Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

Интерпретируемые методы NLP

Александров Валентин Валерьевич

Москва, 2021

Содержание

Введение

Современные модели машинного обучения показывают хорошие результаты в своих предсказательных способностях, что их примение в реальных задачах стало обыденным явлением. Однако этого удалось добиться за счет повышения сложности моделей. Это привело к тому, что современные модели машинного обучения являются **черными ящиками** - системами, чье внутреннее устройство и процесс получения ответа являются исключительно трудными для анализа. В противовес прозрачным "простым"моделям вроде линейной регрессии и решающим деревьям, мы зачастую не можем достоверно достоверно ответить на вопрос "как работает эта система"в случае современных нейронных сетей.

Существуют опасения относительно надежности и оправданности предсказаний, которые дают современные ML-системы. Это особенно критичная проблема для областей, в которых принимаются решения с очень большой ответственностью: здравоохранение, правосудие, финансы и т. д. Для этих областей, существуют повышенные требования к безопасности, этичности и ответственности принимаемых решений. Уже существует обильное множество примеров реальных нейросетевых моделей, которые показывают нежелательные поведения и возникающие с этим этические проблемы.

Проблема в удовлетворении к возросшим требованиям к безопасности систем машинного обучения в том, что мы не можем доподлинно покрыть юнит-тестами все возможные поведения модели, как это возможно в сфере разработки программного обеспечения. Чтобы убедиться в надежности и безопасности модели вместо этого приходится полагаться на интерпретируемость. Если модель интерпретируема для нас, то мы опосредованно можем судить о её надежноссти.

К сожалению, общепринятого формального определения интерпретации не существует. В контексте ML систем в [1] дается определение интерпретируемости как "способность объяснить или представить в понятных человеку терминах".

Мотивация

В [2] утверждается, что всю мотивацию интерпретируемости моделей машинного обучения можно свести к **ответственности** за их решения. В случае ошибочного предсказания модели нам необходимо понимать, почему эта ошибка произошла. А в случае когда цена ошибки модели слишком высока, мы хотим минимизировать возможные риски модели перед её развертыванием. Этими двумя случаями мы описали ситуации, когда мы применяем интерпретацию **ретроактивно** и **проактивно** соответственно. Ретроактивные интерпретации получили большое распространение благодаря возможности использовать их в судопроиз-

водстве и в уже существующих структурах оценки и контроля качества в банковской среде.(Bhatt et al., 2019)

В [2] [1] обощаются следующие ключевых фактора мотивации развития методов интерпретируемости:

— **Этичность**. Подразумевает, что поведение модели согласуется с представлениями общества об этике и морали. Так как в целом понятие "этичности поведениярудно оценить и представить объективную метрику@"трудно оценить и представить объективную метрику для этого, обычно это оценивается вручную специально назначенными людьми. Типично для рассмотрения таких вопрос в крупных компаниях существуют целые отделы по этике. Типичный вопрос представляющий опасения об этичности модели - проблема дискримации. И уже существует ряд работ, посвященные формализации метрики "справедливости"(fairness) и методам устранения нежелательных предубеждений. Но так как это является лишь одним из многих возможных этических проблем, нужда во вручной человеческой оценке остается.

— **Безопасность**. Обозначает то, что модель работает в рамках наших ожиданий. Зачастую подразумевает собой также устойчивость к adversarial атакаи и к смещению распределений. В более широком смысле [3] под этим также подразумевается доверие к модели.

— **Подотчетность**. Обозначает способность модели "объясниться"в случае неудачи в production-среде. Требование к способности модели объяснить свое решение сейчас все больше и больше распространяется - сейчас это проявляется в виде постановления GDPR от Европейского Союза и банковской среде.

— **Научный интерес**. Помимо простого удовлетворения человеческой жажды знаний со стороны исследователей, скрывает за собой также более приземленную причину. Если мы понимаем, почему модель должна работать, мы также можем ответить и на вопрос "почему наша конкретная модель не работает как задуманно". То есть, методы интерпретирования моделей очень полезны для процесса дебагинга моделей

1 Метрики интерпретируемости

Из-за того, что для понятий "объяснения"и "интерпретации"не существует формального определения, встает очень большая проблема как оценивать качество объяснений, предложенных тем или иным методом, и как эти методы сравнивать между собой. В рассмотрены различные методики оценивания и выделены три категории: оценивания приложением, людьми и функциональное оценивание (Application-grounded, Human-grounded, Functually-grounded соответственно).

