УДК 004.82 *Научная статья* DOI: 10.18287/2223-9537-2026-16-3-502-514

Программный конвейер для формирования шаблонов онтологий из текстов на естественном языке

© 2026, С.А. Федосин ✉, С.А. Ямашкин, А.А. Сайгин

Национальный исследовательский Мордовский государственный университет им. Н.П. Огарёва (МГУ им. Н.П. Огарёва), Саранск, Россия

Аннотация

Рассматривается задача извлечения и связывания фактов из русскоязычных юридических текстов и конвейерный способ её решения для формирования базы знаний в виде иерархического словаря, как шаблона онтологии предметной области. Предлагаемый программный конвейер включает этапы нормализации входных данных, лексический анализ с извлечением именованных сущностей, синтаксический анализ с построением абстрактного синтаксического дерева и отображение полученного словаря в *RDF/OWL*-представление. В лексическом анализе использован гибридный модуль извлечения сущностей, сочетающий модели на основе машинного обучения и правила, заданные грамматиками и шаблонами. Проведена экспериментальная оценка качества анализа на специализированном корпусе решений региональных подразделений Федеральной антимонопольной службы России: показано, что добавление правил к обученным моделям приводит к приросту интегральных метрик извлечения сущностей около 5%. На этапе синтаксического анализа предложена конфигурируемая грамматика, позволяющая отделять разные дела в одном юридическом документе и связывать извлечённые сущности в структурированный словарь. Такой словарь может быть отображён в онтологический граф и использован как слой онтологизации для хранения в графовой базе знаний и сопоставления с подходами на основе больших языковых моделей.

Ключевые слова: *извлечение фактов, шаблон онтологии, обработка естественного языка, извлечение сущностей, извлечение отношений, машинное обучение, словарь*.

Цитирование: *Федосин С.А., Ямашкин С.А., Сайгин А.А*. Программный конвейер для формирования шаблонов онтологий из текстов на естественном языке. *Онтология проектирования*. 2026. Т.16, №3(61). С.502-514. DOI: 10.18287/2223-9537-2026-16-3-502-514.

Финансирование: Исследование выполнено в рамках НИР «Создание лаборатории искусственного интеллекта» по программе социально-экономического развития Республики Мордовия на 2022-2026 гг.

Вклад авторов: *Федосин С.А.* – экспертиза исходных данных и результатов; *Ямашкин С.А.* – постановка задачи, разработка концепции, формализация модели; *Сайгин А.А.* – разработка программного кода и обработка текстов.

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Введение

Большой объём текстовой информации в Интернете доступен для автоматизированной обработки с целью извлечения новых полезных знаний и моделирования предметных областей (ПрО). Извлечение фактов при этом является основой для построения онтологий, семантического поиска и систем поддержки принятия решений [1].

Цель статьи – разработка программного конвейера (ПрК), обеспечивающего извлечение упорядоченных наборов данных из слабоструктурированных документов, позволяющего повысить точность извлечения данных и снизить ресурсоёмкость процесса обработки. ПрК должен обеспечивать сбор и нормализацию текстовых данных, извлечение именованных

сущностей, выделение отношений между ними и формирование иерархических словарей, которые могут рассматриваться как шаблоны онтологий ПрО.

Для достижения поставленной цели необходимо:

- проанализировать подходы к извлечению фактов из текстов на естественном языке (ЕЯ);
- спроектировать архитектуру ПрК, включающего этапы сбора и нормализации данных, лексического и синтаксического анализа;
- разработать модуль извлечения именованных сущностей и связей на основе сочетания методов машинного обучения (ММО) и правил, заданных грамматиками и шаблонами;
- сформировать иерархические словари в машиночитаемом формате на основе выделенных сущностей и отношений, определить правила их отображения средствами *RDF/OWL*-представления в шаблоны онтологического описания ПрО;
- провести оценку качества извлечения фактов на специализированном наборе данных.

1 Программный конвейер обработки текстов

Процесс обработки текстов на ЕЯ можно осуществить посредством ПрК. На первом этапе проводится извлечение и нормализация данных, очистка их от артефактов вёрстки, обработка кодировок и сегментация текста на предложения. На этапе лексического анализа текста осуществляется семантическая разметка текста. После того, как лексемы найдены, определяются логические связи между ними, превращая изолированные термины в факты. Для каждой пары сущностей в одном логическом блоке проверяется наличие семантической связи. На заключительном этапе осуществляется перевод полученных результатов в машиночитаемый формат. Данные упаковываются в иерархическую структуру (например, *JSON*-объект), и полученный объект экспортируется в требуемый формат (см. рисунок 1).

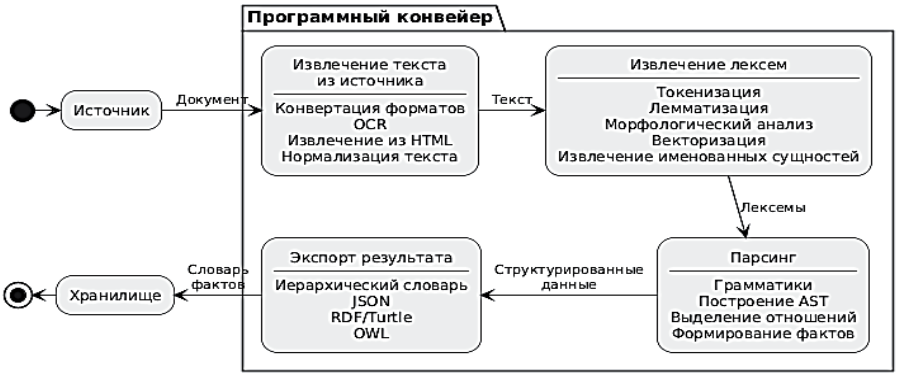

Рисунок 1 – Схема программного конвейера обработки текстов

В ПрК реализованы конверторы для следующих форматов: текстовых (*DOC, RTF, DOCX, ODT, PDF*); табличных (*XLS, XLSX, ODS*); графических (*JPEG, TIFF* и т.п.). Для графических форматов требуется применение оптического распознавания (*Optical Character Recognition*, *OCR*). Задача ПрК – независимо от исходного формата входного файла сформировать текстовый выход в кодировке *UTF*-8 с сохранением логической структуры документа.

