



Ссылка на статью:

// Ученые записки УлГУ. Серия Математика и информационные технологии. 2024, № 2, с. 111-120.

Поступила: 05.11.2024

Окончательный вариант: 08.12.2024

© УлГУ

УДК 159.9.072; 004.8

Автоматизация интерпретации результатов психологического тестирования с применением искусственного интеллекта

Шахбазян К. О.^{1,*}, Цыганова Ю. В.^{1,2}

[*karina.shakhbazyan@gmail.com](mailto:karina.shakhbazyan@gmail.com)

¹УлГУ, Ульяновск, Россия

²Университет Иннополис, Россия

В статье представлена программа, которая позволяет в автоматическом режиме проводить психологическое тестирование «Руки Вагнера» с использованием нейронной сети. Нейронная сеть анализирует ответы пользователя и классифицирует их для определения психологического состояния и возможных расстройств испытуемого. Программа может найти применение в профессиональной психологической практике для автоматизации процесса психологического тестирования.

Ключевые слова: нейронные сети, машинное обучение, Python, психологическое тестирование, модель BERT, машинная обработка естественного языка

Введение

Проективное психологическое тестирование «Руки Вагнера», разработанное в 1962 году Эдвином Е. Вагнером [1], предназначено для определения предрасположенностей и наклонностей личности путем классификации его ответов в соответствии с одной из нескольких категорий, такими как агрессия, коммуникация, эмоциональность и др. Тест «Руки Вагнера» отличается от других психологических тестов тем, что испытуемый не ограничен выбором из представленных готовых вариантов, а придумывает свой ответ на каждый из 10 вопросов. При этом на каждый вопрос он может придумать от одного до четырех ответов.

Возможность применения нейронных сетей для автоматизации процесса психологического тестирования рассматривала, например, С.Т. Ахметова [2]. Автор исследовала способность нейронных сетей к конструированию психологических тестов. Как отмечено

в [2], «средства современной психодиагностики разделяются на две группы: формализованные методики, и методики мало формализованные», при этом автор подчеркивает, что «многие строго формализованные методики доведены до уровня компьютерной реализации» со ссылкой на книгу еще 1994 года издания [2]. К сожалению, современные авторы продолжают в своих работах автоматизировать в основном строго формализованные методики (см., например, [4-6]); публикаций же, обсуждающих применение компьютерных методов к реализации мало формализованных методик, в отечественных изданиях найти не удалось. Предложенные на данный момент онлайн ресурсы психологического тестирования либо прямо указывают, что «Поскольку ответы даются в свободной форме, их автоматическая обработка невозможна. Оценку ответов должен проводить психолог. Полноценная интерпретация этого теста, как и всех подобных проективных методик, является творческим процессом, который во многом определяется квалификацией и опытом специалиста. Самостоятельно проходить онлайн-версию и обрабатывать свои ответы имеет смысл лишь в образовательных целях» [7], либо предлагают выбор ответа из ограниченного списка [8].

В данной работе исследуется возможность применения нейронных сетей к задаче интерпретации одной из проективных методик психологического тестирования - теста «Руки Вагнера». Поскольку методика относится к проективным, ее нельзя назвать строго формализованной. В то же время данная методика имеет строго формализованную инструкцию и подразумевает ее последовательное выполнение по заданному алгоритму, поэтому она представляется перспективной для разработки программы автоматизации интерпретации результатов психологического тестирования с применением искусственного интеллекта.

1. Набор данных для обучения

Обработка и интерпретация результатов тестирования по методике «Руки Вагнера» заключается в том, что испытуемому показывают изображение руки в виде «стоп-кадра» с просьбой описать действие, происходящее на изображении. Затем каждый ответ испытуемого относят к одной из 11 категорий: агрессия, страх, зависимость и т. д [1]. Для генерации тестового набора данных достаточного объема, и при этом содержащего разнообразные ответы, была использована большая языковая модель ChatGPT-4. Приведем примеры запросов, использованных для генерации набора данных:

- Перед тобой на картинке изображена рука. Напиши 30 предложений описания руки без указания на совершаемые ею действия.

