Создание рекомендательной системы, позволяющей соединять акты с внешних источников с локальными актами компании, которые потенциально подлежат изменению

Мекан Ходжаев
Сколковский институт науки и технологии
Руководитель компании: Александра Фролова
Руководитель Сколтеха: Евгений Фролов

Ссылка в github

Аннотация

В данной работе нашей основной задачей является поиск внутренних актов компании, которые потенциально подлежат изменению, когда произошло изменение внешних актов.

Мы разделили работу на две части: первая — найти документы по сходству с помощью spacy, transformers и других моделей и сравнить с моделями, которые использовались ранее. Вторая часть среди тех документов, к которым относится потенциальное изменение, сравнивает абзац за абзацем и рекомендует именно потенциальные абзацы.

На последних этапах мы говорили о планах на будущее по использованию глубокого обучения для решения задачи классификации.

1 Введение

Чтобы понять, с чем мы имеем дело, давайте сначала немного поговорим о ТрэкТрэк.

Компания ТрэкТрэк с 2019 года специализируется на разработке систем по мониторингу и работе с регуляторными рисками для крупных компаний.

Разработанная система позволяет автоматизировать и взять под полный контроль три ключевых направления работы: мониторинг проектов НПА и изменений по ним, организацию совместной работы в рамках системы документооборота между подразделениями и с госорганами (GR, юристы), контроль за исполнением для руководства.

Наша основная задача — выявить наиболее похожие документы внутренней компании из десятков тысяч документов, к которым относится смена. Например, на рисунке 1 для Документа 1 мы рассчитаем сходство с

тремя другими документами. Как видно из рисунка, второй документ наиболее похож на данный документ (относительно выбранной меры). Чтобы сравнить два документа, нам нужны методы и меры для сравнения. Например, самая основная мера состоит в том, чтобы сравнивать слово за словом, не принимая во внимание смысл предложения. Мы подробно обсудим этот подход в разделе 5.1.

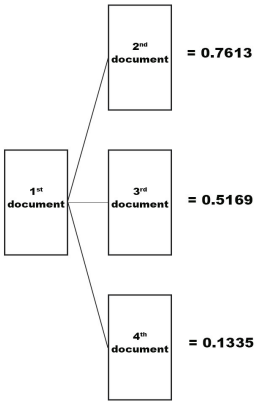

Рис. 1: Сходство документа с документом

Но главный минус такого подхода в том, что он не показывает, какие документы подходят, а какие нет (в некоторых случаях есть документы с потенциалом для изменений). То есть не существует определенного порога того, что показатель сходства выше числа подходит, а ниже — нет. Модель просто возвращает первые n совпадающих документов. Потенциальным решением этой проблемы в будущем могло бы быть обучение системы задаче классификации. Мы обсудим этот метод более подробно в разделе 7.

Оптимальное решение нашей основной задачи — сравнение документов по абзацам. В этом подходе мы делаем две вещи в одной — находим более подходящие документы и показываем абзацы потенциально изменяемым. Но чтобы найти изменения во внешних актах, нам нужно найти и выделить измененные абзацы. Мы подробно рассмотрим это в разделе 2.

Для сравнения моделей нужно учитывать точность, время вычислений, наличие GPU, т.е. докупки. [2] Мы подробно сравним эти характеристики в разделе 5 с экспериментами.

2 Сравнение документов с абзацами, чтобы найти добавленные или измененные абзацы

Для поиска различий в двух документах воспользуемся методом построения матриц. Опишем процесс подробно. Во-первых, как на рисунке 2, мы делим

документ на абзацы (на первом документе у нас есть абзацы: A, B, C, D и на втором документе A, E, C, D, F). Наша задача вывести, если мы поставили первым Документ 1, вывести Б, так как во втором нет Б. Если мы поставили первый Документ 2, то вывести абзацы E и F.

