Метод объяснимости трансформера BERT при решении задачи классификации текстов

П.Л. Николаев

***Аннотация* — Нейронные сети, как и другие методы машинного обучения, являются черными ящиками. Это означает, что невозможно понять, как они принимают то или иное решение. При этом с усложнением нейронных сетей все актуальнее становится развитие методов их объяснимости. В данной работе предлагается новый метод объяснимости трансформеров (вид глубоких нейронных сетей), предназначенных для классификации текстовых данных на русском языке. В статье рассматривается трансформер BERT, дообученный классификации книг по жанрам на основе их аннотаций. Дообученная модель показывает высокую точность при проверке на тестовой выборке – 84%. Кроме того, в работе приводится метод объяснимости модели BERT на основе кластеризации его голов внимания с помощью метода HDBSCAN. С помощью кластерного анализа удалось разбить головы внимания на различные группы, по которым можно проанализировать работу модели.**

***Ключевые слова* — Искусственный интеллект, машинное обучение, искусственные нейронные сети, глубокое обучение, трансформеры, классификация текстов.**

I. Введение

В последнее время глубокие нейронные сети все чаще применяются для решения самых разнообразных задач, среди которых можно выделить распознавание и генерацию изображений, распознавание и генерацию текста, распознавание и генерацию различных звуков. Одним из видов глубоких нейронных сетей являются трансформеры, которые показывают наиболее точные результаты при решении задач, связанных с обработкой естественного языка (их еще называют большими языковыми моделями). Например, трансформеры можно использовать для классификации текстов (модели семейства BERT [1]), для генерации текстов (GPT-подобные модели [2-4]) или для суммаризации текстов (BART [5], T5 [6]). При этом трансформеры, как и другие нейронные сети, являются черными ящиками. Это означает, что невозможно разобраться, как нейронная сеть принимает то или решение. В связи с этим, существует потребность в разработке методов объяснимости нейронных сетей, и, в частности, трансформеров. В рамках работы рассмотрим объяснимость трансформеров, предназначенных для одного из направлений обработки естественного языка – классификации текста.

II. Обзор существующих методов объяснимости

Существующие методы объяснимости нейронных сетей можно разделить на несколько категорий:

- суррогатные – метод LIME [7];
- методы на основе важности признаков – метод интегральных градиентов (Integrated Gradients) [8], SHAP [9];
- модели визуализации – GRAD-CAM [10].

Метод LIME создает простую интерпретируемую модель, приближающую сложную нейронную сеть локально по окрестности входа. Модель строится вокруг конкретного предсказания.

Метод интегральных градиентов и SHAP показывают влияние каждого входного признака на итоговое решение. Метод интегральных градиентов аккумулирует градиенты входа по пути от базового состояния к текущему. А метод SHAP, основанный на теории игр, объясняет вклад каждого признака в прогнозирование конкретного наблюдения [11].

Метод GRAD-CAM создает тепловую карту для сверточной нейронной сети. По этой карте можно определить, какие области изображения активировала сеть.

III. Архитектура трансформеров

Базовые трансформеры имеют энокдерно-декордерную структуру. На вход энкодера поступает последовательность символов, которая преобразуется в числовые векторы. Декодер, основываясь на этих числовых представлениях, создает конечную последовательность. На каждом этапе процесса генерации модель использует как исходную входную последовательность, так и уже сгенерированные символы.

Также в трансформерах используется механизм внимания [12], позволяющий учитывать важные семантические связи между различными частями всей информации, поступающей модели. Обычно используется механизм самовнимания, состоящий из голов. Механизм самовнимания позволяет модели учитывать важность каждого слова или элемента последовательности относительно других. В рамках этого механизма используется несколько голов, которые параллельно вычисляют разные аспекты внимания. Сама голова является отдельным механизмом внимания, работающим с собственными весами и представлениями.

Статья получена 30 декабря 2025.

П.Л. Николаев – старший преподаватель Московского авиационного института (национального исследовательского университета) (e-mail: npavel89@gmail.com).

Все это позволяет трансформерам захватывать разные типы зависимостей и признаков в данных. После этого результаты всех голов объединяются, что увеличивает шансы модели понимать сложные связи.

В рамках данной работы рассматривается вариация сети BERT (Bidirectional Encoder Representations from Transformers) – RuBERT от проекта DeepPavlov [13], решающая задачу классификации текстовых данных на русском языке.