- Перед тобой на картинке изображен человек. Напиши 30 предложений, где рука человека участвует в действии, не связанном с коммуникацией; однако рука должна изменить свое физическое местоположение, приложить усилие.

В Таблице 1 представлены примеры сгенерированных ответов модели на запросы.

Таблица 1. Некоторые примеры из набора данных, созданные с помощью ChatGPT-4

Сгенерированный ответ	Категория
-----------------------	-----------

Человек показывает технику дыхательных упражнений, направляя руки вверх и вниз в соответствии с дыханием.	Демонстративность
Человек делает знак “всё хорошо” рукой, поднимая большой палец вверх.	Коммуникация
Пальцы руки давят на насекомое, пытаясь его уничтожить.	Агрессия

Количество ответов было равномерно распределено по 11 категориям, чтобы избежать искажения при обучении модели (см. Таблицу 2).

Таблица 2. Распределение данных в наборе по количеству ответов в каждой категории

Категория	Количество примеров
Агрессия	957
Указание	903
Страх	917
Эмоциональность	931
Коммуникация	898
Зависимость	913
Демонстративность	939
Увечность	923
Активная Безличность	948
Пассивная Безличность	911
Описание	927
Всего:	10167

В результате работы по генерации ответов был создан размеченный набор данных из 10167 примеров.

2. Обучение модели RuBERT

Для решения задачи классификации ответов на тест «Руки Вагнера» была выбрана нейронная сеть с архитектурой трансформера для обработки естественного языка под названием RuBERT (Russian Bidirectional Encoder Representations from Transformers) [9]. На рис. 1 представлена архитектура модели RuBERT для задачи классификации.

