ISBN 5-94052-066-0 ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ. ПЕРЕСЛАВЛЬ-ЗАЛЕССКИЙ, 2004

УДК 519.767.6

Д. А. Кормалев

# Приложения методов машинного обучения в задачах анализа текста

Аннотация. Статья посвящена применению машинного обучения в задачах анализа текста. Особое внимание уделяется подходам, которые можно эффективно использовать для извлечения информации из текста на естественном языке. Рассмотрены различные этапы и уровни анализа текста и возможность использования методов машинного обучения на каждом из них.

*Ключевые слова и фразы:* Извлечение информации, машинное обучение, анализ текста.

## Введение

Задача извлечения информации заключается в обработке текста на естественном языке с целью извлечения заданных элементов. На входе системы извлечения информации — слабоструктурированный или неструктурированный текст на естественном языке; на выходе — заполненные структуры данных (экзофреймы), позволяющие проводить дальнейшую автоматическую или ручную обработку информации. Извлечение информации можно рассматривать как особый вид аннотирования текстов, когда в роли аннотации выступает специфическая структура данных.

Информация, извлеченная из текста, хранится в экзофрейме, который представляет собой набор целевых слотов. Целевой слот может содержать информацию об объектах (например, персоналии, организации, продукты), отношениях или событиях, их атрибутах, также возможна привязка к фрагменту текста, на основании которого получена данная информация.

Релевантная информация должна быть определена абсолютно точно для того, чтобы автоматическая система извлечения информации показывала хорошие результаты. Хорошей постановкой задачи можно считать такую, для которой согласованность результатов выделения информации вручную для нескольких экспертов предметной области (*inter-annotator agreement*) будет высокой (более 90%). Если же ключевая информация скрыта или настолько плохо определена

в тексте, что несколько человек не могут одинаково ее определить, то автоматическая система анализа тем более не сможет выполнить эту задачу.

Для качественной работы автоматической системы в конкретной предметной области ей необходимо обладать значительными знаниями в этой области. Каждая предметная область предполагает извлечение данных различного характера, свой специфический профессиональный словарь и стиль написания текста.

Каждая конкретная задача извлечения данных из текста предусматривает слоты разных видов: для событий, персоналий, организаций, дат и т. д. Целевой фрейм и правила извлечения информации описывают условия, при которых создается экзофрейм и способ заполнения его слотов.

Рассмотрим типичное применение системы извлечения информации. Задается массив текстов, в каждом из которых потенциально присутствует описание некоторого объекта или события предметной области. Например, это может быть подборка новостей, в которых может встречаться информация о появлении новых товаров на рынке. Другой пример — набор домашних страниц сотрудников какой-либо организации. Помимо этого задано определение целевой информации (можно рассматривать его как список вопросов, относящихся к предметной области). Для каждого текста из массива на основании определения целевой информации требуется выделить ответы на вопросы в виде фрагмента текста. Для подборки новостей целью может быть обнаружение названия товара, названия фирмы-производителя и даты появления товара на рынке; для домашних страниц — обнаружение имени владельца страницы, его домашнего адреса и подразделения, в котором он работает.

На всех этапах обработки текста на естественном языке присутствует неопределенность, которая разрешается разными средствами. Большую проблему представляет построение словарей, тезаурусов, онтологий. Эта работа по большей части выполняется вручную. Попытки автоматизации данной работы проводились с использованием статистических методов и методов машинного обучения. На настоящий момент, по-видимому, нет свободно доступных исследований, описывающих всестороннее решение этой проблемы.

Применение методов машинного обучения может упростить настройку и разработку систем извлечения информации и облегчить

переключение всей системы на новую предметную область. Рассмотрим уровни анализа текста в целом (раздел 1), а затем возможность и эффективность использования применения машинного обучения для контекстного снятия омонимии (раздел 2.1), синтаксического анализа (раздел 2.2), определения семантических классов (раздел 2.3), построения правил извлечения информации и объединения частичных результатов (раздел 2.4), разрешения кореферентности (раздел 2.5). В разделе 3 будет приведена краткая характеристика эталонной модели *TIPSTER* и рассмотрена ценность данной модели для задач извлечения информации в целом и, в частности, для построения правил извлечения информации.

## 1. Общая схема обработки текста при извлечении информации

Системы извлечения информации используют во многом сходные методы. Обратимся к типичной последовательности обработки текста в задачах извлечения информации. Сразу будем отмечать этапы обработки, для которых было бы полезно использовать машинное обучение. К ним относятся, в первую очередь, те этапы, которые требуют тонкой настройки в конкретных приложениях.

