

DOI: 10.18721/JCSTCS.11412
УДК 004

ИЗВЛЕЧЕНИЕ МОДЕЛЕЙ БИЗНЕС-ПРОЦЕССОВ ИЗ ОБУЧАЮЩИХ МАТЕРИАЛОВ

Д.А. Тимофеев, А.В. Самочадин

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Моделирование играет важную роль в анализе и оптимизации бизнес-процессов организации. Автоматизация построения моделей бизнес-процессов особенно значима в тех сферах деятельности, где преобладает интеллектуальный труд, но выполняемые процессы редко документируются – в частности, в области разработки программного обеспечения. В такой ситуации важным источником информации о процессах являются обучающие материалы, создаваемые разработчиками как часть общей документации, а также для обмена опытом и популяризации технологий. Существующие алгоритмы извлечения моделей бизнес-процессов накладывают серьезные ограничения на вид и структуру входных данных. В статье рассмотрен подход к извлечению моделей процессов из обучающих материалов, в которых могут одновременно присутствовать описания различных процессов, а также фрагменты текста, которые не должны учитываться при моделировании. Для учета этих факторов предлагаемый метод принимает во внимание не только стандартные извлекаемые из текста признаки, но и структуру, и разметку документа.

Ключевые слова: бизнес-процесс, моделирование бизнес-процессов, извлечение процессов, обработка естественного языка, анализ текста.

Ссылка при цитировании: Тимофеев Д.А., Самочадин А.В. Извлечение моделей бизнес-процессов из обучающих материалов // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 162–170. DOI: 10.18721/JCSTCS.11412.

PROCESS EXTRACTION FROM EDUCATIONAL TEXTS

D.A. Timofeev, A.V. Samochadin

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

Business process modeling plays an important role in analysis and optimization of organizational processes. Automation of process models is particularly crucial in domains where processes involve mostly intellectual activity that is not properly documented. Software development is an example of a domain with these properties. Educational materials like instructions and guides, blog posts, or conference talks are an important source of information about the processes in this case. Known algorithms of process extraction pose strict requirements to the input text. In this paper, we propose an approach to process extraction from complex sources containing descriptions of multiple processes and text blocks unrelated to the process model. In order to account for these aspects, the method considers not only standard lexical and syntactic properties of the text but also its structure and markup.

Keywords: business process, process modeling, process extraction, natural language processing, text analysis.

Citation: Timofeev D.A., Samochadin A.V. Process extraction from educational texts. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 162–170. DOI: 10.18721/JCSTCS.11412.

Введение

Для анализа и оптимизации процессов, выполняемых работниками в ходе профессиональной деятельности, необходимо иметь возможность описывать эти процессы. В промышленности такие описания (технологические процессы, технологические карты) разрабатываются технологами заранее. В сферах, где преобладает интеллектуальная деятельность, регламентирование последовательности операций применяется существенно реже. Например, в области разработки программного обеспечения конкретные предписания для исполнителя могут создаваться при тестировании программного обеспечения (тест-план). При этом сама возможность строго описать процесс решения задачи приводит к возможности автоматизировать выполнение этого процесса, исключив участие человека. В частности, инженеры по обеспечению качества программного обеспечения в настоящее время преимущественно занимаются не ручным тестированием, а разработкой тестовых планов, которые выполняются с помощью специализированных программных систем, таких как Selenium [1, 2]. Как следствие, большая часть выполняемых процессов явно не описывается, а часто остается скрытой от наблюдателя в силу своего преимущественно ментального характера. В то же время явное описание процессов остается важным не только для их оптимизации, но и с организационной точки зрения: для обучения студентов и начинающих профессиональную деятельность специалистов, адаптации новых сотрудников, обмена опытом, документирования процессов, что в конечном итоге приводит к повышению производительности труда и снижению зависимости организации от знаний конкретных сотрудников. Таким образом, сбор и формализация описаний выполняемых в организации процессов становится важной задачей.

