

УДК 004.912

doi: ... заполняется издательством

СРАВНЕНИЕ ПОДХОДОВ К ИНТЕРПРЕТАЦИИ ЯЗЫКОВЫХ МОДЕЛЕЙ: АНАЛИЗ МЕТОДА НА ОСНОВЕ МАСКИРОВАННОГО ЯЗЫКОВОГО МОДЕЛИРОВАНИЯ И ТРАДИЦИОННЫХ МЕТОДОВ

А.А. Рогов (*rogov.alisher@gmail.com*)^А

Н.В. Лукашевич (*louk_nat@mail.ru*)^{А,В}

^А МГТУ им. Н.Э. Баумана, Москва

^В МГУ им. М. В. Ломоносова, Москва

С развитием предобученных языковых моделей, таких как BERT, их применение охватывает всё более ответственные сферы — от рекомендательных систем до медицинской диагностики. Однако рост сложности моделей требует не менее активного развития методов интерпретации, позволяющих понять, на основе каких признаков модель принимает решения. В данной работе исследуются подходы к объяснению поведения BERT в задачах текстовой классификации. Основное внимание уделено сравнению двух парадигм: современных методов, основанных на маскированном языковом моделировании и промпт-обучении, и традиционных техник, таких как LIME и методы на основе векторной близости. Экспериментальный анализ проводится на датасетах Web of Science и 20Newsgroups. Для оценки качества интерпретаций используется построение графиков активации, что позволяет визуализировать значимость входных токенов для конечного предсказания.

Ключевые слова: интерпретация нейросетевых моделей, метод LIME, маскированное языковое моделирование, вербализатор, классификация текста.

Введение

Современные языковые модели, такие как BERT [Devlin et. al., 2019], представляют собой сложные системы, которые часто воспринимаются

как «чёрные ящики». Это создаёт барьеры для их применения в критически важных областях, где требуется уверенность в том, как модель принимает решения.

Одним из ключевых факторов, способных повысить доверие к моделям машинного обучения, является интерпретируемость. Чем лучше пользователь понимает логику работы системы, тем выше вероятность её принятия и эффективного использования. Особенно это важно в таких сферах, как медицина, финансы или юриспруденция, где ошибка может иметь серьёзные последствия.

В рамках этой работы мы продолжили наше предыдущее исследование [Rogov et. al., 2024] по интерпретации моделей в задаче классификации текста, делая акцент на человеко-ориентированном подходе. Подразумевается, что качественное объяснение должно быть семантически связано с предсказанной категорией — пользователь должен видеть явную связь между выделенными моделью признаками и смыслом класса.

Для исследования были выбраны три метода: PromptExplainer [Feng et. al., 2024], LIME [Ribeiro et. al., 2016] и метод на основе векторной близости. Все они предоставляют ранжированный список слов с весами, отражающими степень влияния каждого слова на финальное решение модели. Для сравнительного анализа использовался метод активационных графиков: в модель последовательно подавались фрагменты текста, содержащие по 10% самых значимых слов, и оценивалась вероятность сохранения исходного предсказания. Эксперименты проводились на датасетах Web of Science и 20NewsGroups.

1. Методы интерпретации

1.1. Метод на основе векторного представления слов

Для построения интерпретаций текста на основе семантической близости мы использовали подход, основанный на сравнении векторных представлений слов из текста с вектором, соответствующим целевой категории. Пусть $D = \{d_1, d_2, \dots, d_n\}$ — множество слов текста, подлежащего анализу, а q_i — вектор, представляющий метку категории i . В качестве меры схожести между словами и классом применялось косинусное расстояние между их векторными представлениями:

$$\text{similarity}(d_j, q_i) = \cos(\vec{d_j}, \vec{q_i}),$$

где $\vec{d_j}$ — вектор слова d_j из текста, а $\vec{q_i}$ — вектор метки класса q_i .

В качестве источников векторных представлений были выбраны две хорошо зарекомендовавшие себя модели: GloVe¹ [13] и fastText² [5]. Эти модели имеют широкое применение в задачах NLP благодаря своей способности отражать лексическую и семантическую информацию. GloVe строит вложения на основе статистики совместной встречаемости слов, тогда как fastText учитывает внутреннюю структуру слов, что позволяет ему эффективно обрабатывать омонимы и редкие слова.

