

DOI: 10.5862/JCSTCS.229.7

УДК 004

К.В. Соколов, Д.А. Тимофеев, А.В. Самочадин

ИЗВЛЕЧЕНИЕ ОПИСАНИЙ БИЗНЕС-ПРОЦЕССОВ ИЗ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

K.V. Sokolov, D.A. Timofeev, A.V. Samochadin

EXTRACTION OF BUSINESS PROCESS MODELS FROM TEXTS

Для автоматизации бизнес-процессов необходимо наличие их формальных моделей. Создание таких моделей – трудоемкий процесс, требующий привлечения квалифицированных экспертов. В то же время для многих процессов уже существуют описания на естественном языке в виде регламентов, инструкций, положений и других официальных документов, а также в виде менее формализованных описаний, составляемых участниками процессов. Рассмотрена методика извлечения описаний бизнес-процессов из набора текстов и используемые для этого алгоритмы. Отличительной особенностью предлагаемого подхода является применение методов глубинного анализа процессов (process mining).

БИЗНЕС-ПРОЦЕССЫ; ИЗВЛЕЧЕНИЕ ПРОЦЕССОВ; ГЛУБИННЫЙ АНАЛИЗ ПРОЦЕССОВ; СЕМАНТИЧЕСКИЙ АНАЛИЗ; ОБРАБОТКА ЕСТЕСТВЕННОГО ЯЗЫКА.

Formal models of business processes are necessary for process analysis and for developing process-oriented services. Process modeling is a time-consuming task which should be performed by an expert. A large part of the expert's work consists of gathering information about the process. In most cases, process descriptions already exist in textual form. The automated method of extracting process models from texts is required to put this knowledge into use. In this paper we develop a process extraction method for texts like records, reports, or user stories. The main feature of the proposed method is the application of process mining techniques to text analysis.

BUSINESS PROCESS; PROCESS EXTRACTION; PROCESS MINING; SEMANTIC ANALYSIS; NATURAL LANGUAGE PROCESSING.

Анализ и автоматизация бизнес-процессов требуют использования их формальных моделей. Разработка формальных описаний процессов – это трудоемкая задача, требующая привлечения экспертов-аналитиков. Как следствие, актуальной является задача автоматизации моделирования процессов.

В ходе работы аналитик использует доступные источники информации об исследуемом процессе. Во многих случаях

такими источниками служат тексты: спецификации, регламенты, инструкции, методические указания. Кроме официальных документов могут использоваться тексты, составленные самими участниками процесса, такие как отчеты или отзывы, описывающие конкретный опыт участника. В некоторых случаях (например, при телефонном обращении в службу технической поддержки или в органы государственной власти) могут записываться сеансы взаимодействия

с клиентами. Средства автоматизированного анализа таких источников позволили бы не только сократить временные затраты на построение формального описания процесса, но и повысить полноту анализа за счет увеличения количества информации о процессе.

Одной из областей, в которых автоматизированное извлечение описаний бизнес-процессов из текстов представляется особенно полезным, является разработка процессно-ориентированных мобильных сервисов. Пример такого рода сервисов – мобильные помощники, подсказывающие пользователю последовательность шагов, которые необходимо выполнить для достижения цели.

В университетской среде такое приложение может способствовать адаптации недавно поступивших студентов, предоставляя им персонализированную информацию о порядке оформления в качестве студента и доступе к ресурсам университета (общежития, библиотеки, спортивные секции), а также помогать выполнению студентами и сотрудниками стандартных процедур (оформление академического отпуска, восстановление, оформление командировок, участие в конкурсах на получение грантов). Каждому такому процессу соответствует общая модель, единая для всех процессов соответствующего класса, и частная модель, учитывающая особенности данного процесса: например, запись в спортивные секции выполняется по общим правилам, но для каждой конкретной секции детали процесса записи могут отличаться.

Поскольку частные особенности конкретных процессов не всегда описаны в официальных документах и могут со временем меняться, представляется полезным уточнять модели процессов на основе обратной связи от самих участников процессов. Не будучи экспертами в моделировании бизнес-процессов, пользователи могли бы уточнять модели процессов путем отправки текстов, описывающих их личный опыт реализации соответствующего процесса. Возможность анализа таких частичных описаний процессов и объединения извлеченной из них информации с существующими мо-

делями позволила бы лучше адаптировать систему к изменениям процессов.

