ВЕСТНИК НАЦИОНАЛЬНОГО ИССЛЕДОВАТЕЛЬСКОГО ЯДЕРНОГО УНИВЕРСИТЕТА “МИФИ”, 2020, том 9, № 4, с. 376–384

ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА

УДК 004.032.26

ОБРАБОТКА НЕСТРУКТУРИРОВАННОЙ ТЕКСТОВОЙ ИНФОРМАЦИИ С ПОМОЩЬЮ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ

© 2020 г. И. А. Лисенков[1,*], В. А. Кузнецов[1], Н. М. Леонова[1]

[1] *Национальный исследовательский ядерный университет “МИФИ”, Москва, 115409, Россия*

**e-mail: ivan.lisenkov@gmail.com*

Поступила в редакцию 07.09.2020 г.
После доработки 07.09.2020 г.
Принята к публикации 12.10.2020 г.

Объем документов, которые обрабатываются в ходе операционных процессов организации, растет с каждым годом. В результате возникает острая потребность в автоматической обработке подобной информации. К сожалению, значительная часть подобных документов является неструктурированной. Форматы неструктурированных документов очень вариативны и сильно отличаются от документа к документу. В этом случае автоматизация, с помощью разработки правил и шаблонов извлечения данных, сложна и неэффективна для последующей поддержки. На практике подобные документы обрабатываются вручную, но это отнимает много времени и ресурсов. В данной работе представлена общая архитектура системы извлечения информации, основанной на современных подходах обработки естественного языка (Natural Language Processing – NLP) и машинного обучения. В статье приведены статистические результаты проведенных экспериментов решения ряда практических задач обработки неструктурированных документов. Представленное решение позволяет обрабатывать большое количество неструктурированной информации без написания кода и подготовки шаблонов синтаксического анализа.

Ключевые слова: извлечение информации, обработка естественного языка, машинное обучение, неструктурированная информация

DOI: 10.1134/S2304487X20040057

1. ВВЕДЕНИЕ

В настоящее время мы повсеместно наблюдаем тенденцию к цифровизации процессов в практически всех сферах деятельности, которая позволяет повысить производительность труда и эффективность производственных процессов. Поэтому большинство организаций стремится уйти от физических бумажных документов к цифровым, и от документов в произвольном формате к формализованным записям: программным интерфейсам (API) или смарт-контрактам [1]. К сожалению, полностью перейти к работе с документами в электронном виде, по крайней мере на данный момент, невозможно. Далеко не все люди, вовлеченные в производство, обеспечены цифровыми устройствами или могут эффективно работать с ними. Кроме этого, принимая во внимание соображения безопасности и существующее законодательство, принятое в большинстве стран, большая часть заключаемых финансовых операций между организациями должна быть продублирована в бумажном виде. Как следствие, большинство крупных организаций и государственные учреждения создают инфраструктуру для поточной обработки и перевода в электронный вид входящих бумажных документов. Согласно [2], можно выделить следующие типовые компоненты данной инфраструктуры:

1. Поточный сканер документов;
2. ПО оптические распознавания текста (OCR);
3. Интеграционный слой (шлюз);
4. Внутренние системы организации.

Современные аппаратные средства сканирования документов показывают достаточно хорошие результаты. Так же на рынке существуют системы оптического распознавания текста (OCR), которые достаточно качественно выделяют текстовый слой из изображения, но OCR-система может предоставить только “набор символов из документа”. Полученная информация не формализована и не имеет семантической связи и структуры, поэтому до передачи данных в целевые системы организации необходимо формали-

зовать информацию, другими словами перевести ее в формат понятный информационным системам организации. В общем виде для этого информацию необходимо преобразовать в формализованный вид, содержащий следующие наборы данных:

• тип информации (для определения, как и в какую систему передавать информацию);

• данные вида “ключ”–“значение”;

• таблицы с описанием колонок и значением ячеек.

На практике в большинстве случае данная задача решается доработкой интеграционных шлюзов систем, путем разработки скриптов или шаблонов разбора текста по ключевым словам и местоположению информации. Такая разработка ведется под конкретную прикладную задачу.

