УДК 004.45

Методы и технологии конструирования эффективных и надежных программ и программных систем на основе графовых моделей и семантических преобразований

Касьянов В.Н. (Институт систем информатики СО РАН),
Касьянова Е.В. (Институт систем информатики СО РАН)

В Институте систем информатики СО РАН с 2021 года выполняется проект «Методы и технологии конструирования эффективных и надежных программ и программных систем на основе графовых моделей и семантических преобразований» коллективом сотрудников лабораторий «Конструирования и оптимизации программ» и «Системное программирование» под научным руководством д.ф.-м.н., профессора В. Н. Касьянова. Ответственными исполнителями проекта являются к.ф.-м.н., доцент Е. В. Касьянова и к.т.н. В. И. Шелехов. В данной статье кратко представлены те результаты выполнения первого этапа данного проекта, которые были получены сотрудниками лаборатории «Конструирования и оптимизации программ».

Ключевые слова: *визуальная обработка, конструирование программ, оптимизирующая трансляция, параллельное программирование, предикатное программирование, программное обеспечение, теоретико-графовые методы, трансформационное программирование, функциональное программирование, языки и системы программирования.*

1. Введение

Тенденция развития программирования состоит в том, что все более разнообразные процессы обработки программ и данных и все в большей степени поддерживаются машиной. Большинство из этих процессов обработки программ и данных реализуется в существующих инструментах как текстовые или языковые, но является семантическими. В них, как правило, требуется сохранение некоторого инварианта, определенным образом связанного с семантикой обрабатываемых объектов (например, трансляция и другие функционально эквивалентные преобразования программ сохраняют функцию, реализуемую программой). Поэтому без всестороннего изучения и глубокого использования в инструментальных системах семантических преобразований нельзя достичь ни надежного, ни эффективного

решения задач автоматизации программирования, перейти от кустарного производства программ к технологии и массовому производству.

Трансформационный подход трактует программирование как систематическое применение фундаментальных процессов семантической обработки программ, сохраняющих при преобразовании программы определенный ее семантический инвариант и образующих в совокупности «сумму технологий». Трансформационные методы используются в качестве основного средства для достижения эффективности при автоматизации программирования методами трансляции, особенно в связи с появлением ЭВМ новых архитектур. Они являются перспективным направлением в создании новых, более мощных средств автоматизации конструирования эффективных и надежных программ.

Работы по теоретическому обоснованию трансформационного подхода к разработке программного обеспечения активно развиваются во всем мире. Вместе с тем перед исследователями все еще стоит задача разработать «алгебру программ», позволяющую манипулировать программными фрагментами в рамках формального исчисления программ с целью автоматизации конструирования эффективных и надежных программ для перспективных ЭВМ. Эта очень заманчивая цель вряд ли будет достигнута в ближайшем будущем ввиду разнообразия используемых языков программирования и спецификаций, а также архитектур ЭВМ. Однако если алгоритмический уровень для начальной спецификации зафиксирован, то разработка методов и средств для преобразования данной программы в корректную и эффективную версию для компьютеров с различными архитектурами может считаться реалистичной задачей. В этом направлении учеными новосибирской школы программирования, основанной и долгое время руководимой академиком А. П. Ершовым, были получены значительные результаты, образующие хороший фундамент для активного и целенаправленного продолжения работ [1–3, 10–14].

С 2021 в Институте систем информатики СО РАН выполняется проект «Методы и технологии конструирования эффективных и надежных программ и программных систем на основе графовых моделей и семантических преобразований» коллективом сотрудников лабораторий «Конструирования и оптимизации программ» и «Системное программирование» под научным руководством д.ф.-м.н., профессора В. Н. Касьянова. Ответственными исполнителями проекта являются к.ф.-м.н., доцент Е. В. Касьянова и к.т.н. В. И. Шелехов.

Цель проекта — повышение эффективности и надежности компьютерного решения прикладных задач за счет совершенствования программного обеспечения перспективных вычислительных систем. Задачей проекта является развитие теории, методов и технологий

оптимизирующей трансляции и конструирования эффективного, надежного, переносимого и адаптивного программного обеспечения для суперкомпьютеров и компьютерных сетей на основе трансформационного и объектно-ориентированных подходов, теоретико-графовых методов, аннотирования программ, функциональных и логических спецификаций, средств специализации и визуальной обработки.

Расширение трансформационного подхода на логические и аннотированные программы и использование теоретико-графовых методов и средств визуализации, развиваемые авторами проекта, позволяют создать единую основу для сочетания различных видов семантической обработки, включая анализ, преобразование и синтез, а также для объединения автоматических и автоматизируемых процессов обработки.

В данной статье кратко представлены те результаты выполнения первого этапа проекта, которые были получены сотрудниками лаборатории «Конструирования и оптимизации программ» [4–9, 15–22]. Оставшаяся часть статьи состоит из трех разделов и заключения. Раздел 2 содержит обзор результатов по развитию методов и технологий конструирования эффективных и надежных параллельных программ на основе функциональных спецификаций и семантических преобразований. В Разделе 3 рассматриваются разработанные теоретико-графовые методы и инструменты для поддержки конструирования эффективных и надежных программ. Разработке методов, алгоритмов и систем для исследования сложных больших данных, систем и процессов через их визуальные представления с использованием атрибутированных иерархических графовых моделей посвящен Раздел 4.

2. Развитие методов и технологии конструирования эффективных и надежных параллельных программ на основе функциональных спецификаций и семантических преобразований

Проведено исследование методов и средств конструирования эффективных и надежных параллельных программ на основе функциональных спецификаций и семантических преобразований.

Разработана программа CPPS, которая формирует начальную версию облачной расширяемой интегрированной визуальной среды поддержки параллельного программирования на языке Cloud Sisal (см. Рис. 1). Разработанный входной язык Cloud Sisal программы CPPS, оставаясь функциональным потоковым языком с неявным параллелизмом,

содержит такие средства написания научных программ, как циклы и массивы. Среда CPPS доступна к использованию через веб-браузер, использует для всех своих компонент единое семантическое представление Cloud Sisal программы в виде графовой модели и ориентирована на расширение средствами разработки, отладки, верификации и исполнения параллельных программ по их функциональным спецификациям на языке Cloud Sisal. Вместе с созданными компилятором и профилировщиком (см. Рис. 2) программа CPPS позволяет пользователю на любом устройстве, имеющем выход в Интернет, разрабатывать и исполнять функциональные программы на языке Cloud Sisal.

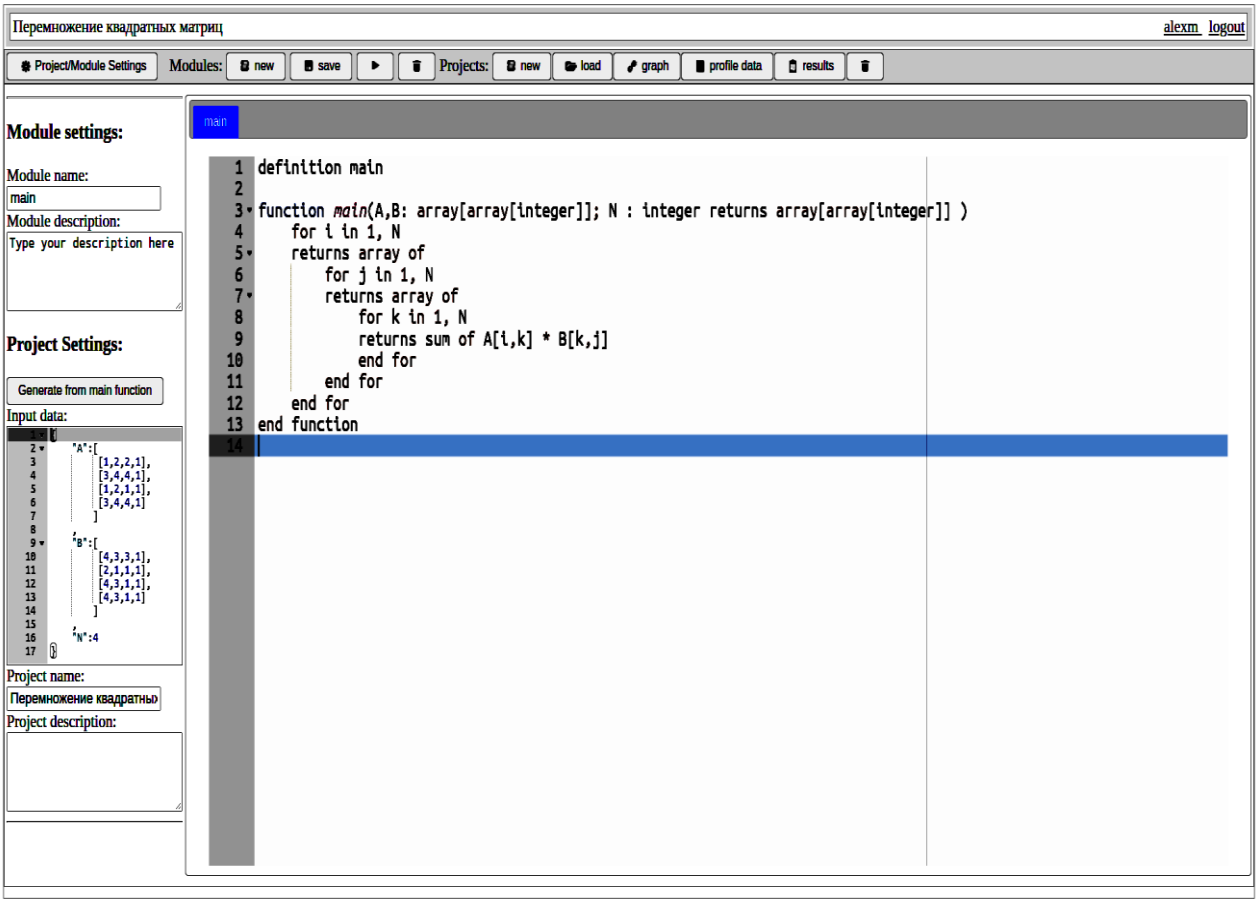


Рис. 1. Система CPPS

Проведено исследование задач визуализации графового внутреннего представления Cloud Sisal программ, а также визуализации процесса вычислений и отладки Cloud Sisal программ. Предложена модель визуализации графовой модели с портами и атрибутами с помощью статических изображений в формате векторной графики SVG. Описана модель отображения изменений графовой модели с портами и атрибутами с помощью анимаций, поддерживаемых форматом векторной графики SVG. Соединение графических анимаций,

отображающих изменения в визуальных стилях и изменений в атрибутах графовой модели с портами, реализовано с помощью безопасных сетей Петри. Описано моделирование вычислений, соответствующих функциям заданной Cloud Sisal программы с помощью иерархических сетей Петри, где переходы соответствуют функциям, а места аргументам и параметрам соответствующих функций. Также описаны модификации иерархических сетей Петри, обеспечивающие реализацию функциональности точек останова и редактирования аргументов или результатов функций при активированных точках останова в целях отладки с помощью добавления дополнительных мест и переходов.

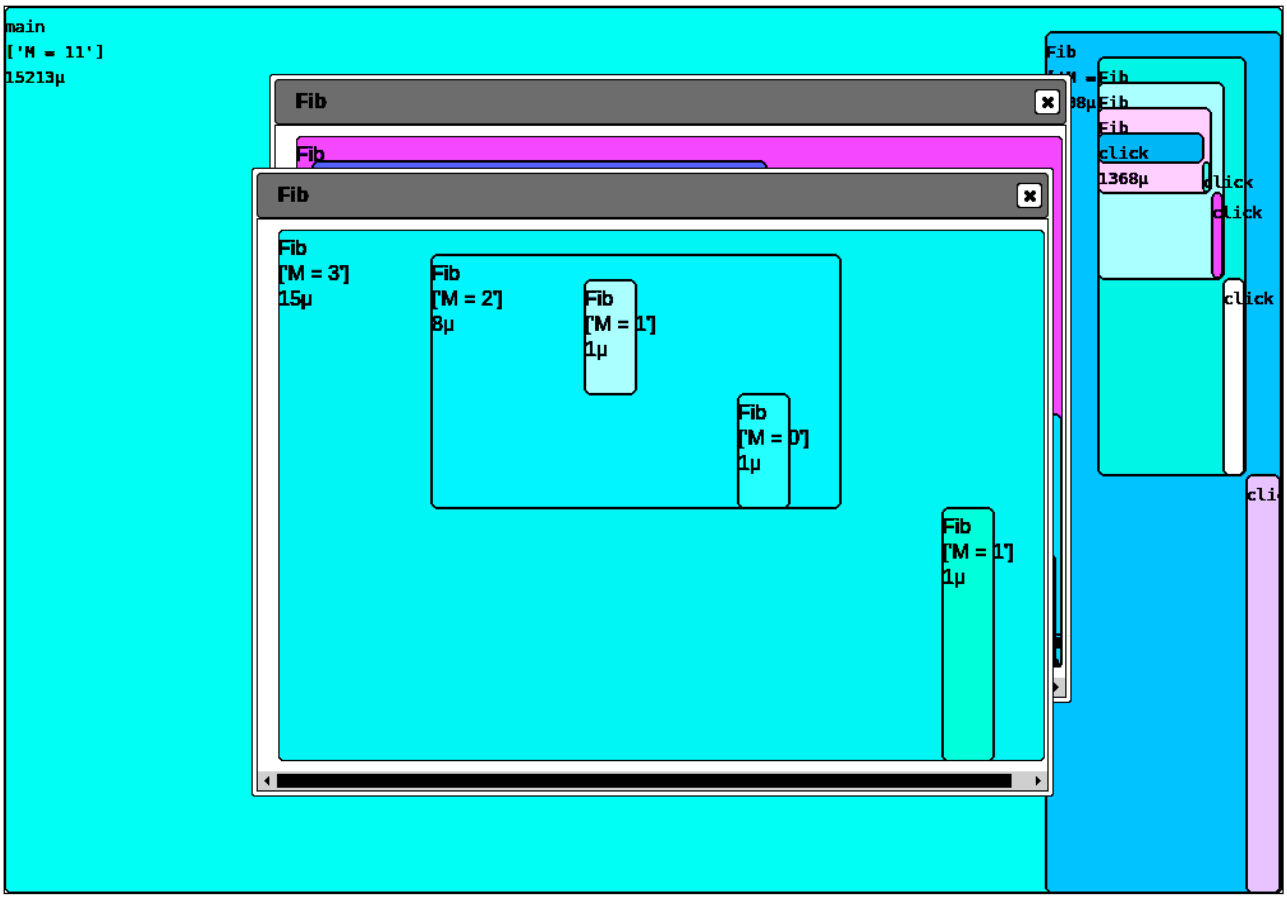


Рис. 2. Интерактивный просмотр результата работы профилировщика

На основе предложенных методов и модели разработана программа CSIRI, поддерживающая визуальную отладку Cloud Sisal программ, представленных в виде графового внутреннего представления на языке GraphML. Областью применения программы CSIRI является разработка Cloud Sisal программ, предназначенных для решения вычислительно-ёмких задач с помощью облачных вычислителей. Функциями программы CSIRI являются: интерпретация Cloud Sisal программы на данных малой размерности,

отладка с помощью механизма точек останова, позволяющего просматривать и изменять промежуточные значения в ходе вычислений, генерация трассировочных данных процесса вычислений.

