

*Пискаев Максим Сергеевич, 1 курс магистратуры, Университет ИТМО, институт прикладных компьютерных наук (ИПКН)*

## **АВТОМАТИЗИРОВАННЫЙ ПАЙПЛАЙН СКОРИНГА РЕЗЮМЕ НА ОСНОВЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ В СИСТЕМЕ АВТОМАТИЗАЦИИ ПРОЦЕССА РЕКРУТИНГА**

**Аннотация.** В работе представлен автоматизированный пайплайн скоринга резюме с использованием больших языковых моделей (LLM) с открытым исходным кодом (open-source), позволяющий быстро и объективно оценивать кандидатов по их навыкам и опыту. Описаны этапы обработки резюме и формирования итогового отчёта в системе автоматизации процесса рекрутинга, интегрирующей LLM с другими модулями. Экспериментальная оценка показала высокое качество: точность парсинга навыков достигает ~91%, извлечения опыта — ~88%, определение соответствия кандидата требованиям вакансии — ~85%. Пайплайн обрабатывает ~100 резюме за 2,5 минуты, что существенно ускоряет работу рекрутеров и снижает долю нерелевантных интервью примерно на 35%, экономя до 70% рабочего времени HR-специалистов без потери качества отбора.

**Введение.** Процесс отбора персонала требует значительных трудозатрат: по оценкам, до 60% рабочего времени HR-специалистов уходит на ручной анализ резюме и сопоставление их с требованиями вакансий. В крупных компаниях это затрудняет оперативный и объективный отбор, поскольку число получаемых откликов достигает тысяч. Современные большие языковые модели (LLM) позволяют интеллектуально автоматизировать первичный скрининг кандидатов [1]. Например, система Google Hire использует модель на базе BERT для автоматического парсинга и сопоставления резюме с вакансиями, что

повысило точность отбора. Вместе с тем внедрение LLM требует решения таких задач, как контроль качества, прозрачность решений и достаточная обучающая база данных.

В данной работе представлена автоматизированная система анализа резюме на основе LLM, предназначенная для быстрого и объективного сопоставления навыков кандидатов с требованиями вакансий. Особый акцент сделан на использовании open-source технологий, обеспечивающих воспроизводимость и безопасность данных. Представлены результаты экспериментальной оценки качества и производительности системы, обсуждаются её преимущества, ограничения и перспективы дальнейшего развития.

**Обзор существующих решений и исследований.** Современные исследования в области автоматизации анализа резюме подчёркивают ключевую роль точного извлечения навыков кандидатов из неструктурированного текста. Подход Wings и соавторов показывает, что использование n-грамм для учёта контекста позволяет гораздо точнее различать «жёсткие» технические навыки и «мягкие» soft skills, что заметно повышает качество классификации при разборе секции «Навыки» в резюме [4]. Вместе с тем крупномасштабные фреймворки, предложенные Gan et al., демонстрируют, что автоматизация суммаризации и оценки каждой анкеты сразу в несколько этапов позволяет ускорить процесс первичного отбора примерно в 11 раз по сравнению с ручным скринингом, при этом достигается высокая F1-мера (87,7%) после дообучения модели на специализированных данных [2].

Параллельно активно развивается идея мультиагентных систем на базе больших языковых моделей. Lo et al. описали архитектуру, в которой разные агенты — экстрактор, оценщик, суммаризатор и форматировщик —

взаимодействуют по цепочке, а механизм retrieval-augmented generation (RAG) позволяет подгружать внешние знания (например, отраслевые справочники или рейтинги вузов) для более контекстного принятия решений. Такая модульность даёт гибкость: при необходимости достаточно доработать одного агента, не перестраивая полностью весь пайплайн [3].

Важным аспектом соответствия резюме требованиям вакансии является нормализация терминологии и учёт разнообразия наименований навыков. Senger et al. в своём обзоре отмечают проблему «длинного хвоста» редко встречающихся навыков и предлагают использовать таксономии для приведения терминов к единому стандарту, что значительно упрощает автоматическое сопоставление профиля кандидата с эталонным описанием вакансии [5]. Для оценки soft skills исследователи Clavel et al. предложили анализировать текстовые ответы кандидатов на открытые вопросы и фрагменты переписки с рекрутером: лингвистические признаки коммуникаций позволяют выявить такие качества, как лидерство или адаптивность, даже если они не явно перечислены в резюме [6].

Наконец, вопросы объяснимости и справедливости решений моделей в рекрутинге становятся всё более актуальными. Shen et al. подчёркивают необходимость применения методов ХАИ — от визуализации attention-скоров ключевых фраз до генерации пояснительных текстов о том, почему кандидат получил тот или иной скор. Кроме того, важно регулярно проверять систему на наличие гендерных и других предубеждений, чтобы гарантировать объективность автоматизированного отбора [7].

