Интеллектуальная информационная система поддержки принятия судебных решений в сфере экономического правосудия

А.А. Алексеев[1], Д.С. Зуев[2], А.С. Катасёв[1], Е.В. Тутубалина[2], А.Ф. Хасьянов[2]

1 Казанский национальный исследовательский технический университет им А.Н. Туполева – КАИ
2 Казанский (Приволжский) федеральный университет

Аннотация. Описана архитектура системы интеллектуального анализа текстов в юриспруденции, которая способна помечать важные места, на которые следует обращать внимание при процессуальных действиях с использованием инструментов текстовой аналитики. Создана модель для извлечения из корпуса юридических текстов значимых сущностей и фактов. На корпусе юридических текстов проведено обучение алгоритма автоматического извлечения сущностей на базе рекуррентной нейронной сети. В результате было выявлено, что рекуррентная нейронная сеть предсказывает сущности, которые не были размечены вручную либо размечены некорректно и требуется проверка разметки. Важной функцией системы является поиск и предоставления аналогичных решений по схожим судебным искам. Для правильного определения категории судебного спора решается задача классификации. Было выявлено более 80 различных категорий судебных споров, которые встречаются с разной частотой. Процесс классификации с ростом количества обрабатываемых документов может быть очень затратным по времени, поэтому данная функциональность реализуется отдельный микросервис с обменом с другими модулями системы в асинхронном режиме. Отдельным модулем системы представлен сайт – шаблонизатор исковых заявлений, позволяющий корректно формировать типовые исковые заявления**.** Разработка системы ведется вместе с практикующими юристами, в качестве специалистов из предметной области привлекаются судьи Арбитражного суда Республики Татарстан, преподаватели и обучающиеся юридического факультета Казанского федерального университета.

Ключевые слова: аналитика и управление данными, интенсивное использование данных, электронные библиотеки, кластеризация, рекомендательная система.

Intellectual information decision support system in the field of economic justice

A.A. Alekseev[1], A.S. Katasev[1], A.F. Khassianov[2], E.V. Tutubalina[2], D.S. Zuev[2]

1 Kazan National Research Technical University
2 Kazan Volga Region Federal University

Annotation. The architecture of the system for the intellectual analysis of texts in jurisprudence, which is capable of marking important places to which attention should be paid during procedural actions using text analytics tools, is described. To extract meaningful entities and facts from a corpus of legal texts we created a model. An algorithm for automatic extraction of entities based on a recurrent neural network was trained on the corpus of legal texts. As a result, it was revealed that the recurrent neural network predicts entities that were not manually labeled or incorrectly labeled and markup checking is required. An important function of the system is to search for and provide similar solutions for similar lawsuits. For the correct determination of the category of lawsuit, the task of classification of entities is solved. More than 80 different categories of lawsuits identified, occurring with varying frequency. The classification process with an increase in the number of processed documents can be very time consuming, so this functionality is implemented by a separate microservice with integration with other modules of the system in asynchronous mode. A separate module of the system is a website - template of claim statements, which allows to filling correctly typical claims. The development of the system is carried out together with practicing lawyers, as experts from the subject area. Judges of the Arbitration Court of the Republic of Tatarstan, teachers and students of the law faculty of Kazan Federal University are also involved.

Keywords: data analytics and data management, data intensive domains, digital libraries, recommender system, decision support systems.

Как известно, информационное общество характеризуется высоким уровнем развития информационно-коммуникационных технологий (ИКТ) и их интенсивным использованием всеми и всюду. Развитие облачных технологий позволило принципиально изменить подходы к созданию сложных программных систем практически для всех предметных областей. Хотя в области судопроизводства в России постоянно повышается уровень использования информационных технологий, но, тем не менее, информационная и производственная нагрузки на судей по-прежнему остаются чрезмерно высокими. Без использования специализированных автоматизированных информационных систем существенное повышение эффективности работы судов просто невозможно.

Одним из приложений технологий машинного обучения в судопроизводстве является создание интеллектуальных систем, способных на имеющейся базе данных судебных документов выявлять общие зависимости, предоставлять судьям для ознакомления близкие по тематике дела, рекомендовать наиболее вероятные исходы или помечать важные места, на которые судебным работникам следует обращать внимание при процессуальных действиях.