С помощью данного подхода можно достаточно качественно обрабатывать документы фиксированной структуры (счета, чеки, паспорта и т.д.), но на практике существуют документы, структура и формат которых очень сильно меняется от документа к документу (например: законодательный акт, договор об оказании услуг, протокол встречи) или от контрагента к контрагенту (например: акты приемки работ, внутренние учетные документы, прейскурант). В подобных документах есть необходимая информация, но местоположение и ключевые слова могут быть различными. Как следствие, разработка специализированных правил для интеграционного слоя сложна и требует постоянной поддержки разработчиками, что очень затратно по времени и ресурсам. В связи с этим такую интеграцию либо не делают, либо сотрудники организации не доверяют ей. Фактически квалифицированные сотрудники просматривают неструктурированные документы и вручную заносят данные во внутренние системы организации.

2. ПОСТАНОВКА ЗАДАЧИ

Согласно исследованию международной компании Accenture, 80% опрошенных компаний считают, что 80% данных в их бизнес-процессах неструктурированные [3]. Такую же оценку давал Merill Lynch (сейчас Merrill a Bank of America) еще в 1998 году [19]. Если принять к сведению, что согласно исследованиям IDC и Seagate [4] объем обрабатываемых данных увеличится в 5 раз к 2025 году, то данная проблема может быть блокирующей для дальнейшей цифровизации организации.

В данной статье приводятся практические рекомендации по формированию программного решения обработки неструктурированных документов организации с помощью современных подходов обработки естественного языка и машинного обучения. Приводятся результаты проведенных экспериментов для решения практических задач классификации и извлечения данных из неструктурированных документов.

Рассмотрим типовые прикладные задачи, когда возникает потребность в обработке неструктурированных документов, на основе которых, проводились эксперименты и оценивалась эффективность применения современных подходов обработки естественного языка.

Первая задача заключается в обработке входящих первичных бухгалтерских документов. Обработка входящих документов должна осуществляться по мере поступления, данные из документов должны поступать в учетную систему как можно быстрее. Эти документы описывают различные соглашения и финансовые операции между организациями по покупке и продаже товаров и услуг. Типовые документы содержат: наименования организаций, наименование товаров и услуг, стоимости, даты совершения операции и другую сопутствующую информацию. В рамках обработки документов производится отнесение каждого документа к одному из нескольких заранее известных типов документа: Акт, Счет-Фактура, Торговая накладная и другие. Извлечение необходимой информации – это нахождение конкретных участков текстового слоя и их соотнесение с одним из типов информации (ключ) для извлечения. Для некоторых типов документов (прежде всего Акты) структура информации может существенно отличаться от случая к случаю и это трудно автоматизировать. Поэтому обработка некоторой части потока осуществляется вручную. Так как поток входящих документов уже обрабатывался специалистами вручную, то эту ретроспективную информацию можно использовать для формирования полноценных примеров обработки документов для обучения и тестирования алгоритмов работы системы исключая дополнительные трудозатраты на подготовку примеров. Другая практическая задача заключается в извлечении и структуризации информации из архива документов, например, протоколов встреч или внутренних документов организации. Принципиальное отличие от предыдущей задачи заключается в том, что в данном случае нет срочности в обработке – главное провести миграцию с минимальными трудозатратами и ожидае-