— **Application-grounded оценивание**. Суть в том, что если у нас есть реальная задача с ML-системой асистирующей человеку, можно оценивать то, насколько хорошо система взаимодействует с человеком. Хорошим примером является сегментация на снимках томографии. Объяснения работают хорошо, если они сокращают время работы эксперта, который работает с системой, позволяют находить ошибки и т. д. Очевидная проблема этого метода оценивания заключается в том, что он требует работы всей системы в production-среде. Более того, это требует трат времени и сил доменного эксперта, что делает этот вид оценивания наиболее дорогостоящим.

— **Human-grounded оценивание**. Суть его в том, что человеку даются *простые* задачи, по которым мы пытаемся понять ценность интерпретации модели. Главное отличие от Application-grounded оценивания заключается в том, что не требуется дорогостоящее время эксперта, чтобы протестировать заданную методику. Примеры экспериментов:

Бинарный выбор, при котором человек должен из пары представленных объяснений выбрать более качественное

Симуляция инференса. Человек при заданном объяснении и входном наблюдении должен корректно "просимулировать"модель и угадать её предсказание

Симуляция контрфактов. Человек при заданом объясненнии, входе и выходе модели должен ответить, как следует поменять вход так, чтобы изменилось предсказание модели.

— **Functually-grounded оценивание**. Выбирается простая вычислимая прокси-метрика, по которой мы опосредованно судим об интерпретируемости модели. Так как её можно считать автоматически, без участия ручного труда человека, её очень удобно использовать для первичного оценивания методик. Её недостаток в том, что зачастую очень сложно выбрать работающую прокси-метрику. Для каждой выбранной прокси-метрики требуется веское обоснование, почему это можно использовать. Пример прокси-метрики - разреженность эмбеддингов, если ранее было показано, что разреженные модели лучше интерпретируются

2 Методы интерпретации

В этой главе мы кратко представим существующие подходы интерпретации моделей машинного обучения.

Следует представить некоторую категоризацию имеющихся методов. Популярным подходом является разделение методов на три широкие группы: локальные, глобальные объяснения и объяснения отдельных классов. **Локальные объяснения** предназначены для ретроактивного объяснения результатов только для одного заданного наблюдения. **Глобальные объяснения** пытаются суммаризовать информацию о всей модели целиком, но обычно по отношению только к одному выбранному аспекту. **Классовые объяснения** также пытаются объяснить всю модель, но только по отношению работы с одним конкретным классом.

Также в [2] дополнительно проводится различие между внутренними (intrinsic) и post-hoc методами. Внутренние методы в своей работы полагаются на устройство рассматриваемой модели. Зачастую это относится к моделям, которые в силу свое простой структуры изначально интерпретируемы. Хотя и существуют примеры для более сложных моделей, например, Attention-слои, при которых мы можем судить о важности определенных токенов при генерации предсказания. Post-hoc же методы предоставляют объяснения только после того как модель обучена и зачастую эти методы агностичны к рассматриваемой модели.

Также методы инетрпретации различаются и по виду их результата. Это может быть обощающая статистика по входным признакам (пример - feature importance в случайном лесе), некоторая обощающая визуализация по входным признакам, интерпретация по весам модели (для линейной регрессии и решающего дерева), некоторое противоположное или похожее наблюдение (контрфакты и adversarial-примеры) или аппроксимация рассматриваемой модели через более простую внутрене интерпретируемую модель.

2.1 Локальные объяснения

2.1.1 Входные признаки

В данной группы методов мы пытаемся понять, насколько важны были те или иные признаки на входе для заданного предсказания. В случае NLP-моделей мы "подсвечиваем"важные токены текста на входе, в случае CV-моделей - пиксели заданного изображения.

Классическим представителем является градиентный метод [4], в котором для классификатора мерой важности признака является значение градиента.

Развитием градиентного метода является специфичный для нейросетей метод **DeepLIFT** [5], который заключается в сравнении работы нейронов с некоторым "примерным"наблюдением. В случае изображения это может быть серое изображение, в случае текста - пустой текст из [PAD] токенов.