Известные подходы к извлечению описаний процессов

Выделяются два класса работ, посвященных автоматическому выводу моделей процессов. Первый, наиболее обширный класс, посвящен задаче извлечения процессов из протоколов работы информационных систем, известной как задача глубинного анализа процессов (process mining) [1].

Протокол представляет собой последовательность событий, каждому из которых сопоставлен идентификатор реализации процесса, одинаковый для всех событий, которые имели место при выполнении процесса в конкретном случае. Например, при анализе процесса обслуживания клиентов интернет-магазина в качестве такого идентификатора можно использовать номер заказа. Часто каждая запись протокола также содержит идентификатор пользователя, выполнившего операцию, и отметку о времени события.

Протоколы описывают процесс только частично, поскольку некоторые необязательные действия могут не иметь места в каждом конкретном случае. Часто протоколы также содержат ошибки, которые могут быть связаны, например, с некорректным выполнением инструкции оператором или с отсутствием части данных.

Цель алгоритмов глубинного анализа – построение формальной модели процесса, наилучшим образом соответствующей данному протоколу. Процессы при этом чаще всего представляются в виде графов переходов или сетей Петри. Область глубинного анализа процессов активно развивается, однако предметом исследований являются в первую очередь методы и алгоритмы анализа протоколов; задача извлечения формальных описаний процессов из данных другой природы, насколько нам известно, не рассматривается.

Второй класс работ посвящен извлечению описаний процессов непосредственно из текстов. Основным подходом, которого придерживается большинство авторов, является выделение в тексте элементов,

составляющих бизнес-процесс, и формирование на их основе описания процесса в соответствии с базовой концептуальной моделью.

В качестве базовой модели часто используется модель CREWS [2]. В ней составляющими процесса являются объекты, которые могут выступать в качестве действующих лиц и ресурсов, и действия, элементарные или составные. Аналогом этих понятий в лингвистическом контексте выступает аргументно-предикатная структура предложения. Выделение сущностей и отношений – стандартная задача обработки естественного языка. Основные отличия между известными подходами к извлечению описаний процессов состоят в используемых авторами методах анализа текста.

В системе R-BPD [3] для извлечения составляющих процесса используется поиск на основе синтаксических шаблонов, определение частей речи и статистическое выделение компонентов предложения.

В работе [4] для анализа сценариев использования реализован более сложный алгоритм, включающий поверхностный синтаксический анализ, идентификацию словарных слов и словосочетаний, разрешение анафоры и анализ контекста.

Похожий метод использован в работах [5, 6], посвященных автоматизированному анализу пользовательских историй.

В работе [7] для анализа текста применен статистический парсер [8], а в качестве источника информации о семантических отношениях (в частности, для учета синонимии) использованы базы данных WordNet [9] и FrameNet [10].

С точки зрения используемых методов выделяется работа [11], авторы которой используют для описания моделей рекурсивные шаблоны, представленные в виде структур признаков. Аналогичные структуры признаков используются на синтаксическом уровне при анализе текста. Поиск соответствия между шаблонами и синтаксическими конструкциями реализуется путем унификации структур признаков.

В работе [12] рассматривается случай линейных процессов, каждое действие в которых описывается глаголом. Модель

процесса строится путем анализа отношения предшествования на множестве пар, состоящих из глагола и следующего за ним существительного (предполагая, что в большинстве случаев такое существительное является прямым дополнением), которые извлекаются из набора документов. Особенность этой работы – метрика соответствия документа линейной структуре процесса.

К извлечению описаний процессов близка задача оценки степени отклонения описания процесса на естественном языке от формальной модели, рассматриваемая в работе [13]. Близость описаний оценивается путем определения соответствия между фрагментами текста, которым сопоставляются множества наиболее информативных слов, и компонентами формальной модели, для которых аналогичные множества слов извлекаются из онтологии.

Анализ и постановка задачи

Рассмотрим, какие сложности возникают в указанных подходах при попытке использовать тексты на естественном языке.