С точки зрения анализа и автоматизации деятельности предпочтительнее описание процессов на формальном языке, таком как BPMN [3] или UML [4]. Моде-

лирование процессов на этом уровне детализации требует привлечения аналитиков, чьей задачей становится не только построение формальной модели процесса, но и в первую очередь извлечение знаний о процессах путем наблюдения, интервьюирования работников и анализа существующих нормативных документов и других видов описаний процессов. Для упрощения моделирования целесообразно автоматизировать эту деятельность, в частности, использовать средства анализа текста на естественном языке для извлечения описаний процессов из документов и записей интервью. При этом необходимо учитывать вид обрабатываемого текста: нормативные документы обычно имеют стандартную структуру и написаны в едином стиле, в то время как описания процессов, сделанные самими работниками, могут существенно отличаться друг от друга и содержать большое количество фрагментов, не относящихся напрямую к выполняемым действиям, а описывающих или обосновывающих выполняемые шаги.

Данная статья посвящена задаче извлечения моделей процессов в области разработки программного обеспечения. Основной особенностью этой области является то, что в ней редко используются нормативные документы, регламентирующие рабочие процессы, в особенности индивидуальные процессы, выполняемые разработчиками: проектирование систем, написание и модификация программного кода, тестирование и отладка. Как следствие, при построении формальных моделей этих процессов необходимо опираться на взаимодействие с самими разработчиками или на описания тех или иных аспектов их деятельности, которые они создают для других целей. Среди таких описаний особое место занимают обучающие материалы: лекции, доклады на профессиональных конференциях, руководства по установке и использованию программ.

Цель создания таких материалов – передача опыта и обучение других людей решению определенных задач или использо-

ванию инструментальных средств. Этот класс документов представляется особенно интересным с точки зрения моделирования индивидуальных процессов: создаваемые разработчиками описания, с одной стороны, отражают часть их повседневной деятельности, обычно не документируемой, а с другой – специально создаются в расчете на восприятие другими людьми, в том числе менее опытными, благодаря чему в явном виде описывают часть очевидных для самого разработчика операций, сведений о контексте и обоснований принятых решений.

Подходы к извлечению моделей процессов из текста

Можно выделить два основных подхода к построению формальных моделей процессов на основе слабоструктурированных данных. Первый подход, обычно называемый извлечением процессов, состоит в построении модели на основе последовательности событий [5]. В качестве исходных данных при этом могут использоваться протоколы взаимодействия пользователей с информационными системами или полученные другими способами записи выполняемых участниками процесса действий. При использовании второго подхода исходными данными служат тексты на естественном языке, которые содержат описания процессов. В этом случае модель процесса восстанавливается путем извлечения из текста конструкций, описывающих участников процесса, выполняемые действия и связи между ними [6, 7]. Результатом является набор правил (в частности, в соответствии со стандартом SBVR [8]), промежуточные текстовые представления процессов [7, 9] или метамодели [10, 11], или непосредственно модели процессов на одном из стандартных языков моделирования [12, 13].

Известные подходы также отличаются друг от друга требованиями, предъявляемыми к исходным данным. Часть решений опирается на контролируемый язык, допускающий только ограниченное подмно-

жество синтаксических конструкций [11, 13]. В работе [10] релевантные фрагменты текстового описания должен выделять пользователь системы. Большая часть систем работает в предположении, что подаваемый на вход документ содержит описание одного процесса и не содержит существенного количества дополнительной информации, которая может быть ошибочно интерпретирована. Частным случаем является работа с исходными документами определенной структуры – например, сценариями использования. В [14] предложен подход, позволяющий выделить релевантные фрагменты текста путем сопоставления извлекаемых из текста слов с набором слов, входящем в концептуальную модель предметной области.

Основной метод извлечения описаний процессов из текста – поиск шаблонов или синтаксических конструкций, соответствующих действующим лицам, действиям и их аргументам, а также маркеров дискурса («и», «или», «затем», «в противном случае» и т. д.) [15, 16]. В работе [12] применяется семантический анализ текста, а для получения информации о семантических отношениях используются базы данных WordNet и FrameNet.