В [1] описаны онтологии и особенности работы с юридическими документами. Есть успешные реализации подобных информационных систем за рубежом. Примером служит система «Case Cruncher Alpha» [2], разрабатываемая в Sidney Sussex College, Cambridge и ориентированная на прогнозирование ре-

шения юридических задач в банках, страховых компаниях и юридических консультациях. Основной ее недостаток, как и многих иностранных систем – отсутствие поддержки русского языка и кириллической транскрипции.

Существующие решения, используемые в юридической области, направлены либо на автоматизацию документооборота в целом, либо представляют собой широчайшие базы данных тематических документов. Поиск необходимой информации не всегда представляется возможным в сжатые сроки, а весь спектр семантических технологий и инструментария текстовой аналитики практически не используется. Увеличение количества дел, рассматриваемых судами, невозможно без качественного изменения эффективности работы судей или существенного увеличения числа сотрудников судебных органов.

2. Цели и задачи создания системы

«Робот-юрист» – это информационная система, которая должна позволять участникам юридического процесса правильно проводить подготовку судебных дел и осуществлять планирование судебной деятельности. Цель системы – помочь определить характер спора, осуществить поиск и проверку действия правовых норм, регулирующих спорные правоотношения, оказывать содействие в установлении компетентного суда (подсудность, подведомственность), статуса участников спора (действующее, ликвидированное, банкрот), определении круга обстоятельств, имеющих значение для рассмотрения спора, характера спорного правоотношения, нормы права, подлежащей применению (действует ли данная норма), а также проверять достаточность и комплектность представляемых документов. Как отдельные функции запланированы обеспечение возможности оформления искового заявления, а также вычисление (по предоставленным исходным данным на основе архива судебных дел) вероятности принятия того либо иного решения.

Проведенные исследования по семантическому структурированию информации в других предметных областях (см., например, [3, 4]), анализ инструментов текстовой аналитики (см, например, [5]) и наработки по применению семантических технологий при работе с юридическими документами [1] говорят о принципиальной реализуемости поставленной задачи.

Разработка системы ведется вместе с практикующими юристами, в частности, в качестве специалистов из предметной области привлекаются судьи Арбитражного суда Республики Татарстан, преподаватели и обучающиеся юридического факультета Казанского федерального университета.

Для достижения поставленных целей поставлены следующие задачи.

- создание портала для формирования шаблонов исковых заявлений с отслеживанием их жизненного цикла;
- разметка и анализ существующей базы судебных решений, исковых заявлений (отбор значимых признаков для определенных категорий судебного спора, классификация заявлений и решений, извлечение сущностей и фактов);

- подбор аналогичных дел и решений, рекомендательный сервис;
- сопоставление исковых заявлений и судебных решений;
- распределение судебных дел между судьями с учетом их специализации и текущей загрузки.

Фактически каждая из выделенных задач является автономным модулем разрабатываемой информационной системы, а сама система – практической демонстрацией совместного использования ряда семантических технологий и инструментов текстовой аналитики.

Текущие парадигмы разработки предусматривают два концептуально различных подхода к дизайну приложений. Первый вариант – «монолитные приложения». Это довольно очевидный способ построения информационных систем, в которой запросы обрабатываются в рамках единственного процесса, при этом используются возможности конкретного языка программирования для разделения приложения на классы и функции. Подобный подход порождает ряд проблем: любые изменения, даже самые небольшие, требуют перекомпиляции всего дистрибутива информационной системы и последующего обновления всех ее модулей, с течением времени изменения в логике работы одного модуля начинают влиять на функции других модулей, возникают проблемы с масштабированием приложения.