С точки зрения структуры описания процесса можно выделить два класса документов. К документам первого класса относятся инструкции, спецификации, регламенты. Их общая черта – рассмотрение различных аспектов процесса и вариантов его выполнения. Второй класс документов включает описания отдельных реализаций процесса. Примерами таких документов являются протоколы, отчеты, некоторые виды отзывов.

Извлечение описаний процессов из документов типа инструкций затруднительно. Анализ документов сложной структуры требует обращения к методам лингвистики текста. Помимо линейной последовательности текст характеризуется существованием структурных единиц особого типа (элементарная дискурсивная единица, сверхфразовое единство и пр.) и ряда взаимосвязанных нелинейных структур, многие из которых представлены в тексте неявно (ср. понятия когезии и когерентности, топика, актуального членения, макроструктуры и макропропозиции, риторической структуры и

др.), а также нетривиальным соотношением локальных и глобальных языковых явлений (пресуппозиции и имплицатуры, контекст, речевые акты). Рассматриваемый подход к построению модели процесса потребует выявления основных единиц текста и экспликации подобного рода структур во всей их взаимосвязи в качестве модели текста, на основе которого должна восстанавливаться модель описываемого в этом тексте процесса. Известные в настоящий момент попытки построения формальных моделей на основе текстов данного типа ориентированы на идентификацию фрагментов заранее определенной модели с использованием извлекаемого из текстов фактического материала (например, [14]). Полностью автоматическая реконструкция процесса по изолированному подробному текстовому описанию нетривиальной структуры в настоящее время не представляется возможной (но ср. работы по автоматическому реферированию [15], машинному переводу и автоматическому пониманию текстов [16])

Тексты второго класса (отчеты и протоколы) значительно проще по структуре, хотя также могут содержать нетривиальные конструкции как на уровне предложения, так и на уровне всего текста в целом. Здесь основным содержанием текста является последовательность действий, выполненных в конкретном случае. Это свойство сближает такие тексты с протоколами работы информационных систем.

Формализация семантики такого рода текстов может опираться на иную концепцию значения, отличающуюся от традиционной для статической формальной семантики концепции значения как условий истинности. В данном случае интерпретация текста может опираться на последовательность сообщений, модифицирующих информированность адресата (т. н. принцип «потенциала изменения контекста»), и носит выраженный динамический характер. В рамках динамической семантики известны такие подходы, как теория репрезентации дискурса и динамическая логика предикатов. Данная концепция значения позволяет предложить решения ряда упоминавшихся выше проблем, в частности, разрешения

анафоры как одного из механизмов обеспечения связности текста [17].

Если основная информация в тексте будет соответствующим образом формализована, для извлечения модели процесса можно использовать алгоритмы глубинного анализа процессов. В данной статье мы рассматриваем реализацию этого подхода. Мы начинаем с рассмотрения частного случая: документов, описывающих конкретную последовательность действий, которая должна быть выполнена — или действительно была выполнена автором — для достижения цели.

На первом этапе должна быть решена задача преобразования текста, описывающего последовательность действий, в структурированный протокол, к которому можно применять алгоритмы извлечения процессов. В этой статье мы рассматриваем алгоритм анализа текстов, в которых описание последовательности действий усложняется языковыми конструкциями «если — то», «и», «или», выражающими их зависимость, совместность и альтернативность.

В настоящее время мы решаем задачу извлечения описаний процессов только для английского языка, что связано в первую очередь с доступностью лингвистических ресурсов. Тем не менее мы строим алгоритмы таким образом, чтобы разработанный метод мог использоваться для анализа текстов на русском и других языках.

Предлагаемое решение

Мы предлагаем следующую последовательность шагов для извлечения формального описания процесса из текста.

1. Формирование семантического представления для каждого предложения с помощью семантического парсера.
2. Идентификация сущностей в полученных представлениях.
3. Генерация линейной последовательности событий в формате протокола.
4. Формирование соответствующей протоколу модели процесса в виде сети Петри.

