер». Для каждого из описанных этапов, в предыдущих наших работах были составлены собственные наборы данных на основе разработанного генератора русскоязычных текстовых команд, дополнительно были использованы технологии краудсорсинга и данные из открытых источников. На этих наборах данных была проведена точная настройка языковых моделей нейронных сетей. В предлагаемой работе, полученные настроенные языковые модели были имплементированы в управляющий интерфейс, и оценено влияние этапа поиска и замены местоимений на эффективность преобразования команд. На базе разработанной в Национальном исследовательском центре «Курчатовский институт» виртуальной трехмерной модели робототехнического устройства было продемонстрировано, что процесс преобразования сложных русскоязычных команд в составе человеко-машинного интерфейса позволяет эффективно управлять робототехнической платформой при помощи естественного языка.

Ключевые слова: Взаимодействие человека и робота; Обработка естественного языка; Глубокое обучение; Искусственный интеллект; Интерфейс человек-машина

Информация для цитирования: Сбоев А. Г., Грязнов А. В., Рыбка Р. Б., Скороходов М. С., Молошников И. А. Метод глубокого обучения на основе языковых моделей для обработки русскоязычных команд естественного языка при взаимодействии человека и робота // Научный результат. Вопросы теоретической и прикладной лингвистики. 2023. Т. 9. № 1. С. 174-191. DOI: 10.18413/2313-8912-2023-9-1-1

Introduction

State-of-the-art approaches to the development of human-machine interfaces for controlling robotic platforms are used to process commands in natural language and analyze information from different sensors. A field condition control, including by inexperienced operators, requires the creation of a flexible and accurate system for processing commands in natural language into

a formalized format of commands recognized by a robotic platform.

The command representation can be formalized as a logical representation or a graph reflecting the semantic relation between entities. A popular formalized format is the semantic graph data representation model RDF (McBride, 2004). It operates with statements of the form “subject” - “predicate” - “object”. The set of statements forms a

directed graph with vertices "subjects" and "objects", and the edges reflect "predicates". The main difference between a formalized format and a natural language is the absence of ambiguities in the command interpretation. It poses a relevant task of developing a system for processing Russian language commands into a formalized format of commands, including the processing of anaphors, free word order, neologism, homonyms, synonyms etc.

Natural language processing methods are divided into rule-based approaches and methods using neural network language models. Rule-based approaches include writing hard-coded instructions based on semantic analysis. Systems based on such approaches are very sensitive to input data, require a lot of development time, and face difficulties in handling ambiguous phrases.

Language models based on neural networks have a complex architecture, often models have a transformer topology (Vaswani et al., 2017). These models are pre-trained on huge sets of text corpora, solving problems of determining the next word, recovering missing words, etc. The pre-trained language models are trained using fine-tuning (changing the neural network model parameters), prompt-tuning (training embeddings using gradient descent without changing the language model parameters) and few-shot (preparing a hint on natural language without model training), which require a significantly smaller amount of data corpora than at the preliminary pre-trained stage. Using methods based on neural networks is a promising area of focus in natural language processing. Compared to rule-based approaches, language model training requires less time for its development.

Our work presents a deep learning method based on language models with a transformer architecture for processing Russian language commands into a formalized RDF graph format for controlling a robotic platform. The method includes a model for finding and replacing pronouns and a neural network interface for processing

complex Russian-language commands. The neural network interface, presented in our previous paper (Sboev et al., 2022), includes the steps of restoring missing verbs, decomposing complex commands into simple ones, and classifying the attributes of simple commands. Section 1 describes the state-of-the-art on the topic of controlling robotic platforms using natural language based on neural networks. Section 2 provides a description of the data used in model training. Section 3 shows the model architectures. Section 4 describes the complete system of processing a Russian command on natural language and precision when controlling a robotic platform.

1 Related works

At the present time, there are no publications in the literature on the topic of controlling a robotic platform using Russian commands on natural language. At the same time, there are a number of works in English (Gubbi, Upadrashta and Amrutur, 2020; Min et al., 2021; Ahn et al., 2022; Choi et al., 2021) related to the control of robotic platform using language models of neural network.

In the paper (Gubbi, Upadrashta and Amrutur, 2020), the authors use neural network models. The first model is LSTM (Hochreiter and Schmidhuber, 1997) (Long Short-Term Memory). The second model is BERT (Devlin et al., 2018) with a transformer topology. These models transform natural language into Python 3 programming language functions (Van Rossum and Drake, 2009) to control a robotic arm with grip and speed control Dobot Magician.¹ Dobot Magician is a universal platform for in-depth study of industrial robotics. The authors show that their method works better than training neural network models to predict the actions of a robotic platform directly.

Another work (Min et al., 2021) uses the above-mentioned language models BERT with transformer topology to classify

¹ DOBOT, available at: <https://en.dobot.cn/> (Accessed 08 November 2022).

command attributes in natural language and compile a sequence of subtasks to control a home assistant robot in a virtual three-dimensional environment. The assistant robot is capable to perform various sub-tasks: pick up and place, take and place, clean and place, heat and place, examine under light. BERT language models are part of a module control system, which allows controlling a robotic platform without expert trajectories or low-level instructions. The module system includes the Mask R-CNN convolutional neural network (He et al., 2017), as well as search and navigation policies (Chaplot et al., 2020) to control the assistant robot.

The work (Ahn et al., 2022) proposes a neural network approach called SayCan using the knowledge of the large language model PaLM (Chowdhery et al., 2022) to control the robot from Everyday Robots company.² The robot has a mobile manipulator with a hand with 7 degrees of freedom and two-fingered grip, and is additionally equipped with RGB camera. For the robot, 3 types of skills are described: take object, find and move to object, pick up and bring an object. The Say method of the SayCan approach proposes low-level tasks grounding to the capabilities of the robot. The Can method chooses, according to the plan, which of the proposed tasks to perform. The authors evaluate the obtained approach on a real robotic device for a number of tasks. As a result, the PaLM language model handles ambiguous natural language commands.