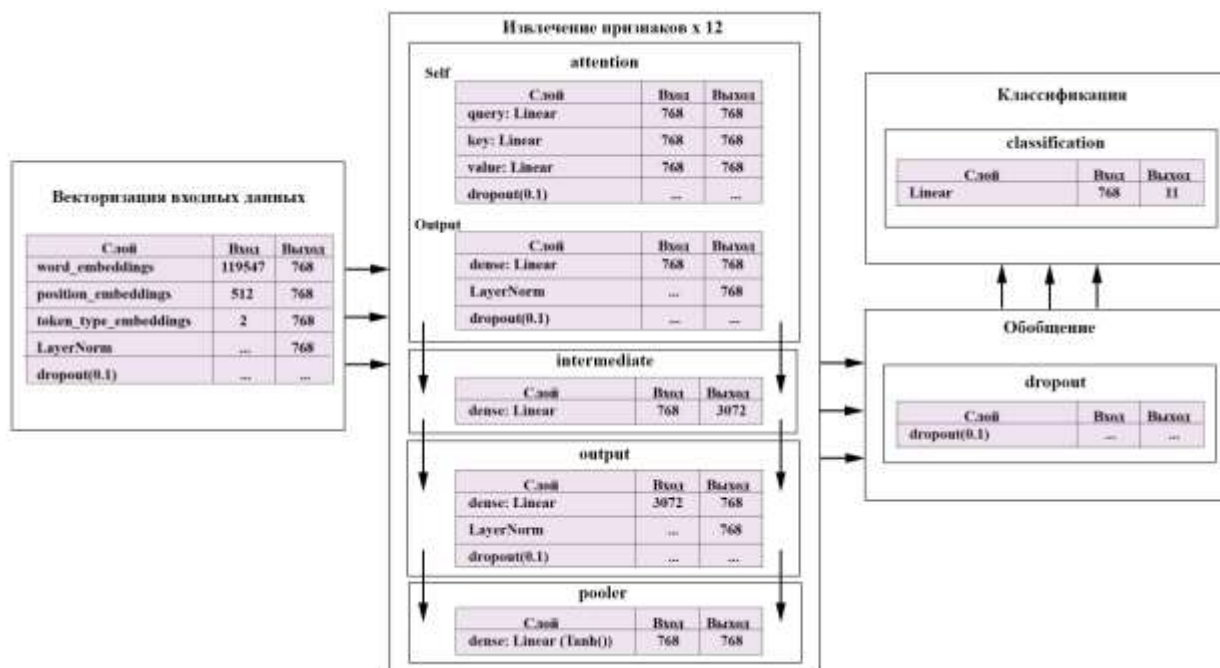


Рис. 1. Архитектура модели RuBERT для задачи классификации

В результате настройки модели для решения задачи классификации был добавлен новый линейный слой прямого распространения [10]. Он имеет 768 входов и 11 выходов, которые соответствуют категориям классификации. Чтобы выполнить обучение полученной модели, необходимо было выполнить предобработку созданного ранее набора данных. Для этого каждое предложение из набора было разбито на отдельные токены и представлено в виде вектора [11]. При выполнении этой задачи был использован AutoTokenizer из библиотеки Transformers. Этот токенизатор позволил использовать метод encode_plus, который включает в себя настройку таких параметров как [12]:

- Add_special_tokens: добавляет специальные токены начала и конца предложения.
- Max_length: устанавливает максимальную длину токена.
- Padding: дополняет остальные токены до максимальной длины.
- Return_tensors: возвращает тензоры токенов.

После токенизации набора данных был получен массив со специальными токенами начала и конца предложения, состоящий из тензоров, которые были дополнены до максимальной длины равной 32. Результаты представлены в Таблице 3.

Таблица 3. Результат преобразования текста с помощью токенизации

Оригинальное предложение	Результат токенизации
Рука крепко сжимает кулак, готовясь нанести удар.	101, 40165, 79073, 60186, 1828, 102932, 128, 91079, 33128, 11570, 132, 102, 0, ... 0
Рука с силой хватает другого человека за руку, останав-	101, 40165, 869, 5563, 15484, 35400, 17743,

ливая его.	6798, 1758, 24700, 128, 23793, 1637, 2752, 132, 102, 0, ... 0
Рука тянется, чтобы дать пощечину.	101, 40165, 59965, 128, 5247, 18769, 64291, 38763, 4650, 132, 102, 0, ... 0

В качестве оптимизатора модели при обучении был выбран AdamW из библиотеки PyTorch со следующими параметрами [13]:

- Learning rate: шаг обучения. Стандартный и равен $2e-5$.
- Eps: эпсилон. Используется для расчета функции потерь. Равен $1e-8$.

Далее происходит сам процесс обучения нейронной сети путем перенастройки весовых коэффициентов с использованием градиентного спуска. Для этого использовались библиотеки Hugging Face, PyTorch и Transformers на языке Python [12,13]. Также были применены CUDA (Compute Unified Device Architecture) и CuDNN (Cuda for Deep Neural Networks) [14]. Это программно-аппаратные архитектуры, позволяющие увеличить вычислительную производительность при обучении нейросети [11].

3. Результаты обучения

Для оценки качества обучения нейронной сети и ее способности верно классифицировать ответы пользователя, а также определения оптимального количества эпох обучения модели, были использованы метрики машинного обучения: точность (accuracy) и результат вычисления функции потерь (loss) [15]. Результаты представлены в Таблице 3.

Таблица 4. Значение метрик шести эпох обучения модели

Эпоха	Потери при обучении	Потери при тестировании	Точность	Время обучения
1	1.12	0.44	0.9	0:05:03
2	0.22	0.17	0.96	0:01:42
3	0.1	0.15	0.96	0:01:46
4	0.05	0.11	0.96	0:01:46
5	0.02	0.13	0.96	0:01:42
6	0.01	0.14	0.96	0:01:46

В качестве функции потерь использовалась категориальная кросс-энтропия [11]. Формула для категориальной кросс-энтропии выглядит следующим образом:

$$L(y, \hat{y}) = - \sum_{i=1}^c y_i \log(\hat{y}_i) \quad (1)$$

где C – кол-во классов, y_i – истинный класс (one-hot вектор, где 1 соответствует правильному классу, а остальные 0), \hat{y}_i – предсказанная вероятность для класса i . Значение функции потерь на обучающей выборке постепенно уменьшается, тогда как на тесто-

вой выборке после четвертой эпохи начинает расти. При этом точность модели остается неизменной. Этот результат указывает на явное переобучение, поскольку потери на тестовой выборке играют более значимую роль, чем на обучающей [14]. Было принято решение остановиться на четырех эпохах обучения. На рис. 2 изображен график зависимости функции потерь от количества эпох обучения на обучающей и тестовой выборках.