Использование семантического формализма позволяет отвлечься от особенностей конкретного языка. В качестве такого формализма мы используем абстрактное



представление значения (Abstract Meaning Representation, AMR) [18]. AMR представляет собой ориентированный граф с выделенной вершиной. Каждому узлу графа соответствует сущность, а ребрам сопоставляются атрибуты, описывающие отношения между сущностями. Набор атрибутов в AMR представляет собой расширенный набор семантических ролей PropBank [19]. В частности, в AMR представимы интересующие нас конструкции условия, объединения и альтернативы, которым сопоставлены атрибуты «condition», «and» и «or» соответственно. Семантический парсер — это программа, которая принимает на вход последовательность предложений и для каждого из них возвращает граф AMR, представляемый обычно в виде списочной структуры формата ((метка / узел) :атрибут подграф ... :атрибут подграф), где в качестве подграфа может выступать аналогичная списочная структура или метка узла. Для AMR существует несколько семантических парсеров; мы используем парсер, описанный в работе [20].

Генерируемый на основе текста протокол представляется в виде списка пар вида (i, P_i) , где i — номер реализации, а $P_i = [A_1; \dots; A_p]$ — упорядоченный список действий, имеющих место в данной реализации процесса. Мы предполагаем, что действия выполняются в том порядке, в котором следуют предложения в тексте, а порядок совместных действий соответствует порядку их упоминания в предложении. Для упрощения описания алгоритма положим, что если s и t — семантические представления действий, то конструкция $(if\ st)$ описывает зависимость («если выполнено действие s , то нужно выполнить действие

t »), конструкция $(or\ st)$ описывает альтернативность («выполните действие s или действие t »), конструкция $(and\ st)$ описывает совместность («выполните действие s и действие t »).

Алгоритм формирования протокола

1. Выделить в графе подграфы, соответствующие элементарным, зависимым, совместным и альтернативным действиям.

2. Нормализовать представления совместно и альтернативно реализуемых действий. Подграфы, соответствующие конструкциям «выполнить действие s с объектами a и b », преобразуются в пары подграфов «выполнить действие s с объектом a » и «выполнить действие s с объектом b ». Преобразование альтернативных конструкций производится аналогично.

3. Для каждого элементарного действия сгенерировать текстовое описание, которое станет элементом протокола.

4. Сформировать протокол путем последовательного добавления в него действий в порядке их следования в тексте и с учетом их структуры.

Добавление действий в протокол можно описать следующим псевдокодом (см. листинг 1). Здесь квадратные скобки обозначают список, знак «+» обозначает конкатенацию списков, конструкция $[case + x \mid case \leftarrow Protocol]$ обозначает создание нового протокола, который строится из существующего протокола $Protocol$ путем добавления в конец каждой входящей в протокол реализации $case$ последовательности действий x . Конструкция $match\ s\ with$ обозначает сопоставление с образцом: она выбирает тот вариант действия, который соответствует структуре анализируемого объекта s .

```
def AddAction(Action, Protocol) =
  match Action with
  | (and s1 s2) => [case + [s1; s2] | case <- Protocol]
  | (or s1 s2)  => [case + [s1] | case <- Protocol] +
                  [case + [s2] | case <- Protocol]
  | (if s1 s2)  => [case + [s1; s2] | case <- Protocol] +
                  Protocol
  | s          => [case + [s] | case <- Protocol]
end
```

Листинг 1. Алгоритм добавления действий в протокол

Как видно из приведенного псевдокода, альтернативные и зависимые конструкции увеличивают количество реализаций в протоколе: при обработке альтернативной конструкции мы берем все построенные к этому моменту реализации процесса и для каждой из них строим все возможные варианты выполнения действия. Аналогично для зависимых действий рассматриваются

два варианта развития событий: выполнение условия и его невыполнение.

Текстовые описания элементарных действий строятся на основе соответствующих им подграфов с помощью фиксированного набора правил. Генерация полноценных описаний действий является предметом будущего исследования.