Несмотря на существование более современных моделей, таких как MiniLM-L12-H384³, которые демонстрируют улучшенные характеристики по качеству и размеру, мы решили сосредоточиться на проверенных решениях. Это позволило нам получить простой базовый метод, необходимый для корректного сравнения с другими методами.

1.2. LIME

LIME (Local Interpretable Model-agnostic Explanations) — это метод локальной интерпретации, не зависящий от внутренней структуры модели. Его основная идея заключается в аппроксимации поведения сложной модели в окрестности анализируемого примера с помощью более простой и понятной модели, например, линейной регрессии.

В задачах классификации текста входной пример представляется бинарным вектором, где каждая компонента соответствует наличию или отсутствию определённого слова. При этом модель может использовать сложные признаки, такие как эмбединги, недоступные для прямой интерпретации.

Формально пусть $x \in R^d$ — объясняемый пример, $f: R^d \rightarrow R$ — функция, реализуемая моделью, а $g \in G$ — интерпретируемая модель, где G — класс простых моделей. Обычно используется линейная модель вида:

$$g(x') = \phi_0 + \sum_{i=1}^{d'} \phi_i x'_i,$$

где $x' \in \{0,1\}^{d'}$ — упрощённое представление примера, а ϕ_i — веса, характеризующие вклад признаков.

Для построения интерпретации минимизируется функционал:

¹ <http://nlp.stanford.edu/data/glove.840B.300d.zip>

² <https://dl.fbaipublicfiles.com/fasttext/vectors-wiki/wiki.en.zip>

³ <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>

$$\xi(x) = \arg \min_{g \in G} L(f, g, \pi_x) + \Omega(g),$$

где L — мера ошибки аппроксимации, $\pi_x(z)$ — функция близости между образцами, а $\Omega(g)$ — штраф за сложность модели.

Таким образом, LIME позволяет выделить ключевые признаки, повлиявшие на конкретное предсказание, что особенно важно при работе с текстовыми данными.

1.3. PromptExplainer

PromptExplainer — метод интерпретации языковых моделей, основанный на парадигме промпт-обучения (prompt-based learning). Его ключевая идея — использовать внутреннюю задачу маскированного языкового моделирования (masked language modeling, MLM) для оценки значимости токенов, вместо внешних аппроксимаций (таких как градиентные или внимательные методы). Такой подход позволяет получать объяснения, согласованные с реальными механизмами принятия решений модели. Метод состоит из двух основных этапов.

Первым этапом является проекция представлений токенов в пространство словаря. Представления всех токенов (включая незамаскированные) подаются в MLM-голову модели, которая проецирует их в пространство размером, равным количеству слов в словаре. Формально:

$$H_v = M_h(H),$$

где $H \in R^{n \times d}$ — матрица представлений токенов, M_h — MLM-голова, а $H_v \in R^{n \times V}$ — проекция в пространство словаря размера V . Каждый токен теперь представлен вектором, элементы которого указывают на вероятность его связи с каждым словом словаря.

Вторым этапом является извлечение дискриминативных признаков с помощью вербализатора. После получения пространства объясняющих признаков используется вербализатор — отображение между классами и набором слов-меток. Вербализатор помогает выделить те слова из пространства H_v , которые наиболее коррелируют с конкретным классом. Это позволяет сформировать окончательное объяснение в виде списка слов, релевантных данному предсказанию:

$$H_D = V(H_v),$$

где V — функция вербализации, а $H_D \in R^{n \times p}$ — матрица дискриминативных признаков для p классов. Затем для каждого класса

вычисляется softmax-нормализованная вероятность которая служит мерой влияния токена на предсказание:

$$E_i = \text{Softmax}(H_D)[\cdot, c_i].$$

В исследовании использовались два типа вербализаторов: КРТ [Hu et al., 2022] и LogReg. Оба подхода формируют набор слов, связанных с каждым классом задачи, что позволяет строить интерпретации предсказаний модели.

КРТ представляет собой расширенный вербализатор, использующий внешние источники знаний. Такой подход не ограничивается одним фиксированным словом на класс, а создаёт богатое семантическое представление, охватывающее разные уровни абстракции и связи между понятиями. Для построения вербализатора мы опирались на следующие источники⁴⁵⁶. Эти ресурсы, хотя и работают независимо, внутри используют такие базы знаний, как WordNet и ConceptNet, а также учитывают информацию из предобученных векторных представлений слов. Автоматически собранные слова могут содержать шум, поэтому после этапа построения проводилось несколько шагов фильтрации:

- Relevance Refinement (RR) — отбор слов по релевантности целевому классу (удаляются слова с весом ниже порога).
- Frequency Refinement (FR) — удаление слов, редко встречающихся в корпусе. Это помогло отсеять термины, которые, несмотря на семантическую связь, практически не встречались в реальных примерах.
- Contextualized Calibration (CC) — корректировка с учётом частотности в MLM, что снижает риск систематических ошибок.
- Learnable Refinement (LR) — обучение весовых коэффициентов слов для их ранжирования по влиянию на итоговое предсказание.