Проведено исследование существующих методов отладки функциональных программ на языках Haskell, F# и Common Lisp. На их основе разработаны методы отладки Cloud Sisal программ и выполнена их экспериментальная реализация для некоторого представительного подмножества языка Cloud Sisal виде отдельного компонента системы CPPS. Успешная практика применения подобных методов для других функциональных языков и успешные проведенные эксперименты показывают, что предложенные методы представляют интерес для их применения в отладчике системы CPPS. При их встраивании в отладчик они могут стать важным дополнением к используемым в отладчике методам визуальной отладки Cloud Sisal программ на основе их графового представления, поддерживаемым программой CSIRI.

Начата работа по государственной регистрации программ CPPS и CSIRI.

3. Развитие теоретико-графовых методов и инструментов для поддержки конструирования эффективных и надежных программ

Проведено исследование в области онтологии применения теоретико-графовых методов в информатике и программировании. Продолжались работы по пополнению электронного толкового словаря по графам в информатике WikiGRAPP новыми базовыми терминами и по его развитию путем улучшения описаний терминов и представления отношений между ними. Создана начальная версия электронного толкового словаря WikiGRAPP по теории графов и ее применениям в информатике и программировании, пригодная для научного и учебного применения (см. Рис. 3).

Проведено изучение с целью систематизации алгоритмов обработки, визуализации и применения теоретико-графовых методов в информатике и программировании. Продолжались работы по пополнению электронной энциклопедии WEGA алгоритмов решения задач информатики и программирования новыми статьями и по ее развитию путем улучшения статей и структуры энциклопедии. Среди новых аналитических статей, расширивших энциклопедию, выделим следующие: «Сравнение с шаблоном для сжатого текста», «Отказоустойчивые квантовые вычисления», «Сложность биматричного равновесия Нэша», «Сложность ядра», «Локальные аппроксимации задач об упаковке и покрытии», «Балансировка нагрузки», «Списочное планирование», «Локальные вычисления в

неструктурированных радиосетях», «Локальные аппроксимации задач об упаковке и покрытии», «Локальное выравнивание (с вогнутыми штрафами за открытие гэпа)», «Разработка алгоритмов для вычислительной биологии», «Точные алгоритмы решения задачи о выполнимости формулы в КНФ общего вида», «Индексирование сжатого текста», «Обмен пакетами при переключении между несколькими очередями», «Маршрутизация», «Разработка высокоэффективных алгоритмов для крупномасштабных задач», «Покрытие множества почти последовательными подмножествами», «Приближенное сравнение регулярных выражений».

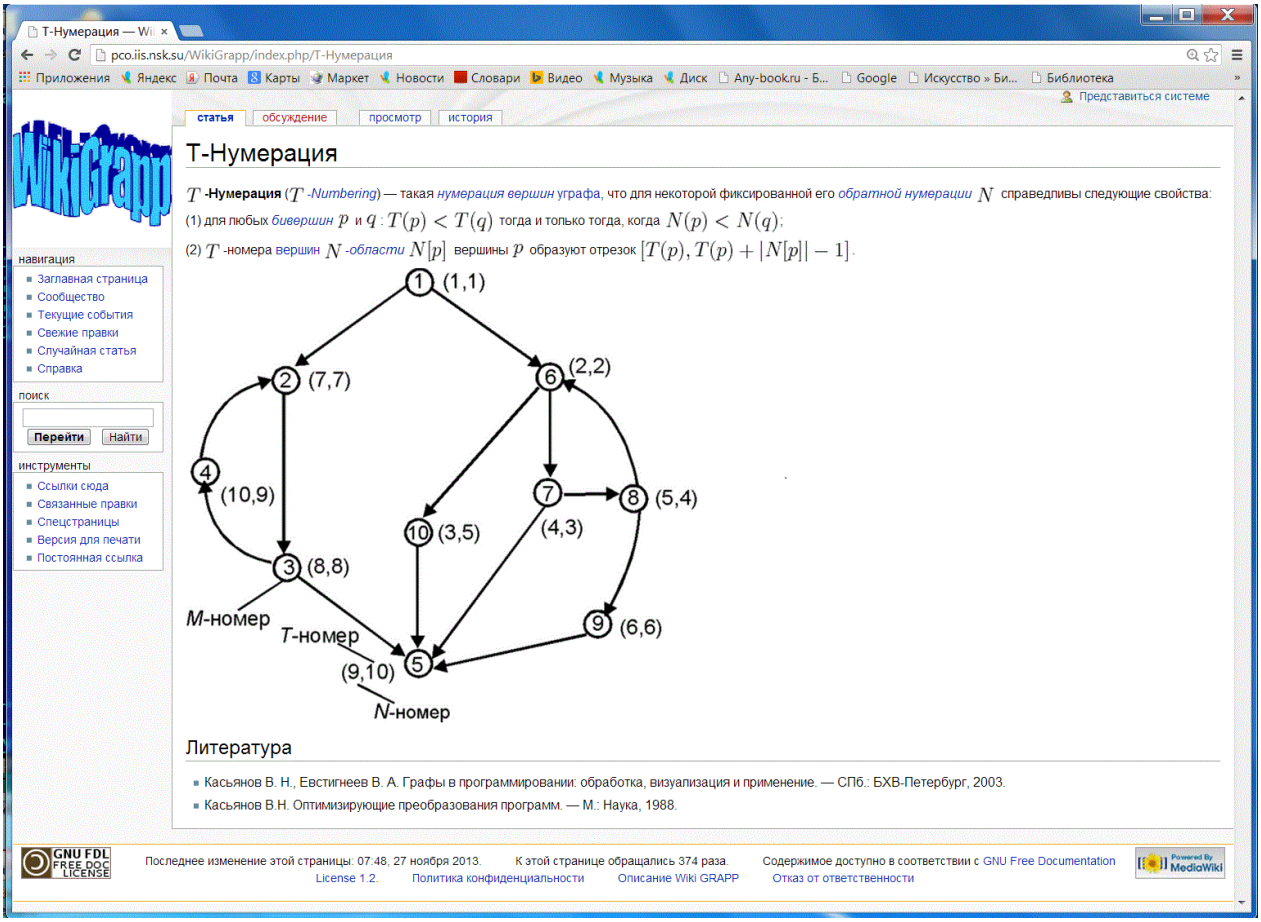


Рис. 3. Словарь WikiGRAPP

Подготовлена начальная версия электронной энциклопедии теоретико-графовых алгоритмов решения задач информатики и программирования WEGA, пригодная для учебных и научных применений (см. Рис. 4).

Разработан проект по развитию словаря WikiGRAPP и энциклопедии WEGA как систем для поддержки накопления и широкого использования знаний по теоретико-графовым моделям и методам решения задач информатики и программирования. В нем помимо

дальнейшего развития словаря WikiGRAPP и энциклопедии WEGA, обусловленного постоянным развитием теоретико-графовых алгоритмов и моделей решения задач программирования, а также расширением их применения на новые предметные области, также предполагается разработка методов и средства визуализации систем и процессов на основе иерархических графовых моделей и высокоуровневых описаний алгоритмов.

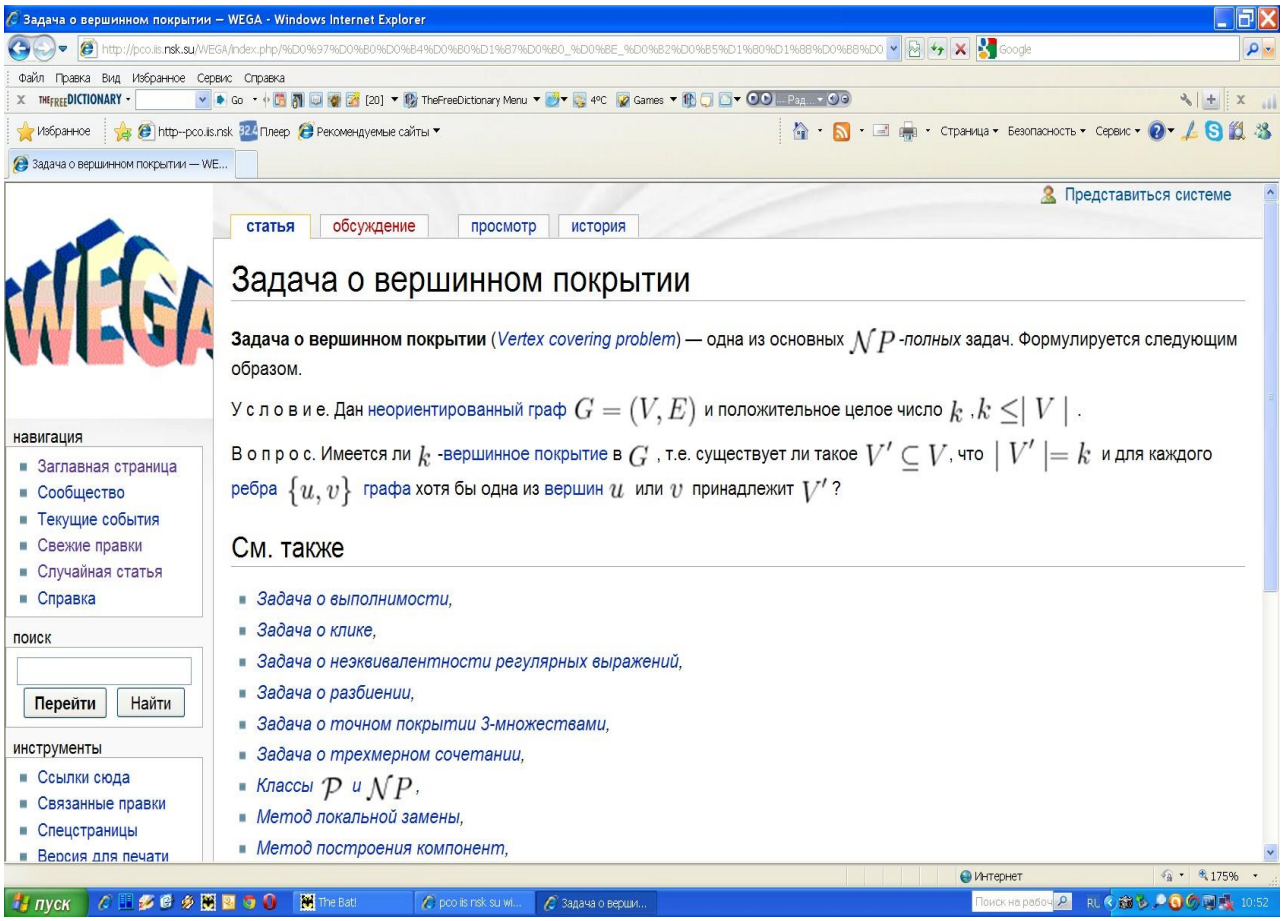


Рис. 4. Энциклопедия WEGA

Проект предполагает исследование задач построения, просмотра и анализа визуальных представлений структурированных данных большого размера на основе иерархических графовых моделей, а также задач построения на основе высокоуровневых описаний графовых алгоритмов, просмотра и анализа таких интерактивных визуализаций исполнений графовых алгоритмов, которые просты для восприятия и управления и наглядно демонстрируют их основные свойства. Цель исследований — построение расширяемой веб-системы для поддержки визуализации структурированной информации о системах и процессах на основе иерархических графовых моделей и высокоуровневых описаний графовых алгоритмов, которая будет интегрирована со словарем WikiGRAPP и

энциклопедией WEGA и могла бы быть использована для создания статических, динамических и интерактивных иллюстраций для вики-словаря WikiGRAPP и вики-энциклопедии WEGA.

Создаваемая веб-система, вики-словарь WikiGRAPP и вики-энциклопедия WEGA будут открыты для общего использования и позволят вовлечь в использование и накопление знаний по теоретико-графовым моделям и методам широкие массы программистской общественности. Свободное использование созданными инструментами всем программистским сообществом приведет к повышению эффективности и надежности компьютерного решения широкого класса прикладных задач, использующих теоретико-графовые модели и методы.

4. Разработка методов, алгоритмов и систем для исследования сложных больших данных, систем и процессов через их визуальные представления с использованием атрибутированных иерархических графовых моделей

Проведены исследования методов и средств визуального представления структурной информации с использованием графовых моделей. Разработаны новые методы и эффективные алгоритмы анализа и визуализации сложно организованной информации большого объема на основе атрибутированной иерархической графовой модели с портами.

Во многих приложениях объекты, моделируемые вершинами графа, содержат непересекающиеся логические местоположения (так называемые порты), через которые они (объекты) находятся во взаимосвязи, моделируемой дугами. Например, в графе программы, моделирующем поток данных в программе, операторы программы представляются вершинами графа, операнды операторов (их аргументы и результаты) моделируются портами вершин, а поток данных между результатами и аргументами операторов представлен дугами, соединяющим соответствующие порты.

Наглядность полученного изображения графа сильно зависит от того, как его элементы (вершины и дуги) расположены на плоскости. Циркулярное изображение графа — это такая укладка графа на плоскости, при которой все вершины графа помещаются на окружность некоторого круга, а каждая дуга рисуется внутри этого круга обычно в виде прямой линии. Однако проблема построения циркулярного изображения с минимальным количеством пересечением дуг является NP-полной. Циркулярная укладка находит свое применение в тех приложениях, где объекты, моделируемые вершинами графа, имеют равный приоритет, и ни

один из них не занимает привилегированное положение. Циркулярные изображения графов используются для визуализации топологий кольцевых и звездных сетей, биологических и социальных сетей, а также небольших кластеров в больших графах. Поскольку эти приложения работают с большими данными, проблема разработки алгоритма циркулярной укладки для общих иерархических графов очень актуальна, но в настоящее время только для кластерных графов, представляющих собой простые иерархические графы без портов, был разработан алгоритм циркулярной укладки.

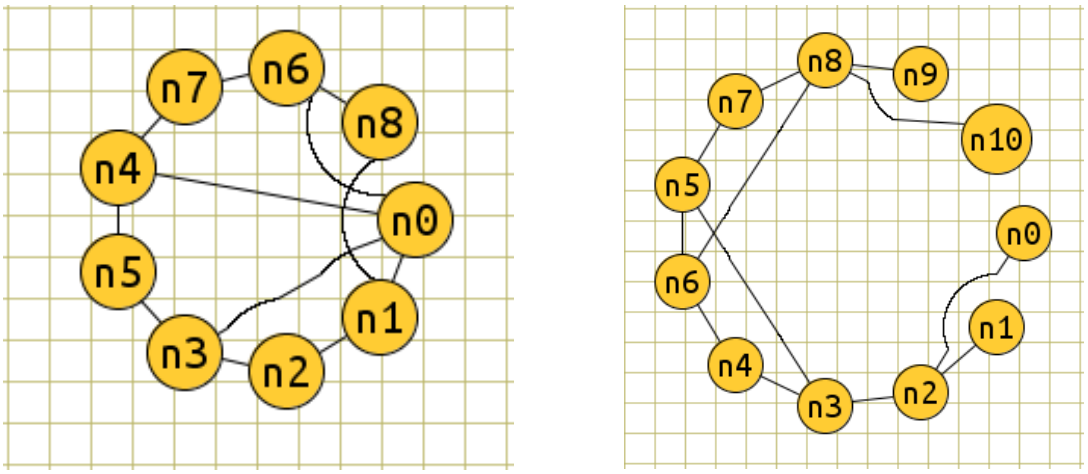


Рис. 5. Циркулярная укладка простых графов

На базе введенного понятия структурного расширения атрибутированного иерархического графа с портами разработан эффективный алгоритм циркулярной укладки иерархического графа с портами на плоскости и выполнена его реализация в рамках системы визуализации Visual Graph. Алгоритм использует специальную эвристику для минимизации пересечений «дуга-дуга», а также округлые вставки в дуги для решения проблемы пересечений «дуга-вершина» (см. Рис. 5).