Исходный текст подвергается *графематическому анализу*; происходит выделение слов и предложений. На следующих этапах происходит обнаружение составных слов, которые должны рассматриваться как одно (с точки зрения морфологического анализатора). Графематический анализ обычно не требует настройки, зависящей от предметной области, поскольку реализация общего алгоритма графематического анализа подходит для большинства реальных приложений.

*Морфологический анализ* обычно работает на уровне отдельных слов (возможно, составных) и возвращает морфологические атрибуты данного слова. В случае, когда атрибуты не могут быть установлены однозначно, возвращается несколько возможных вариантов морфологического анализа. Использование методов машинного обучения для морфологического анализа не принесет пользы, так как существует множество высококачественных словарных и бессловарных решений этой задачи, которые могут применяться в широком спектре приложений.

Результаты морфологического анализа используются при *микро-* и *макросинтаксическом анализе*. Микросинтаксический анализ осуществляет построение ограниченного набора синтаксических связей

(например, выделение именных групп). Задача макросинтаксического анализа состоит в выделении в предложении крупных синтаксических единиц — фрагментов — и в установлении иерархии на множестве этих фрагментов. Разбиение на микро- и макросинтаксический анализ условно, оно отображает тот факт, что для большинства задач извлечения информации достаточно поверхностного (микросинтаксического анализа).

Эксперименты показывают, что лингвистический анализатор, обладающий богатыми выразительными возможностями, дает больше ошибок из-за того, что почти каждый уровень анализа представляет собой задачу, которая не имеет строгого, а тем более формализуемого, решения. В наибольшей мере это относится к синтаксическому анализу. Поэтому в предметной области, где достаточно простого синтаксического анализа, мощный анализатор будет лишь вносить нежелательный шум, а производительность будет падать. В то же время существуют предметные области, в которых для извлечения информации требуются развитые возможности представления лингвистической информации. В таких предметных областях примитивный анализатор не сможет предоставить необходимых для извлечения целевой информации лингвистических атрибутов. Настройка выполняется вручную, поэтому данный этап анализа выиграл бы от применения машинного обучения.

Поскольку у каждого слова после выполнения морфологического анализа может присутствовать несколько омонимичных словоформ, то для улучшения качества синтаксического анализа и повышения его производительности можно использовать алгоритмы *устранения омонимии*, которые сокращают количество вариантов морфологического анализа. Часто задача снятия омонимии решается при помощи наборов правил, составление которых очень трудоемко, поскольку практически применимые наборы оказываются довольно крупными. Кроме того, для каждой предметной области набор правил приходится модифицировать. Снятие омонимии — еще одна область анализа текста, которая может быть улучшена при помощи машинного обучения.

В дальнейшем происходит *выделение семантических классов* (составных типов). При выделении составных типов осуществляется пометка фрагментов текста, которые позже (например, при применении правил) рассматриваются как единое целое (например, даты, имена, должности). Выделение семантических классов осуществляется

на основе тезаурусов или правил, подобных правилам извлечения информации. Оба варианта представляют интерес с точки зрения методов машинного обучения. Первый, к сожалению, практически невозможно автоматизировать, а второй мы рассмотрим ниже.

Затем осуществляется *применение правил* извлечения информации к тексту. При выполнении условий и ограничений, описанных в правилах, выполняется функциональная часть правил. Функциональная часть позволяет строить целевые структуры данных или сохранять дополнительную информацию, которая будет использована на последующих этапах. Чаще всего правила группируются по фазам: правила последующих фаз имеют доступ к информации, порожденной правилами предыдущих. Построение и тестирование наборов правил извлечения информации, особенно для сложной предметной области — трудоемкая задача, для которой предлагается ряд удовлетворительных решений с применением машинного обучения.

Целевые фреймы могут быть подвергнуты дополнительной обработке с целью повышения качества работы системы. Для этого используются средства разрешения кореферентности и объединения частичных результатов.

При *разрешении кореферентности* в целевых фреймах особым образом помечаются объекты, которые описываются разными фрагментами текста, но указывают на одну сущность реального мира. Исследования показывают, что нет общего решения проблемы кореферентности, однако существуют общие подходы, которые приемлемо работают во множестве предметных областей[1], но требуют настройки при переходе от одной области к другой, следовательно здесь также потенциально может быть использовано машинное обучение.