LogReg - вербализатор на основе логистической регрессии, который строился без привлечения внешних источников знаний. На первом этапе тексты датасета векторизовались с помощью TF-IDF. Затем обучалась модель логистической регрессии, которая для каждого класса сохраняла веса признаков, отражающие вклад слова в вероятность отнесения текста к этому классу. Слова с наибольшими по модулю весами формировали список ключевых терминов класса. Такой подход позволяет учитывать специфику корпуса и адаптировать интерпретации под конкретную задачу.

⁴ relatedwords.org

⁵ describingwords.io

⁶ reversededictionary.org

2. Методы оценки объяснений: активация

Для оценки качества построенных интерпретаций мы использовали метод возмущения входного текста [Ali et. al., 2022] — один из подходов для анализа объяснимости моделей. Основная идея заключается в том, чтобы проверить, насколько точно интерпретация отражает реальные признаки, которыми модель руководствуется при принятии решения.

Процедура оценки:

1. Исходный текст заменяется на «пустую» версию, где все токены заменены на `<unk>`.
2. Слова исходного текста ранжируются по значимости на основе весов, выданных методом интерпретации.
3. В текст постепенно возвращаются наиболее важные слова — порциями по 10% от общего числа слов.
4. После каждого шага измеряется вероятность того, что модель выдаст исходное предсказание для целевого класса. Этот показатель называется активационной вероятностью.
5. Для итоговой оценки используется метрика активации — усреднённое значение этой вероятности по всем шагам добавления слов, что отражает скорость и качество восстановления правильного предсказания на основе ключевых признаков.

Чем быстрее растёт активационная вероятность при добавлении небольшого числа слов, тем точнее считается объяснение, что указывает на верное выделение моделью ключевых признаков.

Подход, ранее использованный в работах Ali et al. (2022), показал чувствительность к качеству интерпретаций, устойчивость к шуму и универсальность для различных методов. В наших экспериментах все интерпретации строились на одной модели с использованием официальных реализаций, что исключало влияние архитектурных различий и позволяло сравнивать именно способность методов выделять значимые признаки.

3. Наборы данных

Для проведения экспериментов были выбраны два широко известных датасета: 20Newsgroups⁷ и Web of Science (WOS) [Kowsari et. al., 2017]. Оба датасета предназначены для задачи многоклассовой классификации текста и имеют различия как в структуре данных, так и в сложности задачи.

⁷ <http://people.csail.mit.edu/jrennie/20Newsgroups/>

Датасет WOS представляет собой коллекцию аннотаций научных публикаций, взятых из базы данных Web of Science. Он содержит три версии корпусов разного размера: 5736, 11967 и 46985 документов, соответствующих 11, 34 и 134 темам соответственно. В рамках исследования мы работали с версией, которая включает 11967 текстовых примеров, относящихся к 34 темам. Выборка была разделена в соотношении 70% / 30% на обучающую и валидационную части.

Особую ценность этого датасета представляет наличие двух уровней классификации, что позволяет анализировать методы интерпретации как на уровне широких научных областей, так и при работе с узкими специализациями:

- Первый уровень (WOS_L1) классификации включает семь обширных научных направлений, таких как «Информатика», «Электротехника», «Психология», «Машиностроение», «Строительная инженерия», «Медицинские науки» и «Биохимия».
- Второй уровень (WOS) классификации содержит 34 более специализированные категории, среди которых можно выделить такие темы, как «Обработка изображений», «Машинное обучение», «Социальное восприятие», «Гидравлика», «Генетика» и другие.

Для работы с составными названиями категорий, такими как «Machine learning» или «Water Pollution», мы использовали усреднение векторов отдельных слов. Это обеспечило корректную работу методов, основанных на векторных представлениях (например, GloVe и fastText), и позволило точно отразить семантику сложных меток.