Создана начальная версия системы визуализации Visual Graph, пригодная для научного и учебного применения (см. Рис. 6). Помимо существенного развития реализованных в системе методов и алгоритмов визуализации информации на основе графовых моделей и анализа ее структурных свойств, было произведено целый ряд изменений, основными из которых являются следующие.

Изменился пользовательский интерфейс системы. Панель с атрибутами была перенесена из нижней секции пользовательского интерфейса в секцию слева, что позволило увеличить просматриваемую область графового изображения.

Значения атрибутов были убраны с панели атрибутов и перенесены в область графового изображения, и теперь, когда пользователь выделяет элементы графового изображения, автоматически появляется информация о выбранных элементах. Данная информация представлена в виде сплошного текста, в котором можно осуществлять поиск, а так же копировать и переносить в другие сторонние утилиты. При желании можно скрыть данное меню. Миникарта переехала из отдельной секции пользовательского интерфейса и разместилась на графовом изображении, сверху справа. При желании теперь можно ее отключить.

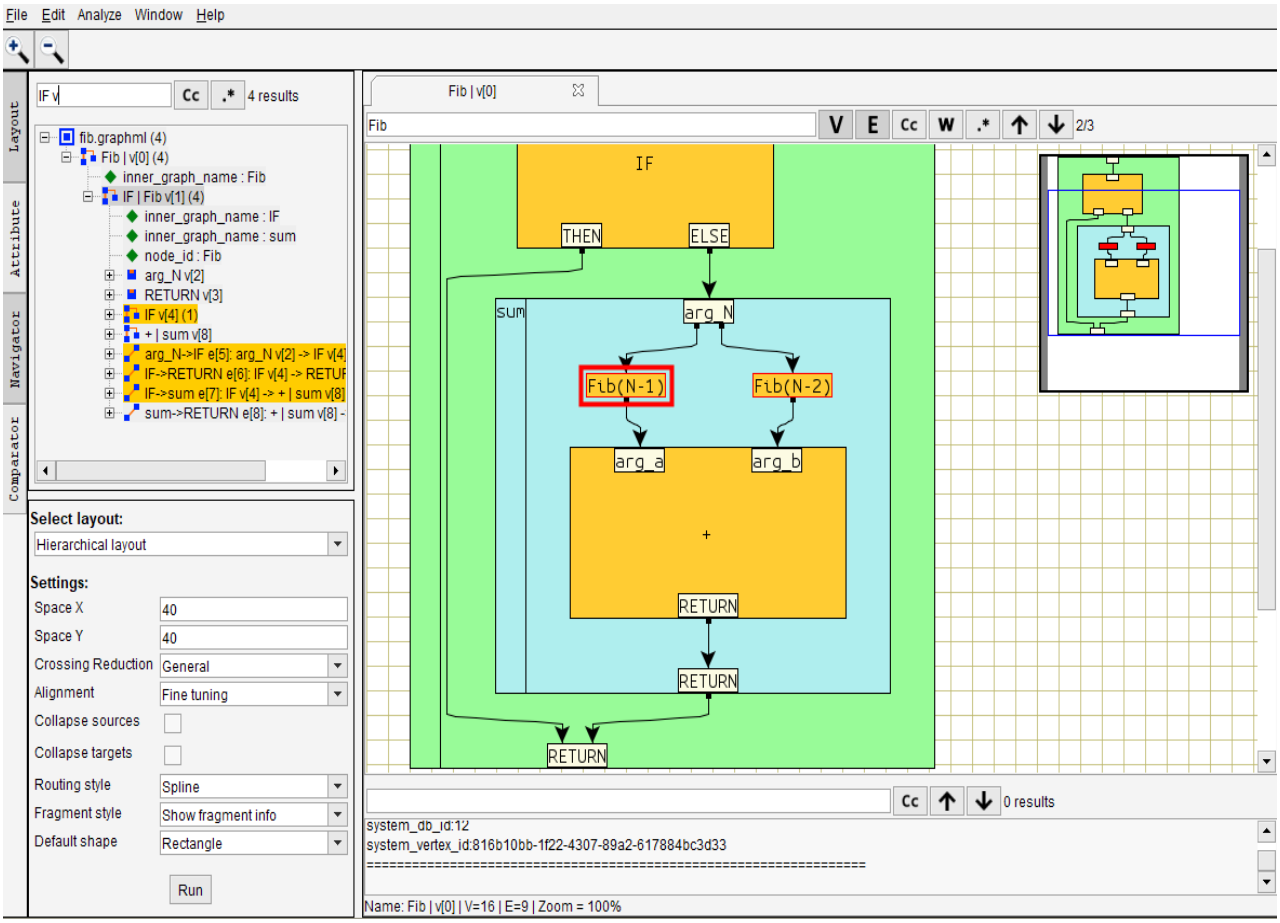


Рис. 6. Система Visual Graph

Изменилась работа с графовыми изображениями. В первоначальной версии системы Visual Graph, для отображения графов использовалась библиотека JGraph. Данная библиотека помимо отображения графов умела укладывать граф на плоскости, используя различные алгоритмы. Но эта библиотека обладала и рядом ограничений, которые не позволяли в полном объеме реализовать работу с иерархическими атрибутированными графами с портами. Основными из этих ограничений являлись проблемы при отображении

графов с большим количеством элементов, а также отсутствие возможностей изменения способов отображения портов и фрагментов. Поэтому в новой версии было принято отказаться от данной библиотеки и реализовать собственный модуль для отображения графов, который бы удовлетворял всем необходимым требованиям.

Расширены возможности импорта и экспорта графов. Полностью реализован импорт графов по их представлениям в форматах Gml, Dot и GraphML. Завершается работа над инструментом по экспорту графов и/или выбранных их частей в файлы указанных форматов. Сейчас ведется активная доработка данного инструмента и его тестирование.

5. Заключение

На первом этапе выполнения проекта все поставленные задачи были достигнуты и получены следующие основные результаты. Разработаны новые методы и эффективные алгоритмы анализа и визуализации сложно организованной информации большого объема на основе атрибутированной иерархической графовой модели с портами. Создана начальная версия системы визуализации Visual Graph, пригодная для научного и учебного применения. Помимо существенного развития реализованных в системе методов и алгоритмов визуализации информации на основе графовых моделей и анализа ее структурных свойств, новая версия системы поддерживает улучшенные возможности по работе с графовыми изображениями, более удобный пользовательский интерфейс и расширенные возможности импорта и экспорта графов. Разработаны методы и программные средства, поддерживающие конструирование и визуальную отладку Cloud Sisal программ, представленных в виде их графового внутреннего представления на языке GraphML.

Список литературы

1. Images of Programming: Dedicated to the Memory of A. P. Ershov / Eds. D. Bjorner and V. Kotov. Amsterdam: North-Holland, 1991. 190 p.
2. Kasyanov V. N. A support tool for annotated program manipulation // Proc. of Fifth European Conf. on Software Maintenance and Reengineering. IEEE Computer Society Press, 2001.P. 85–94.
3. Kasyanov V. N. Transformational approach to program concretization // Theoretical Computer Science. 1991. Vol. 90, № 1. P.37–46.
4. Kasyanov V. N., Kasyanova E. V., Malishev A. A. Support tools for functional programming distance learning and teaching // J. Phys.: Conf. Ser. 2021. Vol.2099, 012052.
5. Kasyanov V. N., Kasyanova E. V., Malishev A. A. Support tools for functional programming distance learning and teaching // Marchuk Scientific Readings-2021: Abstracts of the Intern. conf., October 4–

8, 2021. Novosibirsk: Institute of comput. mathematics and mat. geophysics SB RAS, 2021. P.154–155.

6. Kasyanov V. N., Merculov A. M., Zolotuhin T. A. A circular layout algorithm for attributed hierarchical graphs with ports // J. Phys.: Conf. Ser. 2021. Vol.2099, 012051.
7. Kasyanov V. N., Merculov A. M., Zolotuhin T. A. A circular layout algorithm for attributed hierarchical graphs with ports // Marchuk Scientific Readings-2021: Abstracts of the Intern. conf., October 4–8, 2021. Novosibirsk: Institute of comput. mathematics and mat. geophysics SB RAS, 2021. P.155.
8. Андрей Петрович Ершов: ученый и человек / Отв. ред. А. Г. Марчук. Новосибирск: Издательство СО РАН, 2001. 504 с.
9. Гордеев Д. А. Модель визуализации и отладки на графовом представлении Cloud-Sisal программ // GraphiCon 2021: труды 31-й Междунар. конф. по компьютерной графике и машинному зрению (Нижний Новгород, 27–30 сент., 2021 г.), Нижний Новгород: Нижегород. гос. тех. ун-т, 2021. С.54–62.
10. Ершов А. П. Избранные труды / Отв. ред. И. В. Поттосин. Новосибирск: Наука, 1994. 413 с.
11. Касьянов В. Н. Оптимизирующие преобразования программ. М.: Наука, 1988, 335 с.
12. Касьянов В. Н., Гордеев Т. А., Золотухин Т. А. и др. Система облачного параллельного программирования CPPS: визуализация и верификация Cloud Sisal программ / Отв. ред. В. Н. Касьянов. Новосибирск: ИПЦ НГУ, 2020. 256 с.
13. Касьянов В. Н., Золотухин Т. А. Visual Graph — система для визуализации сложно структурированной информации большого объема на основе графовых моделей // Научная визуализация. 2015. Т. 7, № 4. С. 44–59.
14. Касьянов В. Н., Евстигнеев В. А. Графы в программировании: обработка, визуализация и применение. СПб.: БХВ-Петербург, 2003. 1104 с.
15. Касьянов В. Н., Касьянова Е. В. Опыт преподавания программирования в пандемию // Информатика: проблемы, методы, технологии. Материалы XXI Международной научно-методической конференции. Воронеж: ООО «Вэлборн», 2021. С.1689–1698.
16. Касьянов В.Н., Касьянова Е.В. Особенности преподавания программирования в пандемию // Преподавание информационных технологий в Российской Федерации: Материалы Девятнадцатой открытой Всероссийской конференции. М.: ООО «1С-Паблишинг», 2021. С.229–231.
17. Касьянов В.Н., Малышев А.А. Программные средства поддержки дистанционного обучения функциональному программированию // Информатика: проблемы, методы, технологии. Материалы XXI Международной научно-методической конференции. Воронеж: ООО «Вэлборн», 2021. С.1834–1842.
18. Касьянов В.Н., Малышев А.А. Программные средства поддержки дистанционного обучения функциональному программированию // Преподавание информационных технологий в

Российской Федерации: Материалы Девятнадцатой открытой Всероссийской конференции. М.: ООО «1С-Паблишинг», 2021. С.129–131.

19. Кламбоцкий К. А. Методы и средства отладки Cloud Sisal программ // Математика: Материалы 59-й Междунар. науч. студ. конф. 12–23 апреля 2021 г. / Новосиб. гос. ун-т. Новосибирск : ИПЦ НГУ, 2021. С.143.
20. Меркулов А. М. Циркулярная укладка атрибутированного иерархического графа с портами // Информационные технологии: Материалы 59-й Междунар. науч. студ. конф. 12–23 апреля 2021 г. / Новосиб. гос. ун-т. Новосибирск : ИПЦ НГУ, 2021. С. 26.
21. Электронная энциклопедия WEGA [Электронный ресурс] URL: http://pco.iis.nsk.su/wega (дата обращения: 11.04.2022)
22. Электронный словарь WikiGRAPP [Электронный ресурс] URL: http://pco.iis.nsk.su/grapp (дата обращения: 11.04.2022)

УДК 004.45

Лаборатория информационных систем. Основные научные результаты, полученные в 2021 году

Марчук А.Г. (Институт систем информатики СО РАН),

Андреева Т.А. (Институт систем информатики СО РАН),

Городняя Л.В. (Институт систем информатики СО РАН),

Демин А.В. (Институт систем информатики СО РАН),

Крайнева И.А. (Институт систем информатики СО РАН),

Пономарев Д.К. (Институт систем информатики СО РАН),

Тихонова Т.И. (Институт систем информатики СО РАН),

Филиппова М.Я. (Институт систем информатики СО РАН)

В статье изложены основные научные результаты, полученные лабораторией информационных систем ИСИ СО РАН в 2021, их связь с мировыми исследованиями и работами предыдущих лет.

Ключевые слова: *Лаборатория информационных систем ИСИ СО РАН, семантические методы, исследования по истории науки, методики обучения программированию, технологии обработки данных, информационные ресурсы.*

.

1. Введение

Обязательный раздел, в конце которого можно поместить информацию о поддержке исследования грантами.

Выполняемый в лаборатории проект направлен на: создание фундаментальных подходов и технологий для автоматизации предметных областей, решения задач в сфере анализа данных, прогнозирования, принятия решений; формирование принципов, методологии и технологии для историко-ориентированных информационных систем, в том числе электронных архивов, реализующих фактографический подход; описание, формализация и

развитие моделей, методик и технологий обучения программированию, востребованных в современной ИТ-индустрии; изучение важных классов задач, связанных с обработкой больших данных, предложение и техническая реализация эффективных алгоритмов.

Более конкретно, в фокусе интереса исследований были: нахождение и обоснование новых способов комбинирования логических, алгебраических и вероятностных методов описания предметных областей, интеграция символьных и субсимвольных моделей анализа данных.

Изучались особенности создания электронных архивов, в частности архивов СО РАН, на основе междисциплинарного взаимодействия гуманитаристики и точных наук (математики, информатики), акцентируя внимание на источниковедческом аспекте. Формулировались технологические и организационные проблемы, выявленные при создании источнико-ориентированных ИС (инженерный и научный подходы).

Продолжалось создание новых методик обучения программированию, методики опробоваются на Летних школах юных программистов (ЛШЮП) и в спецкурсах НГУ.

Производилась классификация задач и схем структуризации, характерных для различных предметных областей. Это касается классических задач делопроизводства, социальных сетей, обработки физической, биологической, астрофизической информации. Анализ существующих реализаций различных схем структуризации и алгоритмов обработки. Выделение модельных задач, формирование бенчмарков.

