Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования
“Московский физико-технический институт (государственный университет)”

Факультет инноваций и высоких технологий

Кафедра управления технологическими проектами

РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЧЕСКОГО СТРУКТУРИРОВАНИЯ И КЛАССИФИКАЦИИ АВТОРСТВА НАУЧНЫХ ПУБЛИКАЦИЙ (НА ПРИМЕРЕ ТАСС)

Выпускная квалификационная работа (магистерская диссертация)

Направление подготовки: 27.04.07.

“Наукоемкие технологии и экономика инноваций”

Выполнил:

студент 391а группы Охлопков Даниил Олегович

Научный руководитель:

асс. Аммосов Юрий Павлович

Научный консультант:

_____

1. Список сокращений и обозначений

1. *ТАСС* - российское государственное информационное агентство федерального уровня. Информационные продукты ТАСС получают более 5 тысяч корпоративных подписчиков в России и за рубежом, в том числе более 1000 СМИ, 200 диппредставительств, более 250 финансовых компаний и банков, более 2000 промышленных предприятий, научных и учебных заведений, библиотек [54].
2. *Парсер* (to parse - разбирать) - программа или часть программы, извлекающая информацию из содержимого веб-страниц или другого источника неструктурированных данных [2]. В данной работе будет говориться о парсерах страниц Интернет-изданий и PDF-файлов научных публикаций.
3. *PDF-файлы* (Portable Document Format, портативный формат документов) - формат электронных документов, предназначенный для представления полиграфической продукции в электронном виде [3].
4. *Прокси-сервер* (proxy - доверенное лицо) - промежуточный сервер, выполняющий роль посредника между пользовательской системой и целевой [4]. В рамках данной работы, прокси-серверы будут использоваться для предотвращения блокировки разрабатываемой системы на стороне научных Интернет-издательств из-за потенциально частых запросов.
5. *API* (Application Programming Interface, программный интерфейс приложения) - описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой, в частности, с веб-сервером [5].
6. *RSS-лента* (Rich Site Summary - обогащенная сводка сайта) - семейство форматов предназначенных для описания лент новостей, анонсов статей, изменений в блогах и тп. [6]. Используется агрегаторами для удобного сбора данных из разных источников в

одном формате. В рамках данной работы научные публикации из интернет журналов будут извлекаться через RSS-ленту, если такой функционал будет поддерживаться.

2. Введение

2.1. Важность экстракции метаданных

В научном мире распространение новых идей и открытий осуществляется через публикации и чтение научной литературы. За последние несколько десятилетий, после перехода к электронным средствам массовой информации, научная коммуникация существенно увеличила свой объем [11].

В настоящее время отслеживание научных достижений затруднительно: научная коммуникация и дальнейшее распространение знаний в академических кругах замедлилось в связи с ограниченностью автоматических средств обработки потока научных публикаций [12]. Современные исследовательские порталы, такие как Arxiv [56], Researchgate [60], Google Scholar [61], облегчают изучение научной литературы за счет дополнительного библиографического инструментария. Средства работы с текстом включают: интеллектуальные инструменты поиска, перечни аналогичных и связанных с текстом документов, создание и визуализация интерактивных сетей цитирования и авторства, оценки качество и влияние статей с использованием статистики на основе цитирования и т. д.

Для таких инструментов системе необходим доступ не только к текстовому содержанию хранимых документов, но и к их машиночитаемым метаданным. Несмотря на то, что “предоставление метаданных является обязанностью каждого поставщика данных” [43], нужного для тех или иных анализов качества метаданные не всегда доступны. Поэтому существует спрос на надежный автоматический метод извлечения метаданных, поддающихся машинному считыванию, непосредственно из исходных документов.

2.2. Предмет и объект исследования

Предметом работы является создание агрегатора потоков англоязычных научных публикаций, который позволяет выцеплять из этого потока статьи, авторы которых непосредственно или исторически связаны с Россией.

Объектом исследования является корпус научных трудов, опубликованный в Интернет-изданиях и Интернет-агрегаторах, и их использование научной редакцией ТАСС.

2.3. Цель и задачи ВКР

Цель - разработать под задачи научной редакции ТАСС систему автоматического структурирования и классификации авторства научных публикаций с возможной дальнейшей коммерциализацией тем или иным методам.

Задачи:

1) Определить возможность существующих технологий для создания описанной системы,
2) Определить требования к такой системе,
3) Разработать описанную систему и провести первоначальное обучение базы с участием сотрудников ТАСС,
4) Предложить способ дальнейшей коммерциализации.

3. Современное состояние машинного алгоритмического анализа авторства научных работ

3.1. Актуальность

Все больше и больше ученых из неанглоговорящих стран выбирают английский язык в качестве основного языка для своих публикаций, поэтому обнаружение таких статей, а также все научные коммуникации значительно усложняются. В результате научные труды соотечественников тонут в постоянно увеличивающимся потоке англоязычных публикаций и не получают должного внимания со стороны научного сообщества. Поэтому необходимость такой фильтрации диктуется научными администрациями неанглоговорящих стран, в частности России. В англоязычных странах ученые большую часть публикаций в научные журналы делают на своем родном языке [15]. По этой причине подобный функционал в таких странах, скорее всего, будет не востребован.

Рисунок 1: Соотношение количества журнальных статей, опубликованных исследователями на английском языке, к числу статей на официальных языках восьми разных стран, 1996–2011 годы [7].

За все время во всех странах было выпущено более 50 миллионов научных трудов [8], а в среднем в год выходит более 2.5 миллиона новых статей [9]. На 2019 год насчитывается более 13 000 научных журналов с открытым доступом к статьям (по данным агрегатора DOAJ.org [10]). Такой поток публикаций нуждается в автоматической агрегации и тегировании.

Сейчас требуется редакторская работа, чтобы вручную отобрать статьи по различным критериям, в частности, по принадлежности к той или иной стране. Для того, чтобы найти публикации российских ученых, сотрудникам ТАСС приходится просматривать каждую статью, у которой есть автор со славянской фамилией, и вручную проверять его аффилиацию. Только по имени автора аффилиацию определить достоверно нельзя: нобелевский лауреат 2010 года Константин Новоселов, выпускник МФТИ 1997 года, публикуется в Манчестерском университете, а Фарит Халили - профессор Физфака МГУ [78]. Статья может остаться незамеченной, если у автора неславянская фамилия. Публикации по биологии могут содержать более десяти авторов, поэтому легко можно пропустить русскую аффилиацию в огромном списке иностранных институтов и исследовательских центров.

Авторы публикаций часто не указывают полную информацию о себе, например, свою аффилиацию, полные ФИО, соавторов. В некоторых случаях недостающую информацию можно восстановить, используя базы знаний, например, можно использовать аффилиационные данные из других публикации авторов.

Один из важнейших трендов развития науки - увеличение мобильности как постдоков и аспирантов, так и ученых[1]. На Рисунке 2 представлены абсолютные данные по академической мобильности российских исследователей, составленные по данным публикациям в Scopus, OECD [1]. По оценкам члена Совета по науке при Минорбнауке Александра Кабанова, Российская научная диаспора “очень крупная” - число ученых, которые получили постоянный контракт преподавания в

университете или директора исследований в крупной компании , “измеряется тысячами” [55].

Рисунок 2: Абсолютный поток мобильности по РФ, человек [1].

Научным и научно-популярным СМИ необходимо узнавать о новых научных статьях как можно раньше: по словам редакторов ТАСС, с которыми работал автор, пятидневная задержка между публикацией и новостью о ней уже недопустима. Несмотря на это, из-за отсутствия возможности отфильтровать статьи по стране авторов, агрегаторы научных публикаций частично решают проблему оперативного поиска и агрегации статей из разных журналов в одном месте: задержки в Google Scholar могут достигать 7 рабочих дней [79].

На основании сказанного есть достаточные причины предположить, что разработка искусственного интеллекта по автоматическому детектированию, тегированию и классификации потока научных публикаций становится актуальной и востребованной.

3.2. Новизна

Существующие инструменты для экстракции аффилиации авторов из научных публикаций (CERMINE [13], GROBID [14] и др.), Интернет-журналы (Arxiv [56], PLOS [59] и др.) и агрегаторы (ResearchGate [60], Google Scholar [61] и др.) не позволяют фильтровать статьи по принадлежности автора к той или иной стране.

3.2.1. Экстракция метаданных из научных публикаций

Существуют как открытые, так и платные решения по экстракции метаданных из научных статей. Обычно аналитиков интересуют следующие метаданные:

- Заголовок,
- Абстракт,
- Библиографические ссылки,
- Имена и аффилиации авторов,
- Данные о журнале или издателе, и др.

Самый распространенные подходы к проблеме поиска и экстракции метаданных из научных публикаций используют регулярные выражения, базы знаний и алгоритмы машинного обучения.

Регулярные выражения предполагают, что текст размечается и анализируется с помощью формальной разбивки по строкам (количество символов, последовательности знаков препинания и тд.). Этот подход обычно основан на наборе разработанных вручную правил, которые способны захватывать несколько полей метаданных в различных текстах. Такая стратегия работает лучше всего, если стиль оформления метаданных в текста (аффилиации, ссылки на источники) соответствует некому стандарту или известен заранее. На практике, поддержание паркера на основе регулярных выражений может быть очень трудоемкой из-за постоянной адаптации набора используемых регулярных выражений на случай выявления новых стандартов оформления метаданных в тексте.

Регулярные выражения часто сочетаются с другими методами, например, дополнительной логикой распознавания или базами знаний. Гупта и др. [16] предлагает сочетание эвристик на основе регулярных выражений и систем на основе баз знаний для парсинга ссылок на другие статьи: подход способен сопоставлять встроенные цитаты с соответствующими библиографическими ссылками. Константин и др. [17] описал систему, основанную на правилах и регулярных выражениях, называемую PDFX и способную извлекать логическую структуру и библиографию научных статей из документов в формате PDF. Дэй и др. [18] предложил использовать иерархическую структуру представления знаний под названием INFOMAP для извлечения метаданных из строк с ссылкой на источник. Они сообщают о точности 92,39% для извлечения автора, названия, журнала, номера тома и выпуска, года и страницы из ссылок, отформатированных с шестью основными ссылочными стилями. Наконец, Кортез и др. [19] описал FLUX-CiM - метод автоматического подбора регулярных выражений для экстракции библиографических метаданных по корпусу размеченных научных публикаций, собранной из открытых источников. Согласно их результатам, FLUX-CiM достигает значений метрик точности и полноты выше 94% для широкого набора возможных вариантов записи метаданных. Эти и подобные им решения используются в коробочных решениях по комплексной экстракции метаданных из научных публикаций и не только.

Машинное обучение с учителем также используется в задачах экстракции метаданных из научных публикаций. При таком подходе обучающие данные используются для тренировки так называемой модели, которая проводит синтаксический анализ и извлекает метаданные из входной строки. Такой подход требует незначительных экспертных знаний и ручной подстройки коэффициентов, поскольку модели находят зависимости и правила самостоятельно непосредственно на основе данных обучения.