Формирование модели процесса на основе извлеченных из текста конструкций производится с помощью маркеров дискурса, ключевых слов или извлекаемых отношений между предикатом (глаголом) и аргументами (именными группами). При этом обычно не гарантируется целостность и непротиворечивость построенной модели. Основная причина этого – свойственная естественному языку неоднозначность. Анализ такой неоднозначности, возникающей при описании бизнес-процессов, проводится в [17], авторы вводят понятие «поведенческого пространства», включающего все возможные интерпретации извлеченной из текста последовательности событий. Результаты исследования корпуса описаний процессов, представленного в [12], выявили неоднозначно интерпретируемые конструкции в 32 из 47 текстов.

Извлечение моделей процессов из обучающих материалов

Хотя существующие методы извлечения процессов позволяют достичь достаточно высокой для практических целей точности извлечения моделей процессов (77 % на корпусе [12]), они имеют ряд существенных ограничений, в первую очередь касающихся объема и содержания текста. Для обучающих материалов (доклады, видеоуроки) характерно наличие отступлений, в которых автор объясняет принятые решения, демонстрирует существующие возможности или высказывается на отвлеченные темы. Эта особенность существенно меньше выражена в пошаговых руководствах, являющихся частью документации продукта (например, инструкции по установке программы), но и они содержат фрагменты, не описывающие действия участников процесса или состояния системы. Такие фрагменты часто включают те языковые конструкции, на которые опирается алгоритм извлечения процессов, что приводит к включению в модель неверных элементов.

Еще одна важная особенность реальных инструкций и обучающих текстов — одновременное присутствие в тексте описаний нескольких процессов, которые могут быть как связаны друг с другом (процесс и подпроцессы), так и быть независимыми (например, процессы установки приложения в операционных системах Windows и Unix).

Для решения этих проблем предлагается следующий подход.

1. Для анализа используется не только текст, но и разметка документа. Инструкции, руководства и обучающие материалы создаются для использования людьми и поэтому применяют стандартные структурные элементы (разделы, нумерованные и маркированные списки), которые выделяются шрифтом, нумерацией, условными обозначениями или положением на странице. Преимущественным способом распространения документов при этом является их публикация в сети Интернет или передача пользователям в электронной форме, что позволяет учитывать при ана-

лизе документов существующую машинно-читаемую разметку, такую как теги и стили. Особенно важную роль играют следующие характеристики фрагментов текста.

- **Заголовки.** Заголовок раздела часто обозначает выполняемое действие или подпроцесс, которые описываются или комментируются в тексте раздела. Пример: руководство по установке системы управления проектами Bitbucket [18] содержит заголовки «Install Bitbucket Server» и «Set up Bitbucket», в соответствующих им разделах описываются соответствующие процессы установки и настройки приложения. Сами заголовки при этом можно рассматривать как наименования шагов процесса более высокого уровня.

- **Нумерованные списки.** Нумерация действий обычно используется для явного указания порядка их выполнения, что позволяет использовать сам факт нумерации в качестве признака, позволяющего классифицировать фрагмент текста как описание действия, а также установить связь между различными действиями.

- **Маркированные (нумерованные) списки.** Как и нумерованные списки, маркированные фрагменты текста обычно соответствуют выполняемым действиям или вариантам ветвления процесса. В отличие от нумерованных элементов, маркированные элементы могут отражать как упорядочение действий в порядке их перечисления, так и равнозначные альтернативы.