*Объединение частичных результатов* заключается в поиске частично заполненных целевых фреймов и принятии решения о возможности объединения результатов. В случае, когда объединение возможно, из нескольких целевых фреймов собирается один, обладающий более полной информацией, чем каждый из исходных. Объединение частичных результатов не имеет общего решения, как и ряд перечисленных выше проблем, а требует настройки на предметную область. Особенность этого этапа заключается в том, что есть ряд подходов, реализующих алгоритмы из области машинного обучения

[1] В основном, для английского языка.

и близкие к ним (часто статистические), но помимо настройки параметров алгоритма, требуется *выбор* алгоритма для каждой предметной области и его творческая «доводка» для решения конкретной задачи. Алгоритмы построения правил объединения частичных результатов часто сходны с алгоритмами построения правил извлечения информации.

## 2. Машинное обучение на разных уровнях анализа текста

**2.1. Контекстное снятие омонимии.** Качество морфологического анализа можно повысить при помощи контекстного анализа. Это позволит в большинстве случаев избавиться от морфологической омонимии. Модуль контекстного анализа можно настраивать на произвольную предметную область. Для этого необходимо обучающей программе модуля предоставить множество текстов — документов целевой предметной области. На этом множестве обучающая программа выделит наиболее характерный контекст для значимых с точки зрения омонимии слов и будет использовать его в дальнейшем для разрешения омонимической неоднозначности.

Контекстный анализ, по-видимому, не решит всех проблем омонимии для русского языка. Например, в русском языке у многих существительных совпадает написание в винительном и именительном падежах (при этом возможный контекст лексемы практически не изменяется); то же касается имен собственных. Но существует множество случаев, когда контекстный анализ отсеет нерелевантные омонимы.

Зарубежные аналоги [1–3] показывают высокую точность работы морфологических процессоров при использовании технологии, основанной на скрытых Марковских моделях и правилах специального вида. Существуют реализации как для супервизорного обучения, так и для обучения «без учителя».

**2.2. Синтаксический анализ.** Для использования машинного обучения при синтаксическом анализе требуется тщательная разметка больших объемов текстов, поэтому супервизорное обучение применять неперспективно.

Эксперименты по настройке синтаксического анализатора с применением машинного обучения «без учителя», показывают, что синтаксическая структура естественного языка слишком выразительна

и сложна, чтобы можно было эффективно строить его модель, не располагая размеченными текстами.

Если говорить о практической стороне, то для реализации синтаксического анализа с использованием машинного обучения «без учителя» самым эффективным подходом представляется статистическое обучение, когда выделение синтаксических структур производится без использования лингвистических знаний. Вместо этого можно подсчитывать частоты совместной встречаемости слов. Подобный подход (для русского языка) был исследован в [4], но и там значительное место занимают жестко заложенные в систему формально-грамматические правила.

Тем не менее, очень значимым для качества работы системы будет адаптивный синтаксический анализ. В зависимости от задач, которые мы хотим решать, не всегда рационально использовать всю мощность синтаксического анализатора. Иногда бывает достаточно разобрать лишь те характеристики предложения, которые нам требуются с прикладной точки зрения и имеют меньшую вероятность ошибки при разборе. Тогда работу синтаксического анализатора можно будет модифицировать в соответствии с прикладными целями (в том числе и средствами машинного обучения).

**2.3. Определение семантических классов.** Важным свойством для системы извлечения информации является ее способность определять семантические классы фрагментов текста. Набор семантических классов может включать в себя разные составляющие — от примитивных вариантов (например, определение дат) до выделения именованных сущностей и определения их класса (например, «Организация», «Персона», «Должность»). Это позволит при задании правил извлечения информации оперировать не отдельными словами и их взаимосвязями, а сущностями, характерными для предметной области.

Машинное обучение в этом контексте, скорее всего, возможно только в супервизорном варианте, поскольку применение кластеризации на множестве семантических классов приведет к результатам, с трудом воспринимаемыми человеком.

**2.4. Правила извлечения информации и объединения результатов.** При извлечении информации ключевым моментом является построение набора шаблонов, или правил, позволяющих определить расположение релевантной информации в тексте и правильно заполнить целевую структуру данных. Построение правил извлечения информации — процесс довольно трудоемкий. Это связано с известным «эффектом хвоста». Эффект хвоста заключается в том, что небольшое количество правил обеспечивают приемлемое качество работы системы, но попытки дальнейшего улучшения качества работы приводят к добавлению большого количества правил. Помимо трудоемкости добавления множества правил, появляется проблема нестабильности системы из-за возможной корреляции между правилами.