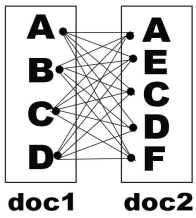

Рис. 2: Параграфы различий документа

Построим матрицу с мерами сходства (можно использовать любой метод, нам нужно только, насколько похожи эти два документа, о мерах мы поговорим в следующем разделе):

$$\begin{pmatrix} (A,A) & (A,E) & (A,C) & (A,D) & (A,F) \\ (B,A) & (B,E) & (B,C) & (B,D) & (B,F) \\ (C,A) & (C,E) & (C,C) & (C,D) & (C,F) \\ (D,A) & (D,E) & (D,C) & (D,D) & (D,F) \end{pmatrix} \rightarrow$$

$$\begin{pmatrix} 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & 1 & \cdot \end{pmatrix}$$

Мы можем видеть из матрицы в первой, третьей и четвертой строке 1, только во второй строке максимальный элемент не равен 1. Мы можем найти, что максимальный элемент этой строки не равен 1. Также, как строки, мы можем найти для столбцов. Для второго и последнего столбца матрицы максимальный элемент не равен 1. Здесь мы найдем абзац B первого документа и абзац E и F второго документа.

3 Найти похожие документы

Как мы уже говорили в первом разделе, наша главная цель — найти похожие документы для данного документа. Прежде чем представить наше решение этой проблемы, давайте обсудим предыдущие работы.

3.1 Предыдущие работы

Базовая функция, которая принимает два строковых параметра, document1 и document2, и возвращает процент сходства между ними. Функция исполь-

зует следующие шаги для вычисления процента сходства:

Во-первых, он удаляет все неалфавитно-цифровые символы из обеих строк с помощью регулярного выражения и преобразует обе строки в нижний регистр. After разбивает обе строки на отдельные слова и сохраняет их в двух отдельных наборах, а также вычисляет общее количество слов в обоих наборах вместе, беря объединение обоих наборов и получая длину результирующего набора. Затем он вычисляет количество слов, присутствующих в обоих наборах, путем пересечения обоих наборов и получения длины результирующего набора. Кроме того, он вычисляет процент похожих слов путем деления количества общих слов на общее количество слов и умножения результата на 100. Наконец, функция возвращает процент сходства.

Например, для этого текста:

What a surprise you find?

and

Our main goal is find surprise

Показатель сходства по словам составляет 22,22%.

Кроме того, давайте поговорим о преимуществах и недостатках этого подхода.

Преимущества:

Точность: сравнение документов слово в слово гарантирует, что ни одна деталь не останется незамеченной. Это позволяет с высокой точностью определять различия и сходства между документами.

Экономия времени: сравнивая документы слово за словом, вы можете быстро определить изменения, внесенные из одной версии в другую, что экономит время в процессе проверки.

Безопасность: Сравнение документов слово за словом может быть полезно для выявления любых несанкционированных или непреднамеренных изменений, внесенных в документ, что особенно важно в юридических документах или контрактах.

Недостатки:

Человеческая ошибка: даже при самом тщательном процессе проверки всегда есть вероятность человеческой ошибки. Важно перепроверить и проверить результаты, прежде чем делать какие-либо выводы.

Ограниченный объем: сравнение документов слово за словом рассматривает только точную формулировку и не принимает во внимание контекст или намерение, стоящее за словами. Это может упустить важные идеи или изменения.

Неэффективно для больших документов. Пословное сравнение может стать неэффективным для больших документов или длинных фрагментов текста, особенно если различия невелики и разбросаны по всему тексту.

4 Меры подобия

Мера сходства — это мера того, насколько похожи два объекта данных. Мера подобия — это контекст интеллектуального анализа данных или ма-

шинного обучения — это расстояние с измерениями, представляющими особенности объектов. Если расстояние небольшое, признаки имеют высокую степень сходства. Тогда как большое расстояние будет низкой степенью подобия. [3]

Меры подобия используются больше в методах предварительной обработки, связанных с текстом, а также в концепциях сходства, используемых в продвинутых методах встраивания слов. Мы можем использовать эти концепции в различных приложениях глубокого обучения. Использует разницу между изображением для проверки данных, созданных с помощью методов увеличения данных.

Сходство субъективно и сильно зависит от предметной области и приложения.

Например, два фрукта похожи по цвету, размеру или вкусу. Следует соблюдать особую осторожность при расчете расстояния между размерами/элементами, которые не связаны друг с другом. Относительные значения каждого элемента должны быть нормализованы, иначе при вычислении расстояния может оказаться доминирующим один признак.

Евклидово расстояние - это расстояние между двумя точками, равное длине соединяющего их пути. Теорема Пифагора дает это расстояние между двумя точками.