The authors of the paper (Choi et al., 2021) propose an adaptation based on language models GPT-2 (Radford et al., 2019) and T5 (Raffel, Shazeer and Roberts, 2019) with transformer topology for processing natural language instructions when controlling industrial robotic devices. In work, the stage of validation of commands coming from a human (feedback) based on the ICARUS cognitive architecture (Choi and Langley, 2018) is proposed. This architecture

assumes a relational representation of knowledge, distinguishes between long-term and short-term memory, in other words, uses knowledge about the environment and the state of the industrial robot, as well as information about the possible actions of the robot to process language instructions and form feedback with the operator if the robot cannot perform task at the moment. The authors show a pre-trained language model can be efficiently fine-tuned to translate verbal instructions into robot platform tasks better than semantic text analysis method.

2 Datasets

Synthetic Dataset

We used our text command generator described in (Sboev et al., 2022) to train neural network models for finding and replacing pronouns, decomposing complex commands into simple ones, and classifying the attributes of simple commands. The generator was modified in this work. It uses dictionaries with synonyms and tools of the pymorphy2 library (Korobov, 2015) such as matching words with numerals and bringing words into the necessary speech form. The modified generator performs the following functions:

1. Creation of commands set in natural language based on given templates.
2. Compilation of vector and labeled representations for each command.
3. Creation of complex commands set in the form of sequences of given templates.

Simple commands are natural language commands consisting of a single action type (see Table 1). A total of 16 templates were compiled (detailed in Appendix A.1) for various commands. These commands are supported by the robotic platform. The generator makes a vector representation depending on the used template and synonyms dictionary to classify the attributes of each command type. In addition to the vector representation, the generator makes a labeled representation (see Table 2). A marker is assigned based on the synonyms dictionary for each word in the generated command (detailed in Appendix A.2).

² Everyday Robots, available at: <https://everydayrobots.com/> (Accessed 08 November 2022).

Table 1. Simple commands templates

Таблица 1. Шаблоны простых команд

№	Template	Example
1	patrol	go on patrol
2	execution termination	stop
3	execution interruption	pause
4	execution continuation	start moving again
5	movement in the direction	move forward

Table 2. Vector and labeled representations

Таблица 2. Векторное и маркированное представления

Attribute	Vector	Marker
Action	0-14	A
Direction	0-8	D
Meters	0-12	M
Degrees	0-24	DS
Hours	0-24	HS
Object 1	0-16	O1

Complex commands are natural language commands contained sequences of simple commands (see Table 3). In total, 4 templates were compiled, including sequential commands separated by keywords, as well as commands using pronouns.

Additionally, the template of complex commands with pronouns was modified for the task of finding and replacing pronouns in accordance with Table 4.

Table 3. Complex command templates

Таблица 3. Шаблоны сложных команд

№	Template	Example
1	Sequential 2 commands	Drive to the house, then drive to the human
2	Sequential 3 commands	Analyze this stone, after that drive up to the house, and then find a person
3	Commands "find-move"	Find a tree. Behind the tree is a person to whom you need to drive up
4	Commands with pronouns	Turn to the nearest house and inspect it

Table 4. Markup Variants of Training Output for Pronoun Processing

Таблица 4. Варианты разметки выходных данных для обработки местоимений

№	input sequence	output sequence
1	Move to the house and look at it	Command: move to the house and look at the house
2	Turn to this human, inspect him	Aim: human him

Solving the problem of finding and replacing pronouns in commands is a special case of the coreference problem (search for anaphoras in a text). In addition to the dataset (see Table 4), a corpus was prepared in accordance with the competition (Budnikov et al., 2019) used for evaluation by coreference metrics. In this competition, each word in the text is assigned the position of the word Offset, the number of characters Length and the number of the Chain ID.

Crowdsourced Dataset

Testing of neural network models for decomposing complex commands and classifying attributes is carried out on a dataset collected by crowdsourcing technology. 250 exemplary commands were selected from a synthetic dataset, after that the participants of the crowdsourcing platform were given the task to rephrase these commands (see Table 5) in different ways (Sboev et al., 2022).

Table 5. Summary of Crowdsourced test commands

Таблица 5. Сводка тестировочных команд, собранных в процессе краудсорсинга

Command type	Number of examples
Command without attributes	51
Movement in the direction	89
Interaction with an object	53
Interaction with an object relative to 1 object	130
Interaction with an object relative to 2 objects	995

In addition to the command generator, natural language commands were collected and labeled for training and testing the attribute classifier of simple commands. The target group of participants received instructions with a brief description of the

commands that can be used to control the robotic platform. For each type of simple and complex command, participants compile 100 examples. Each instruction includes a set of possible attributes of a particular instruction and generator examples (see Table 6).

Table 6. Sample instructions for participants

Таблица 6. Пример инструкции для участников

Command description	Movement for a given number of meters in a certain direction
Possible command attributes	action: move; direction: North, South, East, West, Right, Left, Forward, Back; number of meters: 5 meters, 5 m
Template	[action]+[direction]+[number of meters]
Generator examples	Go 2m; Ride 24 m ahead; Start driving straight 14 meters

Open Data

In (Sboev et al., 2022) we used the dataset from the Dialog-21 conference (Smurov et al., 2019) to train the missing verb recovery model. The open dataset uses news,

fiction and technical texts, as well as texts from social networks. The content of the dataset is presented in Table 7, according to (Sboev et al., 2022). Additionally, ~115,000 sentences with automatic markup were used.

Table 7. Open dataset characteristics
Таблица 7. Характеристика открытых данных

Texts	Train	Test	Validation
With missing verb	5542	680	1382
Total	16406	2045	4142

3 Methods

Neural Network Models

Language models based on neural network models with transformer topology (Vaswani et al., 2017) are used for all the steps described below for the processing of Russian natural language commands into a formalised RDF graph.