- **Гиперссылки.** Оформление фрагмента текста в качестве гиперссылки может использоваться для указания выполняемого подпроцесса или действия, но чаще просто указывает на наличие дополнительной информации. При анализе текста фрагмент, помеченный как ссылка, должен в ряде случаев обрабатываться особым образом. Например, в предложении «Check that your system meets the minimum requirements shown nearby; [Installation requirements](#) has more details» [18] текст ссылки «Installation requirements» является названием раздела, буквальная интерпретация этого словосо-

четания затрудняет как синтаксический анализ (использование формы единственного числа глагола *to have* с существительным во множественном числе), так и семантический анализ текста (является ли фрагмент фразы описанием состояния системы или комментарием, который не следует включать в модель процесса).

- Выделение фрагментов кода, вводимых команд, примеров текста. Учет этого вида разметки необходим для корректной интерпретации соответствующих конструкций, в частности, для их исключения из дальнейшего лингвистического анализа.

2. При анализе документов используется гипотеза локальности: предполагается, что при наличии в документе описаний нескольких процессов каждый из них описывается отдельно. При этом учитывается деление текста на разделы и подразделы: предполагается, что разделы одного уровня вложенности играют одинаковую роль в описании процесса.

3. Анализ текста выполняется по предложениям. Мы предполагаем, что каждое предложение можно отнести к одному из трех классов: «содержит наименование процесса», «содержит описание действия» и «не содержит описания действия».

Наименование процесса может быть представлено в документе как заголовок раздела, обычно достаточно высокого уровня, или как фрагмент предложения, не содержащего описания действия (пример: «But what if you need to install the software on a machine that does NOT have an Internet connection?» [19]).

Описание действия обязательно включает глагольную группу, определяющую выполняемое действие. В зависимости от действия описание может содержать одно или несколько дополнений, описывающих аргументы (параметры) операции, а также именную группу (подлежащее), которая соответствует выполняющему операцию действующему лицу. Если выполнять операцию должен читатель документа, глагол часто ставится в повелительное наклонение, а подлежащее отсутствует. К описа-

ниям действий относятся также предложения, описывающие текущее состояние или реакцию системы (пример: «You should see a blinking indicator in the upper left corner of Live's screen» [20]), т. к. с точки зрения моделирования процесса изменение состояния является действием, выполняемым одним из участников взаимодействия. При интерпретации предложения бывает необходимо изменить действующее лицо, если действие описывается глаголами восприятия (такими, как *see, hear, feel*).

Хотя данную структуру можно считать обязательной для описания действий, она не является уникальной для них. Большая часть комментариев, обоснований и других, не учитываемых при построении модели процесса фрагментов текста, содержат те же самые элементы. Для принятия решения о принадлежности к одному из классов строится классификатор, использующий следующие признаки:

- векторные представления для входящих в предложение слов;

- фрагменты синтаксического дерева фиксированной глубины;

- наличие кореференциальных связей в рамках контекста;

- наличие или отсутствие предопределенных маркеров дискурса;

- характеристика фрагмента текста (наличие или отсутствие разметки);

- класс предложения, стоящего на предшествующем уровне вложенности (например, класс заголовка раздела при распознавании класса предложений в теле раздела).

Распознавание класса каждого предложения должно выполняться в контексте, т. е. с учетом классов других предложений в заданной окрестности. В соответствии с принципом локальности контекст включает предложения на одном уровне структуры документа. Так как решение в этом случае должно согласованно приниматься для каждого из рассматриваемых предложений, для распознавания используется метод MEMM (Maximum Entropy Markov Model) [21]. Для обучения формируется корпус документов с разметкой классов предложений.

Начало

Инициализировать набор описаний процессов пустым множеством.
Создать новое описание процесса без наименования и сделать его текущим описанием.

Разбить текст на блоки в соответствии с вложенностью разделов документа.

Для каждого блока:

Разбить блок на предложения.

Для каждого предложения:

Выделить вектор признаков предложения.

Выполнить классификацию предложения.

Если предложение классифицировано как наименование процесса:

Добавить текущее описание в набор описаний.

Создать новое описание процесса с наименованием, соответствующим предложению, и сделать его текущим описанием.

Иначе, если предложение классифицировано как содержащее описание действия:

Присвоить предложению номер шага.