Для вывода формального описания про-

```
# ::snt Check your department's deadlines.
(x1 / check-01
  :ARG1 (x5 / deadline
        :poss (x3 / department)))

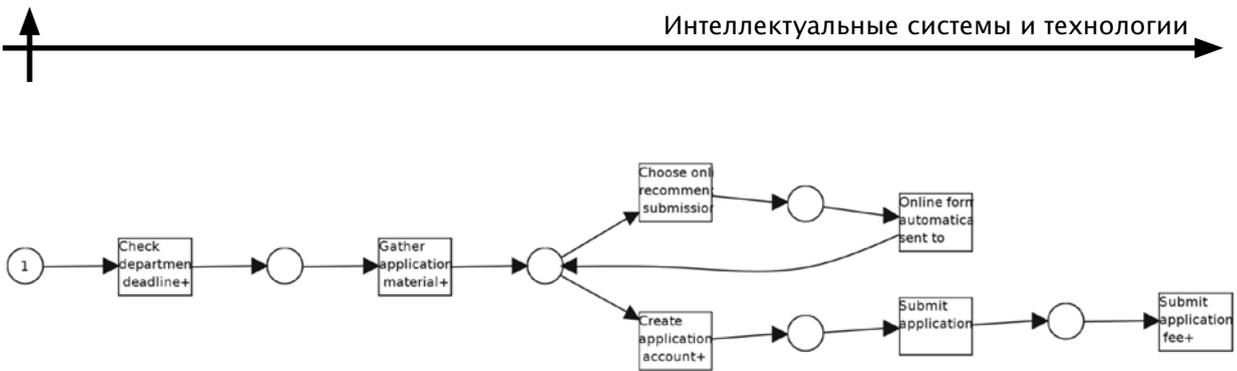
# ::snt Gather necessary application materials.
(x1 / gather-01
  :ARG1 (x4 / material
        :mod (x3 / apply-01)))

# ::snt If you choose online recommendation submission, the online form will
be automatically sent to recommenders.
(x14 / send-01
  :condition (x3 / choose-01
    :ARG1 (x6 / submission
      :mod (x5 / recommend-01
        :mod (x4 / online))))
  :ARG1 (x10 / form
    :mod (x9 / online))
  :manner (x13 / automatically)
  :ARG2 (x16 / recommender))

# ::snt Create an application account and submit the application, and the
application fee.
(x5 / and
  :op1 (x1 / create-01
    :ARG1 (x4 / account
      :mod (x3 / apply-01)))
  :op2 (x6 / submit-01
    :ARG1 (x10 / and
      :op1 (x8 / apply-01)
      :op2 (x13 / fee
        :ARG1 (x12 / apply-01))))))

# ::snt Submit official transcripts from each institution you have attended.
(x1 / submit-01
  :ARG1 (x3 / transcript
    :mod (x2 / official)
    :source (x6 / institution
      :mod (x5 / each)
      :ARG1-of (x9 / attend-01))))
```

Листинг 2. Соответствующие анализируемым предложениям структуры AMR



Выведенная модель процесса

цесса мы используем систему глубинного анализа процессов ProM (<http://promtools.org>), в частности, плагин Mine for a Petri Net using ILP. Анализ применимости и эффективности различных алгоритмов извлечения формальных моделей из протоколов, построенных на основе текста, также остается предметом дальнейшей работы.

Рассмотрим в качестве примера фрагмент текста, описывающий один из этапов подачи документов в университет.

Check your department's deadlines. Gather necessary application materials. If you choose online recommendation submission, the online form will be automatically sent to recommenders. Create an application account and submit the application, and the application fee.

В ходе семантического анализа этих предложений формируется набор структур AMR, показанный в листинге 2.

Применяя к набору графов алгоритм формирования протокола, получаем две реализации процесса: в одной из них действие «choose online recommendation submission» не выполняется, в другом — выполняется, и вслед за этим выполняется обязатель-

ное в этом случае действие «online form automatically sent to recommenders». Итоговый протокол показан в листинге 3.

Подав этот протокол на вход системы ProM, получаем модель процесса в виде сети Петри, приведенной на рисунке.

Открытые вопросы и дальнейшие исследования

Создание метода извлечения формальных описаний процессов из текста даже в ограниченной постановке требует решения ряда проблем. Перечислим наиболее важные из них.