At the stage of finding and replacing pronouns, language models for the Russian language RuT5 are considered based on the language model for generating text sequences (text-to-text) T5, presented in (Raffel, Shazeer and Roberts, 2019). Two versions of the RuT5 model are used. The first version is a «small» model.¹ This version model is based on a multilingual model (Xue et al., 2020) and the Mt5small-ruparaphraser.² The size of the dictionary is reduced from 250 000 to 20 000, the number of parameters is reduced to 65 million, so the total size of the model is 246 MB. The first 5 000 tokens in the new dictionary are taken from the original model, while the remaining 15 000 are tokens obtained by tokenizing the Russian language web corpus from the collection.³ The second version is a «base» model.⁴ This version model is based on the original English T5

model (Raffel, Shazeer and Roberts, 2019). The dictionary size is 32 000, the number of parameters is 222 million, the total size of the model is 892 MB. The model is trained on the Russian corpus, including Wikipedia, books, news, Russian Common Crawl, etc.⁵

The following models are used for the subsequent stages: recovering the missing verbs, decomposing the complex command into simple commands, and simple command attribute classification. The first model is the Multilingual BERT model based on multi head attention layers (Devlin et al., 2018), consisting of 12 transformer blocks and the hidden layer dimension 768. The second model is the RuBERT-tiny model⁶ based on the BERT model with the following changes: the size of the input dictionary is reduced from 119 000 до 30 000 tokens in Russian and English, the size of the vector representation layer is reduced from 768 to 312, the number of transformer layers is reduced from 12 to 3. The model is obtained as a result of a training procedure using the outputs of pre-trained large neural network models RuBERT (Kuratov and Arkhipov, 2019), LaBSE (Feng et al., 2022), Laser (Artetxe and Schwenk, 2019) and USE (Cer et al., 2018). The last model is the RuBERT-tiny2 model. It is an improved version of the previous RuBERT-tiny model. This model has a large dictionary size (83 000 instead of 30 000), supports longer sequences (2048

³ Dale, D. (2021). Hugging Face: rut5-small, available at: <https://huggingface.co/cointegrated/rut5-small> (Accessed 10 October 2022).

⁴ Fenogenova, A. (2021). Hugging Face: m5small-ruparaphraser, available at: <https://huggingface.co/alensch/mt5small-ruparaphraser> (Accessed 10 October 2022).

⁵ Leipzig Corpora Collection: Russian Web text corpus based on material from 2019. Leipzig Corpora Collection. Dataset, available at: https://corpora.uni-leipzig.de/?corpusId=rus-su_web_2019 (Accessed 24 June 2022).

⁶ Sberbank AI (2021). Hugging Face: rut5-base, available at: <https://huggingface.co/sberbank-ai/rut5-base> (Accessed 10 October 2022).

⁷ Zmitrovich, D. (2021). ruT5, ruRoBERTa, ruBERT: how we trained a series of models for the Russian language, available at: <https://habr.com/ru/company/sberbank/blog/567776/> (Accessed 10 October 2022).

⁸ Dale, D. (2021). Small and fast BERT for Russian language, available at: <https://habr.com/ru/post/562064/> (Accessed 24 June 2022).

instead of 512). For its training, mainly texts in Russian are used (Williams, Nangia and Bowsman, 2017).

Finding and replacing pronouns

Two versions of the model «RuT5-base» and «RuT5-small» are trained, differing in the number of parameters, described in the previous section «Neural Network Models». The fine-tuning of these models is carried out

using the examples from Table 4 with the generation of a full command, in which a target object is replaced in the point of the pronoun, and with the generation of the «object-pronoun» pair for the subsequent replacement of the pronoun with the proposed word in the original text command. During the training process, the hyperparameters values presented in Table 8 are set.

Table 8. Hyperparameters values of RuT5-small and RuT5-base neural network models

Таблица 8. Значения гиперпараметров Ru-T5-small и RuT5-base нейросетевых моделей

Hyperparameter	Value
Batch size	32
Epochs	10
Learning rate	1e-4
Max source text length	48
Max target text length	48

Another used neural network model with transformer topology is the RuBERT model. This model is an adaptation of the model (Joshi et al., 2019) based on the Russian language coreference model (Sboev, Rybka and Gryaznov, 2020). It is taken from Tensorflow library (Abadi et al., 2016) without changing the hyperparameter values. The model is fine-tuned over 5 epochs. For fine-tuning, we used the representation of a synthetic dataset when generating a complex command with pronouns (see Table 3) using the following example:

Text: [SPL]Move to house and look at it [SEP] → Link: [3, 3] [6, 6]

Restoring missing verbs

The method (Belkin, 2019) was used to solve this problem. This method is based on a neural network language model for processing sequences of texts (text-to-text). It classifies the input text tokens into five classes: cV, cR1, cR2, R1, R2. Here cV is a verb (or predicate) that is omitted in the following simple sentences as part of a complex one. cR1 and cR2 are correlates from the non-gap sentence, which are syntactically and

semantically similar to R1 and R2 – the remnants from the gap sentences (Sboev et al., 2022). Example:

«Index [cR1 industrial production cR1] for January-February 2008 [cV amounted to cV] [cR2 106.0% cR2], [R1 fixed capital investment R1] – [R2 120.2% R2] and [R1 retail trade turnover R1] – [R2 116.3% R2] »

The omission of a verb is marked as cV. The start of a gap is marked with either R2 or R1 depending if R2 was not found in the text.

Decomposing a complex command into simple commands

To solve this problem, a token classifier was used. A neural network language model is used as a classifier, which defines the following classes (Sboev et al., 2022):

1. O - the token does not apply to any of the commands.
2. [SEP] - the token is part of the current command.
3. [CMD] - the token is related to the current command, but all subsequent tokens are part of the next command.