Извлечь упоминания участников и заменить их на идентификаторы с учетом ранее найденных участников.

Извлечь подлежащее, сказуемое и дополнение и сформировать на их основе описание действия.

Извлечь маркеры условий и ветвлений.

Если найден маркер ветвления:

Определить номер следующего шага, к которому должен быть выполнен переход.

Добавить пятерку <номер шага, участник, действие, условие, следующий шаг> к текущему описанию.

Иначе:

Пропустить предложение.

Если текущее описание не пусто:

Добавить текущее описание в набор описаний.

Для каждого описания в наборе описаний:

Сформировать описание в нотации BPMN на основе упорядоченного набора пятерок.

Конец

Псевдокод алгоритма извлечения описаний процессов
Pseudocode of process description extraction algorithm

Предложения, классифицированные как не описывающие действия, исключаются из дальнейшего рассмотрения. Предложения, классифицированные как наименования процессов, также исключаются из дальнейшего рассмотрения, однако в ходе их обработки производится разбиение документа на фрагменты, соответствующие каждому процессу. Для оставшихся предложений используется алгоритм построения модели процесса, описанный в [7]. Псевдокод алгоритма приведен на рисунке.

В случае, если в одном документе присутствуют несколько описаний процессов,

возможны два подхода: автоматическое построение моделей всех описанных процессов или явный выбор нужного описания пользователем. Хотя явный выбор релевантных описаний пользователем увеличивает трудоемкость процесса, он позволяет уменьшить количество ошибок, связанных с неверным определением границ описаний процессов. В случаях, когда описания процессов оказываются недостаточно разделены на уровне структуры, явное указание границ процесса пользователем оказывается необходимым для корректного извлечения описаний процессов.

Заключение

Задача извлечения моделей процессов из текстовых описаний играет важную роль для автоматизации работы аналитика при изучении и оптимизации процессов в организации, особенно в тех сферах деятельности, где преобладает интеллектуальный труд, но реализуемые сотрудниками процессы не регламентируются и редко описываются. Характерный пример такой сферы – разработка программного обеспечения.

Известные подходы к извлечению моделей процессов демонстрируют достаточную для практического использования точность, но рассчитаны на работу с текстами, подчиняющимися строгим ограничениям по размеру, структуре и подмножеству используемого языка. В данной статье сделана попытка обобщить методы извлечения моделей на случай текстов, которые могут содержать описания более чем одного процесса (при условии выполнения гипотезы локальности), а также содержать фрагменты, не описывающие никакие аспекты процесса (пояснения, обоснования, примеры). Для этого предлагается использовать не только стандартные для задач анализа текста признаки, такие как леммы, части речи, синтаксические зависимости, но и признаки, основанные на структуре и разметке самого документа.

Предложенный подход предназначен в первую очередь для анализа документов со структурной разметкой (HTML, PDF, Microsoft Word, LaTeX). Такие документы широко представлены в сети Интернет, а

также составляют значительную часть электронного документооборота организаций. Возможно также применение этого подхода для анализа текстов без явной структурной разметки за счет использования знаков препинания, отступов и символов перевода строки в качестве признаков. Тем не менее, существенное использование предположения локальности описания приводит к трудностям при анализе документов, в которых присутствуют описания нескольких процессов. В этом случае рекомендуется привлечение пользователя для указания границ процесса и выбора релевантных процессов.

В дальнейшем предполагается развивать предложенный подход в двух основных направлениях:

расширение набора признаков для классификации предложений, в первую очередь использование маркеров дискурса для идентификации фрагментов, похожих на описания действий, но не относящихся к извлекаемому процессу (в частности, действий, приводимых в качестве примеров или нереализованных альтернатив);

использование тезауруса и онтологии предметной области для принятия решений о присутствии в тексте описаний процессов, соответствующих этой предметной области, и более точного определения границ описания процесса.