2 Лексический анализ текста

Исходный текст преобразуется в формализованное представление потока лексем, пригодное для анализа и извлечения информации. Каждая лексическая единица снабжается набором атрибутов: позиция в тексте; словоформа; нормальная форма; часть речи; морфологические характеристики; признаки, характеризующие контекст употребления. Задача извле-

чения именованных сущностей (*Named Entity Recognition, NER*) рассматривается как задача классификации [2].

Ранние методы автоматизированного извлечения фактов из текстов на ЕЯ базировались на явном кодировании лингвистических и логических правил, а также на использовании шаблонов. В методах, основанных на правилах, знания специалиста ПрО формализуются в виде набора инструкций, которые применяют к тексту для извлечения фактов [3]. Способ прописать правила для извлечения фактов, используя регулярные выражения, эффективен в случаях, когда в тексте есть подстроки, о которых можно сделать вывод о принадлежности строки, её части или соседнего фрагмента к какому-либо классу [4].

В методе концептуального моделирования строится граф, в вершинах которого находятся понятия, а на рёбрах задаются связи между понятиями. Метод включает построение множества концептуальных графов, которые агрегируются, и на этой основе выделяются формальные понятия, между которыми можно выстраивать связи [5].

Распространённым методом извлечения фактов из текстовой информации являются правила, построенные на основе контекстно-свободных грамматик (КС-грамматик) [6]. Шаблоны, написанные с применением КС-грамматик, можно применить для текстов на ЕЯ.

В онтологическом подходе для каждой категории фактов разрабатывается набор шаблонов извлечения, которые обобщают имеющиеся в тексте субъекты, объекты и отношения [7].

На первом этапе работы ПрК текст разбивается на отдельные слова, слоги, знаки препинания, отступы. Это лингвистический процесс, основанный на правилах и регулярных выражениях. Далее текстовые токены преобразуются в векторы из чисел. ПрК анализирует полученные векторы признаков, определяет границы сущностей и их типы (см. рисунок 2).

Рисунок 2 – Схема извлечения именованных сущностей

Системы на базе ММО обладают устойчивостью к вариативности языка, их качество зависит от качества обучающей выборки, результаты сложнее интерпретировать. В гибридном подходе одна часть задачи решается с помощью правил, другая – с помощью ММО. Многие сущности могут быть описаны шаблонами, общими для разных текстов, например: номера телефонов, адреса электронной почты, признаки различных документов. Такие сущности удобно извлекать с помощью регулярных выражений. Наименования, которые не меняются в тестах, такие как топонимы, можно искать с помощью словарей. Другие сущности, не обладающие таким постоянством (имена, фамилии, названия организаций и др.), распознаются с помощью обученной модели. Такой подход применён в библиотеке *SpaCy*[1], где для извлечения именованных сущностей используется языковая модель.

Если существует метка, которая моделью не распознаётся, и дообучение не даёт приемлемого результата, то можно прописать набор правил, который будет использоваться совместно с моделью *SpaCy*. Для подтверждения данной гипотезы проведён ряд экспериментов. В *SpaCy* есть набор предобученных моделей, в том числе три для работы с текстами на русском языке: *ru_core_news_sm, ru_core_news_md* и *ru_core_news_lg*. Все три модели обучены на одном корпусе данных – *Nerus*, и содержат одинаковый набор компонентов: токенизатор, морфологию, синтаксический анализатор и *NER*. Различие между моделями состоит в наличии и размере статических векторных представлений слов, что влияет на точность, вес моде-

[1] SpaCy – открытая библиотека *Python* для продвинутой обработки ЕЯ. https://spacy.io/.

ли и её возможности. В маленькой модели (*sm*) нет статических векторов, в средней (*md*) – около 50000 векторов, в большой (*lg*) – 500000 векторов.

Для проверки предложенного подхода использован размеченный набор данных *slm-ct/fas_ad_practice_dataset*[2], который представляет собой коллекцию текстов решений региональных подразделений Федеральной антимонопольной службы (ФАС), подвергнутую предварительной подготовке. Анализ набора данных позволил составить список меток, приведённый в таблице 1.

Полученный набор данных содержит 4332 записи с 45305 размеченными метками. Обучающая выборка содержит 3118 записей, валидационная – 780, тестовая – 434. Для проверки обученных моделей используются стандартные классификационные метрики.

Точность (*precision*) показывает, какая доля ответов спрогнозирована верно. Метрика характеризует способность отличать класс от других. Полнота (*recall*) показывает долю верно спрогнозированных объектов метки из общего объёма метки. Эти метрики не зависят от соотношения классов и применимы в условиях несбалансированных выборок. На их основе выводится единый критерий – гармоническое среднее точности и полноты.

Таблица 1 – Список извлекаемых сущностей

Метка	Описание	Количество
PER	ФИО людей	13049
LOC	Локация (страна, область, город и т.д.)	10380
ORG	Название организации	9562
DATE	Дата	5766
URL	Адрес в Интернете	178
EMAIL	Электронная почта	1725
PHONE	Номер телефона	1739
LAW	Закон	1792
CRIME	Правонарушение	663
PUNISH	Решение по делу	451

Известны различные подходы к расчёту метрик [8].

Для извлечения сущностей *EMAIL, URL, PHONE* и *LAW* написаны правила. В *SpaCy* правила легко встраиваются с помощью класса *EntityRuler* в общий конвейер обработки. Сравнение оценки моделей на тестовой выборке приведены в таблице 2. Добавление правил улучшило работу моделей, прирост метрик составил от 2 до 6%.