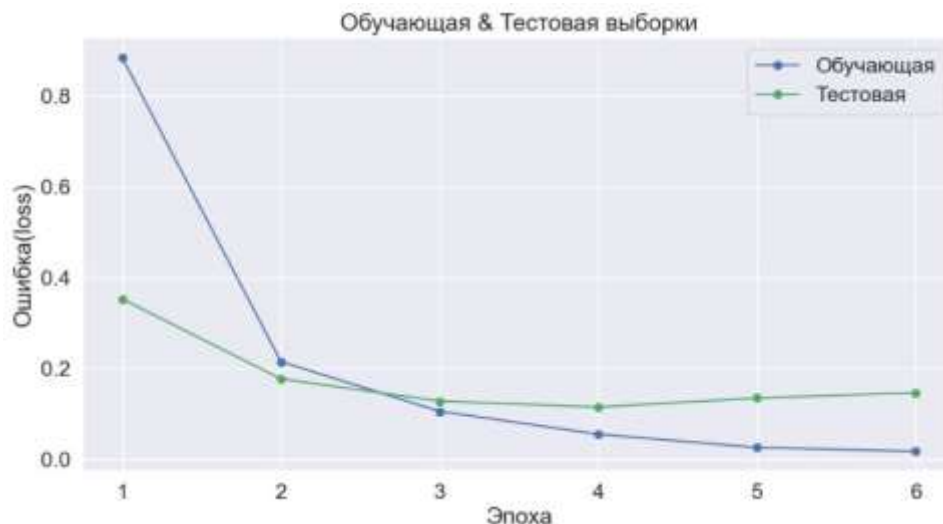


Рис. 2. График функции потерь при шести эпохах обучения модели

Затем модель была протестирована на примерах, не входивших в обучающий и тестовый наборы данных. На рис. 3 представлен результат работы модели на реальных данных.