Датасет 20Newsgroups состоит из 18846 сообщений из новостных групп, охватывающих 20 различных тем. Данные отличаются большей свободой стиля и разнообразием тем, что усложняет классификацию по сравнению с WOS. Использовалось стандартное разбиение: 14846 примеров для обучения и 4000 для валидации.

Перед обучением из сообщений удалялись метаданные (заголовки, служебная информация), чтобы модель ориентировалась на содержимое текста. Длина документа была ограничена 1000 слов для ускорения обучения и оценки.

Оба датасета были выбраны по нескольким причинам:

- Они покрывают разные домены: научные аннотации (WOS) и пользовательские сообщения (20Newsgroups).
- Содержат разное количество классов и примеров, что позволяет оценить эффективность методов на задачах разной сложности.

- Хорошо исследованы и часто используются в литературе, что обеспечивает возможность сравнения с результатами других работ.

4. Эксперименты

В данной работе был реализован метод интерпретации на основе семантической близости слов из текста и названия класса, для чего использовались предобученные модели векторных представлений слов GloVe и fastText. Каждое слово сравнивалось с вектором класса по косинусной мере близости, после чего слова ранжировались по убыванию этой меры, формируя список наиболее релевантных слов для данной категории.

Для сравнения мы также применили традиционные методы интерпретации. Модель BERT (bert-base-uncased) [2] была дообучена на наших датасетах в стандартной задаче классификации. На основе этой дообученной модели с помощью библиотеки LIME строились объяснения для каждого текста. В LIME передавалась фактическая метка класса, чтобы обеспечить корректное сравнение и исключить влияние ошибок модели на результаты интерпретации. Параметры LIME были установлены следующим образом: максимальное количество признаков в объяснении — 256, размер окрестности для локальной модели — 300.

Для реализации PromptExplainer мы использовали фреймворк OpenPrompt [Ding et. al., 2022], который предоставляет удобную инфраструктуру для промпт-обучения. Здесь использовалась модель BERT (bert-base-uncased) и обучение проходило в условиях few-shot — по 5 примеров на класс. Формирование промпта осуществлялось по шаблону «[Category : <MASK>] Текст». После обучения вычислялась значимость слов относительно фактической метки класса, что позволяло исключить влияние ошибок классификации на качество интерпретаций. В качестве вербализаторов использовались два типа: основанный на внешних знаниях KPT и построенный на весах модели логистической регрессии LR. Объяснения формировались с помощью softmax-весов слов из вербализатора, что позволяло ранжировать слова по степени их связи с целевой категорией.

В таблице 1 представлены результаты классификации с помощью промпт-обучения с вербализаторами KPT, LogReg (Prompt_KPT, Prompt_LogReg) и с помощью дообученной (Fine-tuned) модели BERT, измеренные метрикой accuracy. Таблица 2 показывает результаты оценки качества объяснений с помощью метрики активации. В таблице сравниваются методы PromptExplainer с двумя вербализаторами (LogReg и KPT), классический LIME, семантические методы на базе GloVe и fastText.

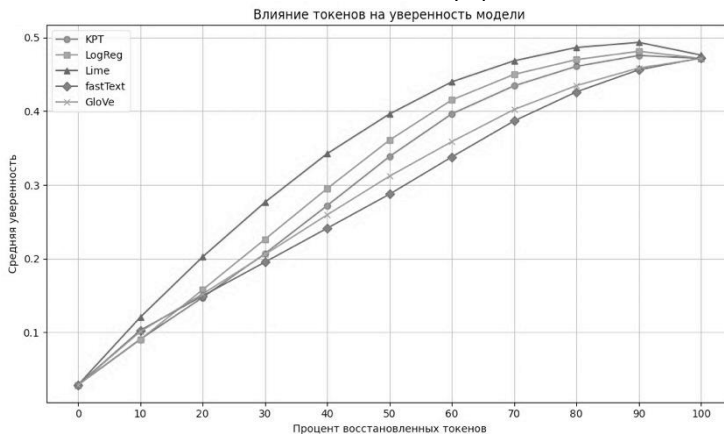
Табл. 1.

	WOS	WOS L1	20Newsgroups
Fine-tuned	86.3	93.1	71.3
Prompt KPT	43.6	65.3	52.7
Prompt LogReg	52.6	70.1	56.4

Табл. 2.