Другой подход – это построение среды, в которой отдается предпочтение слабым связям, абстрагированию низкоуровневой логики, гибкости, а также возможности многократного использования и обнаружения компонентов [6, 7], сервис-ориентированной архитектуре (Service-Oriented Architecture, SOA). Сервисы – это программные компоненты, предоставляющие четко определенную функциональность и используемые в составе многих приложений. Каждый сервис представляет собой изолированную сущность с минимумом зависимостей от других совместно используемых ресурсов: баз данных, традиционных приложений и интерфейсов программирования. Таким образом, возникает возможность изменять отдельные сервисы, не затрагивая при этом всю систему. Развитием парадигмы сервис-ориентированной архитектуры можно считать появление архитектуры микросервисов [8]. Термин «Microservice Architecture» получил распространение в последние несколько лет для описания способа проектирования приложений в виде набора независимо развертываемых сервисов.

С учетом достаточно большого количества модулей системы необходимо было выбрать подход к организации всего приложения и минимизировать зависимости, связанные с изменениями внутри отдельных модулей. При этом очевидно, что модули текстовой аналитики со временем будут изменяться, возможна реализация различных алгоритмов классификации и аналитики в зависимости от массива обрабатываемых документов. Для создания «Робота-юриста» нами был выбран архитектурный стиль микросервисов. Архитектура разрабатываемой системы приведена в [9], где выделено несколько групп сервисов, взаимодействующих между собой с помощью программного интерфейса (API).

Ниже представлены основные подсистемы и сервисы разрабатываемой системы.

3. Сервисы системы

Разметка документов и извлечение сущностей. Для построения зависимостей и извлечения сущностей и фактов создан сервис разметки и анализа документов и выделено несколько подзадач:

- разметка существующего массива документа для поиска зависимостей и построения модели, списка справочников служебной онтологии;
- извлечение сущностей и фактов;
- запись в базу данных и отображение массива документов с выделенными сущностями для дальнейшей обработки.

Разметка существующего массива документов необходима для дальнейшего обучения сервисов системы. Для реализации этой задачи использовался инструмент для быстрого структурированного аннотирования текстов BRAT [10].

Для разметки документов были привлечены специалисты из предметной области, размечено около 3000 судебных актов с ручным выделением значимых сущностей. Необходимо заметить, что процесс разметки документов весьма затратен по времени, хотя и не требует глубокого знания предметной области и вполне по силам студентам старших курсов профильных факультетов. Однако надо понимать, что точность алгоритма извлечения сущностей зависит от количества выборки и качества разметки текстов.

На данном корпусе текстов проведено обучение алгоритма автоматического извлечения сущностей на базе рекуррентной нейронной сети. Общий массив размеченных документов был разделен на обучающую и тестовую выборки в соотношении 80% на 20%, что соответствует общей практике решения подобных задач. В результате на текущем шаге было выявлено, что рекуррентная нейронная сеть предсказывает сущности, которые не были размечены вручную либо размечены некорректно. Таким образом, были подготовлены данные, требующие дальнейшего уточнения. На данный момент производятся повторная проверка размеченных документов и корректировка для дальнейшего обучения сети.

Рекомендательный сервис. Этот сервис предназначен для поиска и предоставления аналогичных решений по схожим судебным искам.

Существуют два основных типа рекомендательных систем: контент-ориентированные и социальные (коллаборативной фильтрации) (см., например, [11]). Первые основаны на представлении предпочтений пользователей путем анализа содержимого рекомендательных элементов. Системы второго типа моделируют предпочтения, оценивая близость профилей пользователей. Ниже под рекомендательным сервисом будем понимать информационную систему, которая:

1) формирует модель предметной области на основе массива документов (включая подготовительные операции – приведение к векторному виду, кластеризацию и т. п.);

2) получает на вход документ и выдает список документов, близких к входному.

По сравнению с поисковыми системами рекомендательные системы наиболее полезны, когда у пользователя возникают трудности с формулировкой эффективного поискового запроса.

Подходы к организации рекомендательных сервисов могут быть разными, в [3] описан подход с использованием онтологий и предпочтений пользователей. Учитывая специфику предметной области и разрабатываемой системы, использовать предпочтения пользователей не корректно.

Алгоритм работы сервиса можно разделить на два этапа:

- подготовительный этап – обработка массива документов и обучение модели;
- основной этап – получение аналогов входного текста по заданному идентификатору документа.