Другим популярным методом является **LIME** [6]. Суть его заключается в семплировании похожих объектов и и обучения на их предсказаниях логистической регрессии. Чем больше параметры локальной логистической регресии - тем важнее этот объект. В NLP объекты семплируются с помощью BoW с косинусным расстоянием и с помощью маскирования токенов. Для изображений используются суперпиксели.

Проблемой LIME является то, что при наличии мультиколлинеарности признаков значения весов линейной модели становится трудно интерпретировать. В методе **SHAP** [7] используются теоретически наработки с векторами Шепли для решения этой проблемы. От этого появляется другой недостаток - для работы SHAP требуется оценить большое количество сгенерированных наблюдений.

Существует критика таких подходов, которая заключается в том, что показанная важность определенных токенов может не отражать реальную значимость для предсказаний. В частности, [8] показан метод, как можно сфабриковать нужные важности токенов используя Attention-слои. Также [9] удалось для LIME и SHAP подделать результаты, чтобы скрыть реальную работу модели.

2.1.2 Adversarial примеры

В этой категории методов мы пытаемся "обмануть"модель, подобрав похожий объект, на котором она выдала бы другое предсказание. Adversarial-атаки произошли из области исследований, посвященной теме устойчивости моделей. Однако дает нам некоторое понимание работы модели.

В [10] утверждается, что контрастные суждения о модели, которые сравнивают заданный пример с другим, более понятны для человека, чем важность признаков.

Примеры adversarial методов:

— **HotFlip** [11]. В этом методе изменяются токены в предложении так, чтобы максимизировать изменение функции потерь. В качестве меры важности токенов используется величина градиента при изменении. Процедура измения одного токена применяется на одном и том же предложении множество раз с помощью beam search процедуры. В оригинальной статье предлагается интерпретация модели, работающей на символьном уровне, но метод можно адаптировать на любые

последовательности, подбирая кандидатов на замену с помощью косинусного расстояния.

— **Semantically Equivalent Adversaries (SEA)** [12]. Используется метрика семантической эквивалентности текстов, которая вычисляется как условное правдоподобие текста относительно оригинального текста. Наиболее похожий по этой метрике текст заменяется и процедура может повторяться множество раз, пока не будет удовлетворен критерий останова. Важно отметить различие от HotFlip метода, парафразирующая модель может удалять, добавлять и изименять сразу множество токенов за раз.

— **BERT-ATTACK** [13] для генерации примеров использует специальная предобученную BERT модель. Генерация происходит в два этапа: BERT определяет наиболее уязвимые слова в заданном предложении и затем эти слова маскируются, чтобы BERT предсказал другие токены на их место.

2.1.3 Контрфакты (Counterfactuals)

Пытаемся отредактировать вход так, чтобы поменялось предсказание. Counterfactual должны минимально отличаться от исходного примера. Похожий по смыслу, на Adversarial примеры, Но в Adversarial примерах мы пытаемся подобрать парафразу, чтобы изменить предсказать, контрфакты должны семантически совпадать.

Примером метода генерации контрфактов является **Polyjuice** [14]. Методика модель-агноситчна. Выполняется с помощью файнтюнинга GPT-2 на существующем датасете контрфактов, генерируя для обучения специально построенные предложения из пар оригинального и контрфактических примеров. Изучаемая модель взаимодействует с системой только на этапе фильтрации контрфактов - выбираются только те, которые значительно меняют предсказание модели.

В отличие от Polyjuice методика **MiCE** предложенная в [15] более тесно работает с объясняемой моделью.

Для генерации контфрактов используется seq2sec модель, пытающаяся предсказать замаскированные слова по метке класса. Например, для положительной метки, модель должна для предложения "*Фильм был [BLANK]*"предсказать "*хороший*". Токены для маскирования выбираются по их важности, определенной с помощью градиентного метода.

Для генерации контрфактов используется обученная модель, с подмененной меткой класса. Так, модель берет некоторый важный токен, маскирует его и заменяет на противоположный по смыслу. Процедура генерации повторяется множество раз на одном предложении с помощью beam search процедуры.

2.1.4 Похожие примеры

По заданному наблюдению, пытаемся найти похожие на него, с точки зрения модели. Эти методы дополнительно полезен тем, что мы берем объекты из датасета, что дает нам возможность также лучше понять рабочий датасет.

В [16] для отслеживания похожих примеров используется классическая техника функций влиятельности, которая в состоит измерении изменении значении функции потерь при удалении некоторого объекта из обучающего набора данных. Эта техника позволяет отследить причины предсказания модели к её тренировочному датасету. В данной работе применение функции влиятельности было расширена на BERT модель.