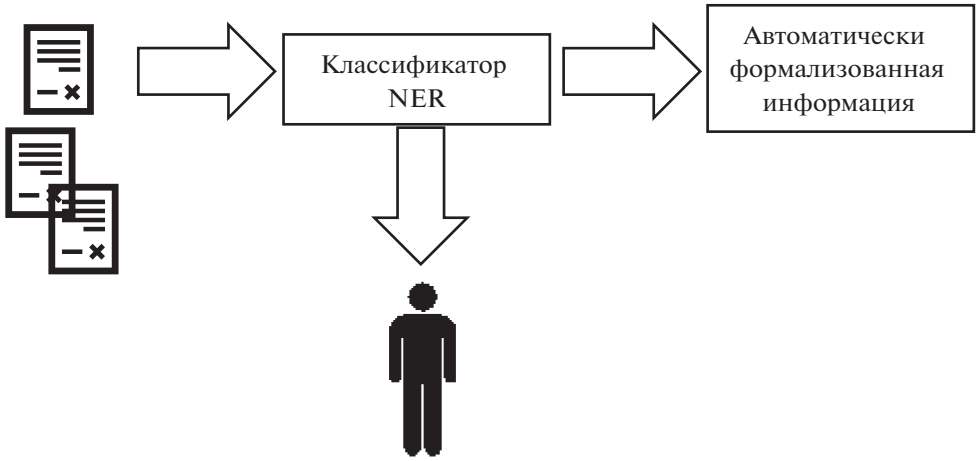

Рис. 1. Схема работы обработки неструктурированного потока документов.

мые сроки. Например, в случае архива протоколов необходимо извлечь следующие данные:

- список участников;
- обсуждаемые вопросы встречи;
- принятые решения;
- некоторые заданные количественные показатели, связанные с принятыми решениями;
- Дополнительная информация.

Еще одной особенностью данной задачи, является то, что каждый документ может содержать множество принятых решений. Каждый документ может содержать более чем одно извлечение каждого типа, или вовсе не содержать искомую информацию в рамках данного эксперимента.

В отличие от первого примера, у заказчика нет ретроспективных данных для подготовки примеров для обучения и валидации. Перед началом процесса миграции эксперты должны подготовить первоначальное множество примеров обработки документов архива. Как следствие, первоначальное множество примеров будет ограниченными, что не позволяет полноценно оценить качество извлечения для всех документов из архива. Тем не менее, можно сформулировать подход статистической оценки качества на ограниченном множестве проверок.

3. НАШ МЕТОД И ПОДХОД

3.1. Общая архитектура решения

Ранее в [6] уже была приведена верхнеуровневая концепция системы проведения исследований с помощью алгоритмов машинного обучения. В данной статье приведено описание практического применения данной концепции для обработки неструктурированных документов, путем применения адаптивных алгоритмов на базе машинного обучения (ML – Machine Learning) и понимания естественного языка (NLP – Natural Language Processing). Основной задачей применения данного подхода является исключение необходимости программировать и создавать шаблоны для обработки документов. Рассматриваемые алгоритмы настраиваются (или обучаются) на основе подготовленных примерах обработки документов. Сотрудники организации не должны быть разработчиками или экспертами в машинном обучении. Для подстройки алгоритмов обработки необходимо предоставлять примеры по извлечению информации и ее классификации, которые охватывают наибольшее количество возможных случаев. В некоторых случаях нет необходимости первоначальной подготовки примеров – можно использовать данные прошлой обработки документов специалистами. Важным преимуществом данного подхода является возможность регулярного дополнительного обучения алгоритмов новыми примерами. Это позволяет автоматически адаптировать обработку к изменениям структуры документов.

В общем, задача обработки неструктурированных документов представляет из себя применение к тексту документа алгоритмов отнесения документа к одной или нескольких категорий (TC – Text Classification) и разметки отдельных элементов текста определенными тэгами или метками (NER – Named Entity Recognition). Фактически с помощью TC мы определяем “Тип информации”, а с помощью NER формируем данные в формате “ключ”–“значение”. Например, рассмотрим задачу извлечения суммы документа из Счета на выполнение работ. Прежде всего система должна отнести документы: “а” к категории

“Иное”, а документ “b” к категории “Счет”, а потом отметить “сумму документа” в тексте документа “b” (см. рис. 1 отмечено желтым).

Таким образом, формируется ансамбль адаптивных алгоритмов, которые обрабатывают неструктурированные документы. Для решения задач TC и NER существует достаточно большой класс адаптивных алгоритмов, построенных на основе классических подходов NLP [7], нейросетевых моделей [8], а также различных эвристических моделей. Например, в [9], [10] приведены подходы использования генетического алгоритма поиска регулярного выражения для разбора текста.