На подготовительном этапе обрабатываются все имеющиеся документы: вырезаются знаки пунктуации, термы приводятся к единому виду (для слов с разными окончаниями и суффиксами). Далее документ приводится к векторному виду. Для представления массива документов в виде числовых векторов, отражающих важность использования каждого слова из некоторого набора слов (количество слов набора определяет размерность вектора), в каждом документе используется мера TF-IDF [5, 11]. На основе массива векторов происходит кластеризация. Напомним, что кластеризация [5, 12] – это процесс разбиения множества объектов на группы, которые заранее неизвестны. В результаты члены каждой группы должны быть похожи друг на друга по признакам разбиения и отличаться от членов других групп. Такие группы называются кластерами.

На первом шаге необходимо определить количество K кластеров, логичным представляется использовать для этого формулу $K = \frac{N_{doc}}{10}$, где N_{doc} – общее количество обрабатываемых документов. Далее производится собственно кластерный анализ по методу *K-means* (метод К-средних, [5, 13]). Полученные результаты сохраняются для дальнейшего использования.

На основном этапе работы на вход сервису подается идентификатор документа. Производится приведение его к векторной форме, которая обрабатывается моделью, причисляется к определенному кластеру. На выход алгоритм выдает первые 10 документов из того же кластера, что и входной документ, хотя данный параметр является настраиваемым и может быть изменен в настройках сервиса.

Процесс переобучения модели следует проводить периодически, например, раз в сутки, либо после существенного изменения всего корпуса документов.

Сервис реализован на языке Python, взаимодействие с другими модулями системы происходит по внутреннему согласованному протоколу взаимодействия.

Классификация судебных дел. Одной из проблем судебного делопроизводства является процедура определения категории и характера спора. Правильное определение категории судебного спора важно, поскольку влияет на назначение судьи на соответствующий процесс, а назначаемый судья должен иметь опыт рассмотрения подобных споров, знать и понимать их особенности. На текущий момент выявлено более 80 различных категорий судебных споров, которые встречаются с разной частотой. Процесс классификации с ростом количества обрабатываемых документов может быть очень затратным по времени, поэтому с архитектурной точки зрения было решено вынести данную функциональность как отдельный микросервис с реализацией обмена с другими модулями системы в асинхронном режиме. К тому же определение категории спора (судебного дела) не является задачей, требующей мгновенного ответа.

Для первого этапа реализации системы было принято решение провести анализ судебных документов по четырем категориям: оспаривание решений антимонопольных органов, оспаривание действий судебных приставов, привлечение к ответственности за нарушение условий лицензирования, споры о неисполнении или ненадлежащем исполнении обязательств по договорам поставки.

Методы классификации текстовой информации основаны на предположении, что документы, относящиеся к одной категории, содержат одинаковые признаки (слова или словосочетания), и наличие или отсутствие таких признаков в документе говорит о его принадлежности или непринадлежности к той или иной теме [14]. Для выявления схожей структуры и одинаковых признаков (терминов) документов одного класса применялся латентно-семантический анализ (ЛСА) [15].

Нами были рассмотрены несколько известных методов классификации и проведены испытания на тестовой выборке: наивный байесовский классификатор [14], метод k ближайшего соседа (k-means) [13] и деревья решений [16]. Результаты работы классификаторов показали наличие ошибок на тестовой выборке: классификатор Байеса – 4%, точность классификации 96%; k-means – 2%, точность классификации – 98%; деревьев решений – 2%, точность классификации – 98%, что не является приемлемым результатом в рамках системы разрабатываемой системы, поскольку с увеличением количества рассматриваемых категорий дел (а их около 100), получаемые ошибки будут накапливаться, и точность работы сервиса в целом будет падать.

Для улучшения точности классификации разрабатывается алгоритм на основе искусственной нейронной сети. Алгоритм реализуется на языке R с помощью искусственной нейронной сети и имеет следующие параметры: 40 нейронов во входном слое, 1 скрытый слой с 4 нейронами, 4 выходных нейрона, активационная функция: сигмоида.

Алгоритм обработки документа выглядит следующим образом:

- на вход подается идентификатор документа;
- из документа выделяются ключевые слова и их количество;
- проводятся анализ и подбор класса дела;
- в качестве результата алгоритм возвращает идентификатор класса судебного дела, который становится дополнительным свойством документа.