Работа подготовлена в ходе реализации проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0247 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. Holmes A., Kellogg M. Automating functional tests using selenium // Agile Conference 2006. IEEE, 2006.
2. SeleniumHQ Browser Automation // URL: <https://www.seleniumhq.org/> (Дата обращения: 10.11.2018).
3. Object Management Group. Business Process Model And Notation, version 2.0.2. – 2014 // URL: <https://www.omg.org/spec/BPMN/2.0.2/> (Дата обращения: 10.11.2018).
4. Object Management Group. Unified Modeling Language, version 2.5.1. – 2017 // URL: <https://www.omg.org/spec/UML> (Дата обращения: 10.11.2018).
5. van der Aalst W.M.P. Process mining: Discovery, conformance and enhancement of business processes. Berlin: Springer-Verlag, 2011.
6. Riefer M., Ternis S.F., Thaler T. Mining process models from natural language text: A state-of-the-art analysis // Multikonferenz Wirtschaftsinformatik. March 2016. Pp. 9–11.

7. **Honkisz K., Kluza K., Wiśniewski P.** A concept for generating business process models from natural language description // *Internat. Conf. on Knowledge Science, Engineering and Management*. Springer, Cham, 2018. Pp. 91–103.
8. Object Management Group. *Semantics of Business Vocabulary and Rules*, version 1.4. – 2017 // URL: <https://www.omg.org/spec/SBVR/1.4/> (Дата обращения: 10.11.2018).
9. **Mogos A.H., Urzica A.** TN4PM: A textual notation for process modelling // *Intelligent Distributed Computing III*. Springer, Berlin, Heidelberg, 2009. Pp. 263–268.
10. **Kop C., et al.** Tool Supported Extraction of Behavior Models // *ISTA*. 2005. Vol. 63. Pp. 114–123.
11. **Yue T., Briand L.C., Labiche Y.** Automatically deriving a UML analysis model from a use case model. Carleton University, 2010.
12. **Friedrich F., Mendling J., Puhmann F.** Process model generation from natural language text // *Internat. Conf. on Advanced Information Systems Engineering*. Springer, Berlin, Heidelberg, 2011. Pp. 482–496.
13. **Caporale T.** A tool for natural language oriented business process modeling // *ZEUS*. 2016. Pp. 49–52.
14. **Annervaz K.M., et al.** Natural language requirements quality analysis based on business domain models // *Proc. of the 28th IEEE/ACM Internat. Conf. on Automated Software Engineering*. IEEE Press, 2013. Pp. 676–681.
15. **Ghose A., Koliadis G., Chueng A.** Process discovery from model and text artefacts // *Services, 2007 IEEE Congress on*. IEEE, 2007. Pp. 167–174.
16. **Gonçalves J.C.A.R., Santoro F.M., Bairo F.A.** Business process mining from group stories // *Proc. of the 2009 13th Internat. Conf. on Computer Supported Cooperative Work in Design*, 2009.
17. **van Der A.H., Leopold H., Reijers H.A.** Dealing with behavioral ambiguity in textual process descriptions // *Internat. Conf. on Business Process Management*. Springer, Cham, 2016. Pp. 271–288.
18. Atlassian. *Install Bitbucket Server on Linux* // URL: <https://confluence.atlassian.com/bi-tbucketserver/install-bitbucket-server-on-linux-868976991.html> (Дата обращения: 10.11.2018).
19. Nuts About Nets FAQ. *How to Register Software Without An Internet Connection* // URL: <http://nutsaboutnets.com/faqs/register-without-internet/> (Дата обращения: 21.11.2018).
20. HomeRecording.com Forum. *SeaGtGruff post: Help – basic DAW setup help* // URL: <https://homerecording.com/bbs/general-discussions/newbies/help-basic-daw-setup-help-384095/3/#post4359360> (Дата обращения: 21.11.2018).
21. **McCallum A., Freitag D., Pereira F.C.N.** Maximum entropy Markov models for information extraction and segmentation // *ICML*. 2000. Vol. 17. No. 2000. Pp. 591–598.