Следует отметить еще одну важную характеристику функционирования алгоритмов – это оценка степени уверенности (действительное число от 0 до 1) отнесения к классу или разметке данных в документе (Confidence Level). С помощью данного показателя можно управлять потоком документов для дополнительной ручной валидации специалистом. В случае Confidence Level ниже заданного порога (∂), документ отправляется на ручную обработку специалистом. Таким образом, можно итеративно получать примеры с новыми случаями для дополнительного обучения и повысить качество обработки потока. Очевидно, что чем ниже Confidence Level, тем больше документов обрабатывается автоматически и тем больше вероятная ошибка алгоритмов при извлечении и классификации. С другой стороны, чем выше Confidence Level, тем больше документов обрабатывается вручную и тем меньше ошибка работы алгоритмов при извлечении и классификации. Таким образом, можно сформулировать два противоположных критерия оценки качества функционирования процесса обработки неструктурированных документов:

• параметр N: неверное извлечение информации из не более N документов по каждому из параметров для каждой M документов;

• параметр K: допускается невозможность извлечения информации по каждому из параметров из не более K документов для каждой M документов, где M – общее количество документов на котором производится оценка N и K.

3.2. Модуль для классификации документов

В рамках задачи отнесения документа к одной или нескольким категориям (TC – Text Classification) входная информация представляет собой текст. Однако для автоматизации процесса классификации необходимо перевести информацию в численный вид. В [11] достаточно подробно описан этот процесс, который так же называется Feature Extraction или Feature Encoding.

Кроме этого, для классификации необязательно названия классов должны присутствовать в тексте: суждение об отнесении документа к классу делается на основе общего вида текста, его структуры и набора ключевых слов, которые наиболее характерны для каждого из классов. Таким образом, для классификации необходимо сформировать уникальный образ документа, по которому можно однозначно принять решение об отнесении к одной из категорий.

Принимая во внимание вышеуказанные особенности, нет необходимости сохранять точную структуру текстового документа и всю текстовую информацию для обработки. Один из простых и практически применимых способов – это использовать типовой подход, применяемый в Language Processing, который называется Bag of Word (BOW) [11, 12]. Принцип данного похода заключается в том, что на основании определенного множества примеров документов, формируется словарь (уникальный перечень) сочетаний слов которые встречаются в документах, так называемые n-граммы. Далее для новых документов подсчитывается количество вхождений в документы n-грамм. С целью повышения качества классификации, перед формированием словаря n-грамм, следует убрать из входного текста, так называемые стоп-слова (stop words). Стоп-слова (иначе называемые шумовыми) – это слова, знаки, символы, которые самостоятельно не несут никакой смысловой нагрузки, желательно такие слова игнорировать при осуществлении ранжирования или индексации N-грамм для уменьшения размерности задачи. Подобная техника используется в SEO-оптимизации сайтов.

Далее, получив числовое представление текстовых документов, для выполнения классификации можно применять следующие алгоритмы машинного обучения [13]:

• деревья решений;

• наивный байесовский классификатор;

• метод k-ближайших соседей;

• модели нейронных сетей (LSTM, Сверточные сети и др.);

• метод опорных векторов [12].

На практике хорошо себя зарекомендовали деревья решений, в частности модель Random Forest [14]. Несмотря на свою достаточную простоту, данный алгоритм хорошо подходит для задачи классификации текстовых документов большого объема принимая во внимание следующие достоинства алгоритма:

• способность эффективно обрабатывать данные с большим числом признаков и классов;

• нечувствительность к масштабированию (и вообще к любым монотонным преобразованиям) значений признаков;

• внутренняя оценка способности модели к обобщению (тест по неотобранным образцам out-of-bag);

• высокая масштабируемость.

Использование функции Softmax вместе с Random Forest позволяет сформировать оценку подобия принадлежности рассматриваемого вектора к каждой из категорий. В этом случае целевая функция классификатора будет следующей $\Phi' : \mathfrak{C} \times \mathfrak{D} \to [0,1]$. Возможность оценки степени подобия определенному классу, позволяет гибко управлять процессом валидации входящих документов. Если значение Φ' больше заданного порога ∂, отнесение к заданному классу можно считать достоверным и автоматически присваивать определенную категорию документу. В противном случае документ требует проверки специалистом и возможной корректировки результатов автоматической классификации. Значение порога ∂ определяется экспериментальным путем в рамках валидации обученной модели классификации и согласно требованиям к качеству классификации и уровню автоматизации процесса. Очевидно, чем больше значение ∂, тем выше качество классификации, но и возрастает количество документов требующих ручной проверки специалистом.