Таблица 2 – Результаты оценки моделей

Модель	Обучение					Обучение + правила				
	Micro_p	*Micro_r*	*Micro_F1*	*Macro_F1*	*Weighted_F1*	*Micro_p*	*Micro_r*	*Micro_F1*	*Macro_F1*	*Weighted_F1*
SpaCy sm	0,78	0,71	0,74	0,65	0,73	0,81	0,71	0,76	0,70	0,75
SpaCy md	0,78	0,73	0,76	0,66	0,74	0,78	0,74	0,76	0,72	0,77
SpaCy lg	0,78	0,73	0,75	0,66	0,74	0,79	0,75	0,77	0,73	0,78

Для задач извлечения структуры из текстов на ЕЯ применяются большие языковые модели (БЯМ), способные выделять сущности, отношения и формировать *JSON*-подобные представления без дообучения. БЯМ целесообразно рассматривать как дополнительный семантический модуль, применяемый для обработки неоднозначных фрагментов, редких конструкций и сложных ссылок.

При выборе подхода для извлечения структуры из русскоязычных юридических документов необходимо учитывать не только качество выделения сущностей и связей, но и ресурсоёмкость решения, включая требования к локальному оборудованию, стоимость эксплуатации и зависимость от внешней вычислительной инфраструктуры.

3 Извлечение отношений

Для консолидации разрозненных сущностей в связные факты необходимо решить задачу выявление отношений между парами сущностей в неструктурированных текстовых данных.

[2] Hugging Face. https://huggingface.co/datasets/slm-ct/fas_ad_practice_dataset.

Автоматическое распознавание связей востребовано при анализе документов, справочников и новостных лент [9].

Известны два подхода к решению задачи извлечения отношений. Первый – синтаксический анализ: осуществляется процесс сопоставления линейной последовательности лексем некоторой формальной грамматике. Другой подход связан с применением ММО, в котором используются нейронные сети с архитектурой *seq2seq*, предназначенные для преобразования последовательностей из одной области в последовательности другой области [10]. Нейронные сети рассматривают извлечение отношений как задачу машинного перевода, что делает невозможной интерпретацию результатов. По этим причинам применяют методы на основе грамматик или совместно с ММО [11].

Для связывания лексем, полученных на этапе извлечения именованных сущностей инструментом синтаксического анализа является абстрактное синтаксическое дерево (АСД) [12]. В настоящей работе под АСД понимается дерево, вершины которого соответствуют нормализованным фактам и их компонентам (предикатам и аргументам), а листья – выделенным именованным сущностям и ключевым лексическим единицам, непосредственно участвующим в формировании факта. При построении АСД опускаются вспомогательные грамматические узлы (служебные части речи, промежуточные нетерминалы, знаки пунктуации) и свёртываются сложные синтаксические конструкции в узлы «нормализованного действия/факта», что позволяет рассматривать полученную структуру как промежуточное представление между деревом зависимостей и семантическим графом. Такое АСД сохраняет древовидную структуру на уровне предложения, но ориентировано на последующее извлечение и отображение фактов в иерархические словари и онтологические шаблоны.

Пусть имеется упорядоченная последовательность лексем: $S=\langle t_1,t_2,\ldots,t_n\rangle$. Каждая лексема t_i – это кортеж из трёх элементов: c – тип лексемы; v – строковое значение лексемы; pos – позиция в исходном тексте. На основе последовательности S строится дерево $T = (V, E)$, где V – множество узлов, а E – связи между ними.

При обработке текста узлы дерева делятся на два непересекающихся множества: $V = V_{int} \cup V_{leaf}$. Внутренние узлы V_{int} – это узлы, у которых есть дочерние элементы. Они не существуют в исходном тексте. Это грамматические или логические конструкции, которые распознаны на основе правил. Листья V_{leaf} – это конечные узлы дерева. В АСД листья ссылаются на значимые лексемы из последовательности S, извлекая значение v.

АСД в ПрК – это результат функции редукции, которая удаляет синтаксический мусор. Синтаксический анализатор *фильтрует лексемы* и *иерархизирует остаток* (оставшиеся лексемы становятся листьями, подвешенными к узлам-концептам).

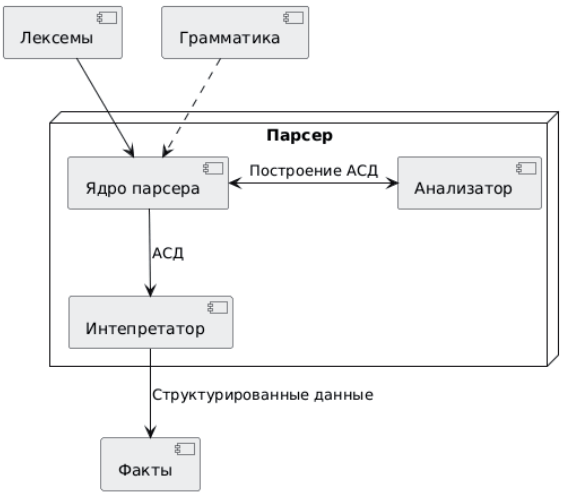

Рисунок 3 – Схема синтаксического анализатора

Схема предлагаемого синтаксического анализатора представлена на рисунке 3. На первом этапе загружается грамматика и задаются множество типов узлов дерева и интерпретационные правила, определяющие, какие узлы преобразуются в какие поля целевой записи. На втором этапе главный модуль анализатора загружает поток лексем, применяет грамматику и строит АСД, отражающее логическую структуру документа. На третьем этапе интерпретатор АСД проходит по дереву и превращает узлы в элементы структурированных данных: поля,

значения, записи и составные структуры. Четвёртый этап заключается в формировании итогового объекта структурированных данных. Для примера рассмотрен следующий документ[3].