В подходах, основанных на машинном обучении, анализ библиографических ссылок и других метаданных обычно формально определяется как задача тегирования последовательности. В задаче тегирования последовательности на входе дается последовательность объектов, в нашем случае - слова, а целью

является присвоение меток элементам последовательности с учетом не только самих объектов, но и их соседей в последовательности. Данная задача может решаться несколькими алгоритмами машинного обучения, в частности методами опорных векторов (SVM), скрытыми марковские модели (HMM), условными случайными полями (CRF) и другими. SVM - это метод классификации общего назначения, в то время как HMM и CRF могут быть непосредственно использованы в качестве тегирователя последовательностей.

Хетцнер [20] предложил решение, основанное на скрытых марковских цепях HMM для извлечения полей метаданных из ссылок. Инь и др. [21] используют модификацию традиционного HMM - биграммные марковские цепи, который использует последовательные пары слов и информацию об их положении. Ойоко и др. [22] исследовали применимость триграммной версии HMM и добились значений точности, полноты и F1-метрики более 95%.

Коунчилл и др. [23] разработали ParsCit, одну из самых известных и широко используемых систем на основе CRF с открытым исходным кодом для извлечения метаданных из ссылок. Система GROBID, созданная Лопесом [24], является еще одним примером системы на основе CRF, функционал которой не ограничивается анализом библиографических ссылок: GROBID является более крупным инструментом, способным извлекать широкий спектр метаданных и логическую структуру из научных работ в формате PDF. Автор сообщает о точности тегирования на уровне 95,7%. GROBID был также внедрен такими компаниями, как ResearchGate, Mendeley [25]. CERMINE, предложенный Tkaczyk et al. [26], также является большой системой, способной извлекать метаданные и структуру, включая разобранную библиографию, из научных работ в формате PDF. Функциональность синтаксического анализа CERMINE также основана на методике CRF [27]. В 2015 году CERMINE выиграла Semantic Publishing Challenge [53], которая включала задачи, требующие точного извлечения информации о названии и годе из библиографических ссылок. Мацука и др. [28] также предлагает основанный на CRF метод парсинга ссылок, который использует как лексические функции, так и лексиконы. Наконец, Чжан и соавт. [29] применили алгоритм CRF для задачи

извлечения информации об авторах, их аффилиаций, журнале и годе из справочных строк, сообщая в целом о 97,95% F1 по данным PubMed Central.

Zou и др. [30] сравнили системы, которые используют CRF и SVM, и получили сходную общую точность для обоих подходов: точность свыше 97%. Чжан и соавт. [31] предлагают SVM с контекстными признаками в качестве входных параметров и сравнивают их с обычными SVM и CRF. Они также сообщают о схожей точности для всех трех подходов: точность классификации токенов выше 98% и 95% для детекции расположения данных. Наконец, Ким и др. [32] описывают систему под названием BILBO и сравнивают ее с другими популярными инструментами анализа ссылок (ParsCit, Biblio, free_cite и GROBID). Согласно их исследованию, лучшие результаты были получены с помощью BILBO (F1 0,64), за которым следует GROBID (F1 0,63).

Часть известных автору открытых решений работают исключительно с библиографией. К ним относятся:

- Anystyle-Parser (основано на CRF и написано на языке Ruby) [62].
- Biblio (основано на регулярных выражениях и написано на языке Perl) [63].
- Citation (основано на регулярных выражениях и эвристиках, написана на Ruby) [64].
- Citation-Parser (основано на предопределенных эвристиках и написана на языке Python) [65].
- Free_cite (основано на CRF и написано на языке Ruby) [66].
- Neural Parscit (основано на методе глубинного обучения - LSTM) [67].
- Reference Tagger (основано на CRF и написана на Python) [68].

Существуют также комплексные решения с открытым кодом, которые решают не только проблему детекции и экстракции библиографических ссылок, но и позволяют получить и другие метаданные, такие как имена и аффилиации соавторов, данные о журнале и др:

- CERMINE [33] [69],
- GROBID [34][70],
- ParsCit [35][71],

- PDFSSA4MET [72],
- Science Parse [73].

Таблица 1: Сравнение комплексных открытых решений по экстракции метаданных из печатных научных публикаций [35].

Решение	**Основано на**	**Извлекает метаданные**
CERMINE	CRF	Авторы, аффилиации, данные о журнале, библиографические ссылки, цифровой идентификатор объекта (DOI), страницы, заголовки
GROBID	CRF	Авторы, рецензент, редактор, данные о журнале и издания, организации, библиографические ссылки, страницы, заголовки, даты
ParsCit	CRF	Авторы, редакторы, аффилиации, заголовки, даты, данные о журнале и издании, библиографические ссылки, локации
Neural ParsCit	LSTM	Авторы, редакторы, аффилиации, заголовки, даты, данные о журнале и издании, библиографические ссылки, локации
PDFSSA4MET	Регулярные выражения	Страницы, заголовки, издания, года
Science Parse	CRF	Авторы, заголовки, данные о журнале, даты

Ткачук, Липинский и др. провели сравнения производительностей перечисленных решений [36]: в результате наивысшие показатели по метрикам точности, полноты и F1-метрики показала система GROBID. Было также проведено несколько исследований по сравнению самых популярных существующих решений по извлечению метаданных из PDF-файлов [25], в которых программа GROBID также показала наивысшую точность.

Парсеры могут извлекать только явно указанные в тексте данные [39]. В частности, аффилиации соавторов могут состоять из упоминания научного центра без явного указания страны, парсер не сможет выдать недостающие данные.

Подобная проблема может решаться базами знаний. В случаях, когда страна не была указана, ее можно получить по наименованию научного заведения, при условии наличии данных, связывающих институты и страны. Подобные датасеты можно вручную собрать из открытых источников, например, Википедии [40].

Опыт тестовой эксплуатации GROBID на ноутбуке Macbook (типичная конфигурация персонального компьютера современного типа 1,1 GHz Intel Core M (4 ядра) и 8 GB RAM) показал, что время обработки одного документа может составить несколько секунд. Подобное время работы может оказаться недостаточно быстрым при обработке всего потока выпускаемых научных публикаций.

Проблема низкой производительности вследствии избыточности решений может решаться несколькими альтернативными способами. Первый способ - за счет грубой силы - увеличить число и мощности серверов, на которых запущены парсеры. Более того, современные облачные инфраструктуры, такие как GCP или AWS, позволяют динамически увеличивать количества обрабатывающих запросы машин в кластере в зависимости от их нагрузки.

Второй способ - использовать сокращение избыточности, за счет того, что метаданные извлекаются из того Интернет-ресурса, где статья была опубликована. Зачастую на сайте Интернет-издания публикуются не только PDF-файлы, но и некоторая метаинформация о статье, которая была либо добавлена авторами публикации самостоятельно, либо была извлечена из статьи парсерами на стороне Интернет-издания. Например, Интернет-журналы, такие как BioRxiv [74] и PLOS [59], публикуют информацию об авторах, их аффилиациях и др. Доставая эти

данные из веб-сайтов, необходимость парсить PDF-файлы (печатные копии статей) отпадает, что позволяет экономить время обработки и вычислительные ресурсы.

3.2.2. Структурирование информации с веб-публикаций научных работ

Существует спрос на структурирование данных, опубликованных в интернете: данные одной природы, в частности, научные публикации, могут быть оформлены по-разному и опубликованы в разных Интернет-площадках. Ученые разных стран отправляют в англоязычные журналы свои труды, впоследствии создавая неудобства научным администрациям в поиске публикаций своих соотечественников.

Существуют агрегаторы научных публикаций из разных открытых Интернет-журналов такие как ResearchGate [60], Google Scholar [61] и др. Такие порталы периодически и автоматически собирают информацию о новых научных публикациях из различных источников и приводят данные в единый формат.

Для уменьшения нагрузки на сайт Интернет-ресурса, его владельцы внедряют возможность третьей стороне извлекать полезные данные по специальным интерфейсам: API или RSS ленте. Однако такие интерфейсы позволяют получать только те данные, которые в него были заложены разработчиками, а они не всегда совпадают с желаемыми. Более того, далеко не все сайты предоставляют такой интерфейс для механического доступа к данным.

Для построения системы сбора и структурирования данных из сети Интернет используют различные топологии, в зависимости от поставленной задачи [41]. Одна из самый простых в имплементации, но при этом масштабируемых архитектур таких систем приведена на схеме ниже.

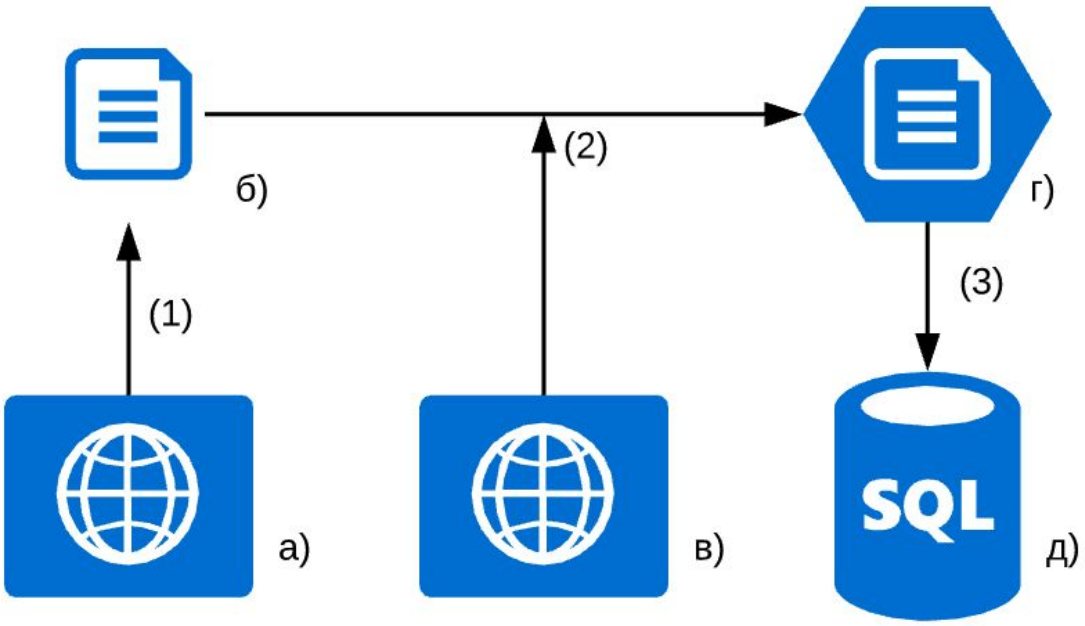

Рисунок 3: обобщенная схема построения масштабируемого парсера.

Элементы на схеме:

а) Интернет-ресурс, на котором расположены интересующие данные,

б) Структурированная информация из а),

в) Интернет-ресурс, данные которого будут дополнением к уже собранным. Может совпадать с а),

г) Обогащенная структурированная информация,

д) База данных для хранения полученных данных.

Алгоритм масштабируемого парсинга данных следующий:

1) Извлекаются интересующие данные о научной публикации из Интернет-ресурса. На этом этапе формируется список из объектов, каждый из которых еще может не содержать все необходимые метаданные.
2) Обогащение информации о каждой статье дополнительными данными из этого же или другого интернет ресурса.
3) Загрузка в базу данных всей полученной информации.