2. Семантические методы интеллектуального анализа, инженерии знаний и управления

Цель исследования заключается в разработке фундаментальных подходов и технологий для автоматизации предметных областей и решения задач в сфере анализа данных, прогнозирования, принятия решений.

Актуальность проблемы, предлагаемой к решению, заключается в общественной потребности в создании методов интеллектуального анализа данных и знаний в построении систем поддержки принятия решений. Конкурентное преимущество обеспечивают технологии, которые позволяют производить аналитику и прогнозирование на данных большого объема, легко интерпретируемы и универсальны в настройке на предметную область. Это обеспечивается интеграцией семантических методов моделирования предметных областей и вероятностных моделей обработки данных, развитием фундаментальной и инженерной базы.

2.1 Обнаружение причин и причинно-следственных связей в информационных моделях событий

Большинство направлений в информатике связаны с математическими моделями, которые дают ответ на вопрос «что» (например, ответ на запрос из базы данных, результат автоматизированного логического вывода и пр.) и лишь немногие позволяют отвечать на вопросы «почему не…», «что если…». При взаимодействии пользователя с системой такие вопросы возникают часто. Способность отвечать на них зачастую продуктивнее, чем возможность системы предоставить объяснение в виде последовательности сделанных своих шагов. В информационных моделях, которые представляют сложные взаимосвязи событий, подобная функциональность является ключевой, а задача установления причинно-следственных связей – весьма нетривиальной.

Цель исследования: создание технологии интеллектуальных систем-ассистентов, позволяющих рассуждать о причинно-следственных связях событий, обнаруживать причины, производить контрфактический анализ событий.

Само понятие причины трудно поддается осмыслению с общих позиций и, тем более, формализации. Примеры из юриспруденции наглядно показывают, что установление причины является комплексным процессом: причиной может быть как действие, так и бездействие одного или нескольких акторов. Более ста лет в литературе идет дискуссия по поводу того, как определить понятие причины. Многие подходы и модели за это время потерпели поражение от найденных интуитивных контрпримеров, которые опровергают ту или иную теорию причин. В последнее время были развиты пропозициональные модели событий, в которых делается попытка построения алгоритмов для выявления причин. Например, J. Pearl с соавторами в [1] развил подход на основе баесовских сетей. В [2] J. Halpern реализовал концепции каузальности Дэвида Юма в рамках систем присваиваний, формулирующих зависимости одних переменных от других. К сожалению, и эти подходы неустойчивы к ряду наглядных контрпримеров. По-видимому, пропозициональные подходы не способны ухватить широкую специфику каузальности, поскольку их язык недостаточно выразителен.

Задача исследования: построение математической модели событий, не уступающей в функциональности пропозициональным моделям и превосходящей их в спектре охватываемых предметных областей и приложений. Построение алгоритмов обнаружения причин, контрфактического анализа событий в рамках модели.

Гипотеза исследования: для анализа причинно-следственных связей необходима модель,

включающая такие ингредиенты, как: действие, понятия шкалы времени, выраженное через последовательность действий, параллельные и последовательные действия, пред/постусловия действий, способность рассуждать о будущем и прошлом (forward/backward reasoning). Все они, кроме параллельных действий, присутствуют в формализме Исчисления Ситуаций [3], основанном на логике первого порядка. В работе изучается применимость этого подхода, как математической основы для причинно-следственного анализа. В сотрудничестве с факультетом информатики Ryerson University, Toronto разрабатывается расширение Исчисления Ситуаций параллельными действиями, развиваются новые методы причинно-следственного анализа, разрабатываются алгоритмы обнаружения причин в информационных моделях событий. Ранее в сотрудничестве ИСИ СО РАН-Ryerson University были разработаны новые методы компонентного представления и анализа теорий действий в Исчислении Ситуаций [4-5], которые сформировали основу для настоящих исследований.

Результаты проведенных за 2021 год исследований опубликованы в [6-8].

2.2 Интерпретируемые логико-вероятностные модели обучения с подкреплением

Направление исследований "обучение с подкреплением" (reinforcement learning, RL) [9] занимается проблемой обучения агента путем его взаимодействия с окружающей средой методом "проб и ошибок". В последнее время доминирующим в RL стал подход "глубокое обучение с подкреплением" ("deep reinforcement learning", DRL), объединяющий классический RL и глубокие нейронные сети. К примеру, алгоритмы DRL позволили машине автоматически обучиться играть в игры Atari, получая на вход лишь сырые пиксельные данные [10], а также побить чемпионов мира в игру го [11]. Однако, при более детальном рассмотрении можно заметить, что своим успехом многие современные эффектные DRL приложения в основном обязаны использованию моделей глубоких нейронных сетей для анализа входной сенсорной информации. В частности, Atari DRL эффективно использует глубокие сети для преобразования сырых пиксельных данных в сжатое представление, пригодное для использования RL методами. При этом по прежнему остались нерешенными ряд традиционных проблем обучения с подкреплением, в частности: 1) проблема разбиения задачи на подзадачи (обнаружение подцелей) и 2) проблема интерпретируемости модели.

Решению первой проблемы занимается направление RL, называемое "иерархическое обучение с подкреплением" (hierarchical reinforcement learning, HRL) [12, 13], которое объединяет различные подходы к группировке элементарных действий для более

эффективного обучения агента и решения им задач. Однако одна из основных проблем большинства этих подходов заключается в необходимости заранее задавать подцели. Таким образом, задача автоматического обнаружения подцелей по прежнему остается крайне актуальной.

Вторая проблема, интерпретируемость модели, также является очень актуальной, поскольку доминирующие в RL нейросетевые модели работают по принципу "черного ящика". В настоящее время предлагается два основных решения: 1) обучение интерпретируемых моделей (деревья решений, набор правил и т.д.) по выходам уже обученной нейросетевой модели (к примеру, метод PIRL) [14]; 2) использование генетических методов [15]. Оба подхода нельзя напрямую отнести к RL методам, поскольку в первом случае необходимо сначала обучить нейросеть, а во втором требуется наличие популяции агентов. Таким образом, можно утверждать, что в настоящее время не существует RL подходов, дающих интерпретируемую модель сразу в процессе обучения агента.

Целью данного исследования является разработка интерпретируемых методов обучения с подкреплением с автоматическим формированием иерархии целей. В исследовании предлагается альтернативный подход, которой, с одной стороны, использует идеи организации управления из нейрофизиологической Теории функциональных систем, а с другой стороны, логико-вероятностный методы (ЛВ-методы) машинного обучения для формализации модели и тренировки агента. Одно из преимуществ использования ЛВ-методов состоит в том, что обнаруженные в результате обучения закономерности имеют явную форму, т.е. представлены в виде логических формул. Это, во-первых, сразу дает интерпретируемую модель, во-вторых, позволяет анализировать и использовать полученные закономерности для построения мета-алгоритмов (извлечение иерархии подцелей, построение плана, автоматическое обнаружение категорий, не сформулированных человеком).

В предыдущих работах [16-19] были предложены логико-вероянностные RL-модели, основанных на орбучении прогнозированию оценок результатов ствоих действий в определенных ситуациях (model-free модели). Данные модели были способны обнаруживать явные подцели и успешно решали двухэтапную задачу фуражирования. Однако недостатками предложенных ранее моделей являлось отсутствие поддержки обучения модели среды и способность обнарживать только явные подцели (подцели, информация о достижении которых постоянно присутсвует в сенсорном поле агента).

Были проведены следующие теоретические исследования:

1) Разработка логико-вероятностных RL-моделей с иерархией подцелей: model-free

(основанные на обучении отображению состояний и действий в прогноз награды, соответствуют большинству классических RL-моделей), model-based (основанные на обучении модели среды и формирующие прогноз награды на основе полученной модели).

2) Разработка логико-вероятностных алгоритмов обучения для разных типов RL-моделей с учетом иерархии целей.

3) Разработка методов автоматического построения иерархии целей, включая неявные подцели, на основе анализа закономерностей, полученных ЛВ-методами обучения, и истории работы агента.

4) Разработка методов автоматического построения концептов на основе анализа закономерностей (для формирования концептуального представления о среде и рефлексии агента с целью ускорения обучения, сокращения пространства перебора действий при исследовании сред, переноса опыта на новые среды).

Теоретические исследования подкреплялись практическими исследованиями и экспериментальными системами:

1) Программная реализация моделей и алгоритмов.

2) Разработка сред для экспериментального тестирования моделей.

3) Проведение экспериментальных исследований с целью апробации моделей и реализаций. Постановка задач для корректировки и доработки моделей и алгоритмов.

4) Сравнение предложенных моделей с другими RL-подходами на общедоступном бенчмарке (из репозитория https://gym.openai.com/).

Были получены следующие научные результаты:

1) Предложена логико-вероятностная RL-модель, основанная на обучении модели среды, и новый метод обнаружения глубоких неявных подцелей.

2) Проведены экспериментальные исследования предложенной модели на примере многоэтапной задачи фуражирования с выделением подцелей различной глубины. По результатам исследований были выполнены доработки модели и метода извлечения подцелей. Были выделены мета-параметры системы и исследовано их влияние на работу модели.

Результаты исследований изложены в [20].

В рамках научного сотрудничества с Mike Soutchansk (Department of Computer Science, Ryerson University, Toronto, Canada) проведены совместные исследования по математической формализации причин и причинно-следственных связей, в том числе для юридических кейсов. Начата работа над совместной статьей.

3 Системы и технологии поддержки исследований по истории науки, техники и образования

Цель исследования заключается в том, чтобы оценить современные потребности науки и образования в области создания и применения источнико-ориентированных информационных систем, обобщить теоретический и практический опыт в области понятийного аппарата, подходов к их проектированию, методов и технологий создания, принципов использования в исследовательской и образовательной практике.

Сформировать принципы, методологии и технологии для историко-ориентированных информационных систем, в том числе электронных архивов, реализующих фактографический подход.

Актуальность данной работы базируется на успешной практической реализации в конкретных исследованиях инструментального подхода к публикации источников по истории науки, техники и образования. Высокие требования к публикационной активности гуманитариев, требования новизны и актуальности исследований с новой силой ставят вопрос о большей и оперативной доступности исследователю архивного, библиотечного и прочего контента наследия. В этом случае особую актуальность приобретает доступность, открытость архивов. Опыт ИСИ СО РАН, который в числе первых научных коллективов осуществил несколько проектов по созданию, научной интерпретации, методической и организационной работе в области академических цифровых архивов может быть актуализирован как технологичный, научно обоснованный и успешно апробированный.

3.1 Сохранение научного наследия на базе ИТ и исследования исторического характера

Большая часть научного наследия Сибирского отделения АН СССР/РАН с момента его образования в 1957 г., является объектом хранения Научного архива СО РАН (НАСО). С 2014 г. после сокращения штатов в результате реформы РАН, НАСО прекратил формирование научных коллекций. Перед коллективом Лаборатории информационных систем ИСИ СО РАН стоит задача мессианского характера: сохранить наследие Отделения насколько это возможно, разместив документы в электронных архивах. Нами выявляются как объекты наследия первостепенной значимости, так и случайные коллекции, которые нам предоставляют фондообразователи или их наследники (фондохранители). Таким образом наполняются электронные архивы СО РАН: Открытый архив и Фоторахив.

В настоящее время сформировано несколько направлений, по которым идет наполнение контента электронных архивов: персональные фонды, институциональные фонды, фонды общественных организаций, фонд устной истории. Наиболее активно формируются персональные фонды. В 2021 г. Открытый архив СО РАН пополнился несколькими значимыми коллекциями:

- Фонд академика Н.Л. Добрецова (1936-2020), заместителя председателя СО РАН (1989-1997), председателя СО РАН (1997-2008). (3697 сканов документов)
- Фонд д.ф.-м.н. Л.А. Боярского (1933-2020) – ведущего научного сотрудника Института неорганической химии СО РАН, профессора, зав. кафедрой физики низких температур НГУ, руководителя клуба любителе кино «Сигма» в ДУ СО РАН (1965-2020). (515 ск.)
- Фонд чл.-корр. А.А. Ляпунова (1911-1973) пополнился ценной перепиской с учениками, присланной нам их Чехословакии. (196 ск.)
- Фонд академика Г.И. Марчука (1925-2013) – положено начало формирования данного фонда, который планируется создать к 100-летию со дня рождения ученого. (224 ск.)
- История Дома ученых СО РАН пополнилась документами от нашего корреспондента из США. (62 ск.)

Всего в 2021 г. на платформе открытых архивов размещено 6674 скана документов (см. Фотоархив СО РАН – 430 сканов фото.

Открытый архив СО РАН – 6244 сканов документов.

Наши исследования лежат в русле мировых трендов. Информатизация научно-исследовательской деятельности – актуальное направление в области развития коммуникативных процессов, которое осуществляются через сетевую организацию вычислительной техники, и поддерживает доступность контента наследия, размещенного в Сети. Фокус-группа проектов – научное сообщество. Глобально и институционально это направление исследований и практических шагов связано с возрастанием потока информации. В то же время, мировое научное сообщество озабочено обеспечением качественной информацией, поэтому идея большого виртуального архива науки для историко-научных и прочих исследований гуманитарного направления вполне своевременна.

Идея архивации наследия оказалась настолько плодотворной, что вызвала к жизни несколько международных проектов по унификации подходов его оформлению. Одним из них стал общеевропейский проект The Open Archives Initiative (OIE, 1999) [21]. OIE «разрабатывает и продвигает стандарты функциональной совместимости, которые

направлены на содействие эффективному распространению контента. OIE стремится улучшить доступ к электронным архивам как к средству повышения доступности научного общения. Однако фундаментальные технологические рамки и стандарты, которые разрабатываются для поддержки этой работы, не зависят ни от типа предлагаемого контента, ни от экономических механизмов, которые окружают этот контент, и обещают иметь гораздо более широкое значение для открытия доступа к целому ряду цифровых источников. По мере того, как OIE получает больше знаний о масштабах применимости разрабатываемых базовых технологий и стандартов, понимает структуру и культуру различных сообществ последователей, предполагается внесение постоянные эволюционных изменений как в миссию, так и в организацию OIE [22]. Группа OIE разработала некоторые спецификации, такие как Протокол сбора метаданных, Руководство по внедрению Протокола OIE для сбора метаданных и т. д. Ассоциация предложила кодирование XML в качестве механизма упаковки для собранных метаданных.

Практически одновременно был запущен проект euroCRIS – The International Organization of Research Information. EuroCRIS предусматривает, что Общий европейский формат исследовательской информации (CERIF) является всеобъемлющей информационной моделью в области научных исследований. Он предназначен для поддержки обмена исследовательской информацией между платформами CRISs. На этом основаны Руководящие принципы OpenAIRE для CRIS-менеджеров [23]. Технический комитет по взаимодействию и стандартам (TCIS) был создан в сентябре. 2020, параллельно с CERIF TG, и направлен на процесс принятия решений для развития CERIF и связанных с ним технических продуктов. TCIS призван установить дорожную карту и стратегию для широкого внедрения CERIF и его согласования с другими соответствующими технологиями, моделями и стандартами.