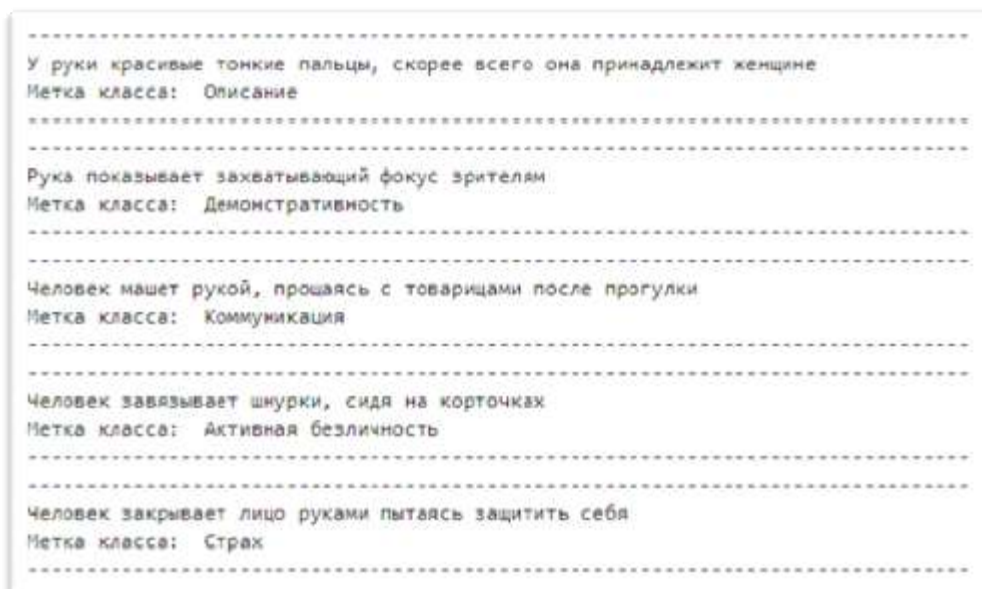
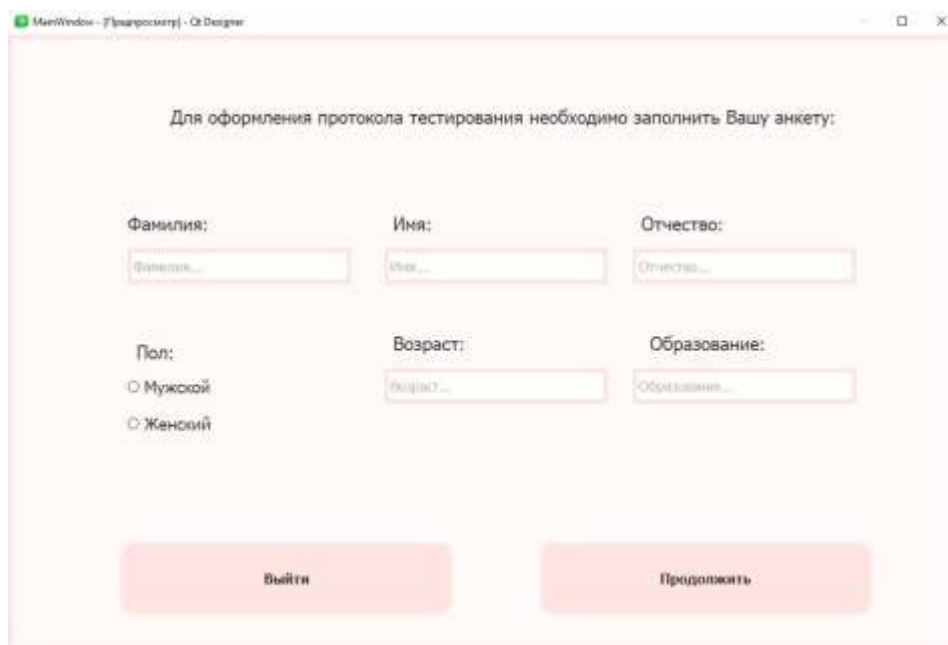


Рис. 3. Результат классификации на реальных примерах

В итоге оптимальным количеством эпох для обучения является четыре. Этот выбор обоснован как метриками оценки качества модели, так и результатами работы нейронной сети на реальных данных.

4. Пользовательский интерфейс

Спроектированный интерфейс состоит из пяти частей: главная страница, анкета пользователя, тестирование, результат тестирования, страница с результатами предыдущих пользователей (рис. 4 – 6). По ним пользователь может переходить по мере взаимодействия с программой с помощью кнопок навигации.



The screenshot shows a window titled "MainWindow - [Проектиров] - Qt Designer". The main text reads: "Для оформления протокола тестирования необходимо заполнить Вашу анкету:". Below this, there are six input fields arranged in two rows of three. The first row contains fields for "Фамилия:" (with a placeholder "Фамилия..."), "Имя:" (with a placeholder "Имя..."), and "Отчество:" (with a placeholder "Отчество..."). The second row contains a "Пол:" section with radio buttons for "Мужской" and "Женский", a "Возраст:" field (with a placeholder "Возраст..."), and an "Образование:" field (with a placeholder "Образование..."). At the bottom, there are two large buttons: "Выйти" on the left and "Продолжить" on the right.

Рис. 4. Интерфейс. Страница анкеты пользователя



The screenshot shows a window titled "MainWindow - [Проектиров] - Qt Designer". The main text reads: "Опишите какое действие выполняет рука или человек, которому она принадлежит?". In the center, there is a line drawing of a right hand, palm facing up. To the left and right of the hand are buttons labeled "на 90°" with circular arrows. Below the hand, the text "1/9" is displayed. Underneath, there are four empty text input fields, each with a placeholder "Заполните эту форму одно из этих полей...". At the bottom, there are two large buttons: "Назад" on the left and "Продолжить" on the right.