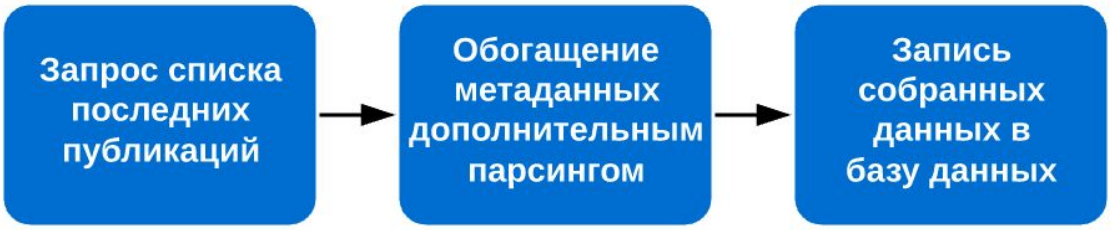

Рисунок 4: Обобщенный алгоритм масштабируемого парсинга

3.2.3. Функционирование при наличие защит от доступа

Автоматизированный сбор информации третьей стороной нагружает серверы, на которых она расположена. Поэтому владельцы подобных ресурсов внедряют системы защиты от парсеров, которые блокируют доступ к ресурсам для определенных IP-адресов из-за частых запросов.

Существует несколько способов обходить подобные защиты от парсинга. Одно решение заключается в том, чтобы парсить сайты насколько медленно, насколько позволяет их защита от автоматического сбора данных, притворяясь пользователем, который зашел на сайт. Такое решение зачастую не требует много времени разработки, однако скорости обработки веб-сайтов может не хватать для бизнес задач.

Другое решение заключается в использовании прокси-серверов (туннелей, позволяющий использовать чужой ip адрес для Интернет-запросов) или нескольких серверов для сбора данных (запросы параллелятся по разным машинам). Для частого смена ip адреса можно использовать Tor сеть [75], которая на Апрель 2019 года состоит из 7500 числа добровольно подключенных компьютеров [76].

3.2.4. Способы анализ собранной информации

Цель тегирования текстов или любых других объектов заключается в упрощении поиска и фильтрации элементов списка однородных сущностей. Тегирование позволяет произвести отображение оригинальных объектов во множество с меньшей мощностью - в заранее определенное количество категорий. Например, музыкальным композициям могут быть проставлены теги, соответствующие жанру, сложности гармонии, языку текста песни, наличию

непристойных слов и тд, а авторам научных публикаций можно присвоить тег - страну его аффилиации.

В большинстве случаев теги проставляются авторами контента самостоятельно - для повышения охвата и упрощения поиска среди огромного потока другой информации. Наличие правильно выставленных мета тегов не гарантирует желаемый охват, так как зачастую множество допустимых к заполнению тегов неполно: невозможно предугадать все возможные варианты.

Для получения результата на выходе искусственного интеллекта, нужно подготовить данные для его обучения: датасет “правильных ответов” -- пар “документ - правильный класс”, и основная проблема -- это собрать достаточный объем таких размеченных объектов [77]. В то время как обычно легко собрать не размеченные документы, их не так просто классифицировать вручную для создания достаточного объема данных для обучения.

Качество работы искусственного интеллекта зачастую ограничено размерами обучающей выборки [44]. Существует несколько способов собрать больше размеченных данных, один из них - позволить пользователям оставлять отзыв о качестве работы алгоритмов с просьбой указать правильные ответы. В дальнейшем, собранные данные можно учитывать при дообучении алгоритма тегирования.

Для увеличения обучающей выборки для задач машинного обучения используют несколько подходов. Один из них - сбор пользовательских реакций на результаты тегирования. Если пользователь увидел статью, которая была отнесена не в ту группу, или не отнесена ни к какому классу, он может назначить этой статье правильный тег, который потом будет использоваться для улучшения качества классификации.

Также существуют другие методы увеличения размера обучающей выборки, например, использующие подход “обучение без учителя”. Предлагаемый метод, описанный в статье [45], делит документы на предложения и классифицирует каждое из них, используя списки ключевых слов каждой категории и меру близости между ними. После выполнения этой операции, категоризированные предложения используются для обучения. Предложенный метод показывает аналогичную степень качества, по сравнению с традиционными методами обучения с учителем. Поэтому

этот метод можно использовать в областях, где необходима дешевая и допустима не очень точная классификация текста. Его также можно применить для расширения обучающей выборки.

Современные методы классификации текстов, основанные на глубоких нейронных сетях [46], показывают наилучшие результаты в сравнении с другими алгоритмами, в основе которых лежат классические методы машинного обучения [47]. При этом архитектурные детали нейросетевых алгоритмов, а также подходы по улучшению их точности разнятся в зависимости от условий, в которых они применяются [48].

Большинство готовых решений по анализу текстов, основанных на алгоритмах машинного обучения, работают исключительно с английским языком. Для перестройки алгоритмов на другой язык необходимы соответствующие датасеты. Для некоторых языков уже можно найти “переведенные” нейронные сети, однако их число мало. Для русского языка, в частности, нет готового решения по экстракции метаданных из научных публикаций, поэтому при необходимости решения подобной задачи, нужно размечать данные вручную.

Улучшить качества моделей можно тремя способами [49]:

1) Изменить архитектуру алгоритма машинного обучения,
2) Оптимизировать значения гиперпараметров модели,
3) Улучшить обучающую выборку.

Первые два способа обычно применяются на самых ранних этапах исследования: подбора алгоритма и его настройка под конкретную задачу, в то время как последний подход можно с легкостью использовать в любое время жизни проекта. Если взять ту же самую модель, которая была подобрана на первых этапах, и обучить ее на более полном и точном датасете, качество модели с большой вероятностью улучшится.

Таким образом, если дать пользователям возможность исправлять теги, которые были проставлены алгоритмом тегирования, можно через некоторое время обновить тренировочный датасет, добавив в него примеры неправильной работы, и, впоследствии, переобучить классификатор [50]. При этом переобучать после каждого пользовательского ответа не рекомендуется, так как некоторые модели

могут сильно изменить свои предсказания после необычного с их точки зрения пользовательского ответа [37][38]. Более того, частое переобучение может быть ресурсозатратным. Рекомендуется сначала собрать большой датасет пользовательских исправлений, и только потом вручную перетренировать модель, проверяя работоспособность на отложенной тестовой выборке.

3.3. Постановка задачи

Проведенный анализ технической и деловой литературы демонстрирует потребность в разработке системы автоматического структурирования и классификации научных публикаций. Эта система - инновационная и научно ценная, так как существующие решения по агрегации и тегированию не позволяют фильтровать поток статей по необходимым метаданным.

По итогам интервью, проведенных автором с редакторами ТАСС, стало возможным сделать вывод, что статья представляет интерес для российской редакции ТАСС (далее “интересной ТАСС” [80]), если

1. Имеет российскую аффилиацию,
2. Или является соотечественником за рубежом.

По аналогии назовем соавтора научной статьи “интересным ТАСС”, если он удовлетворяет условиям:

1) Хотя бы одна его аффилиация - Российская,
2) Или он является русским соотечественником за рубежом.

В итоге, можно назвать статью “интересной ТАСС”, если хотя бы один из ее соавторов “интересен ТАСС”.

Если первый пункт определения исключительно технический (необходимо извлекать информацию из метаданных публикации обо всех аффилиациях автора и проклассифицировать их), то для решения второго нужно собрать базу знаний о российских соотечественников за рубежом. Для этого планируется внедрять функционал сбора пользовательских указаний о том, является ли данная статья или автор статьи “интересным ТАСС”.

Необходимо разработать агрегатор потоков англоязычных научных публикаций, который позволяет извлекать из этого потока и показывать пользователям статьи “интересные ТАСС”.

По словам ТАСС, у них есть ежегодный KPI на количество опубликованных новостей о “интересных ТАСС” научных работах. Из-за того, что на данный момент не существует агрегатора научных статей с фильтром по стране аффилиации соавторов, сотрудникам ТАСС крайне сложно находить интересующие их статьи. Поэтому на данный момент они пишут про любые научные публикации, надеясь, что рано или поздно попадется именно “интересная ТАСС”.

4. Методология

В этой главе будет обоснован выбор базового алгоритма экстракции метаданных из научных публикаций, а также определены пути его дальнейшей доработки, будет описана подходящая модель машинного обучения, ее последующее обучение и сбор необходимых данных. Далее будет описан созданный прототип системы.

4.1. Описание системы

4.1.1. Архитектура системы

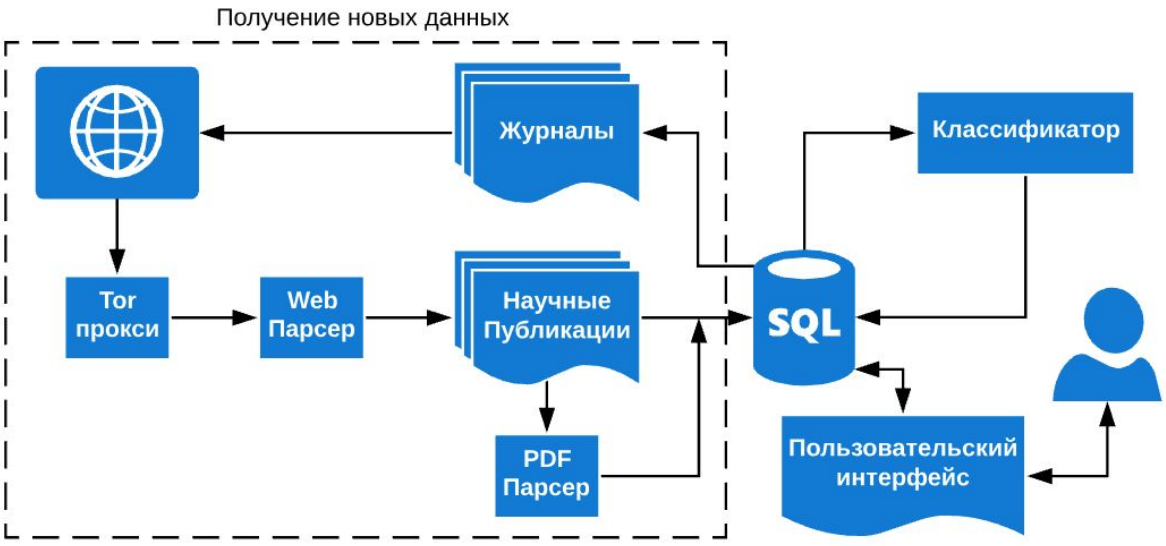

Рисунок 5: Архитектура разрабатываемой системы.

Легенда:

- *Журналы* - список Интернет-изданий, для которых был разработан парсер веб-страниц.
- *Tor-прокси* - микросервис, позволяющий использовать серверы, подключенные в сеть Tor, в качестве прокси серверов для увеличения робастности парсера.