Даже если ограничиться анализом одного документа, одни и те же сущности нередко могут обозначаться в тексте несколькими разными способами (синонимичные именные группы, местоимения в анафорическом употреблении). Задача становится еще более трудной при работе с несколькими текстами, написанными разными авторами. Необходим систематический способ поиска похожих описаний и проверки гипотезы об их соответствии одной и той же сущности. Необходим также метод со-

CaseId	Action
1	Check department's deadline
1	Gather application material
1	Choose online recommendation submission
1	Online form is automatically sent to recommender
1	Create application account
1	Submit application
1	Submit application fee
2	Check department's deadline
2	Gather application material
2	Create application account
2	Submit application
2	Submit application fee

Листинг 3. Извлеченная из текста последовательность действий

гласования частичных описаний процесса, включающих только подмножества путей, по которым может идти выполнение, и этапов, из которых состоит полный процесс.

Разработанный алгоритм опирается на существенное ограничение: мы полагаем, что все действия выполняются в порядке их описания в тексте. Обобщение полученных нами результатов на более широкий класс текстов требует более точного анализа временной последовательности событий. Еще более важным оказывается упорядочение событий при анализе нескольких описаний одного процесса.

В данной статье мы ограничились рассмотрением простых процессов, состоящих из элементарных действий или событий. Интересен вопрос, насколько предлагаемый нами подход может быть обобщен на случай иерархических процессов, в качестве действий в которых могут выступать вложенные процессы, а также на случай, когда текст содержит описание нескольких взаимосвязанных или одновременно выполняющихся процессов.

Наконец, одной из наиболее важных проблем является формирование метрик, позволяющих сравнивать модели процессов и оценивать качество разрабатываемых алгоритмов. В работе [21] приведен ряд метрик, используемых при оценке результатов работы алгоритмов глубинного анализа процессов, но эти метрики оценивают соответствие процессов протоколам, полученным при реальном выполнении процесса. В нашем случае оценка качества рабо-

ты алгоритма может производиться путем сравнения выведенного описания процесса с эталоном, построенным экспертом-аналитиком. При этом необходимо учитывать, что синтаксического сравнения моделей недостаточно: один и тот же процесс может быть описан несколькими различными, но эквивалентными способами. Более точными представляются метрики, основанные на сравнении множеств протоколов, допускаемых каждым описанием. Если рассматривать модель процесса как грамматику, процессы эквивалентны, если они порождают один и тот же язык. Исследование методов сравнения моделей процессов представляет собой одну из наиболее приоритетных задач дальнейшей работы.

В настоящей статье предложен метод построения формальных описаний процессов путем анализа текстов на английском языке, описывающих выполняемые в рамках каждого процесса действия. Отличительной особенностью представленного подхода является применение методов глубинного анализа процессов (process mining). В статье также рассмотрены некоторые открытые вопросы, решение которых позволит обобщить полученные результаты на более широкий класс описаний процессов.

Работа выполнена в ходе реализации комплексного проекта в рамках Постановления Правительства РФ от 09.04.2010 №218 при финансовой поддержке Минобрнауки России. Договор № 02.G25.31.0024 от 12.02.2013.

СПИСОК ЛИТЕРАТУРЫ

1. **van der Aalst W.M.P.** Process Mining: Discovery, Conformance and Enhancement of Business Processes. Berlin: Springer-Verlag, 2011.
2. **Achour C.B.** Guiding Scenario Authoring // European-Japanese Conf. on Information Modelling and Knowledge Bases. 1998.
3. **Ghose A., Koliadis G., Chueng A.** Process discovery from model and text artefacts // IEEE Congress on Services. 2007. Pp. 167–174.
4. **Sinha A., Paradkar A., Kumanan P., Boguraev B.** An Analysis Engine for Dependable Elicitation on Natural Language Use Case Description and its Application to Industrial Use Cases // IBM Research Report RC24712 (W0812-106). 2008.
5. **Gonçalves J.C.A.R., Santoro F.M., Baião F.A.** Business Process Mining from Group Stories // Proc. of the 13th Internat. Conf. on Computer Supported Cooperative Work in Design. 2009.
6. **Gonçalves J.C.A.R., Santoro F.M., Baião F.A.** Let Me Tell You a Story – On How to Build. Process Models // J. of Universal Computer Science. 2011. Vol. 17. Pp. 276–295.
7. **Friedrich F., Mendling J., Puhmann, F.** Process model generation from natural language text // Advanced Information Systems Engineering. 2011. Pp. 482–496.
8. **Manning D.** Natural Language Parsing. Advances in Neural Information Processing Systems



// Proc. of the 2002 Conf. 2003. Vol. 15.