Рассмотрим архитектуру сети BERT. Модель имеет только энкодер с 12 слоями по 12 голов внимания. Отличительная особенность модели заключается в способности обрабатывать текст с двух сторон. Это достигается путем использования маскированного языкового моделирования – часть слов в предложении заменяется на маски (например, «Я [MASK] статью про глубокое обучение»), и BERT учится предсказывать эти пропущенные слова, используя контекст с обеих сторон. Все это позволяет лучше понимать контекст и взаимодействие слов в предложении.

IV. Предлагаемый метод объяснимости трансформеров

Для понимания внутренних структур и функционирования трансформеров можно использовать методы кластеризации данных. Этот подход особенно полезен при исследовании внимания: поскольку оно распределяется по входным токенам, такие распределения можно разбить на группы с помощью кластеризации. Аналогично кластеризация помогает интерпретировать векторные представления (эмбеддинги), формируемые на различных уровнях модели. Это позволяет выявить, какие характеристики текста выделяет модель и как они сгруппированы внутри модели.

Существует множество различных методов кластеризации, среди которых можно выделить методы DBSCAN и HDBSCAN. Данные методы выгодно отличаются от других способностью выделять сложные геометрические структуры, не ограничиваясь сферическими кластерами. Также оба метода не требуют указания числа получаемых кластеров, их количество определяется на основе данных. Помимо этого, DBSCAN и HDBSCAN способны выделять шум – это данные, не принадлежащие ни одному кластеру, что повышает устойчивость к выбросам. Вдобавок к этому, методы позволяет находить кластеры с различной плотностью. Также они отличается высокой скоростью работы. Из двух рассмотренных методов предпочтительнее использовать HDBSCAN, поскольку он требует задания меньшего количества гиперпараметров.

Зададим матрицу внимания A: $A(l,h,i) \in R^{n_i \times n_i}$, где i – пример, l – слой, h – голова, n_i – длина последовательности примера i.

$$A = [A_{jk}], j,k = 1,\dots,n \quad (1)$$

Определим признаки для модели кластеризации, извлекаемые из матрицы A:

1. $x_1(A)$ – среднее значение матрицы внимания, по которой можно увидеть, насколько в среднем токены обращаются друг к другу.
2. $x_2(A)$ – стандартное отклонение внимания (необходимо для исследования разброса значений в матрице внимания).
3. $x_3(A)$ – максимальный элемент матрицы (отражает наибольшее внимание к одному токену):
4. $x_4(A)$ – энтропия матрицы (показывает разбросанность внимания). Если она высокая, то внимание распределено равномерно, в противном случае – сосредоточено:

$$x_4(A) = -\frac{1}{n^2}\sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}\log A_{jk} + \epsilon, \quad (2)$$

где ϵ – число для численной стабилизации, возьмем его равным 10^{-10}.

5. $x_5(A)$ – диагональные элементы матрицы, показывающие, насколько токен обращается к самому себе:

$$x_5(A) = \frac{1}{n}\sum_{j=1}^{n} A_{jj} \quad (3)$$

6. $x_6(A)$ – внимание к токену классификации ([CLS] токен) от других токенов:

$$x_6(A) = \frac{1}{n}\sum_{j=1}^{n} A_{j,1} \quad (4)$$

7. $x_7(A)$ – локальное внимание к соседним токенам:

$$x_7(A) = \frac{1}{n}\sum_{j=1}^{n}\sum_{k=\max(1,j-2)}^{\min(n,j+2)} A_{jk} \quad (5)$$

8. $x_8(A)$ – показатель того, насколько для каждого токена в среднем выделяется один сильный источник внимания:

$$x_8(A) = \frac{1}{n}\sum_{j=1}^{n}\max_{k=1,\dots,n} A_{jk} \quad (6)$$

Для каждого слоя l, головы h и примера i:

$$X^{(l,h,i)} = \left[x_1^{(l,h,i)}, x_2^{(l,h,i)}, \dots, x_8^{(l,h,i)}\right] \quad (7)$$

Среднее значение по всему набору данных:

$$\bar{X}^{(l,h)} = \frac{1}{N}\sum_{i=1}^{N} X^{(l,h,i)} \quad (8)$$

Итоговая матрица признаков для кластеризации:

$$F = \left[\bar{X}^{(0,0)}; \bar{X}^{(0,1)}; \dots; \bar{X}^{(11,11)}\right] \in R^{144\times 8} \quad (9)$$

V. Дообучение модели BERT

Для дообучения и анализа модели BERT использовался язык программирования Python с библиотеками глубокого обучения TensorFlow с Keras API и transformers, содержащей обученные модели трансформеров. Также применялась библиотека машинного обучения scikit-learn.