Среди последних отечественных разработок в области электронного хнаннеия исторических артефактов назовем проект Европейского университета в Санкт-Петербурге «Прожито», запущенный в 2015 г. Здест аккумулируются такие эго-документы, как дневники. Корпус включает датированные тексты на русском и украинском языках. Общий объем корпуса – более полумиллиона записей XVIII–XX вв. Загружено около 500 [4].

3.2 Исследования исторического характера

Пополнение электронных архивов позволяет решать исследовательсие задачи. На базе имеющихся и вновь полученных документов проводятся исследования по истории науки в Сибири. Фокус исследований сосредоточен как на технических и технологических, так и на

гуманитарных вопросах. В частности, акцентируется внимание на персональных биографиях женщин-ученых, истории научных династий, на отдельных периодах истории науки (Великая отечественная война, начальные период становления СО АН СССР). Нами подготовлено и опубликовано несколько статей для журналов, сделаны научные доклады на профильных конференциях.

Так, на основе документов из коллекции академика Татьяны Ивановны Заславской (1927–2013) и доктора филологических наук Майи Ивановны Черемисиной (1924–2013), в девичестве сестер Карповых, проведен краткий анализ данного персонального фонда. В России известно несколько научных династий, которые ведут свою историю с начала XIX века: Ляпуновы-Анри, Капицы-Милевские, Лаврентьевы, Вернадские, Шмальгаузены, Ворожцовы и др. К подобным династиям принадлежали сестры Карповы, достойные потомки первой в семье женщины-ученой Ольги Карловны Крафт, которая получила степень доктора медицины в Париже в 1884 г. Дед по материнской линии Г.Г. Де-Метц (1861–1947) – физик, профессор, был деканом физико-математического факультета, затем ректором Киевского Императорского университета Св. Владимира, одним из организаторов Киевского политехнического университета и Кубанского государственного университета. Отец Иван Васильевич Карпов (1897–1965) – кандидат педагогических наук, историк, в качестве вольноопределяющегося воевал в Первой мировой. Исследуя историю семьи Де-Метц–Крафтов–Карповых, мы ставили задачу «прочесть» тексты персональных историй в контексте культуры и социума, выявить глубинное влияние семейной истории на личную. Кроме того мы хотели привлечь внимание коллег к данной коллекции [25].

В качестве еще одного примера использования материалов Открытого архива СО РАН с привлечением материалов Государственного архива РФ является реконструкция персональной истории одного из представителей культурной московской семьи Румеров – Исидора Борисовича Румера (1884–1938). Биография прослежена в актуальном контексте взаимоотношения российской интеллигенции и власти. Внимание концентрируется на особенностях репрессивной политики в отношении интеллигенции в довоенный период с акцентом на 1930-е гг. В это время происходили необратимые перемены в экономической, политической и культурной политике Советского государства. Они состояли в насаждении единой идеологической доктрины, командных методов управления учреждениями науки и культуры, в фактическом распоряжении властью всей собственностью. Несогласие сопровождалось массовыми репрессиями в отношении явных и мнимых противников. Среди них – представители отечественной интеллигенции, деятели культуры и науки и техники, в своей массе не являвшиеся непримиримыми противниками Советской власти. Их

критический настрой, как правило, носил латентный характер, являлся проявлением ментальности интеллигентов-гуманистов. Они озвучивали свое мнение в свободном общении в кругу единомышленников. Исследование проведено с привлечением архивных материалов, публикаций по теме, осмысления феномена отечественной интеллигенции [26].

Поскольку Открытые архивы СО РАН продолжают пополняться, они предоставляют неисчерпаемые возможности для исследований в области истории науки и техники, биографики, локальной истории науки. Результаты исследований изложены также в [27-30].

4 Развитие новых моделей, методик и технологий обучения программированию

Целью исследования является описание, формализация и развитие моделей, методик и технологий обучения программированию, накопленных за историю развития вычислительной техники и активно востребованных в современной ИТ-индустрии.

Проблема образования в информатике и программировании упоминается в лекциях многих лауреатов премии Тьюринга, тем не менее, её полного решения пока достичь не удалось, не все достижения этого направления сохранены в наши дни. Прецедент успешного образования программистов под Ершовским лозунгом «Программирование — вторая грамотность» формировался шире, чем чисто университетское или школьное обучение на уровне первичных навыков, дополнением было и самообразование, и дистанционное общение, и факультативные формы типа школ юных программистов (ШЮП).

Актуальность проблемы давно видна флагманам компьютерной индустрии, испытывающим серьёзные трудности в привлечении специалистов, имеющих навыки приаппаратного программирования и обладающих изобретательскими способностями для повторного программирования системных библиотек.

4.1 Развитие парадигмального анализа языков и систем программирования

При исследовании и сравнении парадигм программирования получено два результата:

1) Описана мультипарадигмальность параллельных вычислений. Получение этого результата опирается на методику парадигмального анализа языков и систем программирования, применённую к проблемам организации параллельных вычислений и многопоточных программ для многопроцессорных комплексов и распределённых систем. Такая методика позволила сделать вывод о неявной мультипарадигмальности параллельных

вычислений. На основе этого вывода сформулированы требования к учебно-производственному мультпарадигмальному языку параллельного программирования [31].

2) Представлены принципы функционального программирования. При получении этого результата методика парадигмального анализа языков и систем программирования применена к описанию особенностей функционального программирования как парадигмы, выполняющей функции проектно-конструкторского бюро для производственного и параллельного программирования. В результате сформулированы принципы функционального программирования, позволяющие отличать его от других парадигм программирования. Описаны следствия из этих принципов, конкретизированы принципы и следствия при переходе к проблемам параллельного программирования, а также при выполнении работ по повышению эффективности и производительности программ [31, 32].

В мире подобные задачи рассматривались в [33-36]. В настоящее время происходит расширение двух взаимодействующих, формально почти противоположных, направлений. Первое связано с экстенсивным порождением новых проблемно-ориентированных языков программирования (DSL) — их за последнее десятилетие насчитывается десятки тысяч. Второе отражает рост актуальности проблем производительности ПО, особенно на уровне энергопотребления, использования памяти и немонотонности скоростей исполнения программ, что было не очень заметно ещё пять лет назад.

Первое направление провоцирует избыточный объём производства ПО без принципиального роста производительности программ, связанного с техникой синтаксического конструирования новых ЯиСП преимущественно на базе Clang-LLVM над языками семейства C/C++, что ограничивает реализационное пространство возможностями прежней базы, сложившейся на однопроцессорных конфигурациях ещё в 1970-80-е годы, без перехода к рациональному освоению преимуществ новой элементной базы, многопроцессорных конфигураций и компьютерных сетей. Легко создать новый удобный компьютерный язык, трудно обеспечить заметный рост производительности программ, создаваемых с его помощью.

Второе направление приторможено риском новой реализации приаппаратных решений, для которых даже самые мощные производители аппаратуры испытывают дефицит необходимой высокой квалификации разработчиков. По этой причине по 20-30 лет эксплуатируются неизменные ядра компиляторов и ведутся массовые, не имеющие перспектив успеха, эксперименты по решению приаппаратных проблем изобретением методов высокоуровневого декарирования программ. Таких подходов уже опубликовано более ста, идёт поиск новых при отсутствии предложений по пересмотру языковых моделей

работы с аппаратурой. Легко измерять суммарную производительность программы, сращенной с аппаратурой, операционной системой и системой программирования. Трудно выделить из полученных результатов измерения вклад программируемых решений в производительность программ.

Таким образом, оба направления вышли на уровень трудно решаемых проблем. Обычно решение трудных проблем лежит через поиск альтернативного подхода или синтез с комплексом других направлений, уже имеющих технику решения сложных задач. Прецеденты работоспособной техники решения таких сложных задач имеются в современных оптимизирующих компиляторах и накоплены при исследовании методов верификации и моделирования параллельных процессов. Альтернативный подход может дать пересмотр методов реализации компьютерных языков, что требует дальнейшего исследования, особенно при их синтезе с методами верификации, логического вывода и функционально-информационного моделирования. В рассмотренных материалах доступных международных конференций таких обобщённых попыток обнаружить не удалось.

Анализ современного состояния исследований в области ЯиСП показывает существование трудно преодолимой дистанции между потенциалом современных ИТ, достижениями теоретических исследований и исторически сложившейся практикой создания информационных систем, что тормозит прогресс, особенно заметный в области параллельных вычислений, обретающих практический интерес на базе массового доступа к многопроцессорным архитектурам. Заметен яркий дисбаланс между мощностью средств интеграции программных комплексов и дефицитом инструментальных возможностей корректного выделения программных компонент без потери их функциональности. Это отмечено в материалах ICS International Conference on Supercomputing как ведущего интернационального форума по представлению результатов высокопроизводительных вычислений ,отражающего все аспекты развития, исследования и применения суперскалярных экспериментальных и коммерческих программных систем поиска научных ответов на вызовы современных ИТ.

Это по существу создаёт почву для внедрения в практику результатов теоретических работ в области верификации и системного анализа, практичную основу которых может составить функциональный подход, поддержка жизненного цикла разработки безопасного ПО, инкрементального анализа систем (подразумевает быструю повторную проверку недавно измененного кода) и автоматическую генерацию тестов для сложных программных систем. В этом плане интересен Svace – необходимый инструмент жизненного цикла разработки безопасного ПО, обнаруживающий более 50 классов критических ошибок в

исходном коде. Имеется поддержка навигации по коду — разделение срабатываний на истинные и ложные;— миграция результатов между запусками и сокрытие ложных срабатываний. Привлекает внимание и ИСП Crusher – программный комплекс, комбинирующий несколько методов динамического анализа.

Ранее, в Лаборатории информационных систем ИСИ СО РАН по этому направлению были получены результаты [31, 32], показывающие основу для проведения дальнейших исследований в таком направлении.

Попытки описания принципов функционального программирования и методик обучения в мире регулярно рассматриваются на международных конференциях, посвященных перспективам функционального программирования, поддерживаемых ACM и другими авторитетным организациями. [33-38].

Опыт реализации языков функционального программирования и современная тенденция к мультипарадигмальности новых языков программирования дали достаточные основания для перехода к ЯиСП, одновременно содержащим интерпретатор, компилятор, мемоизатор и декомпозитор программ, а также для создания новых средств более высокого уровня, сравнимого с механизмами языков сверх высокого уровня. Появились попытки выделения типовых семантических систем из определений разных ЯиСП. Само название «The next 700 semantics: a research challenge» [41] можно рассматривать как симптом избыточного разнообразия выделяемых компонент. Избыточность препятствует определению чётких критериев систематизации языков и парадигм программирования.

Исследование подходов к систематизации средств и методов разработки ПО активизировано в сентябре 2009 года Иваром Якобсоном, Бертраном Мейером и Ричардом Соули, выступившим с инициативой SEMAT, основы которой они изложили в своей книге «The Essence of Software Engineering: Applying the SEMAT Kernel». Их идею поддержали многие гуру программирования и включили разные корпорации. В настоящее время Бертран Мейер, автор языка Eiffel и книг по объектно-ориентировнному проектированию, с другими коллегами разворачивает инициативу «PEGS - Project, Environment, Goals and System» по исследованию проблем определения инженерных требований к ПО. Учитывая, что профессиональная карьера Б.Мейера началось с работы по достаточно интересному языку спецификаций Z0, можно предполагать хорошие результаты [36, 37].

Общий вывод из содержания рассмотренных материалов по отношению ЯиСП и проблемам параллельного программирования и производительности программ сводится к констатации ряда не решённых проблем. Значение работ в области ЯиСП освещено на ряде конференций [33-38].

Нет средств удобного представления взаимодействующих потоков и их эффективного перевода в процессы, не происходит преодоление страха перед разработкой и отладкой параллельных алгоритмов, особенно в случае лёгкого доступа к готовым последовательным алгоритмам, не созданы методики решения образовательных проблем, включая осознание реальных возможностей аппаратуры, требующих проявления интуитивной грамматики параллельных процессов. Ещё одна проблема связана с отсутствием общепризнанной методики представления результатов измерения вклада программируемых решений в производительность программ. Хотя специалисты, исследующие эту проблему, отмечают низкий уровень заинтересованности программистского корпусе в измерениях, они потратили 20 лет на измерение производительности программых приложений.

Ранее, в Лаборатории информационных систем ИСИ СО РАН были получены результаты [31, 39-43], позволяющие наглядно представлять результаты анализа определений языков программирования и измерения эксплуатационных характеристик программируемых решений, влияющих на производительность программ.

4.2 Создание автоматизированной системы подготовки материалов для олимпиад и тестов

Целью исследований является разработка информационной системы, предназначенной для полной или частичной автоматизации подготовки материалов (задачных и тестовых наборов) для проведения олимпиад и проверочных работ в различных областях знаний (включая гуманитарные науки), позволяющей снизить трудоемкость и сложность подготовительных процессов и расширить за счет этого снижения целевую аудиторию пользователей. Актуальность темы значительно возросла на фоне массового перехода образования в дистанционный формат. Однако исследования по данной теме практически не проводятся: за последние три года в научной электронной библиотеке e-Library зарегистрировано всего шесть работ по близкой тематике (см. [44-49]). Все они посвящены частным вопросам автоматизации проверок заданий, преимущественно в программировании и близких к нему науках. Попыток обобщить и распространить опыт автоматизации подготовки тестовых заданий на другие области школьного и вузовского образования не производилось.

Изначально целью исследований было создание комплексной информационной системы, предназначенной для подготовки материалов (задачных и тестовых наборов) для проведения олимпиад и тестов, позволяющей снизить трудоемкость и сложность подготовительных процессов и расширить за счет этого снижения целевую аудиторию пользователей.

Ранее система подготовки тестовых комплектов была расширена до системы подготовки задачных комплектов, включающих условие задачи, спецификацию входных и выходных данных, тестовый набор, авторское решение, принцип вынесения вердикта о правильности предоставленного участником решения.

В текущем периоде была продолжена разработка системы, а) формализующей при помощи разработанного математического аппарата спецификации входных и выходных данных, заданных на естественном языке; б) генерирующей тестовые наборы исходя из формализованных спецификаций; в) контролирующей совместимость набора спецификаций и форматов данных; и д) проверяющей непротиворечивость, полноту и неизбыточность тестовых наборов.

Внесены дополнения и уточнения в математическую модель вынесения вердиктов о правильности решений на основе различных способов тестирования решений. Поскольку правильность вердикта существенно зависит от качества тестовых наборов, основное внимание было уделено разбору частных случаев некорректных и псевдо-корректных процедур оценивания.

Особое внимание было уделено возможности коллективного использования системы, поэтому возникла необходимость включить в систему дополнительные операции с тестовыми наборами: проверка соответствия тестов формализованным спецификациям, сведение воедино нескольких тестовых наборов, проверка непротиворечивости, полноты и неизбыточности тестовых наборов в соответствии с формализованными спецификациями.

В рамках исследования параллелизма процессов, возникающих при автоматизации коллективной разработки задачных наборов, были сделаны важные замечания, развитие которых планируется осуществить в дальнейшем.