ПРЕДПИСАНИЕ
О ПРЕКРАЩЕНИИ НАРУШЕНИЯ ЗАКОНОДАТЕЛЬСТВА
РОССИЙСКОЙ ФЕДЕРАЦИИ О РЕКЛАМЕ ПО ДЕЛУ № 076/05/18-364/2024

29 мая 2024 года город Ярославль

Комиссия Межрегионального управления Федеральной антимонопольной службы по Ярославской области и Костромской области по рассмотрению дел по признакам нарушения законодательства о рекламе в составе:
председатель Комиссии – заместитель руководителя Управления Шушкова А. С.,
члены Комиссии – начальник отдела контроля органов власти и рекламы Семенюк А. В., главный специалист-эксперт отдела контроля органов власти и рекламы Солодяшкина С. Д.,
на основании своего решения от «29» мая 2024 г. по делу № 076/05/18-364/2024 о признании ненадлежащей рекламы услуг ООО «ВсеИнструменты.ру» с нарушением требований пунктов 4, 10 части 3 статьи 5, статьи 8, частей 1, 2 статьи 18 Федерального закона от 13.03.2006 г. № 38-ФЗ «О рекламе» (далее – Федеральный закон «О рекламе»), в соответствии с пунктом 2 части 1 статьи 33, частью 1 статьи 36 Федерального закона «О рекламе», пунктами 4, 30, 37-42, 44, 45, 48 Правил рассмотрения антимонопольным органом дел, возбужденных по признакам нарушения законодательства Российской Федерации о рекламе,

ПРЕДПИСЫВАЕТ…

Для рассматриваемого класса документов грамматика задаётся не для полного текста на ЕЯ, а для потока уже выделенных лексем, каждая из которых имеет тип, значение и позицию в документе. В этом случае синтаксический анализатор оперирует последовательностью терминалов вида *CASE_NUMBER, DATE, LOC, PER, ORG, LAW, PHONE, EMAIL, URL*. Для данного примера документа ФАС упрощённая грамматика может быть записана в *BNF*-нотации следующим образом:

```
<root>              ::= <case> | <case> <root>
<case>              ::= <case_number> <date> <city> <commission> <case_body>
<case_number>       ::= "CASE_NUMBER"
<date>              ::= "DATE"
<city>              ::= "LOC"

<commission>        ::= <chairman> <members_loop>
<chairman>          ::= "PER"
<members_loop>      ::= "PER" <members_loop> | <skip_to_law> <members_loop> | ε
<skip_to_law>       ::= "CASE_NUMBER" | "DATE" | "LOC" | "ORG" | "TEL" | "EMAIL" | "URL" | "PUNISH"

<case_body>         ::= <body_item> <case_body> | <skip_to_next_case> <case_body> | ε
<body_item>         ::= <violator> | <address> | <phone> | <email> | <url> | <law> | <punish>
<skip_to_next_case> ::= "DATE" | "LOC" | "PER" | "ORG" | "TEL" | "EMAIL" | "URL" | "LAW" | "PUNISH"

<violator>          ::= "ORG"
<address>           ::= "LOC"
<phone>             ::= "TEL"
<email>             ::= "EMAIL"
<url>               ::= "URL"
<law>               ::= "LAW"
<punish>            ::= "PUNISH"
```

<root> служит корнем словаря, от которого строится иерархия дел; <case> описывает отдельно взятое дело, внутри которого содержатся <commission>, описывающее состав комиссии, и <case_body>, содержащее информацию об участниках дела, нарушениях и т.д. В комиссию попадают только метки PER, остальные пропускаются с помощью <skip_to_law>. Метка LAW является началом описания дела, поэтому начинает работать <case_body>. С помощью

[3]Предписание №7159/24 076/05/18-364/2024 от 7 июня 2024 г. База решений и правовых актов. Федеральная Антимонопольная служба. https://br.fas.gov.ru/to/yaroslavsko-kostromskoe-mufas-rossii/e2972484-96d6-4d38-ae21-8e88a9605e66/?query=%E2%84%96%20076/05/18-364/2024.

<skip_to_next_case> в дело не попадут лишние метки. Работает правило до начала следующего дела, либо до конца документа.

Сформированные на этапе синтаксического анализа иерархические словари могут быть интерпретированы как прикладные *JSON*-структуры и как шаблоны онтологического описания документов. На этом уровне каждому ключевому блоку словаря сопоставляется класс ПрО, а вложенные структуры и массивы интерпретируются как связанные сущности, соединённые объектными свойствами. Схема такого отображения документов приведена на рисунке 4. Центральным элементом шаблона является класс Case, содержащий основные атрибуты дела и связываемый с экземплярами классов Commission, Organization, LegalNorm, Address и Contacts. Состав комиссии детализируется через сущности класса Person, а контактные данные представляются как связанные сущности Phone, Email и WebResource, что обеспечивает единообразное отображение результатов извлечения в *RDF/OWL*-представление. Такая схема позволяет сопоставить *JSON*-структуру и онтологический граф и может использоваться как шаблон при обработке других документов того же класса.

Структура документа представляется формальной грамматикой, которую можно записать в конфигурационный файл. Синтаксический анализатор загружает нужную конфигурацию и применяет её к текущему набору лексем. В такой записи нетерминалы верхнего уровня задают логические блоки документа, а терминальные символы соответствуют результатам этапа *NER*; тем самым связывание сущностей выполняется по их принадлежности к распознанному структурному контексту документа.