3.3. Модуль для извлечения информации

Как было описано ранее, в рамках процесса обработки неструктурированной информации, необходимо произвести разметку отдельных элементов текста документа определенными тэгами или метками (NER – Named Entity Recognition). Другими словами, из документов определенной категории необходимо извлечь желаемую информацию. Информация о том, какую информацию извлекать и где она должна находиться, определяется подготовленными примерами извлечения из образцов документов, которые используются для обучения моделей NER.

Каждому документу соответствует множество извлечений по каждому набору извлекаемых атрибутов. При этом, важное условие – искомое извлечение должно обязательно однозначно находиться в документе, без преобразований. Если в документе не было обнаружено искомое извлечение, то документ не участвует в обучении моделей извлечения информации.

Подготовленный набор исходных данных для обучения включает в себя: образцы документов, набор атрибутов данных для извлечения и множество значений атрибутов извлечений по каждому образцу документов.

Дальнейшая подготовка заключается в переборе всех извлечений, нахождения однозначного соответствия им в документах. Для таких извлечений проставляются начальные и конечные позиции символов в документе.

Помимо подготовки примеров извлечения, текст документов необходимо обработать с помощью классических методов обработки естественного языка (NLP), состоящий из нескольких этапов [20]:

• удаление стоп-слов и стоп-символов;

• токенизация;

• анализ токенов;

• разметка текстового слоя.

В описываемом практическом подходе не применяется удаление стоп-слов. Это связано со специфичностью задачи. Искомые извлечения могут быть длинными строками, и часть информации может содержаться в стоп-словах и стоп-символах.

Токенизация – это процесс разбиения текста на так называемые N-граммы: слова или комбинации нескольких слов (фразы). Уникальные значения N-грамм формируют словарь, каждой N-грамме присваивается уникальный числовой идентификатор. Основная задача токенизации – преобразовать текстовую информацию в вектор числовых значений.

Помимо идентификатора N-граммы в вектор добавляются дополнительные характеристики текста (Features). Они описывают слово, его регистр, численное значение, пунктуацию, особые символы. Features могут быть значения true/false, число, символ, строка.

В нашем подходе используются следующие атрибуты:

• токен в нижнем регистре;

• токен в верхнем регистре;

• 3 последних символа;

• 2 последних символа;

• токены до и после текущего токена;

• начинается ли токен с заглавной буквы;

• является ли токен числом.

Последний этап подготовки обучающей выборки – это присвоение каждому токену и его атрибутам соответствующего тега. Тег имеет отно-

Таблица 1. Результаты работы модуля по извлечению информации

Описание задачи извлечения	Описание тега	Tag	Precision	Re-Call	F-score
Извлечение информации из первичных бухгалтерских документов	Номер документа	B-DOCNUM	0.996	0.912	0.952
	Дата документа	B-DOCDATE	1.000	0.998	0.999
		I-DOCDATE	1.000	0.997	0.999
	Сумма	B-DOCAMOUNT	0.998	0.996	0.997
		I-DOCAMOUNT	0.996	0.993	0.995
	Контрагент	B-DOCCPTY	1.000	1.000	1.000
		I-DOCCPTY	1.000	1.000	1.000
	Номер договора	B-DOCAGRNUM	1.000	0.902	0.949
	Дата договора	B-DOCAGRDATE	1.000	0.970	0.985
		I-DOCAGRDATE	1.000	1.000	1.000
	Организация	B-DOCCUSTOMER	1.000	1.000	1.000
		I-DOCCUSTOMER	1.000	1.000	1.000
	ИНН	B-INN	1.000	1.000	1.000
Извлечение информации из потока электронных писем по заявкам	Номер заявки	B-CUSTOMER_RFQ	0.934	0.989	0.960
		I-CUSTOMER_RFQ	0.957	0.957	0.957
	Строка товарной части заявки	B-GOODSSTRING	0.971	0.960	0.966
		I-GOODSSTRING	0.967	0.985	0.976

шение к примерам извлечений. В основном, в NER присутствуют три типа тега:

- B-tag – начало искомой информации;
- I-tag – продолжение искомой информации;
- O-tag – прочее, не входит в извлечение.