При добавлении нового класса проводятся анализ допустимых ключевых слов и повторное обучение нейронной сети.

Создание шаблонов исковых заявлений. Отдельной задачей является сопоставление судебных актов и заявлений по рассмотренным делам, поскольку сами исковые заявления, в отличие от базы знаний принятых решений, являются закрытыми и не публикуются в интернете. Вообще говоря, установление связи искового заявления и судебного решения в общем случае не представляет сложности и не имеет особого смысла, поскольку они связаны связью один-к-одному, однако в рамках разработки системы «Робот-Юрист» актуальной является задача связывания вновь поданного искового заявления и близких результатов судебных процессов для дальнейшей обработки. В этом случае необходимо иметь заявление в виде, удобном для машинной обработки.

Для получения экземпляров исковых заявлений сразу в электронном виде был предложен механизм веб-портала – шаблонизатора заявлений. При подаче пользователем системы искового заявления система формирует печатную версию заявления в соответствии с регламентирующими нормативными документами РФ, а электронная копия документа автоматически размечается и сохраняется в базе данных системы с определенным статусом. Далее задействуются внутренние функции системы по классификации и кластеризации. После завершения регламентных заданий становится возможным задействовать функции системы для поиска и просмотра близких судебных дел или присваиванию категории судебному спору.

Веб-портал предусматривает несколько ролей пользователей с различной функциональностью, также предложена и реализована статусная модель судебного дела для удобства отслеживания жизненного цикла документа в системе.

Заключение

На данный момент система находится на начальном этапе развития – закончено проектирование системы «Робот-Юрист», произведена первоначальная разметка документов. После первых прогонов алгоритма по извлечению сущностей из документов выявлены неточности в разметке, которые сейчас исправляются.

В рамках решения задачи классификации проведены предварительный анализ судебных документов, отбор значимых признаков для определенных категорий судебного спора, проведен латентно-семантический анализ для выявления общей структуры типовых документов. На тестовой выборке проверены

алгоритмы байесовской классификации, k ближайшего соседа и деревьев решений, разрабатывается модель на основе искусственной нейронной сети. На следующем этапе планируется увеличить выборку исковых арбитражных заявлений и рассмотреть большее число типов возможных судебных споров, а также разработать программные модули, выполняющие задачи отбора информативных признаков и классификации.

Работа выполнена за счет средств субсидии, выделенной Казанскому федеральному университету для выполнения государственного задания в сфере научной деятельности, проект 2.8712.2017/8.9.

Литература

1. S. Peroni. SemanticWeb Technologies and Legal Scholarly Publishing Law, Springer, Governance and Technology Series, vol. 15, 2014. doi 10.1007/978-3-319-04777-5
2. Case Crunch Alfa [Электронный ресурс] Режим доступа: http://www.case-crunch.com, свободный
3. А. М. Елизаров, А. Б. Жижченко, Н. Г. Жильцов, А. В. Кириллович, Е. К. Липачёв. Онтологии математического знания и рекомендательная система для коллекций физико-математических документов //Доклады академии наук. 2016. Т 467, № 4, С. 392–395. doi: 10.1134/S1064562416020174
4. А. М. Елизаров, Е. К. Липачёв, О. А. Невзорова, В. Д. Соловьев. Методы и средства семантического структурирования электронных математических документов //Доклады академии наук. 2014. Т. 457, № 6, С. 642–645. doi 10.7868/S0869565214240049
5. Г. С. Ингерсолл, Т. С. Мортон, Э. Л. Фэррис. Обработка неструктурированных текстов. Поиск, организация и манипулирование. / Пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2015. – 414 с.: ил.
6. N. Gold et al. Understanding Service Oriented Software. IEEE Software, vol. 21, no. 2, 2004, P. 71–77.
7. S. Jones. Toward an Acceptable Definition of Service. IEEE Software, vol. 22, no. 3, 2005, P. 87–93.
8. M. Fowler Microservices a definition of this new architectural term https://martinfowler.com/articles/microservices.html
9. Д. С.Зуев, А. А. Марченко, А. Ф. Хасьянов Применение инструментов интеллектуального анализа текстов в юриспруденции // CEUR Workshop Proceedings. 2017. V. 2022, pp. 214-218. http://ceur-ws.org/Vol-2022/paper35.pdf
10. P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou and J. Tsujii Brat: a Web-based Tool for NLP-Assisted Text Annotation. In Proceedings of the Demonstrations Session at EACL, 2012.
11. TF-IDF https://ru.wikipedia.org/wiki/TF-IDF