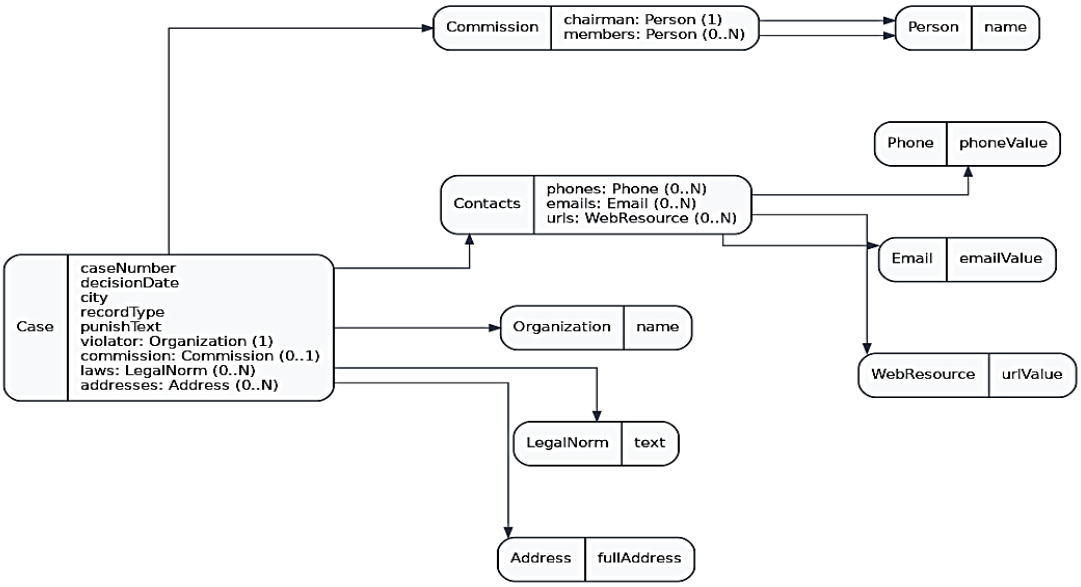

Рисунок 4 – Онтологический шаблон представления дела Федеральной антимонопольной службы

Далее документ обрабатывается лучшей моделью для извлечения именованных сущностей - дообученной *ru_core_news_lg* с правилами. Получается следующий набор сущностей.


```
 [
{"label":"CASE_NUMBER","text":"№ 076/05/18-364/2024","span":{"start":92,"end":112}},
{"label":"DATE","text":"29 мая 2024 года","span":{"start":114,"end":130}},
{"label":"LOC","text":"город Ярославль","span":{"start":228,"end":243}},
{"label":"PER","text":"Шушкова А. С.","span":{"start":504,"end":517}},
{"label":"PER","text":"Семенюк А. В.","span":{"start":588,"end":601}},
{"label":"PER","text":"Солодяшкина С. Д.","span":{"start":671,"end":688}},
{"label":"DATE","text":"«29» мая 2024 г.","span":{"start":722,"end":738}},
{"label":"CASE_NUMBER","text":"№ 076/05/18-364/2024","span":{"start":747,"end":767}},
{"label":"ORG","text":"ООО «ВсеИнструменты.ру»","span":{"start":807,"end":830}},
{"label":"LAW","text":"части 3 статьи 5, статьи 8, частей 1, 2 статьи 18 Федерального закона от 13.03.2006
г. № 38-ФЗ «О рекламе»","span":{"start":869,"end":975}},
```

```
{"label":"LAW","text":"Федеральный закон «О рекламе»","span":{"start":985,"end":1014}},
{"label":"LAW","text":"пунктом 2 части 1 статьи 33","span":{"start":1034,"end":1061}},
{"label":"LAW","text":"частью 1 статьи 36 Федерального закона «О рекла-
ме»","span":{"start":1063,"end":1113}},
{"label":"DATE","text":"37-42","span":{"start":1131,"end":1136}},
```


Этот набор лексем обрабатывается с помощью АСД:


```
[
  {
    "case_number": "№ 076/05/18-364/2024",
    "date": "29 мая 2024 года",
    "city": "город Ярославль",
    "commission": {
      "chairman": "Шушкова А. С.",
      "members": [
        "Семенюк А. В.",
        "Солодяшкина С. Д."
      ]
    },
    "laws": [
      "части 3 статьи 5, статьи 8, частей 1, 2 статьи 18 Федерального закона от 13.03.2006 г. № 38-ФЗ «О
рекламе»",
      "Федеральный закон «О рекламе»",
      "пунктом 2 части 1 статьи 33",
      "частью 1 статьи 36 Федерального закона «О рекламе»",
      "пунктом 2.4 статьи 19.5 Кодекса Российской Федерации об административных правонарушениях",
      "статьей 198 Арбитражного процессуального кодекса Российской Федерации"
    ],
    "violator": "ООО «ВсеИнструменты.ру»",
    "addresses": [
      "109451, г. Москва, вн.тер.г. Муниципальный округ Марьино, ул. Братиславская, д. 16, корп. 1"
    ],
    "phones": [],
    "emails": [],
    "urls": [],
    "punish": "наложить на граждан штраф в размере от трехсот до пятисот рублей, на должностных лиц",
    "type": "case"
  }
]
```


В результате из набора лексем получен факт в виде словаря связанных сущностей, где каждый ключ означает вид связи внутри обработанного документа. Грамматика является универсальной для обработки документов из набора данных и легко изменяемой в случаях изменения структуры текста в документе. Полученное АСД способно отделять разные дела, если они описаны в одном тексте.

4 Шаблон онтологии

После формирования структурированного *JSON*-объекта получается удобочитаемый формат представления данных, но необходимо привести его к онтологически интерпретируемому виду. В ПрК переход к представлениям на языках *RDF/OWL/Turtle* можно выполнить с помощью правил, связывающих поля *JSON* с классами, объектами и свойствами онтологической модели. Такой подход позволяет отделить задачу извлечения и нормализации данных из текста от задачи их формализации в виде онтологических конструкций, пригодных для последующего хранения в графовой базе знаний.

Алгоритм преобразования включает следующие шаги.

1) для корневого *JSON*-объекта определяется его онтологический тип, например *Case, Decision* или *Prescription* в зависимости от класса документа.
2) простые поля *JSON*, содержащие строковые, числовые и календарные значения (*case_number, date, city, type*), отображаются в свойствах соответствующего экземпляра.

3) вложенные объекты и массивы (*commission, members, laws, addresses*) интерпретируются как связанные экземпляры классов *Commission, Person, LegalNorm, Address* и соединяются с корневым объектом через свойства объекта.
4) выполняется нормализация значений: даты переводятся в стандартный формат, повторяющиеся адреса дедуплицируются, а текстовые ссылки на нормы права при необходимости разлагаются на составные элементы: закон, статья, часть, пункт.
5) из полученного графа фактов генерируется *RDF*-представление, которое может быть дополнено *OWL*-аксиомами, а также ограничениями целостности.