- *Web-парсер* - микросервис, позволяющий извлекать из веб-страниц Интернет-изданий информацию в структурированном виде.
- *Научные публикации* - статьи Интернет-изданий, метаданные которых были приведены в единый табличный формат.
- *PDF-парсер* - микросервис, извлекающий из PDF-файлов научных публикаций метаданные. Запускается только в случае нехватки метаданных, извлеченных на этапе парсинга страниц Интернет-журналов.
- *SQL* - база данных, в которой хранятся необходимые извлеченные метаданные научных публикаций.
- *Классификатор* - микросервис, позволяющий для скаченных научных публикаций определять их принадлежность к “интересным ТАСС” на основе не только извлеченных метаданных, но и используя пользовательские исправления результатов работы классификатора.
- *Пользовательский интерфейс* - веб-интерфейс разработанной системы, через который можно не только просматривать найденные статьи российских ученых, но и собирать пользовательские исправления результатов работы классификатора.

Для предотвращения потери доступа к Интернет-журналам из-за потенциально частых запросов необходимо использовать прокси-серверы: они помогут распределить нагрузку на серверы Интернет-издательств, увеличив количество доступных IP-адресов. Было принято решение использовать сеть Tor в качестве поставщика прокси серверов, так как подключение к сети Tor и дальнейшее использование сети Tor бесплатное, а количество доступных нод - потенциально доступных прокси серверов по всему миру - превышает 7000 (на июнь 2019 года) [76].

Все открытые решения получают на вход PDF-файл с печатной версией публикации и извлекают оттуда метаданные, представляя пользователю самому найти статьи, скачать и отправить статьи в парсер PDF-файлов для последующего анализа. Так как ТАСС планирует оперативно писать новости об “интересных

TACC” публикациях, нужно находить статьи сразу после того, как они появились в Интернет-журналах. Следовательно, необходимо разработать свой собственный агрегатор Интернет-издательств научных статей вместо того, чтобы пользоваться сторонним сервисом, таким как Google Scholar, задержки которого могут достигать 7 рабочих дней [79].

4.1.2. Пользовательский интерфейс

Для разработки пользовательского интерфейса были решено провести структурированные интервью в модифицированном виде с учетом специфики клиента. В результате был определен базовый функционал интерфейса, который должен удовлетворять следующим требованиям:

1) Необходимо выводить список найденных статей “интересных TACC”,
2) Необходимо выводить список всех скачанных парсером статей,
3) Каждая статья должна иметь ссылку на первоисточник публикации,
4) Каждую статью в списках можно вручную отметить как “интересную TACC” или снять отметку.

Информацию о пользовании клиентом разрабатываемой системы планируется получать не только во время самих интервью, но и анализируя поведения пользователей на сайте, используя функционал Вебвизора бесплатного сервиса Яндекс Метрика [81].

4.1.3. Извлечение метаданных о научных публикациях

Большинство Интернет-журналов имеют специальный машинный интерфейс, по которому можно получить новые публикации на ресурсе. Самый распространенный вариант - RSS лента, некоторые поддерживают доступ к данным по API. Из-за того, что разработка парсера веб-страниц занимает больше времени, чем извлечение данных по специальным интерфейсам, во всех возможных случаях необходимо получать максимум информации таким способом для экономии времени разработчика. Если через специальные интерфейсы не удастся извлечь все необходимые данные, следующим шагом будет попытка извлечь недостающие данные из веб-страницы с научной публикации в Интернет-журнале. Если после

этой стадии все еще будет недоставать некоторых метаданных - они будут извлечены из PDF-файла печатной версии научного труда.

Извлечение метаданных из веб-страниц Интернет-журналов, где были опубликованы научные работы, выполняется с помощью инструмента Beautiful Soup [51] - популярной библиотеки для экстракции полезных данных с веб-сайтов для языка программирования Python. [52].

Так как не все необходимые метаданные публикуются на сайтах научных Интернет-журналов, необходимо извлекать их из PDF-файлов. В качестве инструмента для выполнения этой задачи была выбрана система GROBID, опередившая по показаниям метрики точности, полноты и F1-метрики другие открытые инструменты [36] [25].

Полный список использованных технологий и фреймворков в проекте:

- Backend - Python Django 2.2.1 [82],
- База данных - PostgreSQL [83],
- Парсер PDF-файлов - GROBID [70],
- Парсер веб-страниц - Beautiful Soup [51],
- Прокси сервер - Tor Rotating Proxy server [75],
- Запуск задач на кластере - Celery [84],
- Frontend - Vuejs [85].

Технологии, такие Django, PostgreSQL, Vuejs, Beautiful Soup, Celery, выбирались исходя из наличия опыта разработки. Использовалась реляционная база данных PostgreSQL, так как ее интеграция идет “из коробки” при использовании библиотеки Django как средства разработки логики бекенда на языке программировании Python.

4.1.4. Поиск публикаций российских ученых за рубежом

GROBID решает одну часть поставленной ТАСС задачи, а именно поиск публикаций российских ученых в зарубежных изданиях. Другая часть - поиск работ ученых-соотечественников за рубежом, которые публикуются из-под зарубежных институтов - может решаться обширной базой знаний российских соотечественников за рубежом. Из-за того, что число таких людей постоянно растет,

эта подзадача не может быть эффективно в долгосрочной перспективе решена словарем. Именно поэтому необходимо разработать самообучающуюся систему, которая, используя пользовательские входные данные, сможет последствии улучшить распознавание и точность детектирования.

Для решения задачи поиска публикаций соотечественников за рубежом предлагается собирать разметку научных статей по группам “русская / не русская”. Правильные ответы будут оставлять пользователи платформы - сотрудники ТАСС. Эти данные поначалу будут использоваться в качестве словаря, который сможет сопоставить зарубежному ученому статус “соотечественника за рубежом”, а в дальнейшем, при накопленном объеме свыше 10000 объектов, использована при обучении ИИ, который уже сам сможет детектировать интересующие статьи и без входных данных от сотрудников клиента.

Был выбран метод обучения с учителем с последующем дообучением на основе пользовательских отзывов о качестве алгоритма классификации, так как “правильные ответы” для модели машинного обучения проставляют пользователи платформы - сотрудники ТАСС. Так как задача тегирования была сведена к задаче бинарной классификации, сотрудники ТАСС будут выбирать правильный класс научной статье в случаях, когда она была проклассифицированна неправильно. Эти данные и будут использованы для улучшения детекции алгоритмов тегирования.

Для того, чтобы наиболее правильно и точно выбрать архитектуру нейросетевого классификатора, который будет находить англоязычные публикации российских соотечественников за рубежом, необходимо будет провести дополнительное исследование после того, как будет собран датасет для обучения.

4.2. Разработанное решение и результаты его эксплуатации

4.2.1. Экстракция метаданных

Было разработано собственное решение по получению аффилиационных данных из сайта-агрегатора научных статей, на котором была опубликована статья. Оно позволяет заметно уменьшить время обработки одной статьи, так как не работает с PDF документом, а исключительно с данными на сайте научного Интернет-издательства. С другой стороны, для того, чтобы добавить новый источник данных (Интернет-журнал с научными публикациями) необходимо разработать парсер его веб страниц.

Для решения поставленной задачи, из научных публикаций необходимо доставать следующий минимальный набор данных:

1) Залоговок
2) Абстракт
3) Дата публикации
4) Список авторов
5) Их аффилиации

Первые три пункта из списка выше необходимы для визуализации результатов работы алгоритма в пользовательском интерфейсе: список статей, отсортированный по убыванию даты публикации. Последние два пункта необходимы для классификации статей на “интересные ТАСС” и нет.

Для извлечения вышеупомянутых метаданных из научных публикаций была выбрана система GROBID, показавшая [25] наивысшие показатели точности, полноты и F1-метрики в сравнении с другими открытыми комплексными решениями.

Иногда в научных публикациях явно указывают принадлежность аффилиации к какой-либо стране. В этих случаях GROBID сможет извлечь и эти данные. Если же такой информации не будет, принадлежность аффилиации и,

соответственно, соавтора научной публикации будет определяться нашим классификатором.

Так как скорость обработки 1 PDF-файла системой GROBID может достигать 7 секунд, а необходимое поле метаданных “дата публикации” обычно не поставляется в печатных PDF-файлах научных статей, было принято решение скомбинировать решения GROBID с самописным парсером Интернет-сайтов, в которых можно найти недостающие метаданные. При этом комбинированном решении и тогда может оказаться, что извлекать метаданные из PDF-файлов необходимости нет - все интересующие данные уже присутствовали на веб-сайте. Отрицательная сторона такого подхода - необходимость разрабатывать парсер под каждый веб-сайт отдельно, что занимает дополнительное время разработчика, однако уменьшает время работы серверов и увеличивает точность извлеченных данных.

В итоге был разработан гибридный алгоритм экстракции метаданных [рисунок 6], в основе которого лежит парсер PDF-файлов GROBID и парсер веб-сайтов Beautiful Soup. Алгоритм старается достать как можно больше полезных данных из веб-страниц с научной публикацией и запускает GROBID только тогда, когда не все интересующие нас данные были опубликованы на сайте научного интернет-журнала.