Simple command Attribute

Classification

To solve this problem, 2 neural network language models based on the RuBERT-tiny2 model are used. The first model solves the problem of classifying tokens. The input string is split into tokens according to the vocabulary of the language model. The classification problem uses the values of vectors and markers from Appendix A.2. After classifying the tokens, substrings for each of the attributes are extracted from the input string.

The second model with the transformer topology solves the problem of matching the selected attribute substrings with the dictionaries of these attributes. For each attribute of the command, a dictionary is assembled containing pairs of a phrase and an attribute value. All dictionary phrases are encoded by the model, i.e. get one vector in correspondence. After that, the attributes selected in the input line are fed to the input of the model, and each attribute also receives one vector. This vector is compared with the vectors of the corresponding dictionary, from which the closest vector is selected according to the selected metric (cosine proximity)

Post-processing of model results:

1. For each action attribute value, the maximum cosine similarity among all phrases in the dictionary with that value is selected.
2. For every other attribute, there is a probability that it will be present in the command. These values are determined by the phrase with the highest similarity from the corresponding dictionary.

4 Experiments

The deep learning method for processing Russian-speaking natural language commands, including the successive steps described in detail in the previous section, is tested using a three-dimensional model of a robotic platform based on the ROS (Quigley et al., 2009) and Gazebo simulator (Koenig and Howard, 2004) developed at the Kurchatov Institute National Research Center. This platform (see Figure 1) is a model of an "arctic" robot and is equipped with the following sensors, transducers and actuators:

1. 4 wheel drives to move around the terrain.
2. Static camera with a resolution of 640x480 pixels.
3. Dynamic camera with a resolution of 640x480, with the ability to rotate in yaw and pitch.
4. Lidar matching real HDL-32E Lidar Sensor.
5. Manipulator for interacting with objects.

The environment of the robot (see Figure 2) is a map of the "Arctic" terrain with a non-uniform terrain and a set of objects that the Arctic robot can interact with: trees, stones, houses, people.

Control commands (the list of possible commands is described in Appendix A.1) are received by the robot in the form of the RDF format. In addition to the possible commands from Appendix A.1, the robot is also capable of executing complex commands – sequential combinations of simple commands in the RDF format.

Figure 1. Three dimensional «arctic» robot model
Рисунок 1. Трехмерная модель арктического робота

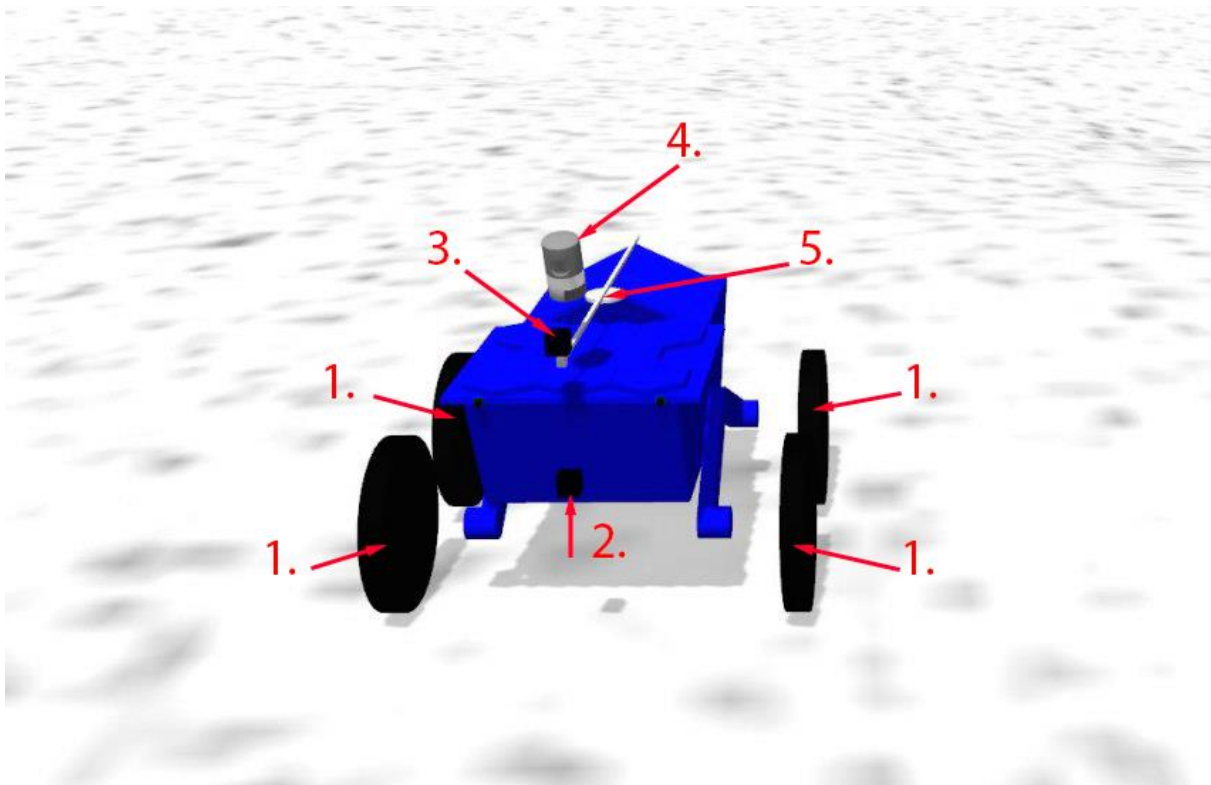
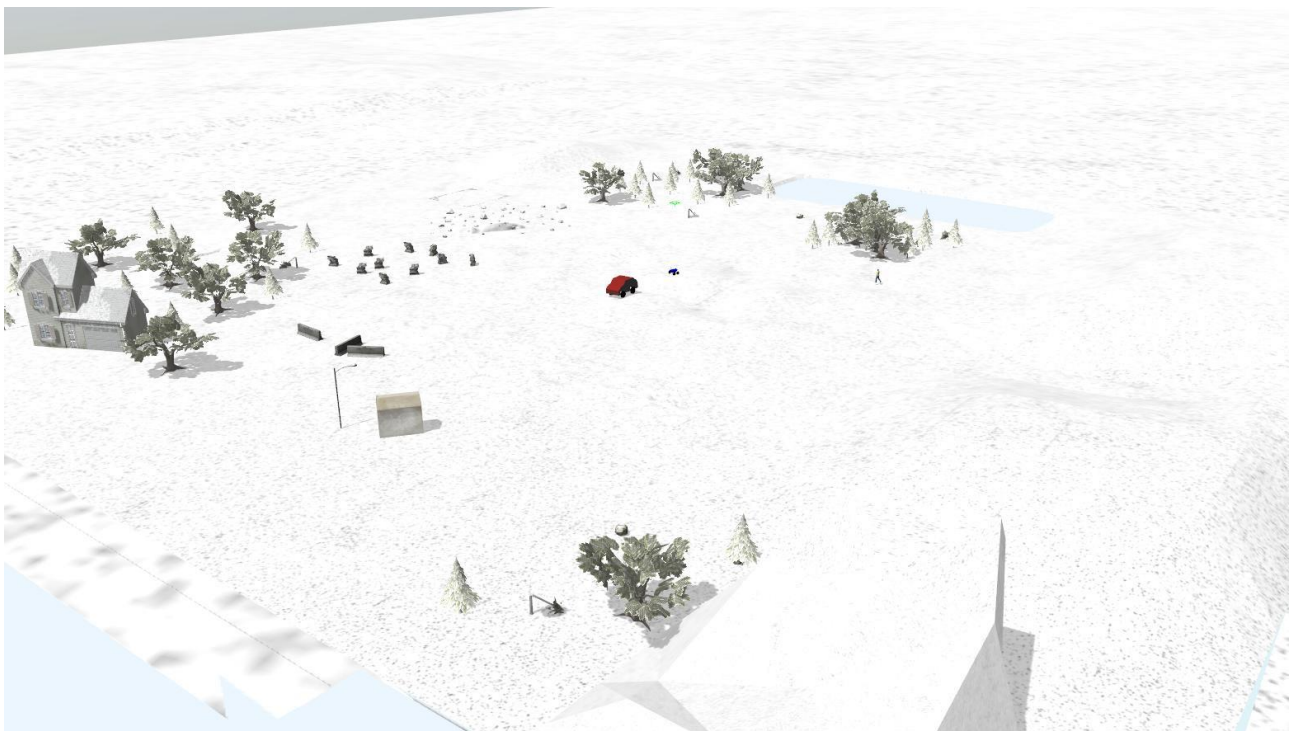