	WOS	WOS L1	20Newsgroups
LogReg	31.3	60.2	39.4
KPT	30.2	58.0	38.3
LIME	33.9	63.4	43.5
GloVe	28.9	52.7	35.6
fastText	28.0	56.4	36.6

Рис. 1. График активации для датасета WOS



На рисунках 1, 2 и 3 представлены графики активации для датасетов WOS, WOS_L1 и 20Newsgroups соответственно. Согласно приведённым данным, метод LIME демонстрирует наилучшие результаты среди всех рассмотренных подходов.

Хотя метод LIME показал лучшие результаты по метрике активации во всех рассмотренных экспериментах, это не означает, что PromptExplainer уступает концептуально. Различия в подходах могут объяснять наблюдаемые результаты.

Рис. 2. График активации для датасета WOS_L1

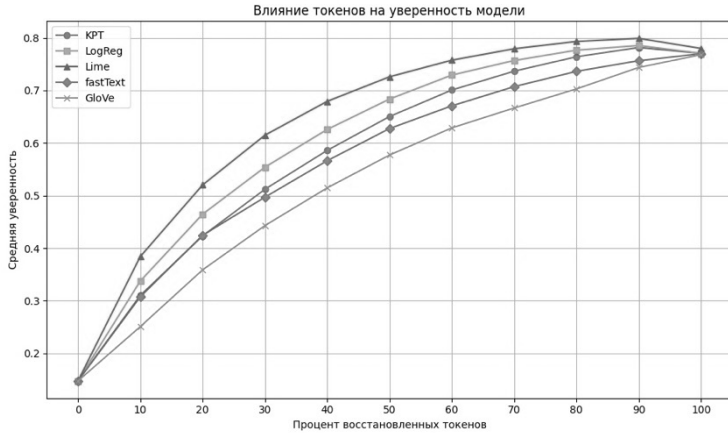
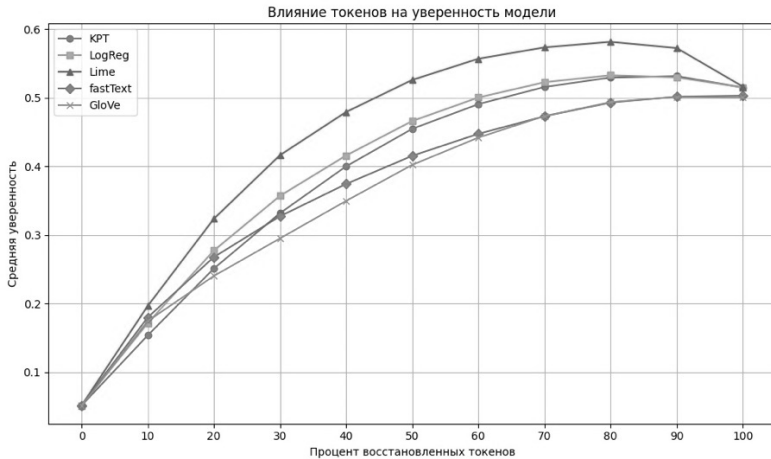


Рис. 3. График активации для датасета 20Newsgroups



LIME является локальным методом, который строит интерпретацию, аппроксимируя поведение модели в окрестности конкретного примера, что позволяет ему точнее выявлять релевантные признаки для данного конкретного текста. В то же время PromptExplainer опирается на глобально сформированные вербализаторы и промпты, которые могут не всегда достаточно гибко учитывать индивидуальные особенности каждого примера.

Кроме того, качество PromptExplainer сильно зависит от выбора и качества вербализатора. В данной работе использовались два типа вербализаторов: один на основе внешних знаний и один на основе весов логистической регрессии. Несмотря на то, что оба подхода обеспечивают адекватные результаты, возможно, они не отражают всех нюансов семантической релевантности для конкретных задач и текстов.

Также стоит отметить, что LIME напрямую использует предобученную BERT-модель, дообученную на конкретном датасете, что даёт ему преимущество в более точной локальной интерпретации. PromptExplainer в экспериментах работал в few-shot режиме, что ограничивает объем доступной информации для обучения промптов и, как следствие, может снижать качество интерпретаций.

Таким образом, разница в результатах скорее отражает особенности реализации и настройки методов, а не фундаментальные ограничения PromptExplainer. Перспективным направлением дальнейших исследований может стать улучшение вербализаторов. Это позволит более полно раскрыть потенциал PromptExplainer и сделать его более конкурентоспособным с существующими методами.

Заключение

В ходе исследования были рассмотрены и сравнены различные методы интерпретации предобученных языковых моделей, с особым акцентом на подход PromptExplainer, который базируется на внутренней задаче MLM и использовании вербализатора для построения объяснений. Также проведено сравнение с традиционными методами, такими как LIME и метод на основе косинусной близости слов.