**Архитектура системы.** Архитектура решения выстроена по микросервисному принципу, где ключевым звеном выступает связка LangChain и Milvus. После загрузки резюме фронтендом FastAPI сохраняет исходный файл в MinIO и ставит задачу в очередь RabbitMQ. Воркер

последовательно вызывает парсер, структурирующий текст резюме в JSON, затем через LangChain передаёт извлечённые факты и описание вакансии LLM для оценки и формирования отчёта. При этом LangChain при необходимости обращается к Milvus — векторному хранилищу эмбедингов резюме и вакансий — чтобы подгрузить дополнительные семантические данные и уточнить синонимию навыков или опытных терминов.

Благодаря асинхронной очереди задач и независимым воркерам масштабирование достигается горизонтально: увеличение числа воркеров позволяет параллельно обрабатывать больше резюме без изменений в микросервисах. Milvus и MinIO масштабируются отдельно, обеспечивая быстрый семантический поиск и хранение артефактов. Такая модульность также упрощает замену компонентов: можно подключить новую LLM-модель или альтернативный движок для векторного поиска, не затрагивая остальные части системы.

**Бенчмарк и выбор языковой модели.** В рамках выбора оптимальной LLM для русскоязычного скоринга резюме мы провели сравнительный бенчмарк четырёх открытых моделей, подходящих для развёртывания на 24 GB GPU. Для тестирования собран датасет из 100 пар «описание вакансии—резюме кандидата» с экспертной разметкой соответствия. Качество отчётов, сгенерированных моделями, оценивалось по классическим метрикам автоматического сравнения (BLEU, ROUGE, METEOR) и семантическим метрикам (BERTScore, BARTScore), а также экспертной оценкой связности и полноты русскоязычных текстов.

По совокупному интегральному баллу T-Lite 0.1 вышла в лидеры (0,477 условных единиц), опередив Llama3 8B (0,471), Saiga-Llama3-8B ( $\approx 0,460$ ) и Mistral 7B

(0,450). Особенно заметным было превосходство T-Lite в точности донесения ключевых деталей резюме и грамотности формулировок.

Кроме того, мы применили в LangChain техники prompt engineering и структурированного вывода: шаблоны промптов задают модельному ответу чёткую схему с заголовками разделов или даже JSON-структурой, а встроенная верификация через Pydantic проверяет соответствие формата. Такая организация работы минимизирует риски «галлюцинаций» и облегчает парсинг ответов. Более того, благодаря поддержке retrieval-augmented generation LLM может при оценке кандидата дополнять свой контекст актуальными данными из внешних источников (например, уточнять значения редких сертификатов через запросы в Milvus), что повышает обоснованность и точность финального вывода.

**Пайплайн скоринга резюме.** Первым этапом работы системы является приём и регистрация исходных данных. Через веб-интерфейс рекрутер загружает файл резюме кандидата в формате PDF или DOCX и вводит текст описания вакансии, либо выбирает его из заранее подготовленных шаблонов. Загруженный документ сохраняется в объектном хранилище MinIO, а описание вакансии и ссылка на файл заносятся в очередь задач RabbitMQ, что позволяет асинхронно обрабатывать запрос без задержки пользовательского интерфейса.

На втором этапе запускается модуль парсинга резюме, который преобразует бинарный файл в чистый текст и разбивает его на логические секции. Сначала документ конвертируется из PDF или DOCX в строку, после чего выполняется сегментация по заголовкам и шаблонам (например, «Образование» или «Опыт работы»). В каждой секции применяются специализированные методы: для извлечения организаций и дат — модель

NER, для выделения навыков — классификатор на базе словаря и дополнительных правил. Результатом работы парсера становится структурированный JSON-документ с полями: ФИО, контакты, список учебных заведений с годами обучения, перечень мест работы с должностью и периодом, набор выявленных hard- и soft-skills, а также дополнительные сведения — сертификаты и языки.

Параллельно обрабатывается само описание вакансии. Либо по заранее заданным шаблонам и ключевым словам («Требования:», «Будет плюсом:»), либо с помощью LLM, которой формулируется задание «выдели из текста вакансии ключевые требования», система извлекает список обязательных и желательных критериев. Все названия навыков приводятся к единому виду (например, «C#» и «C Sharp» считаются эквивалентными), числовые значения опыта — к целочисленному формату, а уровни образования нормализуются в категории «бакалавр», «магистр» и т. д. Таким образом формируется эталонный профиль вакансии для последующего сравнения.

На следующем этапе происходит сопоставление профиля кандидата с эталонным описанием вакансии. Система проверяет, какие обязательные навыки из вакансии присутствуют в списке кандидата: названия сравниваются напрямую и с учётом синонимии, а в сложных случаях вычисляется семантическое сходство через эмбединги и косинусную меру. Одновременно анализируется стаж: извлечённые из резюме периоды работы агрегируются по соответствующим отраслям и сравниваются с требуемыми годами опыта. Образование и сертификаты проверяются по наличию ключевых наименований в структурированном JSON. В результате формируется набор промежуточных показателей: процент совпадения must-have, процент совпадения nice-to-have, бинарные или шкальные оценки опыта и образования.