Для рассматриваемого документа итоговый *JSON*-объект может быть интерпретирован как экземпляр класса *Case*, связанный с экземпляром Commission, нарушителем класса Organization, набором сущностей класса *LegalNorm* и адресом класса *Address*. Фрагмент результата в формате *Turtle* может быть представлен следующим образом:

```
@prefix ex: <http://example.org/legal#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

ex:case_076_05_18_364_2024 a ex:Case ;
    ex:hasCaseNumber "076/05/18-364/2024" ;
    ex:hasDecisionDate "2024-05-29"^^xsd:date ;
    ex:hasCity "Ярославль" ;
    ex:hasCommission ex:commission_076_05_18_364_2024 ;
    ex:hasViolator ex:org_vseinstrumenty ;
    ex:referencesLegalNorm ex:law_1 , ex:law_2 , ex:law_3 ;
    ex:hasAddress ex:address_1 ;
    ex:hasRecordType "case" .
ex:commission_076_05_18_364_2024 a ex:Commission ;
    ex:hasChairman ex:person_shushkova ;
    ex:hasMember ex:person_semenyuk , ex:person_solodyashkina .
ex:person_shushkova a ex:Person ;
    ex:hasName "Шушкова А. С." .
ex:person_semenyuk a ex:Person ;
    ex:hasName "Семенюк А. В." .
ex:person_solodyashkina a ex:Person ;
    ex:hasName "Солодяшкина С. Д." .
ex:org_vseinstrumenty a ex:Organization ;
    ex:hasName "ООО «ВсеИнструменты.ру»".
ex:address_1 a ex:Address ;
    ex:hasFullAddress "109451, г. Москва, вн.тер.г. Муниципальный округ Марьино, ул. Братиславская, д. 16,
корп. 1" .
ex:law_1 a ex:LegalNorm ;
    ex:hasText "части 3 статьи 5, статьи 8, частей 1, 2 статьи 18 Федерального закона от 13.03.2006 г. №
38-ФЗ «О рекламе»".
ex:law_2 a ex:LegalNorm ;
    ex:hasText "пунктом 2 части 1 статьи 33" .
ex:law_3 a ex:LegalNorm ;
    ex:hasText "частью 1 статьи 36 Федерального закона «О рекламе»".
```

В таком представлении *JSON*-структура становится не только техническим форматом обмена данными, но и слоем онтологизации текста. Это позволяет рассматривать сформированный иерархический словарь как прикладной словарь полей документа и как шаблон онтологического описания, в котором выделены сущности ПрО, их атрибуты и отношения.

5 Сравнение программного конвейера с современными подходами

Исследования в области извлечения фактов из текстов на ЕЯ развиваются по нескольким направлениям. Первое направление составляют системы, основанные на правилах и формальных грамматиках [13]. Второе направление образуют обучаемые модели, ориентированные на совместное [14] или поэтапное [15] извлечение сущностей и отношений. Третье направление представлено нейросетевыми и нейросимвольными подходами [7, 16].

ПрК не относится полностью ни к одному из указанных классов. Место предлагаемого решения можно определить как специализированный гибридный конвейер, ориентирован-

ный на извлечение и связывание фактов в документах ограниченного класса с последующим формированием иерархического словаря и его отображением в онтологически интерпретируемое представление. Адаптация предложенного ПрК к другому набору данных выполняется поэтапно и не требует полной переработки, однако её трудоёмкость зависит от конкретной прикладной задачи:

- определяется новая схема целевых сущностей и их соответствие существующим меткам (*PER, ORG, DATE, LAW, PUNISH* и др.);
- корректируются или добавляются правила для формализуемых сущностей (номера дел, телефоны, адреса электронной почты, ссылки и правовые нормы);
- дообучается *NER*-модель на размеченной выборке;
- составляются грамматики синтаксического анализатора и интерпретационные правила, связывающие сущности в итоговую структуру *JSON*-представления;
- *JSON*-документ преобразуется в шаблон онтологии в формате *RDF/OWL/Turtle*.

Заключение

По результатам проведённого исследования сформулированы следующие выводы.

- Разработан ПрК, позволяющий осуществить сбор и нормализацию текстовых данных из различных электронных форматов, лексический анализ с извлечением именованных сущностей, синтаксический анализ с построением АСД и формирование машиночитаемого представления фактов в виде иерархических словарей.
- Показано, что формирование иерархических словарей в формате *JSON* на основе выделенных сущностей и отношений позволяет рассматривать полученные структуры как шаблоны онтологий ПрО, пригодные для интеграции в базы знаний и системы поиска.
- Оценка на специализированном наборе данных решений ФАС подтвердила работоспособность ПрК и показала возможность гибкой его конфигурации для решения задач извлечения и структурирования фактов из текстов на ЕЯ.

Список источников

[1] ***Массель Л.В.*** Экосистема знаний как развитие и специализация цифровой экосистемы. *Труды Международного научно-технического конгресса* «Интеллектуальные системы и информационные технологии–2023». Таганрог: Издатель Ступин С.А., 2023. С.155-164.

[2] ***Большакова Е.И., Воронцов К.В., Ефремова Н.Э., Клышинский Э.С., Лукашевич Н.В., Сапин А.С.*** Автоматическая обработка текстов на естественном языке и анализ данных М.: Изд-во НИУ ВШЭ. 2017. 269 с.

[3] ***van Ginneken B.*** Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. *Radiological physics and technology*. 2017. V.10. No1. P.23-32. DOI: 10.1007/s12194-017-0394-5.

[4] ***Колмогорцев С.В., Сараев П.В.*** Извлечение библиографии из текстов регулярными выражениями. *Новые информационные технологии в автоматизированных системах*. 2017. №20. С.82-88.

[5] ***Богатырев М.Ю.*** Извлечение фактов из текстов естественного языка с применением концептуальных графовых моделей. *Известия Тульского государственного университета*. 2016. Вып.7. Ч.1. С.198-207.