Рис. 5. Интерфейс. Страница тестирования

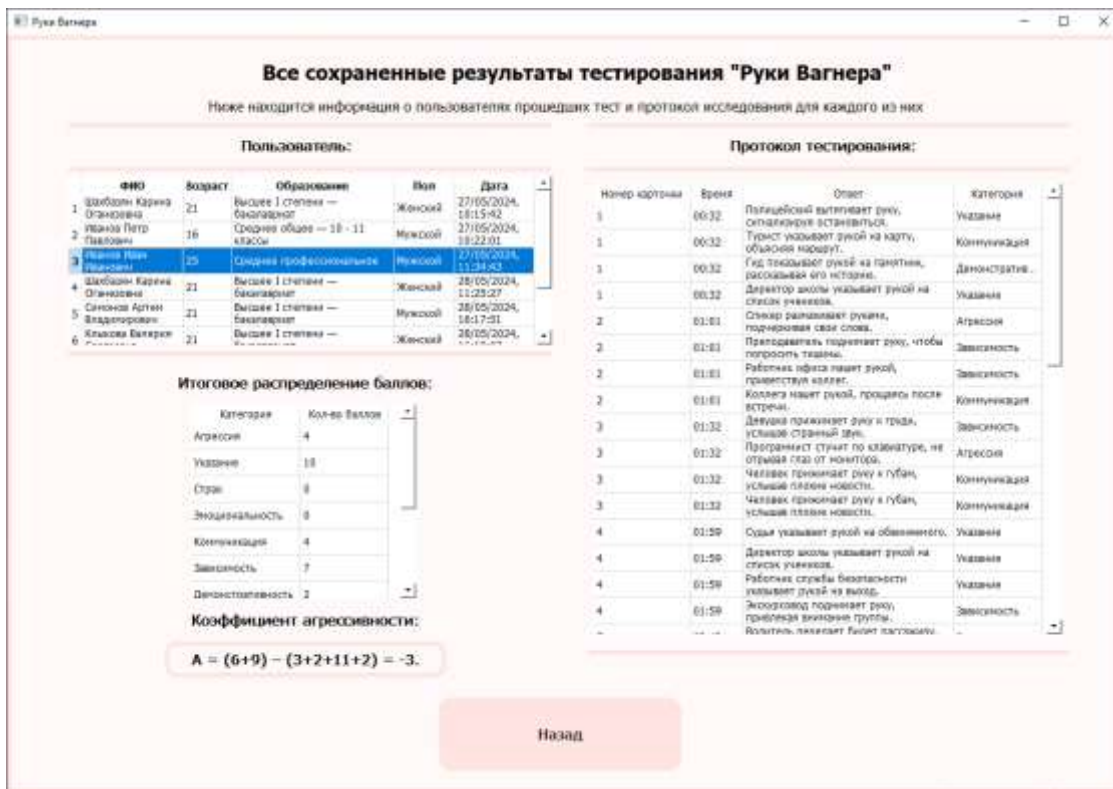


Рис. 6. Интерфейс. Страница сохраненных результатов тестирования

При создании пользовательского интерфейса были использованы изображения рук из методического пособия «Руки Вагнера». Пользователь должен посмотреть каждое изображение и заполнить соответствующие поля для ответов. Программа сохраняет время ответа пользователя относительно начала тестирования и фиксирует его в протоколе. Для создания интерфейса использовались фреймворки PySide6 и PyQt6 Designer [16]. Использование этих фреймворков существенно снизило временные затраты на разработку пользовательского интерфейса.