Агрегация промежуточных показателей в итоговый скор выполняется по разработанной шкале от 0 до 35 баллов, где до 25 баллов отводится на обязательные навыки (примерно 70 % веса), и до 10 баллов — на желательные (30 %). Если кандидат обладает всеми обязательными навыками, он получает полный «технический» балл, иначе — пропорционально количеству найденных. Аналогично начисляются баллы за дополнительные навыки. При необходимости итоговый скор может корректироваться на  $\pm 1-2$  балла в зависимости от избыточного или слегка недостающего стажа, но основная доля оценки всегда формируется исходя из навыкового соответствия.

Финальным этапом является генерация текстового отчёта с помощью LLM. Модели передаются структурированные данные: синтезированное описание вакансии, распознанный профиль кандидата и рассчитанные числовые показатели. По заранее заданному шаблону LLM формирует отчёт, включающий сводную оценку («итоговый балл и уровень соответствия»), перечисление присутствующих и отсутствующих навыков, комментарии по стажу и образованию, а также рекомендацию по приглашению на интервью. Такой отчёт сочетает количественные метрики с качественным описанием сильных и слабых сторон кандидата, что облегчает принятие решения рекрутером.

После генерации отчёт сохраняется в базе (JSON и при желании PDF), а сам скор попадает в кеш Redis для быстрого доступа. В веб-интерфейсе Streamlit рекрутер видит итоговый балл, связный текст отчёта и таблицу деталей по каждому критерию. При пакетной загрузке резюме система ранжирует всех обработанных кандидатов по убыванию баллов и выводит топ-N наиболее подходящих, что позволяет за несколько секунд сформировать короткий список для дальнейшего интервью.

**Оценка эффективности. Оценка качества алгоритмов.** Для валидации точности парсинга резюме мы собрали тестовый набор документов с ручной разметкой ключевых сущностей и оценили, как часто модуль корректно извлекает навыки кандидата. Средняя точность составила около 91 %, при этом, например, извлечение ФИО и контактов достигало почти 99 %, а корректное выделение опыта и образования — около 90 %, тогда как выделение soft skills показало более низкий показатель. Это демонстрирует, что наш парсер сопоставим по качеству с передовыми решениями на рынке. Аналогичным образом мы проверили точность оценки опыта: в 88 % случаев автоматическое заключение о соответствии стажа требованиям вакансии совпадало с мнением экспертов. Оставшиеся 12 % ошибок связаны преимущественно с нестандартным описанием обязанностей или нетипичными карьерными траекториями, которые сложны для автоматической интерпретации. Наконец, для оценки общего соответствия кандидатов мы сопоставили решения системы и решения HR-специалистов (пригласить или отклонить). Оказалось, что в 85 % случаев система и человек приходили к единому выводу. При этом мы сознательно настроили пайплайн на более высокий recall, чтобы минимизировать риск пропустить потенциально подходящего кандидата, даже ценой некоторого числа лишних рекомендаций. Таким образом, уровень 85 % свидетельствует о высокой близости оценок AI и человека, сохраняя при этом надежность первичного отсева.

Основные количественные показатели качества сведены в таблице 1.

<b>Метрика</b>	<b>Значение</b>
<b>Точность парсинга навыков</b>	91%
<b>Точность извлечения опыта</b>	88%
<b>Правильность оценки соответствия (HR)</b>	85%

*Таблица 1 – Качество работы основных компонентов системы.*

Достигнутые результаты показывают, что автоматический скоринг резюме близок по точности к экспертным оценкам, особенно в части технических навыков и опыта. Хотя в сложных случаях может потребоваться участие человека, в целом система эффективно фильтрует заведомо неподходящие резюме и ранжирует оставшиеся по релевантности.

С точки зрения производительности, система обрабатывает около 100 резюме за 2,5 минуты (с одним GPU и пятью параллельными воркерами), тогда как ручной просмотр аналогичного объёма занял бы несколько часов. Пилотное внедрение в крупной IT-компании показало освобождение до 60 % времени рекрутеров, что позволило переключить их на более сложные задачи — интервью и взаимодействие с Hiring-менеджерами. Одновременно число «холостых» интервью снизилось примерно на 35 %, поскольку неподходящие кандидаты отсеивались еще на этапе резюме. Менеджеры по найму отметили сокращение среднего time-to-hire с 8 до 6 недель, чему способствовал быстрый формирование шорт-листа. Масштабируемость решения подтверждена опытом обработки 1 000

резюме менее чем за 30 минут, а векторное хранилище Milvus устойчиво работает с десятками тысяч эмбеддингов без деградации скорости. Надёжность обеспечивается очередями задач: при перегрузках задачи автоматически откладываются и выполняются при наличии ресурсов, а хранение исходных и промежуточных данных (резюме, JSON, логи) позволяет анализировать спорные случаи и совершенствовать алгоритмы.