2.1.5 Объяснения на естественном языке

Пытаемся дать объяснения в виде текста простым языком. При этом объяснения должны быть понятны не-экспертам. Интересной особенностью применения такого метода на NLP-моделях заключается в том, что сгенерериованные объяснения могут использоваться для улучения качества объясняемой модели.

В работе [17] представлена похожая методика CAGE (Commonsense Auto-Generated Explanations) для генерации ответов на вопросы с заданными вариантами ответов. К существующему QA датасету с помощью Mechanical Turk к каждой паре "вопрос-возможные ответы"было подобрано объяснения ответа. Например для вопроса "Можно заняться вязанием, чтобы почувствовать что?"с предлагаемыми ответами "спокойство"и "артрит дается объяснение "Вязание позволяет успокоиться". На этом расширенном датасете дообучается GPT, чтобы он мог генерировать эти объяснения.

Авторы предлагают специальный режим рационализации, при котором обучение GPT дополнительно обусловлено предскзанным сторонней моделью ответом. Таким образом GPT пытается объяснить полученное предсказание. Проблема состоит в том, что GPT очень слабо связана с исходной моделью - смысл методики состоит прежде всего в улучшении качества предсказаний с помощью сгенерированных предсказаний. Полученнные объяснения в этом контексте лишь побочный продукт.

В [18] используется похожий подход в решении задачи QA, при котором модель-генератор пытается предложить гипотетические ответы на предложенный вопрос, на которых модель-классификатор дополнительно обуславливается.

2.2 Глобальные объяснения

2.2.1 Словарные объяснения

В этих методах мы пытаемся понять модель через её словарь. Зачастую мы для этого изучаются вектора-эмбеддинги модели.

Одним из простых подходов к изучению эмбеддингов является их визуализация с помощью сокращения количества измерений векторов методом t-SNE. Такой метод был предложен в [19]. Это позволяет увидеть некоторую структуру в словаре модели. Например, образуются отдельные кластеры семантически похожих слов.

2.2.2 Ансамбли

Этот тип производных методов, при которых мы пытается дать объяснение всей модели как комбинацию локальных объяснений. Технически, это можно выполнить любым представленным локальным методоам, что делает эту категорию широкой. Очевидным недостатком этой категории заключается в дороговизне - получение одного локального объяснения может быть ёмкой задачей само по себе. Поэтому при работе с этими методами пытаются выбрать из датасета малое число наблюдений для интерпретации так, чтобы они могли лучше всего отобразить работу модели.

Следует отметить метод, предложенные авторами LIME [6], в котором производится попытка выбрать ограниченное число наблюдений, которые с помощью LIME метода могли бы лучше всего отразить работу модели.

2.2.3 Лингвистическая информация

Чтобы подтвердить разумность NLP модели, можно согласовать модель с обширной лингвистической теорией.

Методы в этой категории либо исследуют реакцию модели на специальные изменения входа или пытаются показать соответствия латентного и некоторого лигвистического представления. Первое называется поведенческим анализом, второе - структурным анализом

2.2.4 Правила

Эти методы пытаются представить обученную модель как набор простых правил.

В упомянутой ранее работе [12] метод для генерации adversarial примеров SEA расширяется для последующей генерации правил. Для этого анализируются паттерны новых примеров, на которых модель часто начинает менять предсказание на неправильное.

В работе [20] авторы пытаются объяснить формальными правилами представления отдельных нейроной. В случае изображений, для заданного нейрона и наблюдений собираются маски, на которых срабатывает нейрон. Также имеются набор атомарных концептов и для для всех изображений масками отмечено присутствие концептов на изображении. Для полученных масок активации нейронов собираются концепты, присутствующие на этих масках. Затем с помощью процедуры beam search собранные концепты и их области объединяются логическими операторами И, ИЛИ и НЕ так, чтобы как можно точнее покрыть области активации нейрона. Также авторы адоптируют этот подход на задаче natural language inference (NLI).

Заключение

Проблема интерпретируемости для современных моделей машинного обучения является актуальной проблемой и вызывает интерес как со стороны исследователей, так и со стороны регулирующих органов.

Данный обзор коротко затрагивает существующие современные подходы к интерпретации систем машинного обучения, методы их оценивания и мотивацию.