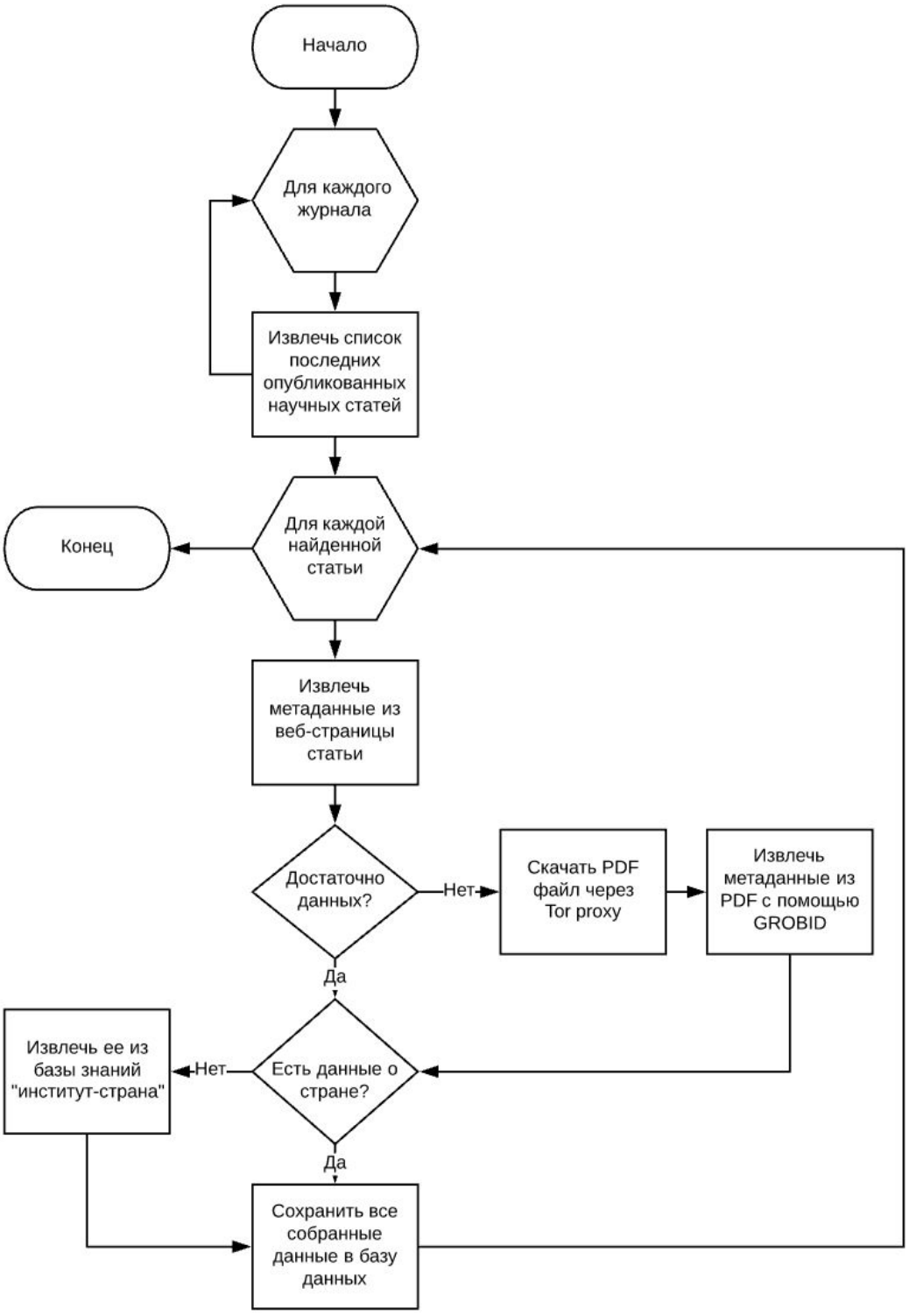

Рисунок 6: Алгоритм парсинга и экстракции метаданных из научных публикаций в Интернет-журналах.

4.2.2. Классификатор принадлежности аффилиации к той или иной стране

Итого, чтобы иметь возможность дать ответ, является ли текущая научная статья “интересной ТАСС” или нет, необходимо разработать парсеры и программы, которые позволят:

1) Для каждой статьи извлечь список соавторов,
2) Для каждого автора извлечь его аффилиации,
3) Для каждой аффилиации определять, находится ли она в России,
4) Для каждого автора определять, является ли он русским соотечественником за рубежом.

На рисунке 7 представлен алгоритм, который позволяет определять, является ли статья “интересной ТАСС”.

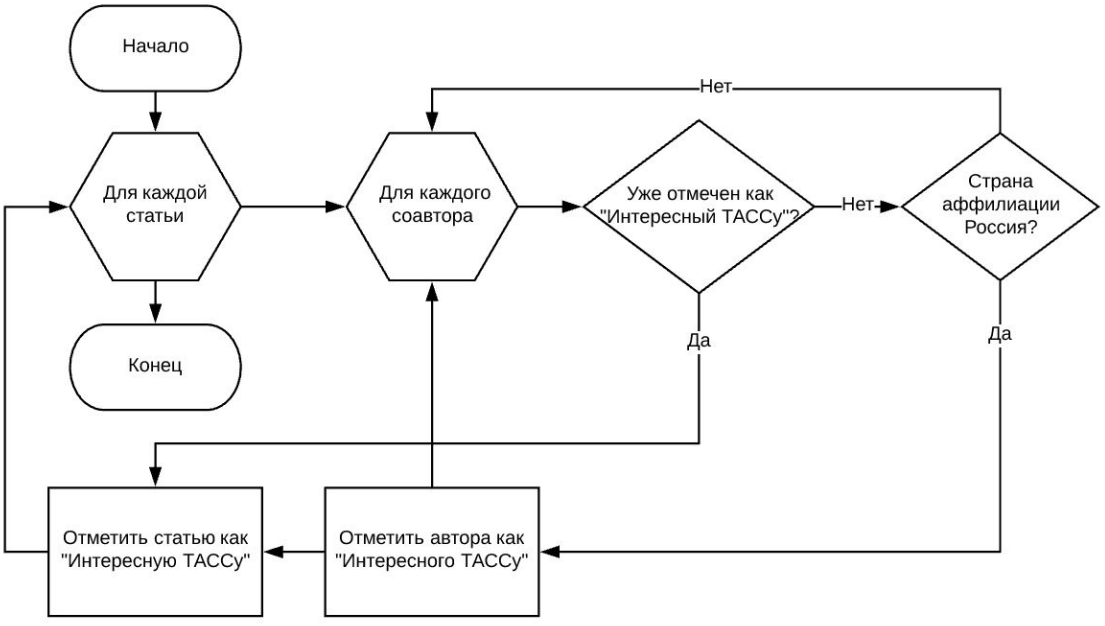

Рисунок 7: Алгоритм классификации авторства научных публикаций.

4.2.3. Детали архитектуры платформы

Для корректного и масштабируемого функционирования системы, были введены следующие сущности, для каждой из которых была создана своя таблица в базе данных проекта.

Ниже перечислены сущности, их краткое описание, а также список полей - столбцов базы данных, которые используются для корректного описания логики системы. Все поля заполняются при наличии соответствующих данных.

1) *Источник данных* - поток новых научных публикаций.
    a) Название потока данных.
    b) Ссылка, по которой получать новые публикации.
    c) Формат данных по ссылке (API или RSS лента).
    d) Дата последнего обновления.
    e) Флаг - получать ли статьи по этому источнику.
2) *Аффилиация* - объект, который сопоставляется научным заведениям.
    a) Название института.
    b) Название факультета.
    c) Адрес и индекс.
    d) Страна.
3) *Автор* - объект, который сопоставляется соавтору публикации.
    a) Полное имя.
    b) Ссылки на объекты типа *Аффилиация.*
    c) Флаг - является ли этот автор “интересным ТАСС”.
4) *Статья* - структурированные метаданные о научной публикации.
    a) Заголовок статьи.
    b) Абстракт.
    c) Ссылка на публикацию в Интернет-журнале.
    d) Ссылка на скачивание PDF-файла.
    e) Дата публикации.
    f) Дата парсинга статьи.

g) Ссылка на единственный объект *Источник данных*, откуда была получены данные этой статьи.
h) Ссылка на объекты типа *Автор.*
i) Флаг - была ли данная статья обогащена данными со страницы публикации в Интернет-журнале.
j) Флаг - была ли данная статья обогащена данными из PDF-файла.
k) Флаг - результат работы классификатора - является ли статья “интересной ТАСС”.
l) Флаг - была ли статья отмечена пользователями системы как “интересная ТАСС” или наоборот.

Указанная схема базы данных обладает важным свойством: если в какой-то момент в систему поступит научная публикация, в которой соавторам не указали аффилиации, недостающие данные подтянутся из предыдущих распарсенных статей, если хотя бы в одной из них у автора была указана аффилиация. Эта особенность позволяет классификатору со временем работать все точнее и точнее.

4.2.4. Работа с пользовательскими исправлениями

Каждой публикации, которая была скачана парсером, пользователи могут проставить вручную информацию о том, является ли текущая статья “интересной ТАСС” или нет. В пользовательском интерфейсе был создан специальный элемент (чек-бокс, см. рисунок ниже), который позволяет не только влиять на результат классификации данной статьи, но и отмечать публикации российских соотечественников за рубежом. Собранные данные будут использованы при разработке детектора статей от российских соотечественников за рубежом и последующего переобучения классификатора.

Пользовательские исправления хранятся в базе данных в виде отдельного флага в модели *Статья.* Этот флаг принимает три возможных значения:

1) Null - эта статья не была отмечена пользователями,
2) True - эта статья была отмечена пользователями как “интересная ТАСС”,

3) False - эта статья была отмечена пользователями как не “интересная ТАСС”.

Количества 10 000 пользовательских отметок достаточно для того, чтобы начать разработку модели, способной самой детектировать статьи российских соотечественников за рубежом. Исследования по подбору архитектуры такой системы детекции возможны только после того, как необходимый объем данных будет собран. По мере того, как данные о “интересных ТАСС” статьях будут накапливаться, планируется вручную прописывать некоторым авторам статус “интересен ТАСС”.

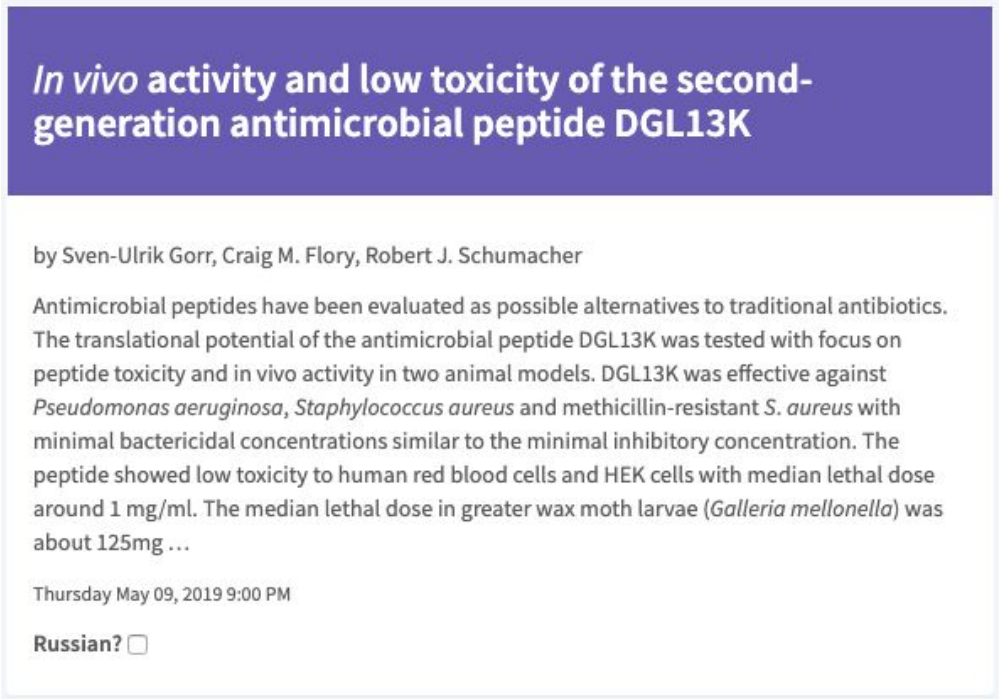

Рисунок 8: Пример редакторского интерфейса с резюме научной публикации и переключателем исправления результатов классификатора.

4.3. Опыт работы с заказчиком

4.3.1. Первое интервью

Задачи на интервью:

1) Выявить “боли” заказчика: с какими проблемами сталкивается компания при работе с текстами,
2) Предложить решение,
3) Определить первую итерацию продукта, техническое задание и дату его реализации.

После интервью была выявлена потребность TACC в решении по оперативному поиску англоязычных публикаций российских ученых. В рамках разработки прототипа, были зафиксированы 4 Интернет-журнала в качестве источников публикаций российских ученых: Arxiv [56], PLOS ONE [57], PLOS Biology [58], BioRxiv [74]. Для каждого из них впоследствии был разработан парсер, позволяющий извлекать максимальное набор метаданных о публикации до того, как PDF-файл отправлялся в GROBID на последующую обработку. Данные о новых публикациях из Интернет-журналов получаются по API или по RSS ленте. При этом сбор метаданных о публикации с веб-сайта собирается отдельными парсерами, написанными с помощью Beautiful Soup.