В зависимости от типа извлекаемой информации, токены именуются различными тегами.

Таким образом, весь текстовый слой одного документа преобразуется в размеченный массив токенов с атрибутами. На основе этого массива можно производить обучение статистической модели, о чем будет сказано далее.

В качестве статистической модели в нашем решении мы используем Conditional Random Fields Model (CRF). Отметим, что данная модель широко используется в NER, например, в таких областях извлечения информации, как медицинские тексты [17, 18], Web-страницы [16] и сельскохозяйственные заметки [15].

4. ЭКСПЕРИМЕНТЫ

В рамках проводимых экспериментов рассматривались следующие практические задачи работы с неструктурированными данными:

– классификация и извлечение из потока электронных писем заявок клиентов;

– классификация и извлечение из сканов первичных бухгалтерских документов;

– классификация разделов многостраничных документов.

4.1. Результаты по извлечению информации

При проведении экспериментов по извлечению информации производилась оценка качества извлечения с помощью F-меры (F-score). F-score, является взвешенной гармонической средней Precision и Re-call, которые в свою очередь определяются в терминах правильных и неправильных решений, сделанных системой [21].

Результаты экспериментов представлены в табл. 1.

4.2. Результаты по классификации информации

Для валидации проводимой классификации используются примеры, не входящие в обучающую выборку. Эти примеры используются для вычисления качества классификации, что может быть вычислено по этой формуле:

$$P = \frac{\text{кол-во верных классификаций } (\Phi' > \partial)}{\text{кол-во всех примеров валидации}}.$$

В то же время уровень автоматизации может быть вычислен следующим образом:

$$A = \frac{\text{кол-во классификаций } (\Phi' > \partial)}{\text{кол-во всех примеров валидации}}.$$

Результаты экспериментов представлены в табл. 2.

Таблица 2. Результаты работы модуля по классификации информации

Название задачи	Точность, P%	Автоматическая обработка, A%	Ручная обработка, %
Классификация потока электронных писем	99.94	86.69	13.31
Классификация первичных бухгалтерских документов	98.31	96.35	3.65
Классификация разделов многостраничных документов	99.35	99.63	0.37

ЗАКЛЮЧЕНИЕ

В рамках данной статьи рассмотрена задача обработки неструктурированных документов и актуальность ее решения. Был предложен подход практического решения задачи на базе адаптивных алгоритмов, который позволяет исключить необходимость программирования шаблонов для извлечения и классификации неструктурированных документов. Было введено понятие степени уверенности функционирования моделей (Confidence level). Также сформулированы критерии по оценке качества обработки неструктурированных документов и описаны практические подходы для извлечения и классификации информации. Проведенные эксперименты применения подхода для решения выбранных практических задач показали точность классификации в среднем более 99% при ручной валидации документов не более 14% от общего объема обрабатываемых документов. При этом средняя оценка F-меры при извлечении данных превышает 0.95. Полученные результаты подтверждают практическую применимость предложенного подхода для обработки неструктурированных документов.