Для обучения использовался набор данных для жанровой классификации книг по аннотациям из [14]. Набор данных содержит данные о 7612 книгах, разбитых на 10 классов. Сама модель BERT была изменена путем добавления слоя классификации с 10 нейронами (по количеству классов). А на вход модели подавались последовательности длины 128.

Нейросеть обучалась со следующим параметрами:

- оптимизатор Adam с коэффициентом обучения, равным 0,00002;
- функция потерь – перекрестная энтропия;
- размер мини-выборки – 64 примера;
- количество эпох обучения – 3.

Минимальное значение функции потерь на валидационной выборке было достигнуто на 2 эпохе.

На рисунке 1 демонстрируется изменение значения потерь во время обучения сети.

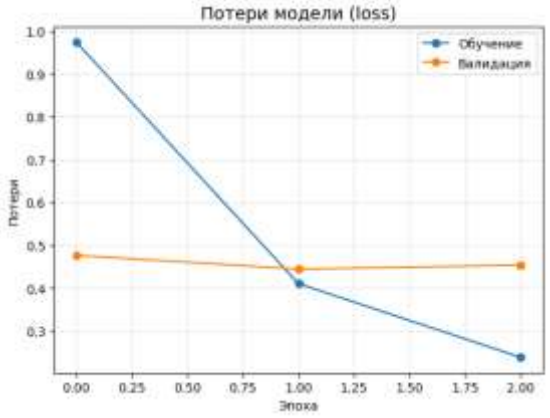

Рисунок 1. Изменение значения потерь во время обучения сети

На рисунке 2 демонстрируется изменения значения точности во время обучения сети.

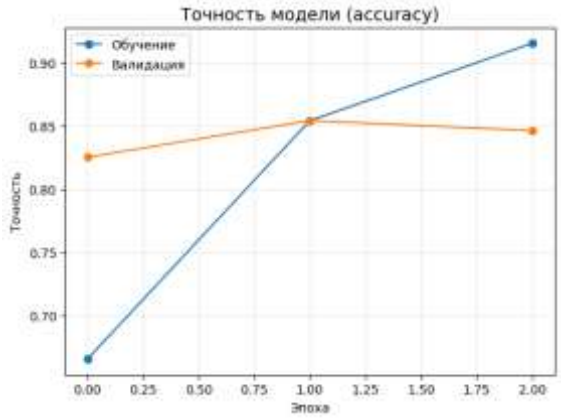

Рисунок 2. Изменение значения точности во время обучения сети

Построим матрицу неточностей при проверке модели на обучающей выборке (рис. 3).

Рисунок 3. Матрица неточностей на тестовой выборке.

В Таблице 1 демонстрируются данные по финальным значениям функции потерь и метрики по всем поднаборам данных. Для сравнения приведены данные по сверточно-рекуррентой модели из [14].

Таблица 1. Результаты обучения и проверки сети.

	Рекуррентно-сверточная сеть [14]		**Сеть на основе трансформера RuBERT**	
Выборка	**Точность (accuracy)**	**Потери (loss)**	**Точность (accuracy)**	**Потери (loss)**
Обучающая (60% данных)	98.07%	0.08	93.12%	0.21
Валидационная (10% данных)	72.80%	0.84	85.41%	0.44
Тестовая (30% данных)	71.11%	0.93	83.92%	0.45

Как видно по результатам сравнения, нейронная сеть на основе BERT оказалась более точной.

VI. Кластеризация голов внимания

После обучения была проведена кластеризация голов внимания с помощью алгоритма HDBSCAN. В Таблице 2 представлены получившиеся кластеры и количество голов внимания в них.

Таблица 2. Распределение голов внимания по кластерам.

Кластер	**Количество голов внимания**
Выбросы	56
Кластер_0	73
Кластер_1	6
Кластер_2	9

В кластер с выбросами попало 56 голов внимания (38.9%). Это означает, что они не влияют на качество модели, и их можно исключить.

Для каждого кластера вычислим средние точки (центроиды) (Таблица 3).

Таблица 3. Центроиды получившихся кластеров.