В последние 15 лет было проведено довольно много исследований в этой области.

Исторический интерес представляют системы *AutoSlog* и *AutoSlog-TS* [5] — средства для автоматического порождения лексикона предметной области. Второе из них особенно интересно тем, что применяется обучение «без учителя». Дальнейшее развитие идеи *AutoSlog* получили в системе *LIEP* [6]. В этой системе уже используются синтаксические связи между фрагментами текста, новые правила порождаются из старых посредством обобщения.

Интересны исследования С. Содерлэнда, отразившиеся в системах *CRYSTAL* [7] и *WHISK* [8]. Первая из них работает с неструктурированным текстом и порождает правила, позволяющие использовать при их применении лексические, морфологические, синтаксические атрибуты, а также принадлежность фрагмента текста к тому или иному семантическому классу. Правило *WHISK* представляет собой особый вид регулярного выражения. Семантические классы также описываются регулярными выражениями.

Для анализа частично структурированного текста были разработаны системы *SRV* [9] и *RAPIER* [10]. Обе системы используют идеи индуктивного логического программирования. В первом случае порождаются собственно хорновские фразы. Предикаты, составляющие правила, предопределены. Алгоритм обучения — покрывающий. Во втором случае те же самые идеи применяются для построения шаблонов особого вида, учитывающих контекст. Каждая итерация алгоритма состоит из 3 шагов: обобщение, специализация контекста слева и специализация справа.

Из новых разработок следует отметить систему *Brief Driven Information Retrieval and Extraction for Strategy (BRIEFS)* [11], позволяющую извлекать заданную целевую информацию из массива текстов. Система построена на основе платформы для разработки *GATE (General Architecture for Text Engineering)* [12], которая сама по себе представляет огромный интерес для разработчиков систем извлечения информации и других приложений анализа текстов. Процесс извлечения основан на правилах. Правила определяют, каким образом информация из текста будет извлекаться для заполнений экзофреймов. Правила используют лингвистическую информацию о тексте, полученную на этапе лингвистического анализа.

Идеальная система извлечения информации должна стремиться к системе с естественно-языковым интерфейсом или, по крайней мере, процесс настройки на новую предметную область должен быть по силам специалисту предметной области, не обладающему навыками программирования или специальными знаниями в области обработки текстов.

Чтобы приблизиться к такому идеалу, предполагается применять методы машинного обучения для полуавтоматической (в отдельных случаях — автоматической) настройки на произвольную предметную область и разнотипную целевую информацию. Для этого методами машинного обучения на основе обучающей выборки должен порождаться набор правил извлечения информации. После обучения правила нужно протестировать и, возможно, модифицировать. Поэтому представляется полезным создание интерактивной среды, которая будет интегрировать в себе этапы обучения, тестирования и модификации. В целях повышения интерактивности и облегчения задачи предварительной разметки текстов можно применить активное обучение, когда очередной пример из неразмеченного множества будет выбираться самой системой на основании определенной стратегии и предлагаться пользователю для разметки. Возможен вариант, когда предварительная разметка производится уже самой системой, на основании порожденных ранее правил. Тем самым можно достичь снижения объема рутинного труда по разметке текстов.

**2.5. Разрешение кореферентности.** По определению, *референция* — сопоставление тех или иных языковых сущностей сущностям внеязыковым. Соответственно, явление *кореферентности* заключается в сопоставлении нескольких (чаще всего, различных) языковых сущностей одной внеязыковой. Частным случаем кореферентной связи является *анафора*, то есть использование языковых выражений, которые могут быть проинтерпретированы лишь с учетом другого, как правило, предшествующего, фрагмента текста (*антецедента*). Наиболее востребованная задача — разрешение кореференции именных групп.

Разрешение кореферентных связей — одна из задач извлечения данных из текстов на естественном языке. При извлечении информации данные по кореферентно связанным сущностям, как правило, объединяются, следовательно, успешное решение данной задачи улучшит качество объединения частичных результатов.

Задача разрешения кореферентности относится к постобработке (с точки зрения базовой системы извлечения информации). Тем не менее, такой компонент представляет большую ценность для системы извлечения информации, потому что позволяет представлять извлеченные данные в унифицированной форме.

