Исследование подходов для оценки семантического сходства текстов в задаче сопоставления вакансий и резюме

A Preprint

Петрова Александра Сергеевна
Московский государственный университет имени М. В. Ломоносова
Научный руководитель: Майсурадзе Арчил Ивериевич
Научный консультант: Колосов Алексей Михайлович

Abstract

Задача сопоставления вакансий и резюме связана с необходимостью работодателей отбирать кандидатов на основе большого количества резюме. Эта задача стала особенно актуальной с развитием онлайн-платформ для поиска работы, где количество резюме может быть значительным. В данной статье рассматривается метод сопоставления вакансий и резюме с использованием векторных представлений, полученных на основе различных моделей архитектуры Transformer, и предлагается метод повышения качества сопоставления.

Keywords Job–candidate matching · рекомендательные модели · предобученные NLP модели · embedding · cosine similarity

1 Введение

Современные технологии электронного рекрутинга привели к росту количества вакансий и заявок, что создало необходимость в эффективных системах рекомендаций. Семантические технологии доказали свою пользу, автоматизируя обработку документов и улучшая подбор кандидатов [5].

Интеграция ИИ значительно продвинула рекрутинг, позволяя системам быстрее и точнее обрабатывать резюме и оценивать кандидатов. Однако такие системы часто зависят от исторических данных, и их точность снижается в случае новых вакансий или навыков. Здесь на помощь приходит Zero-Shot Learning [1]: эта методика помогает адаптировать систему под новые вакансии и профили кандидатов, повышая эффективность найма без переобучения. Основой для её работы являются предварительно обученные языковые модели на архитектуре Transformer, что обеспечивает способность к генерации и пониманию человеческого текста.

В упомянутой ранее статье [1] ранжирование происходило по векторной близости вакансий и резюме. Для данной задачи использовался набор моделей Sentence Transformers, которые обладают хорошей производительностью для создания векторных представлений предложений, необходимых для сопоставления вакансий и резюме.

Sentence Transformers обычно создают представления на уровне предложений, что может приводить к усреднению информации и, как следствие, к потере важных деталей, специфичных для вакансий и резюме. Оригинальный Transformer, наоборот, анализирует текст более детально, на уровне токенов, что может улучшить качество сопоставления.

В данной статье также рассматривается подход к ранжированию по векторной близости вакансий и резюме. Для создания векторных представлений выбраны несколько моделей из семейства BERT и GPT, способные работать с текстами на русском языке. Для оценки близости вакансий и резюме используется косинусное сходство векторов. В экспериментах задействованы два набора данных: синтетический для обучения и экспертно размеченные данные для расчета метрики качества рекомендаций.

2 Постановка задачи

Задача сопоставления вакансии и резюме представляется как задача ранжирования элементов множества, где элементами выступают резюме, а запросом — вакансия.

В этом исследовании мы ограничиваемся предположением, что и вакансии, и резюме представляют собой текстовые данные. Таким образом, задача сопоставления вакансии и резюме сводится к ранжированию текстов в ответ на текстовый запрос.

Входные данные:

- Множество вакансий $V = \{v_1, v_2, \ldots, v_n\}$, где каждая вакансия v_i описана текстом $T(v_i)$.
- Множество резюме $R = \{r_1, r_2, \ldots, r_m\}$, где каждое резюме r_j описано текстом $T(r_j)$.
- Множество релевантных пар $P = \{(v_i, r_j)\}$, где v_i и r_j представляют собой корректно сопоставленную пару вакансии и резюме.

Необходимо разработать функцию семантического сходства $S(v_i, r_j)$, которая для каждой пары (v_i, r_j) из $V \times R$ вычисляет значение $S(v_i, r_j)$, отражающее степень соответствия текста вакансии $T(v_i)$ и текста резюме $T(r_j)$.

На основе функции сходства $S(v_i, r_j)$ формируется ранжированный список резюме $Rank(v_i) = \{r_{j_1}, r_{j_2}, \ldots, r_{j_m}\}$ для каждой вакансии v_i, отсортированный по убыванию значения $S(v_i, r_j)$.

3 Методология

Для оценки схожести текстов вакансий и резюме предлагается строить их векторные представления и использовать cosine similarity в качестве метрики близости. В качестве основы для векторизации используются модели из семейств BERT и GPT.

3.1 Токенизация

Перед векторизацией текстов вакансий и резюме необходимо разделить их на отдельные единицы, или токены, которые будут являться входными данными для языковых моделей. Токенизация проводится с использованием встроенных механизмов моделей BERT и GPT, которые используют подслова (subword units) для представления как часто встречающихся, так и редких слов. Это позволяет избежать проблемы "неизвестных слов"(out-of-vocabulary), которые могут возникнуть при обработке текстов, содержащих специализированные термины и аббревиатуры, характерные для резюме и описаний вакансий.