Кластеры/ Признак	**Выбросы**	**Кластер_0**	**Кластер_1**	**Кластер_2**
x_1	0.01	0.01	0.01	0.01
x_2	0.05	0.03	0.03	0.04
x_3	0.89	0.58	0.91	0.87
x_4	2.27	3.09	2.75	3.04
x_5	0.07	0.05	0.03	0.02
x_6	0.22	0.03	0.23	**0.37**
x_7	0.29	0.22	0.17	0.05
x_8	0.45	0.23	0.3	**0.38**

Высокие значения (выше 0.3) есть у признака x_6 в Кластере_2, отвечающего за агрегацию по классам. Это означает, что в Кластер_2 попали головы внимания, ответственные за агрегирование глобальной информации для классификации данных. Также в этом кластере есть высокое значение у признака x_8, который отвечает за концентрацию на одном ключевом стимуле.

Для визуализации получившихся кластеров можно использовать алгоритм UMAP, понижающий размерность до двух признаков (рис. 4).

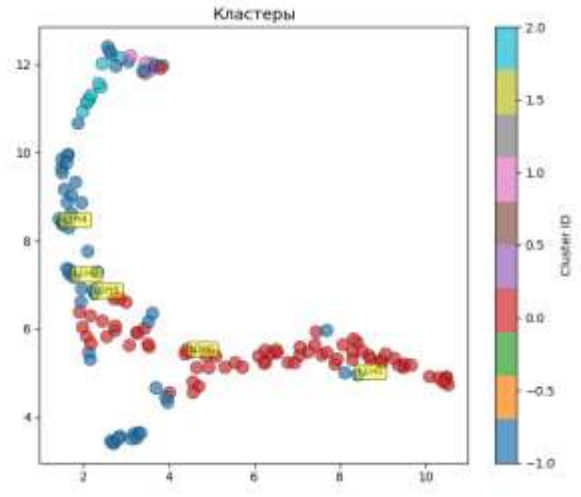

Рисунок 4. Кластеризация голов внимания

На рис. 4 цвета кластеров определяются идентификаторами кластеров (-1, 0, 1, 2). При этом значением -1 обозначаются выбросы.

VII. Заключение

Для решения задачи объяснимости трансформера BERT был предложен метод кластеризации с помощью алгоритмы HDBSCAN. Сначала было проведено обучение русскоязычного варианта BERT – RuBERT на собранном автором наборе данных. При проверке на обучающей выборке модель показала высокую точность – 85% при уровне потерь. Затем был проведен анализ обученной модели путем кластеризации ее голов внимания. В результате анализа были выявлены головы, попавшие в выбросы, а также головы, отвечающие за непосредственную классификацию данных.

В дальнейшем планируется исследовать другие модели трансформеров для решения задач по обработке естественного языка.

Библиография

[1] Devlin J. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin, M. Chang, K. Lee, K. Toutanova // 2019. – URL: https://arxiv.org/abs/1810.04805 (дата обращения: 20.10.2025).

[2] Radford A. Improving Language Understanding by Generative Pre-Training / A. Radford, K. Narasimhan, T. Salimans, I. Sutskever // 2018. – URL: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (дата обращения: 20.10.2025).

[3] Radford A. Language Models are Unsupervised Multitask Learners / A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever // 2019. – URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (дата обращения: 20.10.2025).

[4] Brown T.B. et al. Language Models are Few-Shot Learners // 2020. – URL: https://arxiv.org/abs/2005.14165 (дата обращения: 20.10.2025).

[5] Lewis M. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension / M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer // 2019. – URL: https://arxiv.org/abs/1910.13461 (дата обращения: 20.10.2025).

[6] Raffel C. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer / C. Raffel, N. Shazeer, A. Roberts, K. Lee, S.Narang, M. Matena, Y. Zhou, W. Li, P.J. Liu // 2023. – URL: https://arxiv.org/abs/1910.10683 (дата обращения: 20.10.2025).

[7] Ribeiro M.T. "Why Should I Trust You?": Explaining the Predictions of Any Classifier / M.T. Ribeiro, S. Singh, C. Guestrin // 2016. – URL: https://arxiv.org/abs/1602.04938 (дата обращения: 23.10.2025).

[8] Sundararajan M. Axiomatic attribution for deep networks / M. Sundararajan, A. Taly, Q. Yan // 2017. – URL: https://arxiv.org/abs/1703.01365 (дата обращения: 20.10.2025).

[9] Lundberg S. A Unified Approach to Interpreting Model Predictions / S. Lundberg, S. Lee // 2017. – URL: https://arxiv.org/abs/1705.07874 (дата обращения: 20.10.2025).