Существующие системы [13–17] демонстрируют качество работы от 40% до 85% $F$-метрики для различных текстов (для английского языка). Самый высокий результат показывает система *Resolve*[15] — около 85%. Причина этого, по всей видимости, состоит в использовании всех видов лингвистической информации, в том числе и семантической. Кроме этого в *Resolve* используется комбинированный метод: помимо алгоритма, основанного на машинном обучении, возможна реализация алгоритма разрешения кореферентных связей вручную.

## 3. Эталонная модель *TIPSTER*

Проект *TIPSTER* [18] — инициатива *DARPA (Defense Advanced Research Projects Agency)* по развитию технологий обработки текста. В проекте совместно участвовали исследователи и разработчики научных, промышленных и государственных организаций США. Формально проект завершился осенью 1998 года из-за прекращения финансирования.

*TIPSTER* был ориентирован на решение трех основных задач:

- обнаружение документов — выявление документов, представляющих потенциальный интерес для пользователя, в массиве данных или в текстовом потоке;
- извлечение информации — выделение фрагментов текста, содержащих релевантную информацию, и, возможно, преобразование их в реляционную форму;
- аннотирование — сжатие объема документа за счет выделения наиболее важных фрагментов с сохранением основных идей, изложенных в нем.

Одним из основных результатов проекта стала разработка эталонной архитектуры и идеологии построения систем обработки текста. Проектная документация доступна в сети.

Эталонная модель определена в ряде документов, особый интерес представляет описание архитектуры системы обработки текста. Эталонная модель подразумевает ссылочное аннотирование текста (в противовес аддитивному). Любая лингвистическая информация представляется в виде аннотации. Аннотация сопоставляется фрагменту текста, принадлежит классу аннотаций и обладает атрибутами. Классы и атрибуты аннотаций намеренно не специфицированы в эталонной модели, чтобы можно было применить любой формализм для представления лингвистической или другой информации о тексте. Аннотации добавляются в процессе работы модулей системы, при этом новые аннотации могут строиться на основании полученных на предыдущих этапах анализа.

Аннотации не позволяют в явном виде указывать связи между фрагментами текста, но с этой проблемой можно справиться двумя способами:

(1) использовать расширение, позволяющее идентифицировать аннотации, тогда связи между аннотациями можно выразить через атрибуты аннотаций (в этом случае возможно использование модели дерева зависимостей);
(2) не расширять эталонную модель, а вместо этого воспользоваться представлением структуры в виде системы составляющих, а не дерева зависимостей.

Важное и интересное средство построения и наполнения аннотаций, используемое в *TIPSTER*, — язык *CPSL (Common Pattern Specification Language)*, которые позволяет проводить аннотирование текста на основе сопоставления шаблонов. Перед выполнением шаблоны

преобразуются в конечные автоматы. Левая часть правила описывает контекст и условия, в которых правило должно сработать; правая часть — функциональная, в ней содержатся инструкции по работе с аннотациями. Существуют различные расширения *CPSL* — от использования функций в левой части правил, до использования фрагментов кода (например, на языке *Java*) в правой части.

Сейчас многие библиотеки и приложения обработки текста основываются на эталонной архитектуре или совместимы с ней, общая модель системы, правила *CPSL* и ссылочный подход к аннотированию текстов стали стандартом *de facto* для современных систем обработки текстов на естественном языке. Ценным свойством *CPSL* является то, что существуют алгоритмы построения детерминированных конечных автоматов по правилам (компиляция правил). Конечные автоматы обладают высокой вычислительной эффективностью; их выразительная мощность достаточна для большинства приложений извлечения информации. Кроме того, возможно расширение правил, например, логическими средствами, что даст значительный прирост выразительных возможностей без особого ущерба производительности.

Учитывая приведенные выше особенности и достоинства эталонной модели, надо ориентироваться на построение правил извлечения информации, оперирующие в этой парадигме. Несмотря на недостатки классического *CPSL*, этот язык, с учетом расширений, является мощным стандартным средством описания правил извлечения информации.

## Заключение

Можно сделать вывод о том, что применение классических супервизорных алгоритмов машинного обучения часто не подходит для приложений, связанных с анализом текста по причине трудоемкости разметки. Алгоритмы обучения «без учителя», в свою очередь для большинства приложений дают низкие результаты.