Основные проблемы этой сферы заключаются в отсутствии общепринятых формальных определений и следующая из-за этого сложность измерения качества. Для большого числа предложенных методик сложно подобрать дешевую функциональную метрику и приходится полагаться на тестирования с участием людей, что намного более сложно и дорого.

Интересно заметить, что для большой части методов интерпретируемость является лишь побочным продуктом. Предложенные методы adversarial примеров, контрфактов и объяснений естественным языком прежде всего предназначались для улучшения качества работы моделей.

Список использованных источников

1. *Doshi-Velez, Finale.* Towards A Rigorous Science of Interpretable Machine Learning / Finale Doshi-Velez, Been Kim // *arXiv:1702.08608 [cs, stat].* — 2017. — . — arXiv: 1702.08608. http://arxiv.org/abs/1702.08608.

2. *Madsen, Andreas.* Post-hoc Interpretability for Neural NLP: A Survey / Andreas Madsen, Siva Reddy, Sarath Chandar // *arXiv:2108.04840 [cs].* — 2021. — . — arXiv: 2108.04840. http://arxiv.org/abs/2108.04840.

3. *Lipton, Zachary C.* The mythos of model interpretability / Zachary C. Lipton // *Communications of the ACM.* — 2018. — . — Vol. 61, no. 10. — Pp. 36–43. https://doi.org/10.1145/3233231.

4. How to Explain Individual Classification Decisions / David Baehrens, Timon Schroeter, Stefan Harmeling et al. // *arXiv:0912.1128 [cs, stat].* — 2009. — . — arXiv: 0912.1128. http://arxiv.org/abs/0912.1128.

5. *Shrikumar, Avanti.* Learning Important Features Through Propagating Activation Differences / Avanti Shrikumar, Peyton Greenside, Anshul Kundaje // *arXiv:1704.02685 [cs].* — 2019. — . — arXiv: 1704.02685. http://arxiv.org/abs/1704.02685.

6. *Ribeiro, Marco Tulio.* "Why Should I Trust You?": Explaining the Predictions of Any Classifier / Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin // *arXiv:1602.04938 [cs, stat].* — 2016. — . — arXiv: 1602.04938. http://arxiv.org/abs/1602.04938.

7. *Lundberg, Scott.* A Unified Approach to Interpreting Model Predictions / Scott Lundberg, Su-In Lee // *arXiv:1705.07874 [cs, stat].* — 2017. — . — arXiv: 1705.07874. http://arxiv.org/abs/1705.07874.

8. Learning to Deceive with Attention-Based Explanations / Danish Pruthi, Mansi Gupta, Bhuwan Dhingra et al. // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. — Online: Association for Computational Linguistics, 2020. — . — Pp. 4782–4793. https://aclanthology.org/2020.acl-main.432.

9. Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods / Dylan Slack, Sophie Hilgard, Emily Jia et al. // *arXiv:1911.02508 [cs, stat].* — 2020. — . — arXiv: 1911.02508. http://arxiv.org/abs/1911.02508.

10. *Miller, Tim.* Explanation in artificial intelligence: Insights from the social sciences / Tim Miller // *Artificial Intelligence.* — 2019. — . — Vol. 267. — Pp. 1–38. https://linkinghub.elsevier.com/retrieve/pii/S0004370218305988.

11. HotFlip: White-Box Adversarial Examples for Text Classification / Javid Ebrahimi, Anyi Rao, Daniel Lowd, Dejing Dou // *arXiv:1712.06751 [cs].* — 2018. — . — arXiv: 1712.06751. http://arxiv.org/abs/1712.06751.

12. *Ribeiro, Marco Tulio.* Semantically Equivalent Adversarial Rules for Debugging NLP models / Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Melbourne, Australia: Association for Computational Linguistics, 2018. — . — Pp. 856–865. https://aclanthology.org/P18-1079.

13. BERT-ATTACK: Adversarial Attack Against BERT Using BERT / Linyang Li, Ruotian Ma, Qipeng Guo et al. // *arXiv:2004.09984 [cs].* — 2020. — . — arXiv: 2004.09984. http://arxiv.org/abs/2004.09984.

14. Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models / Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, Daniel S. Weld // *arXiv:2101.00288 [cs].* — 2021. — . — arXiv: 2101.00288. http://arxiv.org/abs/2101.00288.