Манхэттенское расстояние – это метрика, в которой расстояние между двумя точками рассчитывается как сумма абсолютных разностей их декартовых координат. Проще говоря, это общая сумма разницы между координатами x и координатами y.

Предположим, у нас есть две точки A и B. Если мы хотим найти манхэттенское расстояние между ними, нам достаточно суммировать абсолютное отклонение по оси x и по оси y. Это означает, что мы должны найти, как эти две точки A и B изменяются по оси X и оси Y. Говоря более математическим языком, манхэттенское расстояние между двумя точками измеряется вдоль осей под прямым углом.

Расстояние Минковского – это обобщенная метрическая форма евклидова расстояния и манхэттенского расстояния. В уравнении d^{MKD} — расстояние Минковского между записью данных i и j, k — индекс переменной, n — общее количество переменных y, а λ — порядок метрики Минковского. Хотя он определен для любого $\lambda > 0$, он редко используется для значений, отличных от 1, 2 и ∞.

Способ измерения расстояний метрикой Минковского разных порядков между двумя объектами с тремя переменными (На изображении отображается в системе координат с осями x, y, z).

Подобие косинуса находит нормализованное скалярное произведение двух атрибутов. Определив сходство косинусов, мы попытаемся найти косинус угла между двумя объектами. Косинус 0° равен 1, а для любого другого угла он меньше 1. [1]

Таким образом, это суждение об ориентации, а не о величине. Два вектора с одинаковой ориентацией имеют косинусное сходство, равное 1, два вектора под углом 90° имеют сходство, равное 0. В то время как два диа-

метрально противоположных вектора имеют сходство -1, независимо от их величины.

Косинусное подобие особенно используется в положительном пространстве, где результат четко ограничен в [0,1]. Одна из причин популярности косинусного сходства заключается в том, что его очень эффективно оценивать, особенно для разреженных векторов.

Сходство Жаккара. До сих пор обсуждались некоторые метрики для нахождения сходства между объектами, где объекты являются точками или векторами. Когда мы рассматриваем подобие Жаккара, эти объекты будут множествами.

5 Эксперименты

5.1 Сходство doc2doc.

Сходство документов является важной проблемой обработки естественного языка и применяется в различных областях, таких как кластеризация документов, поиск информации и рекомендательные системы. В последние годы подходы, основанные на глубоком обучении, показали многообещающие результаты в задачах на сходство документов. В этом отчете мы исследуем использование модели "ru_core_news_sm"в spaCy для вычисления схожести документов в русскоязычных текстах.

Мы начинаем с предварительной обработки текста, используя функции токенизатора spaCy, лемматизатора и удаления стоп-слов. Затем мы представляем каждый документ в виде вектора, используя модель встраивания Word2Vec, которая фиксирует семантические отношения между словами. Чтобы вычислить сходство между двумя документами, мы используем косинусное сходство между их векторными представлениями. Также мы можем использовать другие показатели, о которых говорилось ранее. Мы оцениваем эффективность нашего метода, используя два набора данных: корпус РИА Новости и корпус русской Википедии.

Эксперименты показывают, что модель "ru_core_news_sm"с вложениями Word2Vec достигает высокой точности в задачах подобия документов, превосходя традиционные методы, такие как взвешивание TF-IDF и латентный семантический анализ. Кроме того, этот метод позволяет выявить семантическое сходство между документами, даже если в них не так уж много точных совпадений слов. Мы демонстрируем применимость нашего метода на двух примерах использования: кластеризация документов и поиск похожих статей в новостном корпусе.

Также мы попробуем использовать модель sber model - мощный и эффективный инструмент для вычисления сходства документов в русскоязычных текстах. Наш метод может быть легко распространен на другие языки, поддерживаемые библиотекой SentenceTransformer, и может применяться в различных реальных задачах, требующих анализа сходства документов. Мы считаем, что модель SentenceTransformer обладает большим потенциа-

лом для улучшения нашего понимания документов на естественном языке и может привести к новым прорывам в обработке и анализе документов.

Другие эксперименты показывают, что модель sber достигает высокой точности в задачах сходства документов, превосходя традиционные методы, такие как взвешивание TF-IDF и латентный семантический анализ. Этот метод позволяет выявить семантическое сходство между документами, даже если в них не так уж много точных совпадений слов.