СПИСОК ЛИТЕРАТУРЫ

1. *Allam Z.* On Smart Contracts and Organisational Performance: A Review of Smart Contracts through the Blockchain Technology, Review of Economic and Business Studies. 2018. V. 11 (2). P. 137–156.
2. *Bozzano G.L.* Introduction to Electronic Document Management Systems. 2012.
3. *Nelson P.* Search and unstructured data analytics: 5 trends to watch in 2020. Accenture Search and Content Analytics Blog. 2020.
4. *Shilakes C., Tylman J.* "Enterprise Information Portals". Merrill Lynch. 1998.
5. *Reinsel D., Gantz J., Rydning J.* The Digitization of the World from Edge to Core. An IDC White Paper. 2018.
6. *Lisenkov I.A., Vertakov P.A.* Architecture of the Software Complex for Full Cycle of Research in the Field of Machine Learning. In *Past, Present and Future Science* [Nauka vchera, segodnya, zavtra]. 2017. V. 9 (43). P. 46–56.
7. *Ming Zh.* Progress in Neural NLP: Modeling, Learning, and Reasoning. ELSEVIER Engineering. 2020. V. 6 (3). P. 275–290.
8. *Li J., Sun A., Han J., Li C.* A Survey on Deep Learning for Named Entity Recognition. In *IEEE Transactions on Knowledge and Data Engineering*. 2020. P. 1–20.
9. *Bartoli A., De Lorenzo A., Medvet E., Tarlao F.* Inference of Regular Expressions for Text Extraction from Examples. In IEEE Transactions on Knowledge and Data Engineering. 2016. V. 28 (5). P. 1217–1230.
10. *Kuznetsov V.A., Lisenkov I.A.* Application of Genetic Algorithm for Information Extraction. In Advanced innovative developments. Prospects and experience of application, problems of implementations in production [Peredovye innovacionnye razrabotki. Perspektivy i opyt ispol'zovanija, problemy vnedrenija v proizvodstvo]. 2019. V. 2. P. 232–234.
11. *Goldberg Y.* Neural Network Methods in Natural Language Processing. Morgan & Claypool, 2017. P. 65.
12. *Nello Cristianini, John Shawe-Taylor.* An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. – Cambridge University Press, 2000. – ISBN 978-1-139-64363-4.
13. *Daniel T.* Larose, Discovering Knowledge in Data: An Introduction to Data Mining (https://web.archive.org/web/20140531051709/http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0471666572.html)
14. *Breiman L., Friedman J., Olshen R., and Stone C.* Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.
15. De-sheng WANG, Jun-zhi LIU, A-xing ZHU, Shu WANG, Can-ying ZENG, Tian-wu MA, Automatic extraction and structuration of soil–environment relationship information from soil survey reports, Journal of Integrative Agriculture, 18, Issue 2, 2019, 328–339.
16. *Etzioni, Oren & Cafarella, Michael & Downey, Doug & Popescu, Ana-Maria & Shaked, Tal & Soderland, Stephen & Weld, Daniel & Yates, Alexander.* Unsupervised named-entity extraction from the Web: An experimental study. Artificial Intelligence. 2005. V. 165. P. 91–134.
17. *Anupama Gupta, Imon Banerjee, Daniel L. Rubin*, Automatic information extraction from unstructured mammography reports using distributed semantics, Journal of Biomedical Informatics. 2018. V. 78. P. 78–86.
18. *Omid Ghiasvand, Rohit J. Kate*, Learning for clinical named entity recognition without manual annotations, Informatics in Medicine Unlocked. 2018. V. 13. P. 122–127.
19. *Shilakes C., Tylman J.* (1998). "Enterprise Information Portals". Merrill Lynch.

20. *Belerao K.* Tweet Segmentation for Named Entity Recognition. Journal of Artificial Intelligence Research. 2017. V. 3. P. 22–25.

21. *Tharwat A.* "Classification assessment methods", Applied Computing and Informatics, Vol. ahead-of-print No. ahead-of-print. 2020.

Vestnik Natsional'nogo issledovatel'skogo yadernogo universiteta "MIFI", 2020, vol. 9, no. 4, pp. 376–384

Processing Unstructured Text Information Using by Machine Learning Algorithms

I. A. Lisenkov[a], V. A. Kuznetsov[a], and N. M. Leonova[a,#]

[a] *National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, 115409 Russia*

[#] *e-mail: ivan.lisenkov@gmail.com*

Received September 7, 2020; revised September 7, 2020; accepted October 12, 2020

Abstract—The number of documents processed during the organization's operational processes is dramatically growing from year to year. As a result, there is a strong demand for automatic processing of such information. Unfortunately, an essential part of such documents is unstructured information. Unstructured document formats are very variable and strongly different from document to document. In this case, automation, through the rule based coding and parsing templates is quite complex and ineffective for further support. It could be analyzed manually, but it is time-consuming and resource intensive. This paper presents the general architecture of an information extraction system based on modern approaches to Natural Language Processing (NLP) and machine learning. The article presents the statistical results of the experiments carried out to solve a number of practical problems of processing unstructured documents. The presented solution allows to process a great amount of unstructured information without coding and preparing parse templates.