4.3 Разработка методов автоматической верификации тестовых наборов

Многочисленные примеры тестовых систем, вошедшие в школьное и вузовское обучение на волне роста дистанционного образования (Moodle, SkySmart, ЯКласс, др.) выявили недостаток внимания к проблеме верификации тестов, создаваемых пользователями этих систем. Проверка правильности создаваемых тестов в этих системах совершенно не автоматизирована. Кроме того, не учитываются различия в типах тестовых заданий, которые могли бы сократить количество ошибочных заданий.

В дополнение к методам автоматической проверки генерируемых тестовых наборов, проводится разработка математически обоснованных методов автоматической и

полуавтоматической верификации полноты и непротиворечивости тестовых заданий, создаваемых пользователями вручную.

Поскольку правильность вердикта существенно зависит от качества тестовых наборов, основное внимание было уделено разбору частных случаев некорректных и псевдо-корректных процедур оценивания.

Для наиболее полного охвата различных типов возможных ошибок проводится сбор примеров ошибочных заданий, созданных пользователями дистанционных образовательных систем (Moodle, SkySmart, ЯКласс, др.)

4.4 Практическая проверка

В условиях дистанционной работы со студентами НГУ промежуточные результаты исследований были подвергнуты практической проверке на примере системы MOODLE (НГУ), предоставляющей недостаточный инструментарий для создания тестов, а также на примере системы автоматической проверки решений, используемой на 1 курсе ФИТ НГУ (getlab.ccfit.nsu.ru, автор Петров Е.С.).

По сформированным в ИСИ СО РАН методикам в 2021 году была проведена очередная Летняя школа юных программистов (ЛШЮП-2021) [50]. В связи с эпидемиологической обстановкой, школа проводилась дистанционно, в on-line режиме.

5 Технологии поддержки обработки больших структурированных данных

Целью исследований является изучение важных классов задач, связанных с обработкой больших объемов данных, предложение и техническая реализация эффективных алгоритмов обработки [51]. Разработка заказных и библиотечных решений, развитие и поддержка системы Polar [52].

Актуальность проблемы, предлагаемой к решению заключается в том, что происходящий процесс информатизации множества предметов и процессов деятельности, увеличивающиеся объемы вовлекаемой в обработку информации, интеграция данных и знаний, определяют актуальность разработки эффективных средств структурирования и обработки структурированных данных. Появилось множество подходов, объединенных общим понятием NoSQL, развиваются распределенные системы хранения/обработки, предлагаются и стандартизуются новые подходы с структуризации данных. Вместе с тем, объемы доступных для данных опережают технические возможности компьютерных систем, сетевые

средства ограничивают производительность и доступность большой обработки, что требует нахождения и реализации новых подходов [53].

Продолжались исследования задачи построения и обработки графа де Брёйна (de Bruijn) как одного из представителей актуальных видов данных большого и сверхбольшого размера. Данная модель используется в задаче восстановления цепочек нуклеотидов по результатам секвенирования [54]. Были рассмотрены варианты работы с графом, причем упор делался на экономное программирование как для одномашинной реализации, так и для многомашинной кластерной конфигурации. Полученные программы позволяют решать задач построения частного варианта графа Де Брёйна и решать на этом графе задачу формирования (отслеживания) цепочек узлов. Созданные программы частично сравнивались с решением с использованием инфраструктуры Hadoop в варианте Spark, полученные результаты свидетельствуют в пользу предложенного подхода [55].

Продолжились исследования по созданию средств обработки структурированных данных. Работы 2021 года концентрировались вокруг проблем обработки структурированных данных предельно большого размера. Такие данные появляются в результате масштабных физических экспериментов, больших систем сквозного измерения, накапливания данных отдельных областей, биологических и медицинских массивов данных и т.д. Была построена модель основной части структурирования больших данных – последовательности однотипных элементов [56]. В частности, было сформировано представление об универсальном индексном построении. Модель позволяет в едином стиле рассматривать сосредоточенные и распределенные базы данных. Часть модели реализована в библиотеки PolarDB.

Было проведено исследование подходов к модернизации электронных архивов, создаваемых в ИСИ СО РАН. Некоторые из них эксплуатируются более 20 лет. Перевод на современные технологические платформы может сопровождаться решением дополнительных задач, таких как интеграция таких архивов. Были проведены эксперименты с Электронным архивом академика А.П.Ершова, Открытым архивом СО РАН и Фотоархивом СО РАН [57].

6 Создание и поддержание информационных ресурсов, ориентированных на поддержку науки и образования

Основными направлениями в 2021 г. были следующие:

Разработка новой концепции и нового графического дизайна сайта ИСИ СО РАН.

Дальнейшее развитие электронного Архива имени академика А.П. Ершова.

Разработка и поддержка сайтов проектов ИСИ СО РАН.

В 2021 году начата подготовка к обновлению сайта ИСИ СО РАН. Была выработана концепция обновленного сайта, запланированы изменения структуры, подготовлены макеты графического дизайна. Конкретно были выполнены следующие работы:

1) Миграция сайта в новое программное окружение – на платформу Drupal 9 и PHP 7;
2) адаптация сайта под мобильные устройства различных разрешений.
3) разработка новой структуры сайта с учетом добавления новых типов материалов, а также изменившихся требований к сайту со стороны вышестоящих организаций;
4) создание современного дизайна взамен устаревшего, который не изменялся с 2010 года.
1) Разработаны несколько вариантов макетов главной страницы, а также и нескольких типов второстепенных страниц для 4 основных разрешений (xs - мобильные телефоны, sm – планшеты, md – компьютеры, lg - компьютеры с широкоформатным монитором).
2) разработаны макеты для иллюстрации оформления отдельных видов материалов и структурных блоков – видео- и фотогалерей, списков, таблиц и т.д.

4) переработана структура расположения материалов и структурных элементов - например, визуализация меню;

5) разработка дизайна велась с учетом того, что основная масса сотрудников института - люди старшей возрастной группы, не очень любят перемены и должны иметь возможность легко ориентироваться на обновленном сайте.

В 2021 г. продолжались работы по дальнейшему Электронного архива академика А.П. Ершова, мигрированного на свободно распространяемое ПО с открытым кодом на платформе Drupal. Ведутся работы по увеличению производительности приложения. Были внедрены изменения в соответствии с запросами пользователей в части бэкенда и фронтенда.

В течение всего отчетного периода продолжались работы по поддержке и обеспечению хостинга сайтов, разработанных в ИСИ СО РАН. Эти работы включают как административные функции

- обеспечение резервного копирования,
- своевременное обновление модулей третьих сторон, что особенно важно при использовании Open Source разработок,
- быстрая реакция на непредвиденные обстоятельства – отключение электроэнергии, сбой аппаратуры и т.д..

так и работы по поддержке – коммуникации с владельцами, обновление материалов по просьбе владельцев, работа с письмами пользователей. В настоящее время на поддержке находятся следующие сайты:

1) Сайт ИСИ СО РАН;
2) Мемориальная библиотека имени А.П. Ершова;
3) Архив академика А.П. Ершова;
4) Сайт Бюллетеня Новосибирского Вычислительного Центра;
5) Сайт серии конференций PSI;
6) Сайт серии конференций PSI до 2017 года;
7) Сайт конференций SORUCOM-2014;
8) Сайт проекта «Информатика и программная инженерия»;
9) Сайт Интеграционного проекта СО РАН № 21 «Веб-пространство»;
10) Сайт «Хроника Сибирского отделения»;
11) Портал MathTree;
12) Коллекция старинных математических книг;
13) Сайт ВНТК "СТАРТ";
14) Сайт «50 лет Отделу программирования».
15) Сайт проекта Кронос;
16) Сайт Музея СО РАН;
17) Исторический портал ММФ НГУ Global MMF;
18) Сайт кафедры программирования ММФ НГУ;
19) Мемориальный сайт Г.И. Марчука;
20) Мемориальный сайт А.Ф. Рара;
21) Мемориальный сайт А.А. Берса;
22) Юбилейный сайт В.Е. Котова;
23) Юбилейный сайт А.Г. Марчука;
24) Исторический сайт «Аллея памяти».

7. Заключение

Все поставленные на 2021 год задачи были выполнены. Были получены существенные результаты по заявленным направлениям:

В исследованиях по причинно-следственному анализу разрабатывается расширение Исчисления Ситуация параллельными действиями, развивается метод причинно-следственного анализа, разрабатываются алгоритмы для цели настоящих исследований.

Была предложена логико-вероятностная RL-модель, основанная на обучении модели среды и новый метод обнаружения глубоких неявных подцелей. Были проведены экспериментальные исследования предложенной модели на примере многоэтапной задачи фуражирования с выделением подцелей различной глубины. По результатам исследований были выполнены доработки модели и метода извлечения подцелей. Были выделены мета-параметры системы и исследовано их влияние на работу модели.

Были выявлены объекты исторического наследия первостепенной значимости и коллекции, которые предоставляют фондообразователи. Открытый архив СО РАН пополнился несколькими значимыми коллекциями.

Выполнен парадигмальный анализ более десяти парадигм и ряда языков параллельного программирования. Создано описание принципов функционального программирования, рассматриваемого как мета-парадигма для организации параллельных вычислений.

С помощью созданного в ИСИ СО РАН программного обеспечения было создано значительное число промышленных и экспериментальных информационных систем. Основные действующие информационные системы в разное время прошли модернизацию и адаптацию к меняющимся технологиям базовых платформ. В 2021 году исследовались принципы сосуществования разных информационных систем, вопросы их интеграции и дезинтеграции. Были проведены содержательные эксперименты по частичному или полному включению ресурсов одного электронного архива в состав другого. Показано, что такое включение может осуществляться без разрушения целостности систем, представляющих авторскую композицию.

Был выполнен большой объем работ по поддержанию и модернизации информационных ресурсов, ориентированных на поддержку науки и образования. Ресурсы созданы в ИСИ СО РАН.

Проведенные исследования выполнены с применением научного метода и, в частности, удовлетворяют требованиям к полноте и опубликованности полученных результатов. В конкретных случаях полученные результаты используются в прикладных исследованиях и разработках или смогут стать основой создания программных продуктов, баз данных и технологий. Технико-экономическая эффективность внедрения результатов не оценивалась. Полученные результаты выполнения НИР находятся на лучшем отечественном и мировом уровне.

Список литературы

1. Judea Pearl, Dana Mackenzie. The Book of Why: The New Science of Cause and Effect // Basic Books, Inc., US, 2018.
2. Joseph Halpern. Actual Causality // MIT Press, 2019.
3. Raymond Reiter. Knowledge in Action // MIT Press, 2001.
4. D. Ponomaryov, M. Soutchanski. Progression of Decomposed Situation Calculus Theories. // Proc. 27th conference on Artificial Intelligence, AAAI'13, July 14-18, Bellevue WA, USA.
5. D. Ponomaryov, M. Soutchanski. Progression of Decomposed Local-Effect Action Theories. // ACM Transactions on Computational Logic (TOCL) 18(2):1-41, 2017.
6. Denis Ponomaryov On the Relationship Between the Complexity of Decidability and Decomposability of First-Order Theories ISSN 1995-0802, Lobachevskii Journal of Mathematics, 2021, Vol. 42, No.

12, pp. 2905–2912.DOI: 10.1134/S199508022112026X Web-ссылка на статью должна появиться в конце ноября-начале декабря. WoS, Scopus

7. P. Emelyanov, M. Krishna, V. Kulkarni, S.K. Nandy, D. Ponomaryov, and S. Raha Factorization of Boolean polynomials: Parallel algorithms and experimental evaluation. Programming and Computer Software 47: 108-118, 2021. https://doi.org/10.1134/S0361768821020043 https://link.springer.com/article/10.1134/S0361768821020043 WoS, Scopus
8. Пономарев, Д. К. Декомпозиция логических теорий: вычислительные проблемы и приложения // Всероссийская научная конференция «Математические основы информатики и информационно-коммуникационных систем». Сборник трудов. — Тверь : ТвГУ, 2021. — С. 57–60. https://doi.org/10.26456/mfcsics-21-7, РИНЦ
9. Sutton R.S., Barto A.G. Reinforcement Learning. London // MIT Press. – 2012. – 320 p.
10. Mnih V., Kavukcuoglu K., Silver D. et al. Human-level control through deep reinforcement learning // Nature 518. – 2015. – pp. 529-533.
11. Silver D., Huang A., Maddison C. et al. Mastering the game of Go with deep neural networks and tree search // Nature 529. – 2016. – pp. 484-489.
12. Al-Emran Mostafa. Hierarchical Reinforcement Learning: A Survey // IJCDS Journal 4(2) . – 2015. – pp. 137-142.
13. Dietterich T.G. Hierarchical reinforcement learning with the MAXQ value function decomposition // Journal of Artificial Intelligence Research, 13. – 2000. – pp. 227–303.
14. Verma, A.; Murali, V.; Singh, R.; Kohli, P.; and Chaudhuri, S. 2018. Programmatically interpretable reinforcement learning // In 35th International Conference on Machine Learning, ICML 2018, volume 11.
15. Juang, C. F.; Lin, J. Y.; and Lin, C. T. 2000. Genetic reinforcement learning through symbiotic evolution for fuzzy controller design // IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 30(2).
16. Vityaev E.E., Demin A.V., Kolonin Y.A. Logical probabilistic biologically inspired cognitive architecture // Artificial General Intelligence - 13th International Conference, AGI 2020, Proceedings. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). – Springer Gabler, 2020. – V. 12177 LNAI. – p. 337-346.
17. Vityaev E.E., Demin A.V. Cognitive architecture based on the functional systems theory // Procedia Computer Science. – Elsevier, 2018. – V. 145. – p. 623-628.
18. Vityaev E.E., Demin A.V. Recursive subgoals discovery based on the Functional Systems Theory // Biologically Inspired Cognitive Architectures 2011, IOS Press, 2011. – p. 425-430.
19. Демин А.В., Витяев Е.Е. Логическая модель адаптивной системы управления // Нейроинформатика. – 2008. – Т. 3. – № 1. – С. 79-107.