12.F. Ricci, L. Rokach, B. Shapira, P.B. Kantor Recommender Systems Handbook. N.Y.: Springer, 2011.
13.https://ru.wikipedia.org/wiki/K-means
14.А. А. Барсегян, М. С. Куприянов, И. И. Холод, М. Д. Тесс, С. И. Елизаров Анализ данных и процессов: учеб. пособие – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2009. – 512 с.: ил. + CD-ROM – (Учебная литература для вузов).
15.T.K. Landauer, P.Foltz,D. Laham An Introduction to Latent Semantic Analysis. Discours Processes, 25, 1998 — P. 259-284.
16.C. C. Aggarwal. Data Classification: Algorithms and Applications. Text Classification. Chapman & Hall/CRC, 2014, ISBN:1466586745 9781466586741

References

1. S. Peroni. SemanticWeb Technologies and Legal Scholarly Publishing Law, Springer, Governance and Technology Series, vol. 15, 2014. doi 10.1007/978-3-319-04777-5
2. Case Crunch Alfa [Электронный ресурс] Режим доступа: http://www.case-crunch.com, свободный
3. A. M. Elizarov, A. B. Zhizhchenko, N. G. Zhil'tsov, A. V. Kirillovich, E. K. Lipachev. Ontologii matematicheskogo znaniya i rekomendatel'naya sistema dlya kollektsiy fiziko-matematicheskikh dokumentov //Doklady akademii nauk. 2016. T 467, № 4, S. 392–395. doi: 10.1134/S1064562416020174
4. A. M. Elizarov, E. K. Lipachev, O. A. Nevzorova, V. D. Solov'yev. Metody i sredstva semanticheskogo strukturirovaniya elektronnykh matematicheskikh dokumentov //Doklady akademii nauk. 2014. T. 457, № 6, S. 642–645. doi 10.7868/S0869565214240049
5. Grant S. Ingersoll, Thomas S. Morton, Drew Farris. Taming Text: How to Find, Organise, and Manipulate it. /Manning Publications, 2013.
6. N. Gold et al. Understanding Service Oriented Software. IEEE Software, vol. 21, no. 2, 2004, P. 71–77.
7. S. Jones. Toward an Acceptable Definition of Service. IEEE Software, vol. 22, no. 3, 2005, P. 87–93.
8. M. Fowler Microservices a definition of this new architectural term https://martinfowler.com/articles/microservices.html
9. D. S. Zuev, A. A. Marchenko, A. F. Khassianov Text Mining Tools in Legal Documents // CEUR Workshop Proceedings. 2017. V. 2022, pp. 214-218. http://ceur-ws.org/Vol-2022/paper35.pdf
10. P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou and J. Tsujii Brat: a Web-based Tool for NLP-Assisted Text Annotation. In Proceedings of the Demonstrations Session at EACL, 2012.
11. TF-IDF https://ru.wikipedia.org/wiki/TF-IDF
12. F. Ricci, L. Rokach, B. Shapira, P.B. Kantor Recommender Systems Handbook. N.Y.: Springer, 2011.

13.https://ru.wikipedia.org/wiki/K-means
14.14. A. A. Barsegyan, M. S. Kupriyanov, I. I. Kholod, M. D. Tess, S. I. Yelizarov Analiz dannykh i protsessov: ucheb. posobiye – 3-ye izd., pererab. i dop. – SPb.: BKHV-Peterburg, 2009. – 512 s.: il. + CD-ROM.
15.T.K. Landauer, P.Foltz,D. Laham An Introduction to Latent Semantic Analysis. Discours Processes, 25, 1998 — P. 259-284.
16.C. C. Aggarwal. Data Classification: Algorithms and Applications. Text Classification. Chapman & Hall/CRC, 2014, ISBN:1466586745 9781466586741