Keywords: information extract, natural language processing, deep learning, unstructured information

DOI: 10.1134/S2304487X20040057

REFERENCES

1. Allam Z. On Smart Contracts and Organisational Performance: A Review of Smart Contracts through the Blockchain Technology, Review of Economic and Business Studies. 2018. 11(2). 137–156.
2. Bozzano G.L. Introduction to Electronic Document Management Systems. 2012.
3. Nelson P. Search and unstructured data analytics: 5 trends to watch in 2020. Accenture Search and Content Analytics Blog. 2020.
4. Shilakes C., Tylman J. "Enterprise Information Portals". Merrill Lynch. 1998.
5. Reinsel D., Gantz J., Rydning J. The Digitization of the World from Edge to Core. An IDC White Paper. 2018.
6. Lisenkov I.A., Vertakov P.A. Architecture of the Software Complex for Full Cycle of Research in the Field of Machine Learning. In *Past, Present and Future Science* [Nauka vchera, segodnya, zavtra]. 2017. 9(43), 46–56.
7. Ming Zh. Progress in Neural NLP: Modeling, Learning, and Reasoning. ELSEVIER Engineering, 2020. 6(3), 275–290.
8. Li J., Sun A., Han J., Li C. A Survey on Deep Learning for Named Entity Recognition. In *IEEE Transactions on Knowledge and Data Engineering*, 2020. 1–20.
9. Bartoli A, De Lorenzo A., Medvet E., Tarlao F. Inference of Regular Expressions for Text Extraction from Examples. In IEEE Transactions on Knowledge and Data Engineering, 2016. 28(5), 1217–1230.
10. Kuznetsov V.A., Lisenkov I.A. Application of Genetic Algorithm for Information Extraction. In Advanced innovative developments. Prospects and experience of application, problems of implementations in production [Peredovye innovacionnye razrabotki. Perspektivy i opyt ispol'zovanija, problemy vnedrenija v proizvodstvo], 2019. 2, 232–234.
11. Goldberg Y. Neural Network Methods in Natural Language Processing. Morgan & Claypool, 2017. P. 65.
12. Nello Cristianini, John Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. – Cambridge University Press, 2000. – ISBN 978-1-139-64363-4.
13. Daniel T. Larose, Discovering Knowledge in Data: An Introduction to Data Mining (https://web.archive.org/web/20140531051709/http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0471666572.html)
14. L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.

15. De-sheng WANG, Jun-zhi LIU, A-xing ZHU, Shu WANG, Can-ying ZENG, Tian-wu MA, Automatic extraction and structuration of soil–environment relationship information from soil survey reports, Journal of Integrative Agriculture, 18, Issue 2, 2019, 328–339.

16. Etzioni, Oren & Cafarella, Michael & Downey, Doug & Popescu, Ana-Maria & Shaked, Tal & Soderland, Stephen & Weld, Daniel & Yates, Alexander. Unsupervised named-entity extraction from the Web: An experimental study. Artificial Intelligence. 165. 2005. 91–134.

17. Anupama Gupta, Imon Banerjee, Daniel L. Rubin, Automatic information extraction from unstructured mammography reports using distributed semantics, Journal of Biomedical Informatics, 78, 2018, 78–86.

18. Omid Ghiasvand, Rohit J. Kate, Learning for clinical named entity recognition without manual annotations, Informatics in Medicine Unlocked, 13, 2018, 122–127.

19. Shilakes C., Tylman J. (1998). "Enterprise Information Portals". Merrill Lynch.

20. Belerao K. Tweet Segmentation for Named Entity Recognition. Journal of Artificial Intelligence Research. 2017. 3. 22–25.

21. Tharwat A. "Classification assessment methods", Applied Computing and Informatics, Vol. ahead-of-print No. ahead-of-print. 2020.