Статья поступила в редакцию 22.11.2018.

REFERENCES

1. **Holmes A., Kellogg M.** Automating functional tests using selenium. *Agile Conference*, IEEE, 2006.
2. SeleniumHQ Browser Automation. Available: <https://www.seleniumhq.org/> (Accessed: 10.11.2018).
3. Object Management Group. *Business Process Model And Notation*, version 2.0.2. – 2014. Available: <https://www.omg.org/spec/BPMN/2.0.2/> (Accessed: 10.11.2018).
4. Object Management Group. *Unified Modeling Language*, version 2.5.1. – 2017. Available: <https://www.omg.org/spec/UML> (Accessed: 10.11.2018).
5. **van der Aalst W.M.P.** *Process mining: Discovery, conformance and enhancement of business processes*. Berlin: Springer-Verlag, 2011.
6. **Riefer M., Ternis S.F., Thaler T.** Mining process models from natural language text: A state-of-the-art analysis. *Multikonferenz Wirtschaftsinformatik*, March 2016, Pp. 9–11.
7. **Honkisz K., Kluza K., Wiśniewski P.** A concept for generating business process models from natural language description. *International Conference on Knowledge Science, Engineering and Management*. Springer, Cham, 2018, Pp. 91–103.
8. Object Management Group. *Semantics of Business Vocabulary and Rules*, version 1.4. – 2017. Available: <https://www.omg.org/spec/SBVR/1.4/> (Accessed: 10.11.2018).
9. **Mogos A.H., Urzica A.** TN4PM: A textual notation for process modeling. *Intelligent Distributed Computing III*. Springer, Berlin, Heidelberg, 2009, Pp. 263–268.
10. **Kop C., et al.** Tool supported extraction of behavior models. *ISTA*, 2005, Vol. 63, Pp. 114–123.
11. **Yue T., Briand L.C., Labiche Y.** *Automatically deriving a UML analysis model from a use case model*. Carleton University, 2010.
12. **Friedrich F., Mendling J., Puhmann F.** Process model generation from natural language text. *International Conference on Advanced Inform-*

mation Systems Engineering. Springer, Berlin, Heidelberg, 2011, Pp. 482–496.

13. **Caporale T.** A tool for natural language oriented business process modeling. *ZEUS*, 2016, Pp. 49–52.

14. **Annervaz K.M., et al.** Natural language requirements quality analysis based on business domain models. *Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering*. IEEE Press, 2013, Pp. 676–681.

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

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

17. **van Der A.H., Leopold H., Reijers H.A.** Dealing with behavioral ambiguity in textual pro-

cess descriptions. *International Conference on Business Process Management*. Springer, Cham, 2016, Pp. 271–288.

18. Atlassian. Install Bitbucket Server on Linux. Available: <https://confluence.atlassian.com/bitbucketserver/install-bitbucket-server-on-linux-868976991.html> (Accessed: 10.11.2018).

19. Nuts About Nets FAQ. How to Register Software Without An Internet Connection. Available: <http://nutsaboutnets.com/faqs/register-without-internet/> (Accessed: 21.11.2018).

20. HomeRecording.com Forum. SeaGtGruff post: Help – basic DAW setup help. Available: <https://homerecording.com/bbs/general-discussions/newbies/help-basic-daw-setup-help-384095/3/#post4359360> (Accessed: 21.11.2018).

21. **McCallum A., Freitag D., Pereira F.C.N.** Maximum entropy Markov models for information extraction and segmentation. *ICML*, 2000, Vol. 17, No. 2000, Pp. 591–598.

Received 22.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ТИМОФЕЕВ Дмитрий Андреевич
TIMOFEEV Dmitrii A.
 E-mail: dtim@dcn.icc.spbstu.ru

САМОЧАДИН Александр Викторович
SAMOCHADIN Alexander V.
 E-mail: samochadin@gmail.com