20. Демин А.В. // Глубокое обучение адаптивных систем управления на основе логико-вероятностного подхода. – Известия Иркутского государственного университета. Серия «Математика» – Иркутск, 2021. – Т. 38. – С. 65-83.
21. Carl Lagose and Herbert Van der Sompel, The Open Archives Initiative: Building a low-barrier interoperability framework // https://www.openarchives.org/documents/jcdl2001-oai.pdf
22. Open Archives Initiative Organization https://www.openarchives.org/organization/
23. Parinov S., International Professional Association of Research Information System Specialists euroCRIS and its Main Product CERIF // http://ceur-ws.org/Vol-1297/6-9_paper-2.pdf
24. Прожито https://prozhito.org/ (дата обращения 04.04.2022).
25. Савелова О.А., Крайнева И.А. Электронный архив сестер Карповых: история научной династии // Исторический курьер. 2021. № 2 (16). С. 25–35. ВАК.
26. Крайнева И.А. Исидор Борисович Румер: страницы биографии // Диалог со временем. 2021. № 74. С. 140-155. Scopus/WoS
27. Крайнева И.А. Академик Будкер: «…будоражить умы и кресла» // История науки и техники. 2021. №3. С. 20-31. ВАК.
28. Крайнева И.А. Ржановы под Ленинградом: выжить, защищая город // Сборник 18-х международных чтений «Право на имя: Биографика XX века» памяти Вениамина Иофе. Санкт-Петербург, 2021. С. 43-54.
29. Krayneva I., Savelova O. The female face of programming (mid 1950s – early 21st century) / HISTELCON-21. Moscow, Nov. 10-12, 2021.
30. Крайнева И.А., Сэнборн К., Мериste М. Энн Тыугу: история эстонского программиста / "Право на имя: Биографика 20 века" 19-е чтения памяти Вениамина Иофе. Санкт-Петербург, 20-22 апреля 2021.
31. Городняя Л.В. О неявной мультипарадигмальности параллельного программирования // Научный сервис в сети Интернет: труды XXIII Всероссийской научной конференции (20-23 сентября 2021 г.). — М.: ИПМ им. М.В.Келдыша, 2021. — С. 104-116.
32. Gorodnyaya L.V. The Role of Functional Programming in the Organization of Parallel Computingь CEUR Workshop Proceedings ICS '19: Proceedings of the ACM International Conference on Supercomputing https://dl.acm.org/conference/ics
33. https://online.isprasopen.ru/docs/ISPRAS_Svace.pdf
34. https://cs.brown.edu/~sk/Publications/Papers/Published/kle-next-700-semantics/paper.pdf
35. Языки программирования и компиляторы — 2017: Труды конференции / Южный федеральный университет ; под ред. Д.В. Дуброва. - Ростов-на-Дону : Издательство Южного федерального университета, 2017. - 282 с. http://plc.sfedu.ru/files/PLC-2017-proceedings.pdf
36. http://keldysh.ru/abrau/2020/ - Доклады, представленные на XIX Всероссийскую научную конференцию «Научный сервис в сети Интернет», 18-23 сентября 2020 года.

37. https://www.sorucom.org/ - 5-я международная конференция «Развитие вычислительной техники в России, странах бывшего СССР и СЭВ (SORUCOM 2020)», 6–8 октября 2020 г. в НИУ ВШЭ, Москва.
38. https://russianscdays.org/ - Суперкомпьютерные дни в России, 21–22 сентября 2020 г., Москва, http://2020.nscf.ru/ - Национальный Суперкомпьютерный Форум (НСКФ-2020). 24-27 ноября 2020, Переславль-Залесский, ИПС имени А.К. Айламазяна РАН
39. Городняя Л.В. Парадигмальный подход к факторизации определений языков и систем программирования // Системная информатика (System Informatics), No. 12 (2018), с. 1- 26. https://system-informatics.ru/files/issue_12_full.pdf (Рукопись поступила в редакцию 10.08.2018)
40. Городняя Л.В. Методы декомпозиции программ // Препринт 182. Новосибирск. 2018. 27 с.3.2
41. Городняя, Л. В. Парадигма программирования : учебное пособие для вузов / Л. В. Городняя. - 2-е изд., стер. - Санкт-Петербург : Лань, 2021. - 232 с. - ISBN 978-5-8114-6680-1 : УДК 004.43 ББК 32.973-18я73
42. М.М. Лаврентьев, Л.В. Городняя, М.А. Держо, Н.А. Иванчева, Д.В. Иртегов, Д.С. Мигинский, Б.Н. Пищик Вопросы мотивации обучения сложным профессиям // Вестник НГУ. Серия: Информационные технологии. 2021 Т.19, №1. С. 80–92. DOI: DOI 10.25205/1818-7900-2021-19-1-80-92
43. Л. В. Городняя Визуализация результатов анализа языков программирования для их поверхностного сравнения // Вестник НГУ. Серия: Информационные технологии. 2021 Т.19, №2. С. 29–52. DOI: 10.25205/1818-7900-2021-19-2-29-52
44. Бесшапошников Н.О., Дьяченко М.С., Леонов А.Г., Матюшин М.А., Орловский А.Е. Использование машинного обучения и нейронных сетей для автоматической верификации заданий в текстовом и графическом представлении и помощи преподавателю. // Успехи кибернетики. 2020. Т. 1. № 2. С. 39-45.
45. Димитриенко Ю.И., Губарева Е.А., Зубарев К.М., Алесин А.В., Иванова Т.Л. Автоматизация проверки математических заданий по курсу «Аналитическая геометрия» в системе NOMOTEX. // В сборнике: Цифровые технологии в инженерном образовании: новые тренды и опыт внедрения. Сборник трудов Международного форума. 2020. С. 206-208.
46. Кацман В.И., Козлов И.А., Новиков Ф.А. Игрофикация процесса решения типовых учебных задач на основе выбора правил преобразования. // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2020. № 9. С. 63-68.
47. Новиков Ф.А., Кацман В.И. Автоматическая проверка решений учебных задач на основе комбинации методов перебора логических правил и тестирования. // В сборнике: Цифровые технологии в инженерном образовании: новые тренды и опыт внедрения. Сборник трудов Международного форума. 2020. С. 266-273.
48. Тузов А.А. Автоматизированный практикум по решению вычислительных задач в среде «Кумир». // В сборнике: Проблемы и перспективы технологического образования в России и за

рубежом. Сборник материалов III Международной научно-практической конференции. Отв. редактор Л.В. Козуб. Ишим, 2021. С. 50-53.

49. Тузов А.А. Практикум с автоматической проверкой решения задач для исполнителя робот системы кумир (расширяем круг задач). // В сборнике: Современный учитель дисциплин естественнонаучного цикла. Сборник материалов Международной научно-практической конференции. Ответственный редактор Т.С. Мамонтова. 2019. С. 184-186.
50. Тихонова Т.И. Организация и проведение Летней школы юных программистов // «Сибирский учитель». Новосибирск, 2021. – № 1 (134). С. 83-88.
51. Виктор Майер-Шенбергер, Кеннет Кукьер. Большие данные. Революция, которая изменит то, как мы живём, работаем и мыслим = Big Data. A Revolution That Will Transform How We Live, Work, and Think / пер. с англ. Инны Гайдюк. — М.: Манн, Иванов, Фербер, 2014. — 240 с. — ISBN 987-5-91657-936-9.
52. Марчук А.Г Архитектура и основные особенности библиотеки PolarDB работы со структурированными данными // Системная информатика, № 13, 2018. Стр. 25-34
53. Min Chen, Shiwen Mao, Yin Zhang, Victor C.M. Leung. Big Data. Related Technologies, Challenges, and Future Prospects. — Spinger, 2014. — 100 p. — ISBN 978-3-319-06244-0. — doi:10.1007/978-3-319-06245-7.
54. Bankevich A., Nurk S. SPAdes: A New Genome Assembly Algorithm and Its Applications to SingleCell Sequencing // Journal of computational biology : a journal of computational molecular cell biology, 19. 2012. P. 455-477
55. Марчук А.Г., Трошков С.Н. Некоторые эксперименты по построению и анализу графа Де Брёйна // Системная информатика, No. 16 (2020), стр. 47- 56
56. Марчук А.Г. Последовательность как абстракция структурированного построения баз данных // Системная информатика, № 18 (2021), стр. 35-52, https://system-informatics.ru/ru/article/284
57. Марчук, А.Г., Трошков С.Н, Крайнева И.А. К вопросу о модернизации и интеграции электронных архивов длительного срока жизни // Сб. конференции Научный сервис в сети Интернет. 2021. № 23. С. 214-227.

УДК 004.942

Исследование и прогнозирование пассажиропотока с помощью системы МИКС-ПРОСТОР

Бульонков М.А. (Институт систем информатики СО РАН)

Малов В.Ю. (Институт экономики и организации промышленного производства СО РАН)

Нестеренко Т.В. (Институт систем информатики СО РАН)

В статье рассматривается задача оценки возможностей ликвидации транспортной дискриминации для населения Азиатской части России. Предлагается расширение системы моделирования транспортных потоков МИКС-ПРОСТОР в части включения в нее оценок вариантов мультимодальных пассажирских перевозок.

Ключевые слова:. *транспортная задача, моделирование, система автоматизации научных исследований.*

Введение

В последнее время резко усилилось внимание к проблеме сокращения населения в регионах азиатской части России, тем более в их северных районах. Более того, анонсируемый проект «Русский ковчег», в основе которого лежит идея создания новых городов в Сибири и на Дальнем Востоке, направлен на реализацию концепции опережающего роста восточных регионов страны [1, 6]. Суммарная численность «нового» населения этих городов оценивается более чем в один млн. человек. Современная тенденция показывает обратное: устойчивое, от года к году, сокращение населения восточных регионов страны. И это на том фоне, что население всей России не спешит расти темпами, характерными для прошлого века.

Постепенно приходит понимание, что даже простое сохранение численности населения в азиатской части России невозможно без существенного и опережающего, по сравнению с регионами Европейской части России, роста уровня жизни. Это тем более верно для создания новых рабочих мест, планируемых в новых городах. Даже если предположить, что под «новыми» городами понимается кардинальная модернизация ранее существующих городов, то и в этом случае требуется серьезное улучшение условий жизни. Вероятно, именно с этим и связаны высказывания о том, что возвращение соотечественников в Россию предполагается

именно в восточные регионы. Для реализации такой политики предполагается вводить разнообразные льготы, вплоть до подъемных, как это уже было в начале прошлого века во времена столыпинских реформ по заселению территории Сибири. В любом случае, для обоснованных предложений по организации нового этапа заселения Сибири и Дальнего Востока требуется оценка условий и возможностей всего комплекса жизнеобеспечения: организация здравоохранения, образования, создание современного ЖКХ, транспортного обеспечения. В данной статье акцент будет сделан на исследование именно последнего – условий обеспечения населения азиатской части России транспортными услугами в целях сокращения транспортной дискриминации, достаточно очевидной в настоящее время.

Понимая, насколько объемная задача (как по информационной составляющей, так и по вычислительной) «дойти» до каждого конкретного населенного пункта, тем более, до вариантов прогнозируемого будущего, невозможно. Мы ограничимся агрегированным представлением центров сосредоточения населения и маршрутов. Более того, упростим постановку задачи предположением, что само понятие «транспортной доступности» относится преимущественно к возможностям жителей Сибири и Дальнего Востока посещать центральные регионы страны: Москву, Санкт-Петербург, Крым, Сочи и т.п.

Допустимость такого упрощения требует дополнительного обоснования. Одно дело – доступность миллионов потенциальных пассажиров из крупных городов до Европейской части страны (о дальнейшем их передвижении в западном направлении пока речи нет). Другое дело – доступность до этих крупных городов со стороны средних и малых городов и еще более мелких поселений. Это, конечно, уже не миллионы людей и, возможно, для последних не так уж и важно дальнейшее путешествие в западном направлении, но требование обеспечения транспортной доступности касается всех граждан России, где бы они не жили. Поэтому настоящая задача касается только первой части. Принимается гипотеза, что в этих городах собрано всё население большого региона.

Подобные упрощения предопределяют и требования к использованию усредненных тарифов на отдельных маршрутах и участках транспорта. Здесь не предполагается учитывать все тонкости формирования тарифов в разных видах транспорта, которые в общем случае зависят и от времени года, и от дальности поездок (в нелинейной зависимости от расстояния), и даже от выбора верхней или нижней полки в железнодорожном вагоне пассажирского поезда. Все эти тонкости в исходной информации мы попытаемся нивелировать, используя

интервальный подход как в собственно «входной» информации, так и в интерпретации получаемых результатов[1].

На территории России выделяются следующие центры концентрации населения (конурбации):

- Москва представляет все регионы европейской части России;
- Владивосток;
- Хабаровск, который условно «объединяет» население и Комсомольска-на-Амуре и Благовещенска;
- Южно-Сахалинск, включая население Камчатки, Чукотки и Магадана;
- Якутск;
- Братск;
- Улан-Удэ;
- Иркутск;
- Красноярск;
- Абакан;
- Новосибирск;
- Новокузнецк, включая население и Кемерово, и Междуреченска;
- Барнаул.

Не охвачены такие населенные пункты, как Омск, Тюмень, Салехард и все города Урала. Такая агрегация объясняется тем, что нас в данной задаче интересует, в первую очередь, ответ на вопрос о том, насколько реализация задачи значительного повышения транспортной доступности повлияет на загруженность транссибирской магистрали пассажирскими поездами. Если, конечно, авиационный транспорт не сможет в рамках приемлемого тарифа обеспечить доступность до всех населенных пунктов Сибири и Дальнего Востока, и людям придется воспользоваться услугами железнодорожного и/или автомобильного транспорта для посещения культурных и/или оздоровительных центров европейской части России. Неявно предполагается, что уже осуществлены социологические исследования, которые определили, при каких условиях жители Сибири и Дальнего Востока готовы остаться в своих регионах на постоянное жительство. Также неявно предполагается, что количество мест работы с

[1] Данная задача имеет исключительно «пассажирскую» направленность, но она является органической частью всей системы МИКС-ПРОСТОР, предполагающей сопоставление и пассажирского, и грузового аспекта оценки возможностей транспортной системы Азиатской России [3, 5]. Предполагается в дальнейшем оценить провозные и пропускные способности отдельных участков разных видов транспорта, возможности организации мультимодальной транспортной системы для всех видов транспортной работы, как пассажирской, так и грузовой. Но это уже выходит за рамки настоящей статьи.

достойной зарплатой (по крайней мере, для покрытия транспортных тарифов) не только сохраняются, но и увеличиваются.

Предполагается, что весомая, может быть, даже основная, часть пассажиропотока направляется из выделенных пунктов в «агрегированную Москву». Местные потоки пассажиров (например, Барнаул – Новосибирск) будут учитываться после решения основной задачи, формулируемой следующим образом:

– определить наиболее рациональные направления и виды используемых транспортных средств, с учетом возможных пересадок, для обеспечения транспортной доступности населения восточных регионов России к центрам культуры и отдыха европейской части страны.

Понятие «рациональные направления» включает в себя такие критерии, как стоимость переезда и время в пути. Эти два критерия рассматриваются как самостоятельные, так и в комбинации. Используемые критерии «рациональности» являются обобщенными для всех пассажиров и всех видов транспорта, что, конечно, является некоторой условностью, но, как нам представляется, допустимой для решения задачи системного представления вариантов транспортных коммуникаций. Многое будет зависеть от интерпретации получаемого решения. Акцент в последнем будет сделан на качественный результат, определяющий интервал изменений тарифов, при выходе из которого может ожидаться смена маршрутов и/или видов транспорта.

Основная цель данного исследования – это оценить возможности сокращения транспортной дискриминации населения азиатской части России. Минимизация затрат на будущие перевозки – также одна из целей (или условий выбора будущих вариантов). Параллельно с этим оцениваются и варианты будущих транспортных сетей, которые включают согласованное развитие всех видов транспорта, создание современной логистики на всем пространстве азиатской части России. Полагаем, что в европейской части страны эта система складывается более рационально и «естественно», так как сделан уже серьезный задел.

Транспортная сеть работает не только для пассажиров, но и, возможно даже в большей степени, для грузоперевозок. Как разделить ограниченные мощности отдельных видов транспорта между грузами и пассажирами? Пример Транссиба говорит о многом: больше скоростных пассажирских поездов – меньше возможности провозки угля. И это на расстоянии 4-5 тыс. км., не говоря уже об оборонной значимости железнодорожных линий.