3.2 Векторизация

После токенизации тексты преобразуются в векторные представления с использованием моделей BERT и GPT. Для этого применяется метод получения эмбеддингов: модели обрабатывают токены и преобразуют их в числовые вектора, отражающие семантическую информацию. Итоговое векторное представление текста формируется путём агрегации векторов токенов с использованием pooling-методов (mean pooling или CLS pooling) для BERT. В случае с GPT векторизация проводится на основе использования скрытых слоёв модели. Эти представления фиксированной размерности будут использоваться на следующих этапах для вычисления метрики схожести.

3.3 Ранжирование

После того как тексты вакансий и резюме преобразованы в векторные представления, для каждого резюме рассчитывается значение cosine similarity по отношению к каждой вакансии. Результатом является ранжированный список резюме, отсортированный по убыванию значения метрики сходства.

3.4 Оценка качества

Для оценки качества сопоставления вакансий и резюме предлагается использовать стандартную метрику для задачи ранжирования - mean average precision at K.

Для того чтобы дать определение mean average precision at K (MAP@K), вводятся следующие понятия:

- Precision at K (P@K)
- Average precision at K (AP@K)

Определение. Допустим, алгоритм ранжирования выдал ранжированный список L_v^K длины K объектов $r \in R$ для элемента $v \in V$. Тогда precision at K (P@K) - это доля релевантных объектов $r \in R$ в списке L_v^K:

$P_v@K = \frac{|L_v^K \cap R_v|}{K}$

Precision at K — базовая метрика качества ранжирования для одного объекта. Она имеет важный недостаток — она не учитывает порядок элементов в «топе».

Этот недостаток нивелирует метрика ранжирования average precision at K (AP@K).

Определение. Допустим, алгоритм ранжирования выдал ранжированный список L_v^K длины K объектов $r \in R$ для элемента $v \in V$. Тогда precision at K (P@K) - это величина, равная сумме P@k по индексам k от 1 до K только для релевантных элементов, деленной на мощность множества R_v:

$AP_v@K = \frac{1}{|R_v|} \sum_{k=1}^{K} 1\left[L_v^K[k] \in R_v\right] P_v@k$

В average precision at K качество ранжирования оценивается для отдельно взятого объекта. Идея mean average precision at K (MAP@K) заключается в том, чтобы посчитать AP@K для каждого объекта и усреднить.

Определение. Mean average precision at K - это усредненная по всем объектам AP@K:

$MAP@K = \frac{1}{N} \sum_{v=1}^{N} AP_v@K$

Замечание. Идея усреднения логична, если все объекты одинаково важны. В случае если это не так, вместо простого усреднения можно использовать взвешенную сумму, домножив AP@K каждого объекта на вес, соответствующий его важности. В данной работе будет принято предположение, что все вакансии имеют одинаковый вес.

4 Эксперименты

4.1 Описание данных

Данные включают набор вакансий и резюме на русском языке, предоставленный HR-отделом компании ACD/Labs. Большинство вакансий и резюме относятся к сфере IT.

Для каждой вакансии известен перечень резюме кандидатов, приглашённых на собеседование. В выборке отсутствуют резюме, не связанные с конкретными вакансиями, и каждое резюме привязано к единственной вакансии. В Таблице 1 представлено распределение вакансий по числу соответствующих им резюме.

Количество резюме	Количество вакансий	Описание
1	1	1 вакансия - 1 резюме
2	2	1 вакансия - 2 резюме
3	1	1 вакансия - 3 резюме
4	2	1 вакансия - 4 резюме
5	1	1 вакансия - 5 резюме
8	1	1 вакансия - 8 резюме
9	1	1 вакансия - 9 резюме
11	2	1 вакансия - 11 резюме
13	2	1 вакансия - 13 резюме
17	1	1 вакансия - 17 резюме
90	13	Total

Таблица 1: Распределение вакансий по количеству резюме

4.2 Предварительно обученные модели NLP

Модели NLP, обученные на обширных и разнообразных наборах данных, содержат в себе широкое понимание естественного языка. Суть эксперимента заключается в использовании этих моделей для сопоставления описаний вакансий и резюме без предварительного обучения этой конкретной задачи.

4.3 Архитектура Transformer

В основе предварительно обученных моделей лежат архитектуры на основе Transformer, которые произвели революцию в обработке естественного языка. Такие модели, как BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pretrained Transformer) и их производные продемонстрировали исключительную способность понимать и генерировать человекоподобный текст. Их способность улавливать контекстуальные связи между словами в предложении привела к значительному прогрессу в решении различных задач NLP.