Перспективное направление исследований — индуктивное построение правил *CPSL* (для извлечения информации) или другого языка с не меньшими выразительными средствами (для объединения результатов). При этом следует обратить внимание на следующие аспекты:

- использование активного обучения;

- повышение степени интерактивности;
- использование гибридных методик обучения;
- самонастройка (*bootstrapping*) — подход, основанный на том, что результаты, полученные на некоторой итерации обучения используются для подготовки входных данных следующей итерации.

При этом можно использовать достижения из области индуктивного логического программирования, а также совмещение нисходящего и восходящего подходов, когда одновременно для каждой гипотезы рассматриваются две границы — наиболее общая и наиболее специализированная, с учетом просмотренных примеров.

## Список литературы


[1] Cutting D., Kupiec J., Pedersen J., Sibun P. *A Practical part-of-speech tagger* // Proceedings of the Third Conference on Applied Natural Language Processing, 1992. ↑2.1

[2] Yarowsky D. *Unsupervised Word Sense Disambiguation Rivaling Supervised Methods* // Meeting of the Association for Computational Linguistics, 1995, с. 189–196. ↑

[3] Brill E. *Unsupervised Learning of Disambiguation Rules for Part of Speech Tagging* // Proceedings of the Third Workshop on Very Large Corpora ред. David Yarovsky and Kenneth Church. — Somerset, New Jersey: Association for Computational Linguistics, 1995, с. 1–13. ↑2.1

[4] Брик А. В.. 2002. *Исследование и разработка вероятностных методов синтаксического анализа текста на естественном языке*, Диссертация на соискание ученой степени кандидата технических наук, МГТУ им. Н. Э. Баумана. ↑2.2

[5] Riloff E. *Information Extraction as a Stepping Stone toward Story Understanding*. — Montreal, Canada: MIT Press, 1999. ↑2.4

[6] Huffman S. B. *Learning information extraction patterns from examples* // Learning for Natural Language Processing, 1995, с. 246–260. ↑2.4

[7] Soderland S. E. 1996. *Learning Text Analysis Rules for Domain-specific Natural Language Processing*, University of Massachusets. ↑2.4

[8] Soderland S. E. *Learning Information Extraction Rules for Semi-Structured and Free Text* // Machine Learning. — **34**, № 1–3, с. 233-272. ↑2.4

[9] Freitag D. *Information Extraction from HTML: Application of a General Machine Learning Approach* // AAAI/IAAI, 1998, с. 517–523. ↑2.4

[10] Califf M. E. 1998. *Relational Learning Techniques for Natural Language Extraction*, Department of Computer Sciences, University of Texas, Austin, TX. ↑2.4

[11] Keijola M. 2003. *On Smart and Natural Language Technology Support of Strategy Work*, Helsinki University of Technology, Helsinki, Finland. ↑2.4

[12] Cunningham H., Humphreys K., Gaizauskas R., Wilks Y. *GATE — a TIPSTER-based General Architecture for Text Engineering* // Proceedings of the TIPSTER Text Program (Phase III) 6 Month Workshop DARPA: Morgan Kaufmann, California, 1997. ↑2.4

[13] Soon W. M., Ng H. T., Lim D. C. Y. *A machine learning approach to coreference resolution of noun phrases* // Computational Linguistics. — **27(4)**, с. 521–544. ↑2.5

[14] Aone Ch., Bennett S. *Evaluating Automated and Manual Acquisition of Anaphora Resolution Strategies* // Meeting of the Association for Computational Linguistics, 1995, с. 122–129. ↑

[15] McCarthy J. F., Lehnert W. G. *Using Decision Trees for Coreference Resolution* // IJCAI, 1995, с. 1050–1055. ↑2.5

[16] Dimitrov M. 2002. *A Light-weight Approach to Coreference Resolution for Named Entities in Text*, MSc Thesis, University of Sofia, Bulgaria. ↑

[17] Kehler A. *Probabilistic Coreference in Information Extraction* // Proceedings of the Second Conference on Empirical Methods in Natural Language Processing ред. Claire Cardie and Ralph Weischedel. — Somerset, New Jersey: Association for Computational Linguistics, 1997, с. 163–173. ↑2.5

[18] Grishman R. // TIPSTER Text Architecture Design. Version 3.1. — New York, NYU, 1998. ↑3

Исследовательский центр искусственного интеллекта ИПС РАН

D. A. Kormalev. *Applications of Machine Learning to Text Analysis.* (in russian.)

Abstract. The article addresses the use of machine learning approaches to natural language processing. Special emphasis is made on the methods and approaches that can be used for information extraction tasks. Various stages of text analysis are considered from machine learning perspective.