15. *Ross, Alexis.* Explaining NLP Models via Minimal Contrastive Editing (MiCE) / Alexis Ross, Ana Marasović, Matthew E. Peters // *arXiv:2012.13985 [cs].* — 2021. — . — arXiv: 2012.13985. http://arxiv.org/abs/2012.13985.

16. *Han, Xiaochuang.* Explaining Black Box Predictions and Unveiling Data Artifacts through Influence Functions / Xiaochuang Han, Byron C. Wallace, Yulia Tsvetkov // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. — Online: Association for Computational Linguistics, 2020. — . — Pp. 5553–5563. https://aclanthology.org/2020.acl-main.492.

17. Explain Yourself! Leveraging Language Models for Commonsense Reasoning / Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, Richard Socher // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. — Florence, Italy: Association for Computational Linguistics, 2019. — . — Pp. 4932–4942. https://aclanthology.org/P19-1487.

18. *Latcinnik, Veronica.* Explaining Question Answering Models through Text Generation / Veronica Latcinnik, Jonathan Berant // *arXiv:2004.05569 [cs].* — 2020. — . — arXiv: 2004.05569. http://arxiv.org/abs/2004.05569.

19. Visualizing and Understanding Neural Models in NLP / Jiwei Li, Xinlei Chen, Eduard Hovy, Dan Jurafsky // Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. — San Diego, California: Association for Computational Linguistics, 2016. — . — Pp. 681–691. https://aclanthology.org/N16-1082.

20. *Mu, Jesse.* Compositional Explanations of Neurons / Jesse Mu, Jacob Andreas // *arXiv:2006.14032 [cs, stat].* — 2021. — . — arXiv: 2006.14032. http://arxiv.org/abs/2006.14032.

21. Concept2vec: Metrics for Evaluating Quality of Embeddings for Ontological Concepts / Faisal Alshargi, Saeedeh Shekarpour, Tommaso Soru,

Amit Sheth // *arXiv:1803.04488 [cs]*. — 2020. — . — arXiv: 1803.04488. http://arxiv.org/abs/1803.04488.

22. *Alatrish, Emhimed*. Building ontologies for different natural languages / Emhimed Alatrish, Dusan Tosic, Nikola Milenkovic // *Computer Science and Information Systems*. — 2014. — Vol. 11, no. 2. — Pp. 623–644. http://www.doiserbia.nb.rs/Article.aspx?ID=1820-02141400023A.

23. DBpedia: A Nucleus for a Web of Open Data / Sören Auer, Christian Bizer, Georgi Kobilarov at al. // The Semantic Web / edited byDavid Hutchison, Takeo Kanade, Josef Kittler at al. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. — Vol. 4825. — Pp. 722–735. — Series Title: Lecture Notes in Computer Science. http://link.springer.com/10.1007/978-3-540-76298-0_52.

24. Ontology-based Interpretable Machine Learning for Textual Data / Phung Lai, NhatHai Phan, Han Hu et al. // *arXiv:2004.00204 [cs, stat]*. — 2020. — . — arXiv: 2004.00204. http://arxiv.org/abs/2004.00204.

25. Improving Disentangled Text Representation Learning with Information-Theoretic Guidance / Pengyu Cheng, Martin Renqiang Min, Dinghan Shen et al. // *arXiv:2006.00693 [cs, stat]*. — 2020. — . — arXiv: 2006.00693. http://arxiv.org/abs/2006.00693.

26. *Zhang, Xiongyi*. Disentangling Representations of Text by Masking Transformers / Xiongyi Zhang, Jan-Willem van de Meent, Byron C. Wallace // *arXiv:2104.07155 [cs]*. — 2021. — . — arXiv: 2104.07155. http://arxiv.org/abs/2104.07155.

27. An Interpretability Illusion for BERT / Tolga Bolukbasi, Adam Pearce, Ann Yuan et al. // *arXiv:2104.07143 [cs]*. — 2021. — . — arXiv: 2104.07143. http://arxiv.org/abs/2104.07143.

28. CEVO: Comprehensive EVent Ontology Enhancing Cognitive Annotation / Saeedeh Shekarpour, Faisal Alshargi, Valerie Shalin et al. // *arXiv:1701.05625 [cs]*. — 2018. — . — arXiv: 1701.05625 version: 2. http://arxiv.org/abs/1701.05625.