9. **Miller G.A.** WordNet: a lexical database for English // *Communications of the ACM*. 1995. Vol. 38. Pp. 39–41.

10. **Baker C.F., Fillmore C.J., Lowe J.B.** The Berkeley FrameNet Project // Proc. of the 17th Internat. Conf. on Computational Linguistics. Association for Computational Linguistics. 1998. Vol. 1. Pp. 86–90.

11. **Ackermann L., Volz B.** Model [NL] generation: natural language model extraction // Proc. of the 2013 ACM Workshop on Domain-Specific Modeling. 2013. Pp. 45–50.

12. **Chen Y., Ding Zh., Sun H.** PEWP: Process Extraction Based on Word Position in Documents // Proc. of the 9th Internat. Conf. on Digital Information Management (ICDIM). 2014. Pp. 135–140.

13. **van der Aa H., Leopold H., Reijers H.A.** Detecting Inconsistencies Between Process Models and Textual Descriptions // *Business Process Management*. Springer, 2015. Pp. 90–105.

14. **Novichkova S., Egorov S., Daraselia N.** MedScan, a natural language processing engine for MEDLINE abstracts // *Bioinformatics*. 2003. Vol. 19. No. 13. Pp. 1699–1706.

15. **Marcu D.** The rhetorical parsing of

unrestricted texts: A surface-based approach // *Computational Linguistics*. 2000. Vol. 26. No. 3. Pp. 395–448.

16. **Леонтьева Н.Н.** Автоматическое понимание текстов. Системы, модели, ресурсы. М.: Академия, 2006.

17. **Gerasimov N., Pyshkin E.** Using Dynamic Predicate Logic for Pronominal Anaphora Resolution in Russian Texts // Proc. of the Internat. Workshop on Applications in Information Technology. Aizu-Wakamatsu, Japan, 2015. Pp. 51–54.

18. **Banarescu L. et al.** Abstract Meaning Representation for Sembanking // Proc. of Linguistic Annotation Workshop. 2013.

19. **Palmer M., Kingsbury P., Gildea D.** The Proposition Bank: An Annotated Corpus of Semantic Roles // *Computational Linguistics*. 2005. Vol. 31. No. 1. Pp. 71–106.

20. **Wang C., Xue N., Pradhan S., Pradhan S.** A Transition-based Algorithm for AMR Parsing // *Human Language Technologies: Annual Conf. of the North American Chapter of the ACL*. 2015. Pp. 366–375.

21. **Rozinat A., Veloso M., van der Aalst W.M.P.** Evaluating the Quality of Discovered Process Models // Proc. of the ECML-PKDD Workshop on Induction of Process Models. 2008. Pp. 45–52.

REFERENCES

1. **van der Aalst W.M.P.** *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Berlin: Springer-Verlag, 2011.

2. **Achour C.B.** Guiding Scenario Authoring. *European-Japanese Conference on Information Modelling and Knowledge Bases*, 1998.

3. **Ghose A., Koliadis G., Chueng A.** Process discovery from model and text artefacts. *IEEE Congress on Services*, 2007, Pp. 167–174.

4. **Sinha A., Paradkar A., Kumanan P., Boguraev B.** An Analysis Engine for Dependable Elicitation on Natural Language Use Case Description and its Application to Industrial Use Cases. *IBM Research Report RC24712 (W0812-106)*, 2008.

5. **Gonçalves J.C.A.R., Santoro F.M., Baião F.A.** Business Process Mining from Group Stories. *Proceedings of the 13th International Conference on Computer Supported Cooperative Work in Design*, 2009.

6. **Gonçalves J.C.A.R., Santoro F.M., Baião F.A.** Let Me Tell You a Story – On How to Build Process Models. *Journal of Universal Computer Science*, 2011, Vol. 17, Pp. 276–295.

7. **Friedrich F., Mendling J., Puhmann F.** Process model generation from natural language text. *Advanced Information Systems Engineering*, 2011, Pp. 482–496.

8. **Manning D.** Natural Language Parsing. *Advances in Neural Information Processing Systems, Proceedings of the 2002 Conference*, 2003, Vol. 15.