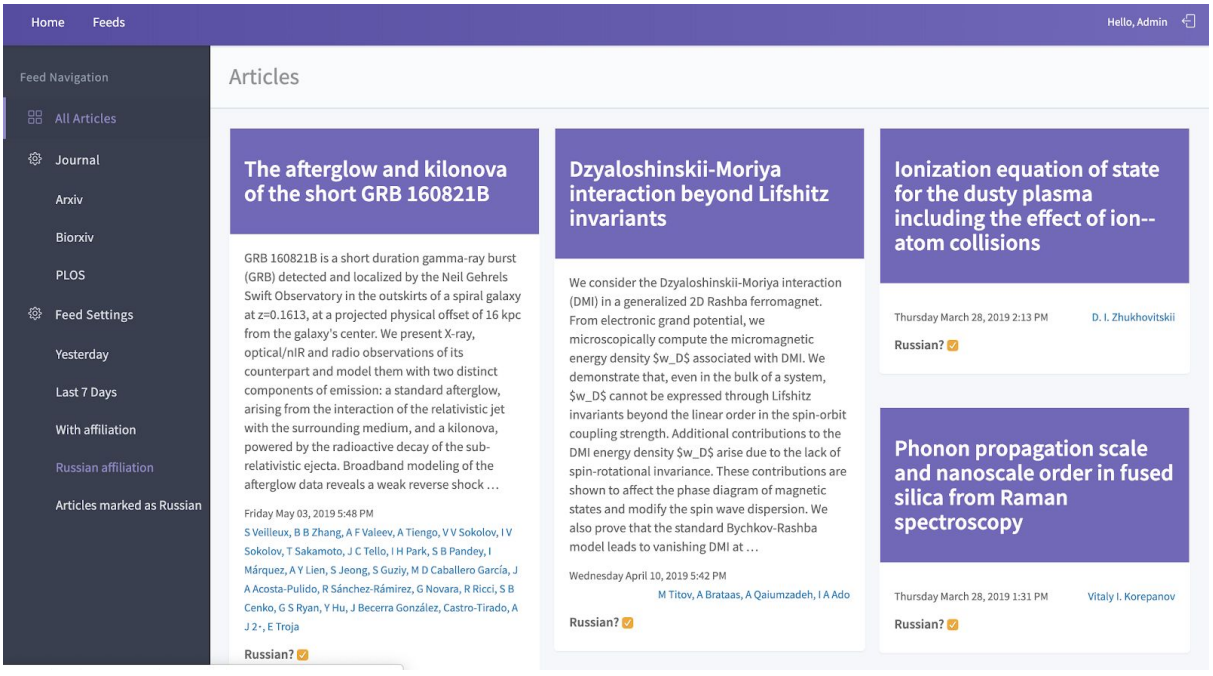

Рисунок 9: Внешний вид интерфейса: список найденных статей “интересных TACC”.

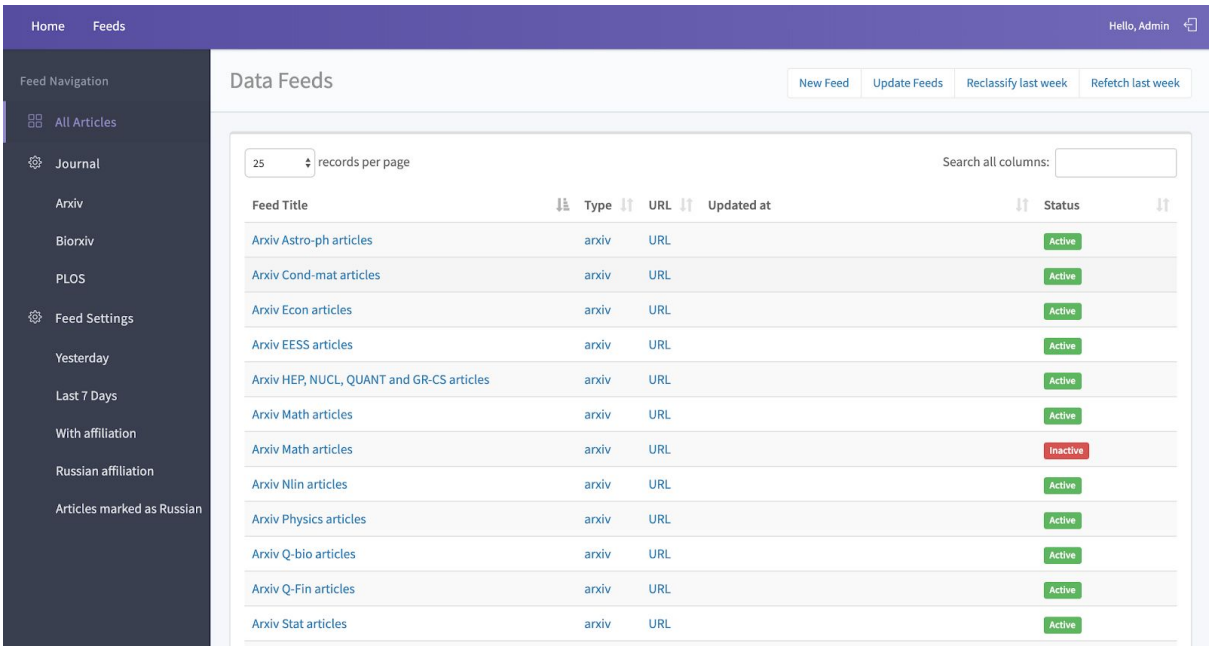

Рисунок 10: Внешний вид интерфейса: панель администратора со списком источников данных.

4.3.2. Второе интервью

Задачи интервью:

- Определить качество детекции,
- Выявить проблемы и предложить их решения на основе прочитанной литературы.

По результатам второго интервью с представителями ТАСС были выявлены следующий проблемы:

1. Несмотря на относительно большой поток статей в 4 выбранных источниках (суммарно в среднем 15 новых публикаций каждый день), из них задетектированных публикаций от российских учёных оказалось менее половины процента. Такое количество не удовлетворяет заказчику, так как, по его опыту, “интересных ТАСС” публикаций должно быть больше.

2. Добавление нового Интернет-журнала - дополнительного источника статей на входе фильтра - занимает много времени разработки: под каждый новый сайт

необходимо писать новый парсер, что заметно замедляет масштабируемость проекта. Данных, которые можно получить из PDF-файлов, может не хватать для поставленной задачи: в PDF-файлах редко упоминается время публикации статьи. Эта информация необходима ТАСС, чтобы публиковать новости только об актуальных исследованиях.

3. Была замечена некорректная работа парсера PDF-файлов GROBID в статьях, в которых был указан рецензент. Неверное срабатывание заключалось в том, что в некоторых случаях рецензент попадал в список авторов, и, если его аффилиация российская, вся публикация отмечалась как “интересная ТАСС”, в то время как авторы самой работы могли быть никак не связаны с Россией.

4. Разработанная система получает данные о новых публикациях в Интернет-журналах через API или RSS интерфейсы. Некоторые журналы, такие как Arxiv, для экономии ресурсов и предотвращения разных атак на их сервисы, кешируют выдачу новых научных работ: обновляют выдачу данных раз в несколько дней. Таким образом, новые научные публикации могут появляться в системе с большой задержкой.

5. При первичном запуске разработанного парсера IP адрес машины, на котором была запущена система, был заблокирован Интернет-порталом Arxiv из-за частых запросов на скачивание PDF-файлов для их дальнейшего парсинга системой GROBID.

Проанализировав выявленные проблемы, были предложены следующие решения:

1. Количество российских статей на выходе можно увеличить двумя способами:

1) Написать парсеры для большего количества Интернет-журналов, тем самым увеличить количество публикаций, которые подаются парсеру и фильтру.
2) Улучшить классификатор, который находит статьи “интересные ТАСС”.

В первом случае необходимо собрать список открытых Интернет-журналов с научными статьями (такие данные можно найти в Википедии [86] или на сайте DOAJ [10] - агрегаторе открытых журналов) и повторить итерацию интервью с ТАСС.

Во втором случае необходимо увеличить базу знаний: добавить больше аффилиаций и институтов, связанных с Россией. Также необходимо расширить список российских соотечественников за рубежом, так как их статьи тоже интересны ТАСС.

2. Сократить время разработки новых парсеров для новых Интернет-журналов можно пересмотром архитектуры: попытаться обобщить подход к экстракции данных из веб-сайтов. Одно из возможных решений - отказаться от извлечения данных с веб-страниц, оставив только сбор метаданных из PDF-файлов с помощью GROBID. В этом случае достаточно будет находить все ссылки на PDF-файлы на веб-страницах Интернет-журнала. Негативные стороны этого подхода состоят в увеличении нагрузки на серверы и времени обработки научных публикаций, а также в невозможности определения некоторых необходимых метаданных, например, дате публикации статьи. Варианты подобного масштабирования планируется обсудить с заказчиком на ближайшем интервью.

3. Открытая система GROBID продолжает развиваться и активно поддерживается сообществом GitHub. Все найденные неточности в работе были донесены до разработчиков. В случаях задержки исправления ошибок в коде системы, возможна интеграция с другим комплексным парсером PDF-файлов, например, CERMINE.

4. Проблема с задержками в обновлениях данных, доступных по API или RSS каналам, может решаться в написании более детального парсера для каждого Интернет-издательства. Текущий парсер веб-сайтов умеет извлекать метаданные только со страницы, где была опубликована сама научная публикация. Но если дополнить его функционал возможностью выдавать список последних опубликованных статей, то можно отказаться от использования API и RSS каналов, однако не все веб-сайты могут иметь веб-страницу с последними опубликованными

статьями. Данный подход требует гораздо больше времени на разработку и не коррелирует с решением проблемы 2.

5. Для решения проблемы блокировки IP адресов из-за частых запросов используют прокси - промежуточные серверы, которые перенаправляют весь входящий и исходящий трафик. В случае парсинга открытых Интернет-изданий отлично подойдет сеть TOR в качестве промежуточного звена между центральным сервером парсера и сервером Интернет-журнала с научными публикациями. На момент написания данной работы, решение на основе TOR сети было имплементировано и показало свою работоспособность и применимость к задачам парсинга.

5. Возможные способы создания продукта на основе разработанной технологии

5.1. Горизонтальное масштабирование

Возможностное горизонтальное масштабирование заключается в выходе на рынки других неанглоязычных стран, то есть искать публикации местных ученых среди публикаций на английском языке. Для этого необходимо дополнить базу знаний “институт - страна” примерами характерными для других неанглоязычных стран. Эти данные можно собрать из Википедии аналогично тому, как получали данные для поиска публикаций российских ученых. По мере внедрения к клиенту и последующего развития продукта, предлагается также пополнять базу российских соотечественников за рубежом, собирая пользовательский ввод для обучения детектора их публикаций. Архитектурных изменений система не претерпит.