Figure 2. Three dimensional world of «arctic» robot in Gazebo simulator
Рисунок 2. Трехмерная модель мира арктического робота в симуляторе Gazebo

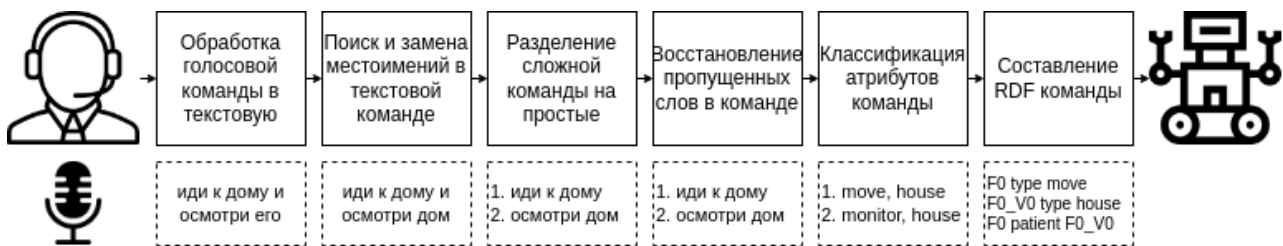


The processing of natural language commands involves the following steps in sequence (detailed in Figure 3):

1. Voice command conversion.
2. Finding and replacing a pronoun in a command.
3. Decomposing a complex command into simple commands.
4. Recovery of missing verbs.
5. Getting commands attributes.
6. Compiling an RDF command.
7. Sending a command to the system of a robotic device.

Figure 3. The system of processing a Russian language command

Рисунок 3. Система обработки русскоязычных команд



To analyze the effectiveness of the presented solution, the accuracy of command execution was assessed both with the use of search and replacement of pronouns in

complex commands, and without this step. While evaluating the accuracy, the attributes of the command are checked, mainly the class of the object (see Table 9).

Table 9. Effect of pronoun substitution in the processing of operator commands

Таблица 9. Влияние замены местоимений при обработке команд оператора

Command	Replaced attributes	Attributes without replacement
Move to house and look at it	1 house; 2 house	1 house; 2 human
Turn to the tree and walk up to it	1 tree; 2 tree	1 tree; 2 gaze
Find a stone and analyze it	1 rock; 2 rock	1 rock; 2 human
Drive to the house near the person and go around it	1 house; 2 house	1 house; 2 gaze human
Find a broken tree and approach it	1 broken-tree; 2 broken-tree	1 broken-tree; 2 gaze
Go to the nearest person and inspect him	1 human; 2 human	1 human; 2 human

Thus, 1 000 pronoun commands are tested for assessing the model coreference (see Section 2 “Datasets”). This test is carried out for each of the models described in

Section 3 "Finding and replacing pronouns". The accuracy of the correct definition of the object in the command with the pronoun is presented in Table 10.