9. **Miller G.A.** WordNet: a lexical database for English. *Communications of the ACM*, 1995, Vol. 38, Pp. 39–41.

10. **Baker C.F., Fillmore C.J., Lowe J.B.** The Berkeley FrameNet project. *Proceedings of the 17th International Conference on Computational linguistics. Association for Computational Linguistics*, 1998, Vol. 1, Pp. 86–90.

11. **Ackermann, L., Volz B.** Model [NL] generation: natural language model extraction. *Proceedings of the 2013 ACM workshop on domain-specific modeling*, 2013, Pp. 45–50.

12. **Chen Y., Ding Zh., Sun H.** PEWP: Process Extraction Based on Word Position in Documents. *Proceedings of the 9th International Conference on Digital Information Management*, 2014, Pp. 135–140.

13. **van der Aa H., Leopold H., Reijers H.A.** Detecting Inconsistencies Between Process Models and Textual Descriptions. *Business Process Management*, Springer, 2015, Pp. 90–105.

14. **Novichkova S., Egorov S., Daraselia N.** MedScan, a natural language processing engine for MEDLINE abstracts. *Bioinformatics*, 2003, Vol. 19,

No. 13, Pp. 1699–1706.

15. **Marcu D.** The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 2000, Vol. 26, No. 3, Pp. 395–448.

16. **Leontyeva N.N.** *Avtomaticheskoye ponimaniye tekstov. Sistemy, modeli, resursy* [Automatic understanding of texts. Systems, models, resources]. Moscow: Akademiya Publ., 2006. (rus)

17. **Gerasimov N., Pyshkin E.** Using Dynamic Predicate Logic for Pronominal Anaphora Resolution in Russian Texts. *Proceedings of the International Workshop on Applications in Information Technology*, Aizu-Wakamatsu, Japan, 2015, Pp. 51–54.

18. **Banarescu L. et al.** Abstract Meaning

Representation for Sembanking. *Proceedings of Linguistic Annotation Workshop*, 2013.

19. **Palmer M., Kingsbury P., Gildea D.** The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 2005, Vol. 31, No. 1, Pp. 71–106.

20. **Wang C., Xue N., Pradhan S., Pradhan S.** A Transition-based Algorithm for AMR Parsing. *Human Language Technologies: Annual Conference of the North American Chapter of the ACL*, 2015, Pp. 366–375.

21. **Rozinat A., Veloso M., van der Aalst W.M.P.** Evaluating the Quality of Discovered Process Models. *Proceedings of the ECML-PKDD Workshop on Induction of Process Models*, 2008, Pp. 45–52.

СОКОЛОВ Константин Владимирович — аспирант кафедры математической лингвистики Санкт-Петербургского государственного университета.

199034, Россия, Санкт-Петербург, Университетская наб., д. 7-9.

E-mail: vtqveant@gmail.com

SOKOLOV Konstantin V. *St. Petersburg State University.*

199034, Universitetskaya Emb. 7-9, St. Petersburg, Russia.

E-mail: vtqveant@gmail.com

ТИМОФЕЕВ Дмитрий Андреевич — старший преподаватель кафедры распределенных вычислений и компьютерных сетей Института компьютерных наук и технологий Санкт-Петербургского политехнического университета Петра Великого.

195251, Россия, Санкт-Петербург, ул. Политехническая, д. 29.

E-mail: dtim@dcn.icc.spbstu.ru

TIMOFEEV Dimitri A. *Peter the Great St. Petersburg Polytechnic University.*

195251, Politekhnikeskaya Str. 29, St. Petersburg, Russia.

E-mail: dtim@dcn.icc.spbstu.ru

САМОЧАДИН Александр Викторович — профессор кафедры распределенных вычислений и компьютерных сетей Института компьютерных наук и технологий Санкт-Петербургского политехнического университета Петра Великого, кандидат технических наук.

195251, Россия, Санкт-Петербург, ул. Политехническая, д. 29.

E-mail: samochadin@gmail.com

SAMOCHADIN Alexander V. *Peter the Great St. Petersburg Polytechnic University.*

195251, Politekhnikeskaya Str. 29, St. Petersburg, Russia.

E-mail: samochadin@gmail.com