Обе модели SentenceTransformer, 'sentence-transformers/all-roberta-large-v1' и 'ai-forever/sbert_large_nlu_ru', являются предварительно обученными моделями для создания высококачественных вложений предложений, но они были обучены на разных корпусах и для разных языков.

«sentence-transformers/all-roberta-large-v1» — это предварительно обученная модель, основанная на архитектуре RoBERTa, которая была обучена на большом наборе данных текста на английском языке. Он состоит из 1,2 миллиона предложений из различных источников, включая Википедию, новостные статьи и книги. Было показано, что эта модель производит высококачественные вложения, которые собирают детализированную семантическую информацию, и достигла современной производительности на ряде эталонных наборов данных.

С другой стороны, ai-forever/sbert_large_nlu_ru — это предварительно обученная модель, разработанная специально для русскоязычного текста. Он был обучен на большом русскоязычном корпусе и предназначен для создания высококачественных эмбеддингов для русскоязычных предложений. Он использует сиамскую архитектуру, которая обучена сочетанию задач вывода на естественном языке, обнаружения парафраз и кластеризации. Также было показано, что эта модель производит высококачественные вложения и достигла современной производительности на ряде эталонных наборов данных на русском языке.

Как правило, выбор используемой модели зависит от типа текстовых данных, с которыми вы работаете. Если вы работаете с англоязычным текстом, то вам может подойти 'sentence-transformers/all-roberta-large-v1', а если вы работаете с русскоязычным текстом, то ai-forever/sbert_large_nlu_ru' будет лучшим вариантом. В конечном счете, обе модели очень эффективны для создания высококачественных вложений предложений, которые можно использовать для широкого круга задач НЛП.

5.2 Разделить документы на абзацы и найти наиболее похожие абзацы

Здесь воспользуемся методом, как в разделе 2. Разобьем документы на части по абзацам (если нужно найти похожие предложения, можно разделить по предложениям). Если в первом документе нам дано несколько абзацев, мы можем выбрать и построить матрицу размером **количество заданных абзацев первого документа** $\times$ **количество абзацев второго документа.** И мы можем найти для каждого заданного абзаца самые крутые топ-3

или топ-5 абзацев второго документа. После этого строим матрицу как на рисунке 3

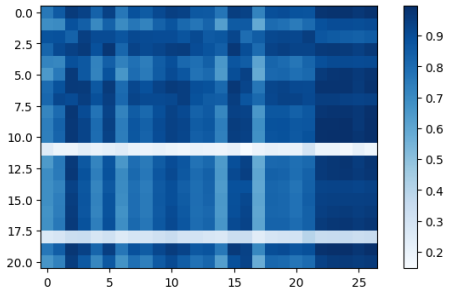

Рис. 3: Сходство документа с разделением на абзацы

Как видно из рисунка выше, в первом документе 21 абзац, а во втором — 27 абзацев. В первом документе 1, 3, 4, 6, 7 абзацы наиболее важны по отношению ко второму документу, а во втором документе 3-й и последние пять абзацев наиболее важны по сравнению с абзацами первого документа.

6 Результаты и обсуждение

Мы пришли к выводу, что модель "ru_core_news_sm"в spaCy с вложениями Word2Vec является мощным и эффективным инструментом для вычисления сходства документов в русскоязычных текстах. Наш метод может быть легко распространен на другие языки, поддерживаемые spaCy, и может применяться в различных реальных задачах, требующих анализа сходства документов.

Кроме того, модель SentenceTransformer('ai-forever/sbert_large_nlu_ru') является мощным и эффективным инструментом для вычисления сходства документов в русскоязычных текстах. Наш метод может быть легко распространен на другие языки, поддерживаемые библиотекой SentenceTransformer, и может применяться в различных реальных задачах, требующих анализа сходства документов. Мы считаем, что модель SentenceTransformer обладает большим потенциалом для улучшения нашего понимания документов на естественном языке и может привести к новым прорывам в обработке и анализе документов.