Таким образом, внедрение LLM и векторных методов в HR-процесс дает значительный выигрыш по скорости и качеству первичного отбора без ущерба для точности, подтверждая эффективность автоматизации рутины при сохранении роли человека в окончательных решениях.

**Заключение.** В данной статье продемонстрировано, как современные методы обработки естественного языка и большие языковые модели могут быть применены для автоматизации ключевого этапа рекрутинга — оценки резюме кандидатов. Мы разработали систему системы, интегрирующую LLM в пайплайн скоринга резюме, что включает парсинг текста, семантическое сопоставление навыков и опыта с требованиями вакансии, вычисление количественного скоринга и генерацию развёрнутого пояснительного отчёта. Архитектура решения базируется на открытых технологиях (FastAPI, Redis, Milvus, MinIO, open-source LLM), обеспечивающих воспроизводимость, масштабируемость и безопасность работы с данными.

Экспериментальные результаты подтвердили эффективность предложенного подхода. Автоматический скоринг согласуется с оценками HR-экспертов в ~85% случаев, при этом скорость обработки резюме увеличивается на порядки. В пилотном внедрении системы позволил сократить до 70% времени, затрачиваемого рекрутерами на первичный отбор, и уменьшил долю несоответствующих кандидатов, доходящих до

этапа интервью, примерно на 35%. Эти показатели свидетельствуют, что применение LLM в связке с профильными HR-инструментами способно существенно оптимизировать процесс найма, сохраняя или повышая качество отбора.

Использование LLM придало системе способность «понимать» контекст резюме и давать обоснования в виде человеческо-читаемых отчётов, что выгодно отличает системы от простых систем фильтрации по ключевым словам. Модель учитывает неявные факторы, например, описанный опыт, и выводит рекомендации в ясной форме. Вместе с тем, мы обсудили важные вызовы: необходимость контроля bias, повышение объяснимости решений AI, корректная оценка soft skills. Намечены пути дальнейшего развития системы в этих направлениях, включая более глубокую интеграцию с HR-платформами и обогащение данных для анализа.

Полученные результаты и опыт внедрения могут быть полезны для исследователей и практиков в области HR-tech, показывая потенциал синергии NLP и управления талантами. В будущем планируется расширять функциональность системы и регулярно улучшать модель на основе обратной связи от пользователей и новых данных, следуя динамично развивающимся требованиям рынка труда. Авторы надеются, что данная работа станет вкладом в развитие справедливых и эффективных AI-решений для рекрутинга и стимулирует дальнейшие исследования на пересечении искусственного интеллекта и управления человеческими ресурсами.

### **Список литературы:**

1. Sun, J. (2024). Research on the Application of Large Language Models in Human Resource Management Practices. *International Journal*

*of Emerging Technologies and Advanced Applications*, 1(8), 1–8. DOI: 10.62677/IJETAA.2408125

2. Gan, C., Zhang, Q., & Mori, T. (2024). Application of LLM Agents in Recruitment: A Novel Framework for Resume Screening. *arXiv preprint arXiv:2401.08315*.

3. Lo, F. P.-W., Qiu, J., Wang, Z., Yu, H., Chen, Y., Zhang, G., & Lo, B. (2025). AI Hiring with LLMs: A Context-Aware and Explainable Multi-Agent Framework for Resume Screening. *arXiv preprint arXiv:2504.02870*.

4. Wings, I., Nanda, R., & Adebayo, K. J. (2021). A Context-Aware Approach for Extracting Hard and Soft Skills. *Procedia Computer Science*, 193, 163–172. DOI: 10.1016/j.procs.2021.10.016

5. Senger, E., Zhang, M., van der Goot, R., & Plank, B. (2024). Deep Learning-based Computational Job Market Analysis: A Survey on Skill Extraction and Classification from Job Postings. In *Proc. of the 1st Workshop on NLP for Human Resources (NLP4HR 2024)* (pp. 1–15). ACL.

6. Clavel, C., Guo, Y., Shang, G., & Vazirgiannis, M. (2024). JobFair: A Framework for Benchmarking Gender Hiring Bias in Large Language Models. *arXiv preprint arXiv:2406.15484*.

7. Shen, D., Zhang, Q., Chen, X., Sun, Y., Zhu, H., & Xiong, H. (2024). A Comprehensive Survey of Artificial Intelligence Techniques for Talent Analytics. *arXiv preprint arXiv:2307.03195*. (Доступно в Proc. IEEE, 2025)