[6] ***Хопкрофт Дж., Мотвани Р., Ульман Дж.*** Введение в теорию автоматов, языков и вычислений. М.: Вильямс. 2015. 528 с.

[7] ***Сидорова Е.А., Иванов А.И., Овчинникова К.А.*** Извлечение информации из текстов на основе онтологии и больших языковых моделей. *Онтология проектирования*. 2025. Т.15, №1(55). С.114-129. DOI 10.18287/2223-9537-2025-15-1-114-129.

[8] ***Ma X., Hovy E.*** End-to-end sequence labeling via bi-directional lstm-cnns-crf. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers). 2016. P.1064-1074. DOI: 10.18653/v1/P16-1101.

[9] ***Moens M.F.*** Information Extraction from Text and Beyond. *Cross-Disciplinary Advances in Applied Natural Language Processing: Issues and Approaches*. IGI Global Scientific Publishing, 2012. P.24-39. DOI: 10.4018/978-1-61350-447-5.ch003.

[10] ***Dong L., Lapata M.*** Language to logical form with neural attention. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers). 2016. P.33-43. DOI: 10.18653/v1/P16-1004.
[11] ***Yin P., Neubig G.*** A syntactic neural model for general-purpose code generation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers). 2017. P.440-450. DOI: 10.48550/arXiv.1704.01696.
[12] ***Falleri J.R., Morandat F., Blanc X., Martinez M., Monperrus M.*** Fine-grained and accurate source code differencing. *Proceedings of the 29th ACM/IEEE international conference on Automated software engineering*. 2014. P.313-324. DOI: 10.1145/2642937.2642982.
[13] ***Chiticariu L., Li Y., Reiss F.*** Rule-based information extraction is dead! Long live rule-based information extraction systems! *Proceedings of the 2013 conference on empirical methods in natural language processing.* Seattle, Washington, USA, 18-21 October 2013. P.827-832.
[14] ***Miwa M., Bansal M.*** End-to-end relation extraction using lstms on sequences and tree structures. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers). 2016. P.1105-1116. DOI: 10.18653/v1/P16-1105.
[15] ***Zhong Z., Chen D.*** A frustratingly easy approach for entity and relation extraction. *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies.* 2021. P.50-61. DOI: 10.18653/v1/2021.naacl-main.5.
[16] ***Saker M.K., Zhou L., Eberhart A., Hitzler P.*** Neuro-symbolic artificial intelligence: Current trends. *Ai Communications.* 2022. V.34. No.3. P.197-209. DOI: 10.48550/arXiv.2105.05330.

Сведения об авторах

Федосин Сергей Алексеевич, 1955 г. рождения. Окончил МГУ им. Н.П. Огарёва в 1978 г., к.т.н. (1983). Профессор кафедры автоматизированных систем обработки информации и управления Института электроники и светотехники МГУ им. Н.П. Огарёва. В списке трудов более 130 публикаций в области проектирования информационных систем. Author ID (Scopus): 57205160591; ORCID: 0000-0003-3481-4101; Author ID (РИНЦ): 101167. *fedosinsa@mrsu.ru.* ✉.

Ямашкин Станислав Анатольевич, 1990 г. рождения. Окончил МГУ им. Н.П. Огарева в 2012 г., к.т.н. (2016). Доцент кафедры автоматизированных систем обработки информации и управления Института электроники и светотехники МГУ им. Н.П. Огарёва, заведующий лабораторией искусственного интеллекта МГУ им. Н.П. Огарёва. В списке трудов около 270 публикаций в области системного анализа. ORCID: 0000-0002-7574-0981; Author ID (РИНЦ): 730394; Author ID (Scopus): 9133286400; Researcher ID (WoS): N-2939-2018. *yamashkinsa@mail.ru.*

Сайгин Андрей Александрович, 1999 г. рождения. Окончил МГУ им. Н.П. Огарёва в 2023 г. Аспирант кафедры автоматизированных систем обработки информации и управления Института электроники и светотехники МГУ им. Н.П. Огарёва. В списке трудов более 10 публикаций в области интеллектуальной обработки данных. ORCID: 0009-0001-4635-6219; Author ID (РИНЦ): 1189605. *andrexsai@mail.ru.*

Поступила в редакцию 11.06.2026, после рецензирования 22.07.2026. Принята к публикации 7.08.2026.

Scientific article DOI: 10.18287/2223-9537-2026-16-3-502-514

Software pipeline for generating ontology templates from natural language texts

© 2026, S.A. Fedosin ✉, S.A. Yamashkin, A.A. Saygin

National Research Mordovia State University named after N. P. Ogarev, Saransk, Russia

Abstract

This paper examines the problem of extracting and linking facts from Russian-language legal texts and presents a pipeline-based approach to solving it for the construction of a knowledge base represented as a hierarchical dictionary serving as an ontology template for the subject domain. The proposed software pipeline comprises input data normalization, lexical analysis with named entity extraction, syntactic analysis with the construction of an abstract syntax tree, and mapping of the resulting dictionary to an RDF/OWL representation. The lexical analysis stage employs a hybrid entity extraction module that combines machine-learning-based models with rules defined by grammars and templates. The quality of the analysis was experimentally evaluated using a specialized corpus of decisions issued by regional divisions of the Federal Antimonopoly Service of Russia. The results demonstrate that augmenting the trained models with rule-based components improves the aggregate entity extraction metrics of approximately 5%. At the syntactic analysis stage, a configurable grammar is proposed that makes it possible to distinguish separate cases contained within a single legal document and link the extracted entities into a structured dictionary. Such a dictionary can be mapped to an ontological graph and used as an ontologization layer for storage in a graph-based knowledge and for comparison with approaches based on large language models.

Keywords: *fact extraction, ontology template, natural language processing, entity extraction, relation extraction, machine learning, dictionary.*

For citation: *Fedosin SA, Yamashkin SA, Saygin AA*. Software pipeline for generating ontology templates from natural language texts [In Russian]. *Ontology of designing*. 2026; 16(3): 502-514. DOI: 10.18287/2223-9537-2026-16-3-502-514.