4.4 Обоснование выбора моделей

Для эксперимента были выбраны следующие модели, способные генерировать высококачественные эмбеддинги текстов на русском языке и учитывать контекстуальные взаимосвязи:

- BERT multilingual: Эта версия модели BERT обучена на множестве языков, включая русский, что позволяет применять её в мультиязычной среде и обрабатывать тексты с элементами иностранных языков. Её способность к извлечению контекстных связей делает её полезной для задач сопоставления на многих языках, что особенно важно при анализе резюме и вакансий с интернациональными терминами.
- BERT Russian: Эта версия BERT специализирована на русском языке и обучена на корпусах, включающих российские тексты. Она способна учитывать специфику русского синтаксиса и грамматики, что делает её подходящей для текстов, в которых присутствуют специфичные для русского языка выражения и лексические особенности.
- DeepPavlov RuBERT: Разработанная на базе BERT для русского языка, модель DeepPavlov RuBERT обучена на большом массиве текстов, включая диалоги, новости и технические статьи. Она хорошо подходит для понимания формального и неформального языка, что важно при сопоставлении описаний вакансий и резюме, где встречаются и формальные, и разговорные выражения.
- RuGPT2 Large: Эта модель, основанная на архитектуре GPT-2 и адаптированная для русского языка, обучена на крупном корпусе русскоязычных текстов. RuGPT2 способна генерировать качественные эмбеддинги и может быть полезна для анализа более длинных текстов, таких как полные описания вакансий и резюме. Её способность к генерации текста также может быть полезна для создания резюмирующих описаний или дополнения информации.
- RuGPT3 Large: Модель RuGPT3 представлена как более мощная версия GPT-3, обученная на русском языке. Она улавливает сложные семантические связи и контекстуальные особенности текста, что особенно полезно для анализа вакансий и резюме с развернутыми требованиями и сложной терминологией. RuGPT3 обеспечивает глубокое понимание текста, что позволяет создавать более точные векторные представления.
- text-embedding-ada-002: Модель от OpenAI, предоставляющая эффективные эмбеддинги для текстов на разных языках, включая русский. Text-embedding-ada-002 создана для извлечения обобщённых текстовых представлений, что делает её подходящей для задач классификации и сопоставления. Её способность к генерации высококачественных эмбеддингов особенно ценна для задач ранжирования и сравнения текстов.

Использование нескольких моделей позволяет провести сравнительный анализ их эффективности в задаче сопоставления вакансий и резюме и выбрать наиболее подходящую. Основные критерии выбора включают точность создания эмбеддингов, адаптацию к русскоязычным текстам, способность улавливать контексты и учитывать терминологические особенности описания вакансий и резюме.

5 Выводы

Задача сопоставления вакансии и резюме была формализована как задача ранжирования всех элементов множества, где элементами являются резюме, а запросом — вакансия.

В исследовании мы ограничились предположением, что и вакансии, и резюме — это тексты. Соответственно, задача сопоставления вакансии и резюме была сведена к задаче ранжирования текстов по текстовому запросу.

Для оценки качества сопоставления вакансий и резюме использовались стандартная метрика для задачи ранжирования: MAP@K. MAP@K дает представление о средней эффективности ранжирования по всей выборке и позволяет учитывать общее качество списка предложенных результатов.

В работе рассматривался метод ранжирования текстов на основе семантического сходства. В работе исследовались подходы для оценки семантического сходства между текстами вакансий и резюме. Ранжирование происходило на основе полученных оценок сходства (или релевантности): резюме сортировались в порядке убывания их релевантности относительно вакансии.

Обзор литературы показал, что все лучшие по качеству ранжирования современные алгоритмы сначала строят векторные представления объектов. Было предложено использовать языковые модели на архитектуре Transformer для получения векторных представлений вакансий и резюме в общем семантическом пространстве, где косинусное сходство между векторами использовалось для измерения семантической близости между запросом и документом.

Наилучшее качество на экспертно размеченных данных показала модель text-embedding-ada-002.

Список литературы

[1] Jarosław Kurek, Tomasz Latkowski, Michał Bukowski, Bartosz Świderski, Mateusz Łępicki, Grzegorz Baranik, Bogusz Nowak, Robert Zakowicz, Łukasz Dobrakowski Zero-Shot Recommendation AI Models for Efficient Job–Candidate Matching in Recruitment Process // MDPI. - 2024

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin Attention Is All You Need // 2017

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // 2018

[4] Gokul Yenduri, Ramalingam M, Chemmalar Selvi G, Supriya Y, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, Deepti Raj G, Rutvij H Jhaveri, Prabadevi B, Weizheng Wang, Athanasios V. Vasilakos, Thippa Reddy Gadekallu Generative Pre-trained Transformer: A Comprehensive Review on Enabling Technologies, Potential Applications, Emerging Challenges, and Future Directions // 2023

[5] Brek, A.; Boufaïda, Z. Semantic Approaches Survey for Job Recommender Systems. DBLP 2022, 1, 1–10