Искать потенциальных клиентов при горизонтальном масштабировании планируется в крупных локальных СМИ.

5.1.1. Канва бизнес-модели Остервальдера для горизонтального масштабирования

1. Key Propositions
    a. Поиск и агрегация англоязычных публикаций российских ученых и соотечественников за рубежом
2. Key Activities
    a. Обучение и улучшение классификатора и парсера
    b. Разработка платформы
    c. Поиск и удержание клиентов
3. Key Resources
    a. Специалисты в Data Science и Software Development

    b. Обученные нейронные сети и методы их обучения
    c. Корпусы размеченных научных публикаций
    d. Платформа в интернете и серверы
4. Customer relationships
    a. Использование услуг
    b. Реализация крупных контрактов
    c. Техническая поддержка
    d. Индивидуальные решения для крупных клиентов
5. Customer Segments
    a. Научные Сми
    b. Научные Администрации
6. Key Partners
    a. Агрегаторы научных публикаций
    b. Научные Интернет-издательства.
    c. Серверы облачных вычислений и хостинга
7. Channels
    a. Software-as-a-Service - платформа в интернете
8. Revenue Streams
    a. Продажа лицензии для доступа к платформе
    b. Несколько типов лицензий для разделения клиентов по потребностям и их величине
    c. Продажа уникальных фич, необходимых конкретному клиенту по согласованию
9. Cost Structure
    a. Использование вычислительных мощностей и хостинг
    b. Разработка и развитие платформы в интернете и нейронных сетей
    c. Накладные расходы

5.2. Вертикальное масштабирование

Возможное вертикальное масштабирование заключается в тегировании не только потока научных статей, но и других данных. По аналогии с потоком научных

публикаций, можно начать агрегировать данные другого рода и начать обучать новый классификатор под новых тип данных. В общем случае, парсер будет иметь схожую архитектуру, а алгоритм экстракции данных будет состоять из этапов:

1) Получение списка новых объектов,
2) Обогащение данных о каждом новом объекте дополнительным парсингом,
3) Тегирование каждого объекта,
4) Приведение собранных данных в единый формат для дальнейшего анализа и визуализации.

Собирая пользовательские исправления некорректной работы алгоритма тегирования, можно автоматически перестраивать классификатор и улучшать его работу даже в случае добавления новых вводных.

В итоге под каждого клиента, которому необходимо периодически агрегировать и классифицировать данные из разных источников, создается отдельная база данных и браузерный интерфейс-дэшборд, в котором пользователи могут

a) Анализировать результаты парсинга и классификации данных,
b) Оставлять отзывы о качестве классификатора, которые будут использоваться для переобучения и улучшения его качества.

5.2.1. Обнаружение текстовых манипуляций

Спроектированный в рамках решения задачи поиска англоязычных научных публикаций российских ученых движок может также быть применен в задаче поиска манипуляций в текстах. Идея состоит в том, чтобы находить и выделять в текстах потенциально опасные места, которые могут ввести читателя в заблуждения, так как он по умолчанию доверяет написанной статье.

Был разработан прототип, позволяющий обнаруживать в англоязычном тексте манипуляции:

1) Излишние обобщения. Пример манипуляции: “**Все** чиновники воруют”.

2) Эмоционально выделенные слова. Пример манипуляции: “**Чиновники** воруют” - слово “воруют” несет в себе негативный подтекст.
3) Главные тезисы в тексте. Алгоритм подчеркивает предложения, которые должны быть доказаны в тексте или ссылками на источники, если этого нет - читателю предлагается проверить подлинность высказываний самостоятельно.

В рамках прототипа для поиска первой манипуляции была составлена база знаний слов обобщений (компиляция англоязычных словарей). Излишне эмоциональные слова детектируются с использованием алгоритмов сентимент анализа Google Cloud Natural Language API [87], а главные тезисы находятся с помощью алгоритмов сжатия текста, которые и позволяют выделить самые важные предложения текста.

Рисунок 11: Экран ввода ссылки на публикацию в интернете. Кнопка "Random Article" в демонстрационных целях найдет случайную публикацию , используя сервис NewsAPI [88].

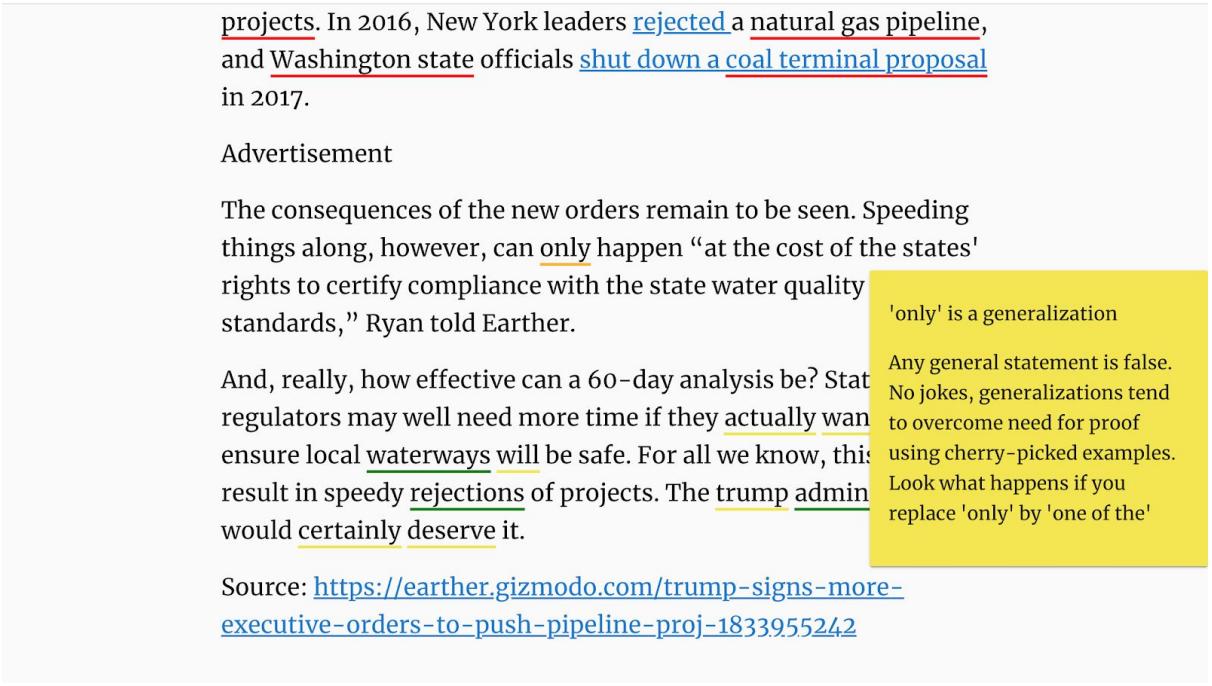

Рисунок 12: Пример работы прототипа: выделение найденного обобщения в публикации.

Предложение о создании стартапа для обнаружения текстовых манипуляций было отправлено осенью 2018 года в акселератор YCombinator [89]. С ней команда проекта прошла до этапа видеоинтервью. Из-за неопределенности в продуктовой части проекта этап не был пройден. Осенью 2019 года планируется доработать бизнес-модели и повторно отправить заявку в акселератор YCombinator.

Возможные применения технологии поиска манипуляций в текстовых материалов:

1. Инструмент для очистки формальных текстов от возможных манипуляций: поиск эмоций там, где они точно не нужны, например, в юридических документах, в договорах.

2. Обучение других людей искать манипуляции в текстах или умело ими пользоваться. Проводить тренинги для отделов продаж компаний, журналистов, психологов.

3. Ранжирование источники информации и трендовые темы по среднему уровню манипулятивности текстов. Детектировать манипулятивные / покупные статьи в интернете.

Разработанный движок экстракции метаданных из Интернет-публикаций также можно применять в потоковой классификации комментариев, которые оставляют под публикациями на сайте потенциального клиента или в его соцсетях. Исходя из требований заказчика, можно индексировать настроения комментаторов или упоминания тех или иных сущностей - компаний, людей, событий.

5.2.2. Канва бизнес-модели Остервальдера для вертикального масштабирования

1. Key Propositions
    a. Очистка формальных текстов от эмоций
    b. Тренинги для отделов продаж, журналистов
    c. Ранжирование СМИ по уровню манипулятивности
2. Key Activities
    a. Обучение и улучшение классификатора и парсера
    b. Разработка платформы
    c. Поиск и удержание клиентов
3. Key Resources
    a. Специалисты в Data Science и Software Development
    b. Обученные нейронные сети и методы их обучения
    c. Корпусы размеченных публикаций в СМИ
    d. Платформа в интернете и серверы
4. Customer relationships
    a. Использование услуг
    b. Реализация крупных контрактов
    c. Техническая поддержка
    d. Индивидуальные решения для крупных клиентов
5. Customer Segments
    a. Отделы продаж

    b. Журналисты
    c. Юристы
    d. СМИ
6. Key Partners
    a. Агрегаторы новостных статей
    b. СМИ
    c. Серверы облачных вычислений и хостинга
7. Channels
    a. Software-as-a-Service - платформа в интернете
    b. Доступ к функционалу через API интерфейс
8. Revenue Streams
    a. Продажа лицензии для доступа к платформе и к API
    b. Несколько типов лицензий для разделения клиентов по потребностям и их величине
    c. Продажа уникальных фич, необходимых конкретному клиенту по согласованию
9. Cost Structure
    a. Использование вычислительных мощностей и хостинг
    b. Разработка и развитие платформы в интернете и нейронных сетей
    c. Накладные расходы

6. Заключение

После анализа технической и деловой литературы была выявлена потребность системе автоматического структурирования и классификации научных публикаций. Была разработана инновационная и научно ценная система, которая позволяет агрегировать и тегировать потоки научных статей по принадлежности авторов статей к той или иной стране.

Архитектурное решение по экстракции метаданных из опубликованных в интернете объектов обладает огромным потенциалом как к горизонтальному, так и к вертикальному масштабированию. Общность архитектурного решения позволяет перенести уже готовую инфраструктуру сбора данных на другие задачи, в частности, задачу детекции манипуляций в СМИ.

Актуальность проблемы не означает, что клиент готов выделять свое время на поиск и внедрение нового решения. Клиенту необходимо инвестировать свои время и усилия на пилотное внедрение, на обучение персонала и тп. Если нет возможности выделять дополнительные ресурсы - инновации не внедряются. Для преодоления барьера нужны усилия не только со стороны инноватора, но и от клиента.

7. Приложение 1. - Перечень используемых публикаций и источников

[1] Петреченко В.А. и др. “Аналитическое исследование “Международные и Российские практики работы с научной диаспорой. Модели для России”. Москва, 2016.

[2] Jensen, Carlos, et al. "Tracking website data-collection and privacy practices with the iWatch web crawler." *Proceedings of the 3rd symposium on Usable privacy and security*. ACM, 2007.

[3] Massand, Deepak. "System and method for reflowing content in a structured portable document format (pdf) file." U.S. Patent Application No. 12/413,486.

[4] Blum, Scott B., and Jonathan Lueker. "Transparent proxy server." U.S. Patent No. 6,182,141. 30 Jan. 2001.