Причина, по которой использование модели sber (русская) для вычисления схожести предложений на русском языке может привести к низкой производительности по сравнению с первым переводом текста на английский язык, а затем с использованием all-roberta-large (английский), вероятно, связана с разницей в качестве двух моделей. all-roberta-large был предварительно обучен на большом корпусе текстов на английском языке и обычно считается высококачественной языковой моделью. С другой стороны, модель sber была предварительно обучена на небольшом массиве текстов на

Paragraphs	Model	Accuracy	Computation Time	Additional Purchases
10	Spacy	77%	3.4s	None
10	SentenceTransformer	92%	8.2s	yes
10	Word-by-word	63%	0.2s	None
100	Spacy	75%	33.7s	None
100	SentenceTransformer	91%	64.5s	yes
100	Word-by-word	56%	2.0s	None
1000	Spacy	70%	5m 37s	None
1000	SentenceTransformer	90%	10m 58s	yes
1000	Word-by-word	45%	20.1s	None

Таблица 1: Сравнение моделей подобия текста в задаче doc2doc

русском языке и может не иметь такого же уровня точности или надежности. Кроме того, стоит отметить, что переводы могут содержать собственные ошибки, поэтому обычно лучше работать с текстом на языке оригинала, если это возможно. Результаты экспериментов мы можем увидеть в таблице 1

7 Будущие эксперименты

7.1 Глубокое обучение для задачи классификации

Как мы уже говорили в разделе 1 общей проблемы наших подходов нет порога, после которого количество документов не имеет значения. Здесь нашим будущим решением будет глубокое обучение. Для этой задачи классификации нам нужна модель обучения более чем 100 000 документов.

Глубокое обучение — это популярный подход к задачам классификации текста, который в последние годы показал большой успех. Для этой задачи обычно используются модели глубокого обучения, такие как сверточные нейронные сети (CNN), рекуррентные нейронные сети (RNN) и модели на основе преобразователя, такие как BERT и GPT.

Одним из преимуществ моделей глубокого обучения для классификации текста является то, что они способны выявлять сложные шаблоны в текстовых данных и могут научиться распознавать тонкие связи между словами и фразами. Это особенно полезно для таких задач, как анализ тональности, где классификация может зависеть от комбинации различных признаков во входном тексте.

Однако модели глубокого обучения могут быть сложными и дорогостоящими в вычислительном отношении для обучения и могут требовать больших объемов размеченных данных для достижения хорошей производительности. Таким образом, альтернативные методы, такие как классификаторы на основе ядра и традиционные алгоритмы машинного обучения, такие как Наивный Байес и SVM, по-прежнему широко используются для задач классификации текста.

7.2 Генерация аннотаций

Как мы уже говорили, больше документов содержит более 1000 абзацев и для использования наших подходов требуется минимум 10-20 минут. Например, для вычисления сходства документов для 10000 абзацев потребуются дни. Кроме того, в тексте не вся информация нужна. Для этого нам нужно извлечь только общую информацию - создать аннотацию или реферат. Мы можем сделать это, используя метод, который мы использовали в разделе 2. Мы можем сравнить все абзацы этого документа с другими абзацами. И извлечь информацию, но общую информацию об изменениях документа относительно второго документа. В будущем аннотация, генерирующая экстракт, решит проблему вычислительного времени и повысит эффективность документа. [4]

8 Заключение

В данной работе перед нами стояла задача найти подобные документы. Мы разделили эту задачу на две части: сравнить по всем документам и сравнить по абзацам. Но сравнивать по абзацам, это все равно решает нашу проблему, какие абзацы нужно изменить или добавить. У нас также был подход сравнения аннотаций, но аннотации должны были создаваться с той же философией. Общий вывод заключается в том, что у каждого подхода есть свои плюсы и минусы, но сравнение по абзацам с пространственной моделью является оптимальным решением задачи со сравнением по абзацам.

Список литературы

[1] Zhang Bingyu and Nikolay Arefyev. The document vectors using cosine similarity revisited. In *Proceedings of the Third Workshop on Insights from Negative Results in NLP*. Association for Computational Linguistics, 2022.

[2] Nicholas Gahman and Vinayak Elangovan. A comparison of document similarity algorithms, 2023.

[3] Jiapeng Wang and Yihong Dong. Measurement of text similarity: A survey. *Information*, 11(9), 2020.

[4] Lu Wang and Wang Ling. Neural network-based abstract generation for opinions and arguments, 2016.