Table 10. Evaluation of the recognition accuracy of commands with pronouns
Таблица 10. Оценка точности распознавания команд с местоимениями

Process	Successfully processed commands, %	Average command processing time, s
ruT5-small, command text generation	32.6	0.232
ruT5-base, command text generation	69.8	0.264
ruT5-small, object-pronoun type generation	95.4	0.132
ruT5-base, object-pronoun type generation	98.7	0.117
Without finding and replacing pronouns	5	0.044

Conclusions

Our work presents a deep learning method based on language models with a transformer topology for processing Russian commands on natural language during the human-robot interaction. The essence of the method lies in the consistent use of language models, each of which solves a specific task of processing a command in a natural language. The method uses neural network models RuT5, RuBERT, MultilingualBERT, RuBERT-tiny2 trained on synthetic data, crowdsourcing data and open data. At the final stage of processing, the received attributes of each command are converted into a formalized RDF graph format. From the standpoint of controlling robotic platforms by people who do not have the necessary skills to handle specialized controller devices, speech control allows to naturally form commands for robotic platforms.

The resulting system based on neural network language models makes it possible to efficiently process complex Russian-language commands and convert them into a formalized graph RDF format for controlling a robotic platform. It is confirmed by testing on a three-dimensional model of the "Arctic" robot developed at the Kurchatov Institute National Research Center. This system works more efficiently with the "RuT5-base" pronoun finding and replacing model. The conducted studies show, the finding and replacing pronouns significantly increases the accuracy of the interpretation of Russian commands on natural language in the system based on the restoring missing verbs, the decomposition of complex command into

simple commands, and the classification of attributes of simple commands.

References

- Abadi, M. et al. (2016). Tensorflow: A system for large-scale machine learning, *OSDI'16: Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation*, 265-283. (In English)
- Ahn, M. et al. (2022). Do As I Can and Not As I Say: Grounding Language in Robotic Affordances, *arXiv preprint arXiv: 2204.01691*. <https://doi.org/10.48550/arXiv.2204.01691> (In English)
- Artetxe, M. and Schwenk, H. (2019). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond, *Transactions of the Association for Computational Linguistics*, 7, 597-610. https://doi.org/10.1162/tacl_a_00288 (In English)
- Belkin, I. (2019). BERT finetuning and graph modeling for gapping resolution, *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2019"*, 63-71. (In English)
- Budnikov, E. A., Toldova, S. Yu., Zvereva, D. S., Maximova, D. M. and Ionov, M. I. (2019). Ru-eval-2019: Evaluating anaphora and coreference resolution for Russian, *Dialogue Evaluation*, available at: <https://www.dialog-21.ru/media/4689/budnikovzverevamaximova2019evaluatinganaphoracoreferenceresolution.pdf> (Accessed 10 October 2022). (In English)
- Cer, D. et al. (2018). Universal sentence encoder, *arXiv preprint arXiv: 1803.11175*. <https://doi.org/10.48550/arXiv.1803.11175> (In English)
- Chaplot, D. S., Gandhi, D., Gupta, A. and Salakhutdinov, R. (2020). Object Goal Navigation using Goal-Oriented Semantic Exploration, *arXiv preprint arXiv: 2007.00643*.

- <https://doi.org/10.48550/arXiv.2007.00643> (In English)
- Choi, D. and Langley, P. (2018). Evolution of the Icarus Cognitive Architecture, *Cognitive Systems Research*, 25-38. <https://doi.org/10.1016/j.cogsys.2017.05.005> (In English)
- Choi, D., Shi, W., Liang, Y. S., Yeo, K. H. and Kim, J.-J. (2021). Controlling Industrial Robots with High-Level Verbal Commands, *International Conference on Social Robotics (ICSR 2021), Social Robotics*, 216-226. https://doi.org/10.1007/978-3-030-90525-5_19 (In English)
- Chowdhery, A. et al. (2022). PaLM: Scaling Language Modeling with Pathways, *arXiv preprint arXiv: 2204.02311*. <https://doi.org/10.48550/arXiv.2204.02311> (In English)
- Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *arXiv preprint arXiv: 1810.04805*. <https://doi.org/10.48550/arXiv.1810.04805> (In English)
- Feng, F., Yang, Y., Cer, D., Arivazhagan, N. and Wang, W. (2022). Language-agnostic bert sentence embedding, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 878-891. <http://dx.doi.org/10.18653/v1/2022.acl-long.62> (In English)
- Gubbi, S. V., Upadrashta, R. and Amrutur, B. (2020). Translating Natural Language Instructions to Computer Programs for Robot Manipulation, *arXiv preprint arXiv: 2012.13695*. <https://doi.org/10.48550/arXiv.2110.12302> (In English)
- He, K., Gkioxari, G., Dollár, P. and Girshick, R. B. (2017). Mask R-CNN, *arXiv preprint arXiv: 1703.06870*. (In English)
- Hochreiter, S. and Schmidhuber, J. (1997). Long Short-term Memory, *Neural computation*, 9 (8), 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735> (In English)
- Joshi, M., Levy, O., Zettlemoyer, L. and Weld, D. (2019). BERT for Coreference Resolution: Baselines and Analysis, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 5803-5808. <http://dx.doi.org/10.18653/v1/D19-1588> (In English)
- Koenig, N. and Howard, A. (2004). Design and use paradigms for Gazebo, an open-source multi-robot simulator, *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, (3), 2149-2154. DOI: 10.1109/IROS.2004.1389727 (In English)
- Korobov, M. (2015). Morphological Analyzer and Generator for Russian and Ukrainian Languages, *Analysis of Images, Social Networks and Texts*, 320-332. https://doi.org/10.1007/978-3-319-26123-2_31 (In English)
- Kuratov, Y. and Arkhipov, M. (2019). Adaptation of deep bidirectional multilingual transformers for Russian language, *arXiv preprint arXiv: 1905.07213*. <https://doi.org/10.48550/arXiv.1905.07213> (In English)
- McBride, B. (2004). The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS, in Staab, S. and Studer, R. (eds.), *Handbook on Ontologies. International Handbooks on Information Systems*, Springer, Berlin, Heidelberg, Germany, 51-65. https://doi.org/10.1007/978-3-540-24750-0_3 (In English)
- Min, S. Y., Chaplot, D. S., Ravikumar, P., Bisk, Y. and Salakhutdinov, R. (2021). FILM: Following Instructions in Language with Modular Methods, *arXiv preprint arXiv: 2110.07342*. <https://doi.org/10.48550/arXiv.2110.07342> (In English)
- Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R. and Ng, A. Y. (2009). ROS: an open-source Robot Operating System, *Workshops at the IEEE International Conference on Robotics and Automation*. (In English)
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I. (2019). Language Models Are Unsupervised Multitask Learners, *OpenAI*. (In English)
- Raffel, C., Shazeer, N. and Roberts, A. (2019). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, *arXiv preprint arXiv: 1910.10683*. <https://doi.org/10.48550/arXiv.1910.10683> (In English)