Анализ экспериментов показал, что PromptExplainer способен формировать осмысленные и достаточно точные объяснения, однако его эффективность существенно зависит от качества и выбора вербализатора. В то же время метод LIME продемонстрировал более стабильные и высокие результаты во всех оценочных метриках, особенно при анализе активационных графиков, что связано с его локальным подходом к интерпретации и использованию дообученной модели BERT.

Полученные результаты указывают на существующие ограничения текущей реализации PromptExplainer и необходимость дальнейших исследований, направленных на улучшение вербализаторов и адаптацию метода к особенностям конкретных задач и данных.

Таким образом, данное исследование подчёркивает важность учёта как архитектурных особенностей предобученных моделей, так и качества вербализаторов при построении интерпретаций, а также открывает новые направления для развития методов объяснимости в области NLP.

Список литературы

- [Ali et. al., 2022] Ali, A., Schnake, T., Eberle, O., Montavon, G., Müller, K. R., & Wolf, L. (2022, June). XAI for transformers: Better explanations through conservative propagation. In International conference on machine learning (pp. 435-451). PMLR. *(статья в сборнике трудов конференции на англ. языке)*
- [Devlin et. al., 2019] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers) (pp. 4171-4186).. *(статья в сборнике трудов конференции на англ. языке)*
- [Ding et. al., 2022] Ding, N., Hu, S., Zhao, W., Chen, Y., Liu, Z., Zheng, H., & Sun, M. (2022, May). OpenPrompt: An Open-source Framework for Prompt-learning. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (pp. 105-113). *(статья в сборнике трудов конференции на англ. языке)*
- [Feng et. al., 2024] Feng, Z., Zhou, H., Zhu, Z., & Mao, K. (2024, March). PromptExplainer: Explaining Language Models through Prompt-based Learning. In *Findings of the Association for Computational Linguistics: EACL 2024* (pp. 882-895). *(книга на англ. языке)*
- [Hu et. al., 2022] Hu, S., Ding, N., Wang, H., Liu, Z., Wang, J., Li, J., ... & Sun, M. (2022, May). Knowledgeable Prompt-tuning: Incorporating Knowledge into Prompt Verbalizer for Text Classification. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 2225-2240). *(статья в журнале на англ. языке)*
- [Kowsari et. al., 2017] Kowsari, K., Brown, D. E., Heidarysafa, M., Meimandi, K. J., Gerber, M. S., & Barnes, L. E. (2017, December). Hdltext: Hierarchical deep learning for text classification. In 2017 16th IEEE international conference on machine learning and applications (ICMLA) (pp. 364-371). IEEE.. *(статья в сборнике трудов конференции на англ. языке)*
- [Ribeiro et. al., 2016] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144). *(статья в сборнике трудов конференции на англ. языке)*
- [Rogov et. al., 2024] Rogov, A. A., & Loukachevitch, N. V. (2024). Evaluating the Performance of Interpretability Methods in Text Categorization Task. Lobachevskii Journal of Mathematics, 45(3), 1234-1245. *(статья в сборнике трудов конференции на англ. языке)*

COMPARISON OF APPROACHES TO INTERPRETING LANGUAGE MODELS: ANALYSIS OF A MASKED LANGUAGE MODELING-BASED METHOD AND TRADITIONAL TECHNIQUES

A.A. Rogov (*rogov.alisher@gmail.com*)^A

N.V. Loukachevitch (*louk_nat@mail.ru*)^{A,B}

^A Bauman Moscow State Technical University, Moscow

^B Lomonosov Moscow State University, Moscow

As pretrained language models such as BERT continue to grow in complexity and application — from recommendation systems to medical diagnostics — the demand for effective interpretation methods becomes increasingly important. Understanding how these models make decisions is crucial for building trust and ensuring their reliable use. In this work, we investigate approaches to explaining the behavior of BERT in text classification tasks. The focus is on comparing two paradigms: modern techniques based on masked language modeling and prompt-based learning, and traditional methods such as LIME and vector similarity-based approaches. The experimental analysis is conducted on the Web of Science and 20Newsgroups datasets. Interpretation quality is evaluated using activation plots, which visualize the significance of input tokens in the final prediction.

Keywords: neural network model interpretation, LIME method, masked language modeling, verbalizer, text classification.