[10] Selvaraju R.R. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization / R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra // 2019. – URL: https://arxiv.org/abs/1610.02391 (дата обращения: 20.10.2025).

[11] LIME и SHAP [Электронный ресурс]. – URL: https://habr.com/ru/companies/otus/articles/779430 (дата обращения: 20.10.2025).

[12] Vaswani A. Attention is all you need / A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin // 2017. – URL: https://arxiv.org/pdf/1706.03762.pdf (дата обращения: 23.10.2025).

[13] BERT in DeepPavlov [Электронный ресурс]. – URL: https://deeppavlov-docs.readthedocs.io/en/latest/features/models/bert.html (дата обращения: 23.10.2025).

[14] Николаев П.Л. Классификация книг по жанрам на основе текстовых описаний посредством глубокого обучения / П. Л. Николаев // International Journal of Open Information Technologies. – 2022. – №1. – С. 36–40.

Explainability method of BERT transformer for solving text classification problem

P.L. Nikolaev

***Abstract* — Neural networks, like other machine learning methods, are black boxes. This means it's impossible to understand how they make decisions. However, as neural networks become increasingly complex, developing methods for their explainability becomes increasingly important. This paper proposes a new method for explaining transformers (a type of deep neural network) designed for classifying Russian-language text data. The article examines the BERT transformer, retrained to classify books by genre based on their annotations. The retrained model demonstrates high accuracy when validated on a test set – 84%. Furthermore, the paper presents the explainability method for the BERT model based on clustering its attention heads using the HDBSCAN method. Cluster analysis enabled us to classify the attention heads into distinct groups, which can be used to analyze the model's performance.**

***Keywords* — Artificial intelligence, machine learning, artificial neural networks, deep learning, transformers, text classification.**

REFERENCES

[1] Devlin J. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin, M. Chang, K. Lee, K. Toutanova // 2019. – URL: https://arxiv.org/abs/1810.04805 (date of access: 20.10.2025).

[2] Radford A. Improving Language Understanding by Generative Pre-Training / A. Radford, K. Narasimhan, T. Salimans, I. Sutskever // 2018. – URL: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (date of access: 20.10.2025).

[3] Radford A. Language Models are Unsupervised Multitask Learners / A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever // 2019. – URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (date of access: 20.10.2025).

[4] Brown T.B. et al. Language Models are Few-Shot Learners // 2020. – URL: https://arxiv.org/abs/2005.14165 (date of access: 20.10.2025).

[5] Lewis M. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension / M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer // 2019. – URL: https://arxiv.org/abs/1910.13461 (date of access: 20.10.2025).

[6] Raffel C. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer / C. Raffel, N. Shazeer, A. Roberts, K. Lee, S.Narang, M. Matena, Y. Zhou, W. Li, P.J. Liu // 2023. – URL: https://arxiv.org/abs/1910.10683 (date of access: 20.10.2025).

[7] Ribeiro M.T. "Why Should I Trust You?": Explaining the Predictions of Any Classifier / M.T. Ribeiro, S. Singh, C. Guestrin // 2016. – URL: https://arxiv.org/abs/1602.04938 (date of access: 23.10.2025).

[8] Sundararajan M. Axiomatic attribution for deep networks / M. Sundararajan, A. Taly, Q. Yan // 2017. – URL: https://arxiv.org/abs/1703.01365 (date of access: 20.10.2025).

[9] Lundberg S. A Unified Approach to Interpreting Model Predictions / S. Lundberg, S. Lee // 2017. – URL: https://arxiv.org/abs/1705.07874 (date of access: 20.10.2025).

[10] Selvaraju R.R. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization / R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra // 2019. – URL: https://arxiv.org/abs/1610.02391 (date of access: 20.10.2025).

[11] LIME и SHAP [Electronic resource]. – URL: https://habr.com/ru/companies/otus/articles/779430 (date of access: 20.10.2025).

[12] Vaswani A. Attention is all you need / A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin // 2017. – URL: https://arxiv.org/pdf/1706.03762.pdf (date of access: 23.10.2025).

[13] BERT in DeepPavlov [Electronic resource]. – URL: https://deeppavlov-docs.readthedocs.io/en/latest/features/models/bert.html (date of access: 23.10.2025).

[14] Nikolaev P.L. Klassifikaciya knig po zhanram na osnove tekstovyh opisanij posredstvom glubokogo obucheniya / P.L. Nikolaev // International Journal of Open Information Technologies. – 2022. – №1. – P. 36–40. (in Russ.)