Financial Support: The study was carried out within the framework of the R&D project “Establishment of an Artificial Intelligence Laboratory” under the Socio-Economic Development Program of the Republic of Mordovia for 2022-2026.

Authors' contributions: *Fedosin S.A.* – expert evaluation of the source data and results*; Yamashkin S.A.* – problem formulation, conceptualization, model formalization*; Saigin A.A.* – software development and text processing.

Conflict of interest: The authors declare no conflict of interest.

List of figures and tables

Figure 1 – Software pipeline for text processing
Figure 2 – Named entity extraction pipeline
Figure 3 – Syntactic analyzer architecture
Figure 4 – Ontology template for representing a Federal Antimonopoly Service case
Table 1 – List of extracted entities
Table 2 – Model evaluation results

References

[1] ***Massel LV***. Knowledge ecosystem as development and specialization of digital ecosystem [In Russian]. Proceedings of the International Scientific and Technical Congress “Intellectual Systems and Information Technologies-2023”. Taganrog: Publisher Stupin S.A., 2023: 155-164.

[2] ***Bolshakova EI, Voroncov KV, Efremova NJe, Klyshinskij JeS, Lukashevich NV, Sapin AS.*** Automatic natural language text processing and data analysis [In Russian]. Moscow: HSE Publishing House. 2017. p. 269.

[3] ***van Ginneken, B.*** Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. Radiol. Phys. Technol. 2017, 10 (1), 23–32. DOI:10.1007/s12194-017-0394-5.

[4] ***Kolmogortsev SV, Sarayev PV.*** Bibliography extraction from texts by regular expressions [In Russian]. New information technologies in automated systems. 2017; (20): 82-88.
[5] ***Bogatyrev MY.*** Fact extraction from natural language texts with conceptual graph models [In Russian]. New information technologies in automated systems. 2016; 7-1: 198-208.
[6] ***Hopcroft JE, Motwani R, Ulman JD.*** Introduction to Automata Theory, Languages and Computation [In Russian]. Moscow: Williams; 2015. 528 p.
[7] ***Sidorova EA, Ivanov AI, Ovchinnikova KA.*** Information extraction from texts based on ontology and large language models [In Russian]. *Ontology of designing*. 2025; 15(1): 114-129. DOI:10.18287/2223-9537-2025-15-1-114- 129.
[8] ***Ma X, Hovy E.*** End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. 2016; 1: 1064-1074. DOI:10.18653/v1/P16-1101.
[9] ***Moens MF.*** Information Extraction from Text and Beyond. Cross-Disciplinary Advances in Applied Natural Language Processing: Issues and Approaches. IGI Global Scientific Publishing. 2012; 24-39. DOI:10.4018/978-1-61350-447-5.ch003.
[10] ***Dong L, Lapata M.*** Language to logical form with neural attention. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. 2016; 1: 33-43. DOI:10.18653/v1/P16-1004.
[11] ***Yin P, Neubig G.*** A syntactic neural model for general-purpose code generation. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017; 1: 440-450. DOI:10.48550/arXiv.1704.01696.
[12] ***Falleri JR, Morandat F, Blanc X, Martinez M, Monperrus M.*** Fine-grained and accurate source code differencing. Proceedings of the 29th ACM/IEEE international conference on Automated software engineering. 2014; 313-324. DOI:10.1145/2642937.2642982.
[13] ***Chiticariu L., Li Y., Reiss F.*** Rule-based information extraction is dead! Long live rule-based information extraction systems! Proceedings of the 2013 conference on empirical methods in natural language processing. 2013; 827-832.
[14] ***Miwa M., Bansal M.*** End-to-end relation extraction using LSTMs on sequences and tree structures. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. 2016; 1: 1105-1116. DOI:10.18653/v1/P16-1105.
[15] ***Zhong Z., Chen D.*** A frustratingly easy approach for entity and relation extraction. Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies. 2021; 50-61. DOI:10.18653/v1/2021.naacl-main.5.
[16] ***Saker M. K., Zhou L., Eberhart A., Hitzler P.*** Neuro-symbolic artificial intelligence: Current trends. Ai Communications. 2022; 34(3): 197-209. DOI:10.48550/arXiv.2105.05330.

About the authors

Sergey Alexeevich Fedosin (b. 1955) graduated from Ogarev Mordovia State University (Saransk, Russia) in 1978; Candidate of Technical Sciences (1983). Professor at the Department of Automated Information Processing and Control Systems, Institute of Electronics and Lighting Engineering, National Research Ogarev Mordovia State University. The list of works includes more than 130 publications in the field of information systems design. Author ID (Scopus): 57205160591; ORCID: 0000-0003-3481-4101; Author ID (RSCI): 101167. *fedosinsa@mrsu.ru*. ✉.

Stanislav Anatolevich Yamashkin (b. 1990) graduated from Ogarev Mordovia State University (Saransk, Russia) in 2012, Candidate of Technical Sciences (2016). Associate Professor at the Department of Automated Information Processing and Control Systems, Institute of Electronics and Lighting Engineering, National Research Ogarev Mordovia State University; Head of the Artificial Intelligence Laboratory at National Research Ogarev Mordovia State University The list of works includes more about 270 publications in the field of systems analysis. ORCID: 0000-0002-7574-0981; Author ID (RSCI): 730394; Author ID (Scopus): 9133286400; Researcher ID (WoS): N-2939-2018. *yamashkinsa@mail.ru.*

Saygin Andrey Alexandrovich (b. 1999) graduated from Ogarev Mordovia State University (Saransk, Russia) in 2023. Postgraduate student at the Department of Automated Information Processing and Control Systems, Institute of Electronics and Lighting Engineering, National Research Ogarev Mordovia State University. The list of works includes more than 10 publications in the field of intelligent data processing. ORCID: 0009-0001-4635-6219; Author ID (RSCI): 1189605. *andrexsai@mail.ru.*

Received June 11, 2026. Revised July 22, 2025. Accepted August 7, 2025.