Заключение

В статье представлена разработанная прикладная программа, которая автоматизирует процесс психологического тестирования по методике «Руки Вагнера», и сокращает время интерпретации ответов с одного часа до нескольких минут. При разработке использовался язык программирования Python, библиотеки PyTorch и Transformers, а также фреймворк PySide6 для создания интерфейса. Модель нейронной сети RuBERT обучалась классификации ответов на специальном наборе данных. В результате модель способна интерпретировать результаты тестирования с точностью 85%, что делает программу пригодной для использования в профессиональной психологической практике.

Список литературы

1. Вагнер, Э. Е. *Проективная методика исследования личности «Hand-тест»: методическое руководство*. Перераб. изд. Лос-Анджелес: Western Psychological Services, 1983. — 120 с.
2. Ахметова С.Т., Ыскак Н.П. Математические модели и алгоритмы психодиагностики // *Известия ВУЗов (Кыргызстан)*. 2012, № 6, с. 5-8.
3. Дюк В. А. *Компьютерная психодиагностика*. СПб: «Братство», 1994. 364 с.
4. Мельников, В.И., Баданов А.А., Прокурова С.В. Применение компьютерного практикума по обработке теста Люшера для диагностики психических состояний личности // *Психопедагогика в правоохранительных органах*. 2023, № 4 (95), с. 416-421.
5. Мельников В.И. Применение информационных технологий в обработке психологических тестов // *Сборник научных статей III Межвузовской научно-практической конференции с международным участием*. 2024. С. 91-95.
6. Павлова С.А. Автоматизация психологических тестовых методик и обработки полученных результатов // *Теория и практика научных исследований: психология, педагогика, экономика и управление*. 2018, № 2, с. 93-97.
7. Психологические тесты онлайн [электронный ресурс]: электрон. ресурс. Режим доступа: <https://psytests.org/classic/hand.html>, свободный (дата обращения: 08.12.2024).
8. Тестоград [электронный ресурс]: электрон. ресурс. Режим доступа: <https://testograd.com/test/ruki-vagnera/>, свободный (дата обращения: 08.12.2024).
9. How to Fine-Tune BERT for Text Classification [электронный ресурс]: электрон. Режим доступа: <https://arxiv.org/abs/1905.05583>, свободный (дата обращения: 26.10.2024).
10. Черкасов А.Н., Туркин Е.А. Выбор оптимальной архитектуры искусственной нейронной сети для задачи классификации текстов // *Ежеквартальный рецензируемый, реферируемый научный журнал «Вестник АГУ»*. 2021, № 1, с. 62-66.
11. Хайкин С. *Нейронные сети: полный курс*. 2-е изд. Москва: Издательский дом “Вильямс”, 2006. 1104 с.
12. Официальная документация Transformers [электронный ресурс]: электрон. документ. Режим доступа: <https://huggingface.co/docs>, свободный (дата обращения: 26.10.2024).
13. Официальная документация PyTorch [электронный ресурс]. Режим доступа: <https://pytorch.org/docs/stable/index.html>, свободный (дата обращения: 29.09.2024).
14. Ростовцев В.С. *Искусственные нейронные сети*. 2-е изд. Санкт-Петербург: Лань, 2021. 216 с.
15. Metrics for Multi-Class Classification: an Overview [электронный ресурс]: электрон. Режим доступа: <https://arxiv.org/abs/2008.05756>, свободный (дата обращения: 17.09.2024).
16. Официальная документация PyQt [электронный ресурс]: электрон. документ. Режим доступа: <https://doc.qt.io/qtforpython-6/>, свободный (дата обращения: 26.10.2024).

Automation of interpretation of psychological testing results using artificial intelligence

Shakhbazyan, K. O.^{1,}, Tsyganova, Yu. V.^{1,2}*

[*karina.shakhbazyan@gmail.com](mailto:karina.shakhbazyan@gmail.com)

¹Ulyanovsk State University, Russia

²Innopolis University, Russia

The paper describes a program that enables automated psychological testing using Wagner's Hand Test with the help of a neural network. The neural network analyzes user responses and classifies them to assess the psychological state and potential disorders of the test subject. The program can be utilized in professional psychological practice to automate the process of psychological testing.

Keywords: neural networks, machine learning, Python, psychological testing, BERT model