[5] Copeland, Bruce W., and Jonathan I. Shuval. "Computer-based uniform data interface (UDI) method and system using an application programming interface (API)." U.S. Patent No. 5,815,703. 29 Sep. 1998.

[6] Liu, Hongzhou, Venugopalan Ramasubramanian, and Emin Gün Sirer. "Client behavior and feed characteristics of RSS, a publish-subscribe system for web micronews." *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. USENIX Association, 2005.

[7] Research Trends (2008) “English as the international language of science”, Research Trends, Issue 6, July 2008.

[8] Jinha, Arif. (2010). Article 50 million: An estimate of the number of scholarly articles in existence. Learned Publishing. 23. 258-263. 10.1087/20100308.

[9] Ware, Mark, and Michael Mabe. "The STM report: An overview of scientific and scholarly journal publishing." (2015).

[10] Morrison, Heather. directory of Open access Journals (dOaJ). Diss. University of British Columbia, 2008.

[11] Beagrie, Neil. "Digital curation for science, digital libraries, and individuals." International Journal of Digital Curation 1.1 (2008): 3-16.

[12] Bollen, Johan, and Herbert Van de Sompel. "An architecture for the aggregation and analysis of scholarly usage data." Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries. ACM, 2006.

[13] Tkaczyk, Dominika, et al. "CERMINE--Automatic Extraction of Metadata and References from Scientific Literature." 2014 11th IAPR International Workshop on Document Analysis Systems. IEEE, 2014.

[14] Lopez, Patrice. "GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications." *International conference on theory and practice of digital libraries*. Springer, Berlin, Heidelberg, 2009.

[15] Falagas, Matthew E., et al. "Comparison of SCImago journal rank indicator with journal impact factor." *The FASEB journal*22.8 (2008): 2623-2628.

[16] D. Gupta, B. Morris, T. Catapano and G. Sautter, "A new approach towards bibliographic reference identification, parsing and inline citation matching," in International Conference on Contemporary Computing, 2009.

[17] A. Constantin, S. Pettifer and A. Voronkov, "PDFX: fully-automated pdf-toxml conversion of scientific literature," ACM Symposium on Document Engineering, pp. 177-180, 2013

[18] M.-Y. Day, R.-H. Tsai, C.-L. Sung, C.-C. Hsieh, C.-W. Lee, S.-H. Wu, K.-P. Wu, C.-S. Ong and W.-L. Hsu, "Reference metadata extraction using a hierarchical knowledge representation framework," Decision Support Systems, vol. 43, no. 1, pp. 152-167, 2007.

[19] E. Cortez, A. S. da Silva, M. A. Gonçalves, F. de Sá Mesquita and E. S. de Moura, "A flexible approach for extracting metadata from bibliographic citations," JASIST, vol. 60, no. 6, pp. 1144-1158, 2009.

[20] E. Hetzner, "A simple method for citation metadata extraction using hidden markov models," in Joint Conference on Digital Libraries, Pittsburgh, 2008.

[21] P. Yin, M. Zhang, Z.-H. Deng and D. Yang, "Metadata Extraction from Bibliographies Using Bigram HMM," in International Conference on Asian Digital Libraries, 2004.

[22] B. A. Ojokoh, M. Zhang and J. Tang, "A trigram hidden Markov model for metadata extraction from heterogeneous references," Inf. Sci., vol. 181, no. 9, pp. 1538-1551, 2011.

[23] I. Councill, C. Giles and M.-Y. Kan, "ParsCit: an open-source CRF reference string parsing package," in International Conference on Language Resources and Evaluation, 2008.

[24] P. Lopez, "GROBID: combining automatic bibliographic data recognition and term extraction for scholarship publications," Research and Advanced Technology for Digital Libraries, pp. 473-474, 2009.

[25] Lipinski, Mario, et al. "Evaluation of header metadata extraction approaches and tools for scientific PDF documents." Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries. ACM, 2013.

[26] D. Tkaczyk, P. Szostek, M. Fedoryszak, P. Dendek and L. Bolikowski, "CERMINE: automatic extraction of structured metadata from scientific literature," International Journal on Document Analysis and Recognition, vol. 18, no. 4, pp. 317-335, 2015.

[27] D. Tkaczyk and L. Bolikowski, "Extracting Contextual Information from Scientific Literature Using CERMINE System," in Semantic Web Evaluation Challenges - Second SemWebEval Challenge at ESWC, 2015..

[28] D. Matsuoka, M. Ohta, A. Takasu and J. Adachi, "Examination of effective features for CRF-based bibliography extraction from reference strings," in International Conference on Digital Information Management, 2016.

[29] Q. Zhang, Y. Cao and H. Yu, "Parsing citations in biomedical articles using conditional random fields," Comp. in Bio. and Med., vol. 41, no. 4, pp. 190-194, 2011.

[30] J. Zou, D. X. Le and G. R. Thoma, "Locating and parsing bibliographic references in HTML medical articles," IJDAR, vol. 13, no. 2, pp. 107-119, 2010.

[31] X. Zhang, J. Zhou, D. X. Le and G. R. Thoma, "A structural SVM approach for reference parsing," BMC Bioinformatics, vol. 12, no. S-3, p. S7, 2011.

[32] Y.-M. Kim, P. Bellot, J. Tavernier, E. Faath and M. Dacos, "Evaluation of BILBO reference parsing in digital humanities via a comparison of different tools," in ACM Symposium on Document Engineering, 2012.

[33] D. Tkaczyk, P. Szostek, M. Fedoryszak, P. Dendek and L. Bolikowski, "CERMINE: automatic extraction of structured metadata from scientific literature," International Journal on Document Analysis and Recognition, vol. 18, no. 4, pp. 317-335, 2015.

[34] P. Lopez, "GROBID: combining automatic bibliographic data recognition and term extraction for scholarship publications," Research and Advanced Technology for Digital Libraries, pp. 473-474, 2009.

[35] I. Councill, C. Giles and M.-Y. Kan, "ParsCit: an open-source CRF reference string parsing package," in International Conference on Language Resources and Evaluation, 2008.

[36] Tkaczyk, Dominika, et al. "Machine learning vs. rules and out-of-the-box vs. retrained: an evaluation of open-source bibliographic reference and citation parsers." *arXiv preprint arXiv:1802.01168* (2018).

[37] Патент номер US6941264B2 “Retraining and updating speech models for speech recognition” США, 2001.

[38] Патент номер US6374221B1 “Automatic retraining of a speech recognizer while using reliable transcripts” США, 1999.

[39] Tkaczyk, Dominika. "New Methods for Metadata Extraction from Scientific Literature." arXiv preprint arXiv:1710.10201(2017)

[40] Wikipedia contributors. "List of institutions of higher education in Russia." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 3 May. 2019. Web. 11 May. 2019.

[41] Seeger, Marc. "Building blocks of a scalable web crawler." Master's thesis, Stuttgart Media University (2010)

[42] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[43] Han, Hui, et al. "Automatic document metadata extraction using support vector machines." *2003 Joint Conference on Digital Libraries, 2003. Proceedings.*. IEEE, 2003.

[44] Banko, Michele, and Eric Brill. "Mitigating the paucity-of-data problem: Exploring the effect of training corpus size on classifier performance for natural language processing." *Proceedings of the first international conference on Human language technology research*. Association for Computational Linguistics, 2001.

[45] Ko, Youngjoong, and Jungyun Seo. "Automatic text categorization by unsupervised learning." *Proceedings of the 18th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 2000.

[46] Zhang, Xiang, Junbo Zhao, and Yann LeCun. "Character-level convolutional networks for text classification." *Advances in neural information processing systems*. 2015.) (Howard, Jeremy, and Sebastian Ruder. "Universal language model fine-tuning for text classification." *arXiv preprint arXiv:1801.06146* (2018).

[47] Pranckevičius, Tomas, and Virginijus Marcinkevičius. "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification." *Baltic Journal of Modern Computing* 5.2 (2017): 221-232.

[48] Joulin, Armand, et al. "Bag of tricks for efficient text classification." *arXiv preprint arXiv:1607.01759* (2016).

[49] Wuest, Thorsten, Christopher Irgens, and Klaus-Dieter Thoben. "An approach to monitoring quality in manufacturing using supervised machine learning on product state data." *Journal of Intelligent Manufacturing* 25.5 (2014): 1167-1180.

[50] Wuest, Thorsten, Christopher Irgens, and Klaus-Dieter Thoben. "An approach to monitoring quality in manufacturing using supervised machine learning on product state data." *Journal of Intelligent Manufacturing* 25.5 (2014): 1167-1180.

[51] Richardson, Leonard. "Beautiful soup documentation." April(2007)) (Nair, Vineeth G. Getting Started with Beautiful Soup. Packt Publishing Ltd, 2014.

[52] Mitchell, Ryan. Web Scraping with Python: Collecting More Data from the Modern Web. " O'Reilly Media, Inc.", 2018.

[53] A. Di Iorio, C. Lange, A. Dimou and S. Vahdati, "Semantic Publishing Challenge - Assessing the Quality of Scientific Output by Information Extraction and Interlinking," in SemWebEval@ESWC, 2015.

8. Приложение 2. - Перечень используемых Интернет-ресурсов

[54] https://tass.ru/tass-today

[55] https://trv-science.ru/2017/07/04/nauchnaya-diaspora-rossii-nuzhna/

[56] https://arxiv.org

[57] https://journals.plos.org/plosone/

[58] https://journals.plos.org/plosbiology/

[59] https://journals.plos.org/

[60] https://researchgate.net

[61] https://scholar.google.ru

[62] https://github.com/inukshuk/anystyle-parser

[63] http://search.cpan.org/~mjewell/Biblio-Citation-Parser-1.10/

[64] https://github.com/nishimuuu/citation

[65] https://github.com/manishbisht/Cittion-Parser

[66] https://github.com/miriam/free_cite

[67] https://github.com/opensourceware/Neural-ParsCit

[68] https://github.com/rmcgibbo/reftagger

[69] https://github.com/CeON/CERMINE

[70] https://github.com/kermitt2/grobid

[71] https://github.com/knmnyn/ParsCit

[72] https://github.com/eliask/pdfssa4met

[73] https://github.com/allenai/science-parse

[74] https://biorxiv.org

[75] https://github.com/zet4/alpine-tor

[76] https://metrics.torproject.org/networksize.html

[77] https://towardsdatascience.com/a-practical-guide-to-collecting-ml-datasets-476f1ecf5e35

[78] https://phys.msu.ru/eng/staff/all/?ID=1448

[79] https://academia.stackexchange.com/questions/93764/how-long-does-it-usually-take-for-published-articles-to-show-up-on-google-schola

[80] https://tass.ru/nauka/5377368

[81] https://metrika.ya.ru

[82] https://www.djangoproject.com/

[83] https://www.postgresql.org/

[84] http://www.celeryproject.org/

[85] https://vuejs.org/

[86] https://en.wikipedia.org/wiki/List_of_open-access_journals

[87] https://cloud.google.com/natural-language/

[88] https://newsapi.org/

[89] https://ycombinator.com