Sboev, A. G., Gryaznov, A. V., Rybka, R. B., Skorokhodov, M. S. and Moloshnikov, I. A. (2022). Neural network interface for converting complex Russian-language text commands into a formalized graph form for controlling robotic devices, *Vestnik Natsional'nogo Issledovatel'skogo Yadernogo Universiteta MIPHI*, 11 (2), 153-163. DOI: 10.56304/S2304487X22020092 (In Russian)

Sboev, A., Rybka, R. and Gryaznov, A. (2020). Deep Neural Networks Ensemble with Word Vector Representation Models to Resolve Coreference Resolution in Russian, *Advanced Technologies in Robotics and Intelligent Systems*, 34-35. https://doi.org/10.1007/978-3-030-33491-8_4 (In English)

Smurov, I. M., Ponomareva, M., Shavrina, T. O. and Drozanova, K. (2019). Agr-2019: Automatic gapping resolution for Russian, *Computational Linguistics and Intellectual Technologies*, 561-575. (In English)

Van Rossum, G. and Drake, F. L. (2009). *Python 3 Reference Manual*, CreateSpace, Scotts Valley, CA. (In English)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I. (2017). Attention Is All You Need, *arXiv preprint arXiv: 1706.03762*. <https://doi.org/10.48550/arXiv.1706.03762> (In English)

Williams, A., Nangia, N. and Bowsman, S. R. (2017). A broad-coverage challenge corpus for sentence understanding through inference, *arXiv preprint arXiv: 1704.05426*. <https://doi.org/10.48550/arXiv.1704.05426> (In English)

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A. and Raffel, C. (2020). mT5: A massively multilingual pre-trained text-to-text transformer, *arXiv preprint arXiv: 2010.11934*. <https://doi.org/10.48550/arXiv.1703.06870> (In English)

Список литературы

Tensorflow: A system for large-scale machine learning / Abadi M. et al. // Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation. 2016. P. 265-283.

Do As I Can and Not As I Say: Grounding Language in Robotic Affordances / Ahn M. et al.

// arXiv preprint arXiv: 2204.01691. 2022. <https://doi.org/10.48550/arXiv.2204.01691>

Artetxe M., Schwenk H. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond // Transactions of the Association for Computational Linguistics. 2019. Vol. 7. P. 597-610. https://doi.org/10.1162/tacl_a_00288

Belkin I. BERT finetuning and graph modeling for gapping resolution // Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2019". 2019. P. 63-71.

Ru-eval-2019: Evaluating anaphora and coreference resolution for Russian / Budnikov E.A. et al. // Dialogue Evaluation. 2019. URL: <https://www.dialog-21.ru/media/4689/budnikovzverevamaximova2019evaluatinganaphoracoreferenceresolution.pdf> (дата обращения: 10.10.2022).

Universal sentence encoder / Cer D. et al. // arXiv preprint arXiv: 1803.11175. 2018. <https://doi.org/10.48550/arXiv.1803.11175>

Object Goal Navigation using Goal-Oriented Semantic Exploration / Chaplot D.S., Gandhi D., Gupta A., Salakhutdinov R. // arXiv preprint arXiv: 2007.00643. 2020. <https://doi.org/10.48550/arXiv.2007.00643>

Choi D., Langley P. Evolution of the Icarus Cognitive Architecture // Cognitive Systems Research. 2018. P. 25-38. <https://doi.org/10.1016/j.cogsys.2017.05.005>

Controlling Industrial Robots with High-Level Verbal Commands / Choi D. et al. // ICSR 2021, Social Robotics. 2021. P. 216-226. https://doi.org/10.1007/978-3-030-90525-5_19

PaLM: Scaling Language Modeling with Pathways / Chowdhery A. et al. // arXiv preprint arXiv: 2204.02311. 2022. <https://doi.org/10.48550/arXiv.2204.02311>

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / Devlin J., Chang M-W., Lee K., Toutanova K. // arXiv preprint arXiv: 1810.04805. 2018. <https://doi.org/10.48550/arXiv.1810.04805>

Language-agnostic bert sentence embedding / Feng, F. et al. // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2022. P. 878-891. <http://dx.doi.org/10.18653/v1/2022.acl-long.62>

Gubbi S.V., Upadrashta R., Amrutur B. Translating Natural Language Instructions to

Computer Programs for Robot Manipulation // arXiv preprint arXiv: 2012.13695. 2020. <https://doi.org/10.48550/arXiv.2110.12302>

Mask R-CNN / He K., Gkioxari G., Dollár P., Girshick R.B. // arXiv preprint arXiv: 1703.06870. 2017.

Hochreiter S., Schmidhuber J. Long Short-term Memory // Neural computation. 1997. № 9. P. 1735-1780.

<https://doi.org/10.1162/neco.1997.9.8.1735>

BERT for Coreference Resolution: Baselines and Analysis / Joshi M., Levy O., Zettlemoyer L., Weld D. // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019. P. 5803-5808.