29. *Paccanaro, Alberto*. Learning distributed representations of concepts from relational data using linear relational embedd / Alberto Paccanaro, G. Hinton // *IEEE Transactions on Knowledge and Data Engineering - TKDE*. — 2000. — .

30. *HE, Shilin*. Interpretable NLP. — 2021. — . — original-date: 2019-07-11T01:35:34Z. https://github.com/ShilinHe/interpretableNLP.

31. Learning to Deceive with Attention-Based Explanations / Danish Pruthi, Mansi Gupta, Bhuwan Dhingra et al. // *arXiv:1909.07913 [cs]*. — 2020. — . — arXiv: 1909.07913. http://arxiv.org/abs/1909.07913.

32. DeBERTa: Decoding-enhanced BERT with Disentangled Attention / Pengcheng He, Xiaodong Liu, Jianfeng Gao, Weizhu Chen // *arXiv:2006.03654*

[cs]. — 2021. — . — arXiv: 2006.03654. http://arxiv.org/abs/2006.03654.

33. *Molnar, Christoph.* Interpretable Machine Learning / Christoph Molnar. https://christophm.github.io/interpretable-ml-book/index.html.

34. Robust Semantic Interpretability: Revisiting Concept Activation Vectors / Jacob Pfau, Albert T. Young, Jerome Wei et al. // *arXiv:2104.02768 [cs, stat]*. — 2021. — . — arXiv: 2104.02768. http://arxiv.org/abs/2104.02768.

35. CausaLM: Causal Model Explanation Through Counterfactual Language Models / Amir Feder, Nadav Oved, Uri Shalit, Roi Reichart // *Computational Linguistics*. — 2021. — . — Pp. 1–54. — arXiv: 2005.13407. http://arxiv.org/abs/2005.13407.

36. How to Explain Individual Classi[U+FB01]cation Decisions / David Baehrens, Timon Schroeter, Stefan Harmeling at al. — P. 29.

37. A Survey on Neural Network Interpretability / Yu Zhang, Peter Tiňo, Aleš Leonardis, Ke Tang // *arXiv:2012.14261 [cs]*. — 2021. — . — arXiv: 2012.14261. http://arxiv.org/abs/2012.14261.

38. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications / Shaoxiong Ji, Shirui Pan, Erik Cambria et al. // *IEEE Transactions on Neural Networks and Learning Systems*. — 2021. — Pp. 1–21. https://ieeexplore.ieee.org/document/9416312/.

39. *Allen, Carl.* Interpreting Knowledge Graph Relation Representation from Word Embeddings / Carl Allen, Ivana Balažević, Timothy Hospedales // *arXiv:1909.11611 [cs, stat]*. — 2021. — . — arXiv: 1909.11611. http://arxiv.org/abs/1909.11611.

40. *Belinkov, Yonatan.* Interpretability and Analysis in Neural NLP / Yonatan Belinkov, Sebastian Gehrmann, Ellie Pavlick // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts. — Online: Association for Computational Linguistics, 2020. — . — Pp. 1–5. https://aclanthology.org/2020.acl-tutorials.1.

41. *Belinkov, Yonatan.* Analysis Methods in Neural Language Processing: A Survey / Yonatan Belinkov, James Glass // *Transactions of the Association for Computational Linguistics*. — 2019. — . — Vol. 7. — Pp. 49–72. https://direct.mit.edu/tacl/article/doi/10.1162/tacl_a_00254/43503/Analysis-Methods-in-Neural-Language-Processing-A.

42. A Survey of Methods for Explaining Black Box Models / Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri at al. // *ACM Computing Surveys*. — 2019. — . — Vol. 51, no. 5. — Pp. 1–42. https://dl.acm.org/doi/10.1145/3236009.

43. A Survey of Methods for Explaining Black Box Models / Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri at al. // *ACM Computing Surveys*. — 2019. — . — Vol. 51, no. 5. — Pp. 1–42. https://dl.acm.org/doi/10.1145/3236009.

44. *Saha, Rupsa.* Using Tsetlin Machine to discover interpretable rules in natural language processing applications / Rupsa Saha, Ole-Christoffer Granmo, Morten Goodwin // *Expert Systems.* — Vol. n/a, no. n/a. — P. e12873. — _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/exsy.12873. https://onlinelibrary.wiley.com/doi/abs/10.1111/exsy.12873.