<http://dx.doi.org/10.18653/v1/D19-1588>

Koenig N., Howard A. Design and use paradigms for Gazebo, an open-source multi-robot simulator // 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, 2004. № 3. P. 2149-2154. DOI: 10.1109/IROS.2004.13897

Korobov M. Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts. 2015. P. 320-332. https://doi.org/10.1007/978-3-319-26123-2_31

Kuratov Y., Arkhipov M. Adaptation of deep bidirectional multilingual transformers for Russian language // arXiv preprint arXiv: 1905.07213. 2019.

<https://doi.org/10.48550/arXiv.1905.07213>

McBride B. The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS // Handbook on Ontologies. International Handbooks on Information Systems / Staab S., Studer R. (eds.) Springer, Berlin, Heidelberg, 2004. P. 51-65.

https://doi.org/10.1007/978-3-540-24750-0_3

FILM: Following Instructions in Language with Modular Methods / Min S.Y. et al. // arXiv preprint arXiv: 2110.07342. 2021.

<https://doi.org/10.48550/arXiv.2110.07342>

ROS: an open-source Robot Operating System / Quigley M., et al. // Workshops at the IEEE International Conference on Robotics and Automation. 2009.

Language Models Are Unsupervised Multitask Learners / Radford A. et al. // OpenAI. 2019.

Raffel C., Shazeer N., Roberts A. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer // arXiv preprint arXiv: 1910.10683. 2019.

<https://doi.org/10.48550/arXiv.1910.10683>

Нейросетевой интерфейс конвертации сложных русскоязычных текстовых команд в формализованный графовый вид для управления робототехническими устройствами / Сбоев А.Г., Грязнов А.В., Рыбка Р.Б., Скороходов М.С., Молошников И.А. // Вестник Национального исследовательского ядерного университета МИФИ. 2022. Т. 11. № 2. P. 153-163. DOI:10.56304/S2304487X22020092.

Sboev A., Rybka R., Gryaznov A. Deep Neural Networks Ensemble with Word Vector Representation Models to Resolve Coreference Resolution in Russian // Advanced Technologies in Robotics and Intelligent Systems. 2020. P. 34-35. https://doi.org/10.1007/978-3-030-33491-8_4

Agrr-2019: Automatic gapping resolution for Russian / Smurov I.M., Ponomareva M., Shavrina T.O., Droganova K. // Computational Linguistics and Intellectual Technologies. 2019. P. 561-575. DOI:10.18653/v1/W19-3705.

Van Rossum G., Drake F.L. Python 3 Reference Manual // CreateSpace, Scotts Valley, CA. 2009.

Attention Is All You Need / Vaswani A. et al. // arXiv preprint arXiv: 1706.03762. 2017. <https://doi.org/10.48550/arXiv.1706.03762>

Williams A., Nangia N., Bowsman S.R. A broad-coverage challenge corpus for sentence understanding through inference // arXiv preprint arXiv: 1704.05426. 2017.

<https://doi.org/10.48550/arXiv.1704.05426>

mT5: A massively multilingual pre-trained text-to-text transformer / Xue L. et al. // arXiv preprint arXiv: 2010.11934. 2020. <https://doi.org/10.48550/arXiv.1703.06870>

Все авторы прочитали и одобрили окончательный вариант рукописи.

All authors have read and approved the final manuscript.

Appendices

Appendix A.1. Simple command templates

Приложение А.1. Шаблоны простых команд

№	Template	Example
1	patrol	go on patrol
2	execution termination	stop
3	execution interruption	pause
4	execution continuation	start moving again
5	movement in the direction	move forward
6	turn in direction	turn left
7	movement in the direction by a number of meters	drive forward 2 meters
8	turn in the direction by a number of degrees	turn left 30 degrees
9	interaction with the object	analyze the stone
10	interaction with the nearest object	find the nearest person
11	interaction with an object relative to 1 object	analyze the stone located to the left of the person
12	interaction with an object relative to 2 objects	go to the person who is near the house, next to the tree
13	interaction with the object relative to the robot	drive up to the house on the left
14	interaction in the direction of gaze	go to this person
15	patrol in a circle / along the route	patrol along the second route
16	following an object	follow the car

Appendix A.2. Vector and labeled representations

Приложение А.2. Векторное и маркированное представления

Attribute	Vector	Marker
Action	0-14	A
Direction	0-8	D
Meters	0-12	M
Degrees	0-24	DS
Hours	0-24	HS
Object 1	0-16	O1
Nearest	0-1	N
Relation between 1 and 2	0-9	R1
Object 2	0-14	O2
Relation between 2 and 3	0-9	R2
Object 3	0-14	O3
Relation between 1 and robot	0-8	S
Gaze direction	0-1	G
Additional words	-	O

Конфликты интересов: у авторов нет конфликтов интересов для декларации.

Conflicts of interests: the authors have no conflicts of interest to declare.

Alexander G. Sboev, Doctor, Leading Research Fellow, Kurchatov Institute National Research Center, Moscow, Russia.

Александр Георгиевич Сбоев, д. ф.-м. н., ведущий научный сотрудник, Национальный исследовательский центр «Курчатовский институт», Москва, Россия.

Artem V. Gryaznov, Junior Research Fellow, Kurchatov Institute National Research Center, Moscow, Russia.

Артем Викторович Грязнов, младший научный сотрудник, Национальный исследовательский центр «Курчатовский институт», Москва, Россия.

Roman V. Rybka, Ph. D., Head of Research Group, Kurchatov Institute National Research Center, Moscow, Russia.

Роман Борисович Рыбка, к. т. н., начальник группы, Национальный исследовательский центр «Курчатовский институт», Москва, Россия.

Maxim S. Skorokhodov, Research Assistant,

Kurchatov Institute National Research Center, Moscow, Russia.

Максим Сергеевич Скороходов, лаборант-исследователь, Национальный исследовательский центр «Курчатовский институт», Москва, Россия.

Ivan A. Moloshnikov, Research Fellow, Kurchatov Institute National Research Center, Moscow, Russia.

Иван Александрович Молошников, научный сотрудник, Национальный исследовательский центр «Курчатовский институт», Москва, Россия.