1. Моделирование

Что мы хотели бы предложить в результате решения нашей задачи, в которой параметрами являются:

- объемы перемещаемых пассажиров,
- ограниченные возможности отдельных участков (плеч) транспорта,
- тарифы на все эти операции.

Ограничениями, собственно, на «плечо», т.е. на участок транспортной линии, для авиаперевозок не имеет смысла, если не принимать во внимание ограничения на диспетчерское сопровождение. С другой стороны, для авиаперевозок существенны:

- ограниченные возможности аэропортов по приему, отправлению и/или транзиту пассажиров,
- ограничение на общий авиапарк.

Ограничения такого вида не рассматривались в наших предыдущих исследованиях [2, 4], но они концептуально не противоречат используемой модели, поскольку также выражаются как линейные ограничения над теми же переменными.

Под «тарифами» в данной задаче понимаются скорее издержки на перевозку, т.е. расходная ставка. Мы не интересуемся эффективностью отдельного участника перевозочного процесса, отдельной компании. Свою цель мы видим в оценке общих (системных, объединенных) издержек на весь перевозочный процесс, кто бы эти издержки не нес. Т.е. именно расходная ставка по всем элементам (погрузка, разгрузка, транзит, перевалка, перевозка) и должна быть главным стоимостным показателем эффективности. Инвестиции в расширение мощностей также будем пытаться перевести в расходные ставки. Стоимость билетов, варьируемых в очень больших пределах в настоящее время, может скорее сбивать с основного курса расчетов: уж больно каждая компания старается перехватить пассажиров, делая всякие «акции», предоставляя «бонусы» и т.п. Конечно, для компаний это крайне важно, но будет сильно осложнять и само решение, и интерпретацию результатов. Еще раз повторим: нам более важно оценить соотношения среднего уровня тарифов, провозных способностей и объемов, требуемых к перевозкам. В дальнейшем предполагается провести сопоставление требований, предъявляемых со стороны производств, к росту перевозимых грузов и повышения транспортной доступности населения азиатской части России. Другими словами, провозные (пропускные) способности отдельных видов транспорта и пунктов отправления (транзита, перевалок и пр.) потребуется каким-то образом «разделить» между грузами и пассажирами,

используя как критерии эффективности (минимум совокупных издержек), так и требования обеспечения транспортной доступности.

Несколько слов об использовании различных видов транспорта:

- трубопроводы – это исключительно для грузового сегмента (газ, нефть). Использование трубопроводов для пассажирских перевозок пока можно отнести к фантастическим проектам;
- морской – практически полностью для экспортных поставок грузов или (очень малая величина) – для круизов (туризм, рекреация);
- водный – возможно, в некоторых регионах, и является единственным, но таких регионов крайне мало и сравнительно мало пассажиров. Это не те суммы издержек, о которых стоит говорить;
- авиация – практически вся для пассажиров, не считая МЧС и военных;
- портовое хозяйство – практически всё нацелено на экспортно-импортные операции, т.е. на грузовой сегмент транспортной сети.

Пассажирские перевозки, по сравнению с грузовыми, имеют свою специфику. Если не учитывать миграцию населения, то в рассматриваемом нами продолжительном промежутке времени все пассажиры возвращаются в исходные пункты отправления. Таким образом, в качестве «оптимального» решения можно было бы взять такое, при котором никто никуда не ездил. Для того, чтобы избежать этого для каждой пары <пункт отправления, пункт прибытия> требуется определить свой «продукт». Это приводит к существенному увеличению количества видов перевозимых продуктов по сравнению с грузовыми перевозками и, как следствие, к увеличению размера решаемой задачи, который квадратично зависит от количества продуктов.

Эту проблему можно смягчить предположением от том, что количество пассажиров, перевозимых «туда» и «обратно», одинаковы. Например, размер пассажиропотока «Новосибирск-Москва» равен пассажиропотоку «Москва-Новосибирск». Таким образом, один из этих «продуктов» можно удвоить, а другой – исключить.

2. Исходные данные

Учитывая, что подавляющее большинство информации будет либо крайне усредненным, либо просто недоступным (секретным, коммерческой тайной, и/или просто не фиксируемой), то более важно оценить соотношение этих данных, при котором тот или иной вариант развития транспортной системы обеспечиваем минимизацию совокупных издержек. Непосредственно для какой-либо транспортной компании результаты наших расчетов могут

иметь только косвенную пользу – в качестве «информации к размышлению». Наши результаты прикладного характера скорее всего были бы полезны для РЖД, Минтранса и Минэкономразвития именно как государственных структур, озабоченные решением имеющих стратегических задач. В качестве инструментария предлагаемый подход может быть использован как имитационная основа для оценки разных вариантов формирования транспортной сети Азиатской России на период 2035-2050 гг. [1]. Интервальный формат входной информации существенно сокращает недостатки ее неточностей, поверхностного представления исходных данных.

Дополнительные требования к упрощению представления пассажиропотоков на разных видах транспорта.

1. Для авиапассажиров одним из основных ограничений является пропускная способность аэропортов в местах посадки, что эквивалентно числу самолетов, подлежащих к обслуживанию. Важен, конечно, и тип самолета, его вместимость, но, вероятно, из крупных аэропортов, представленных в задаче, они приблизительно одинаковые – от 150 до 200 пасс. Поэтому, зная число рейсов в день в аэропорты западных регионов страны (Москва, Санкт-Петербург, Сочи и т.п.), можно ориентировочно определить и объем «погрузки».

2. При пересадках с авиатранспорта на железнодорожный и автомобильный (или обратно) ограничения на провозную способность участков транспорта будет измеряться количеством пассажиров, которые могут быть перевезены по этим участкам. Для железнодорожного транспорта таким ориентиром может стать количество пар пассажирских поездов, проходящих по данному участку. Количество пассажиров в каждом поезде можно ориентировочно определить из качественного состава пассажирских вагонов (общий, плацкартный, СВ, купейный) и экспертно полученную загрузку каждого вагона в разные периоды года. Ограничения по автодорогам также можно экспертно определить по данным о характере движения пассажирского и личного автотранспорта в сопоставлении с потоком грузовых автомобилей.

3. При использовании железнодорожного транспорта требуется несколько иная оценка нагрузок и провозной (пропускной) способности. За эталон возьмем участок железной дороги Новосибирск-Омск — двухпутная электрифицированная линия, по которой есть данные и по объемам провозимых грузов, и по пропускной способности (пар поездов в сутки). Пассажирский состав имеет меньшее количество вагонов, чем, например, угольный или контейнерный, но он также является составной частью той «пары», которая должна быть пропущена, да еще в большем по времени по требованиям безопасности интервале движения. Другими словами, пара пассажирских поездов это, как минимум та же пара угольных составов,

хотя и перевозят существенно меньше тонн «полезного» груза. В задаче же требуется оценить перспективы возможных пересадок с одного вида транспорта на другой. Здесь мы вынуждены принять предположение о том, что одного пассажира условно приравниваем к 4 тоннам перевозимого груза. Это делается по аналогии с пассажирским плацкартным вагоном, перевозящим около 50 пассажиров. А состав из 20 вагонов (в том числе, СВ и купейных) перевозит 750 пассажиров. Тогда один пассажирский состав можно также условно считать за полноценный состав с грузовыми вагонами, весом 3 тыс. тонн (из расчета 50 вагонов по 60 тонн в каждом).

Во всех вышеописанных случаях будет использоваться крайне агрегированная и экспертно представляемая информация, что потребует задействование методов интервального подхода как к самим способам решений, так и к представлению и интерпретации получаемых результатов.

Например, пассажиропоток Новосибирск-ЗАПАД можно оценить следующим образом. Из аэропорта Новосибирска в 2020 г. ежедневно уходило около 40 рейсов, каждый из которых вмещал по 150-200 пассажиров. Предполагая, что заполняемость составляет 100%, получаем ежедневно 6000-8000 пассажиров, а считая, что в году 350 дней (15 дней аэропорт закрыт) – 2,1-2,8 млн. пассажиров. Это и будем принимать за объём «погрузки / разгрузки» в Новосибирске.

Описанный выше метод оценки объёма перевозок не единственный. Имеет место быть гравитационная модель, при которой объем перевозок между двумя узлами пропорционален произведению их совокупного дохода и обратно пропорционален расстоянию между узлами. Кроме того, в качестве минимизируемого показателя может быть использовано суммарное время, затрачиваемое на перевозку. При этом, конечно, надо учитывать не только время, скажем, на перелёт, но и время на дорогу в аэропорт, регистрацию, посадку, а также и время между стыковочными рейсами, например, время переезда от вокзала до аэропорта. Здесь нужно использовать комбинированную стоимость, включающую как собственно расходы на перевозку, так и стоимость потраченного времени, которое, естественно, зависит от доходов населения в конкретном узле.

3. Варьирование параметров

Прогнозирование развития транспортной системы осуществляется путём варьирования выбранных экспертом входных параметров, к которым относятся:

- пропускная способность для данного вида транспорта и выбранному плеча или узла;

- стоимость перевозки (тарифа) данного продукта и вида транспорта по выбранному плечу или его обработки в данном узле.

Оказалось полезным расширение системы реализацией вариаторов новых типов, которые не увеличивают принципиально возможности системы, но существенно упрощают её использование. Введены следующие коэффициенты:

- коэффициент для вида транспорта, на который умножаются все базовые тарифы, связанные с этим транспортом;
- коэффициент для вида транспорта, на который умножаются все пропускные способности, связанные с этим транспортом.
- коэффициент, на который увеличивается пассажиропоток по всем направлениям, входящим или исходящим из данного узла.

Для каждого варьируемого параметра задаётся интервал допустимых значений. Обычно такой интервал содержит значение этого параметра на текущий момент, а границы интервала отражают экспертные оценки на период прогнозирования. Получившееся многомерный прямоугольник [2] покрывается множеством комбинаций значений параметров. МИКС-ПРОСТОР допускает два способа такого покрытия:

- *пошаговое*, при котором каждый параметр изменяется в своём интервале с задаваемым шагом, и
- *стохастическое*, при котором указывается количество точек, случайно выбираемых внутри многомерного прямоугольника.

Стохастический способ может оказаться более эффективным с практической точки зрения.

Для каждой комбинации параметров ищется решение задачи линейного программирования, минимизирующее суммарную стоимость перевозок. В настоящее время МИКС-ПРОСТОР использует для этой цели решатель Google OR [8].

Множество полученных решений может оказаться очень большим, что делает практически невозможным его анализ экспертом. Поэтому мы используем кластеризацию множества решений, сравнивая их по степени схожести транспортных потоков по отдельным плечам. Затем из каждого кластера выделяется наиболее типичный представитель, который и предоставляется эксперту для анализа.

4. Эксперимент и результаты

Ниже рассматривается один из проведённых экспериментов, в котором исследуются влияние на пассажиропоток параметров железнодорожного транспорта. Сначала мы

(достаточно грубо) задаём коэффициенты изменения пропускной способности и тарифов для всей железнодорожной сети, как показано на рис. 1.

Транспортная сеть

Вариатор	Диапазон	#
Железнодорожный - увеличение проп. спос.	1-10	10
Железнодорожный - удорожание	1-10	10

Рис. 1. Задание интервалов значений параметров

Таким образом, всего получается 100 вариантов, в которых как пропускная способность, так и тариф по ж/д увеличиваются вплоть до 10 раз. Далее все эти варианты просчитываются и из множества решений выделяются наиболее типичные способы перевозки

Всё множество решений разделилось на три типичных способа перевозки, обозначаемые красным, жёлтым и зелёными цветами, которые распределяются как показано на рис. 2.

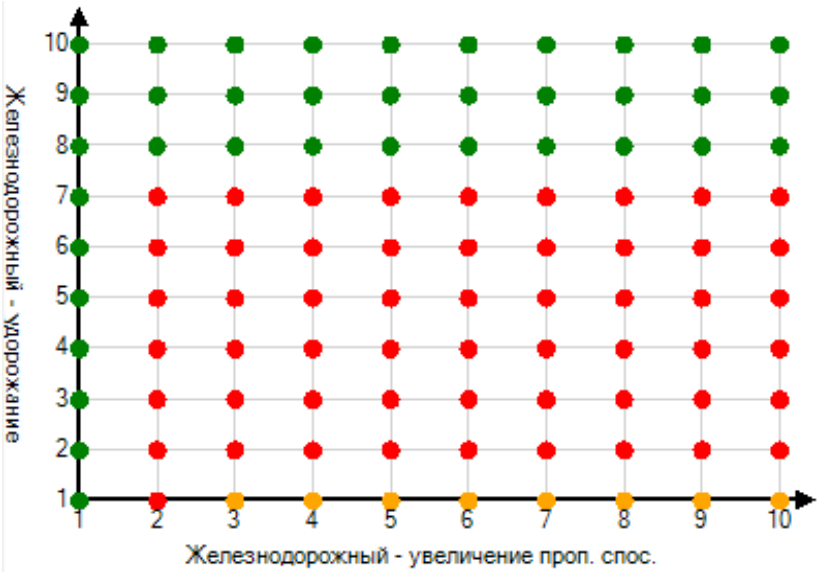


Рис. 2. Разбиение на кластеры при грубой оценке

Эту информацию можно использовать как оценку для более точного определения интервала варьирования, например, ограничив максимальные коэффициенты до 3 и, сохраняя общее количество решений, сделать меньше шаг изменения параметров. В результате разбиение на кластеры получается более детальным (рис. 3).

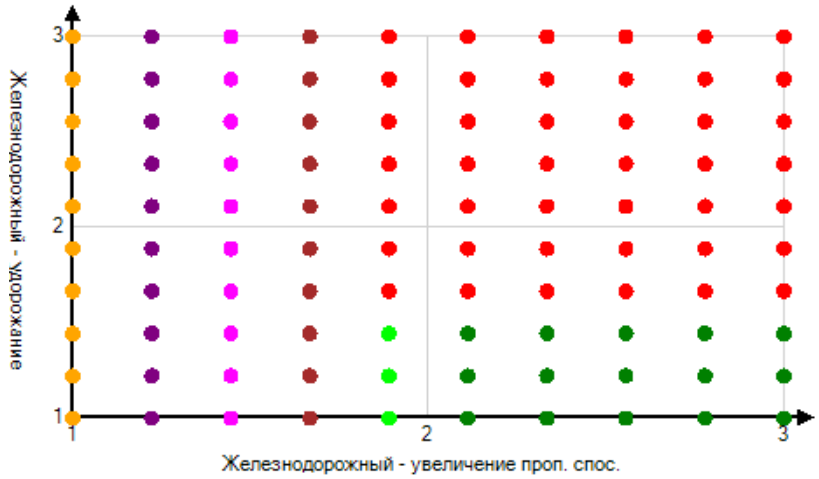


Рис. 3. Разбиение на кластеры при точной оценке

Далее, эксперт может перейти к анализу отдельных решений. Например, при базовом состоянии (оба коэффициента увеличения равны 1), который входит в желтый кластер, транспортная сеть выглядит, как показано на рис. 4.

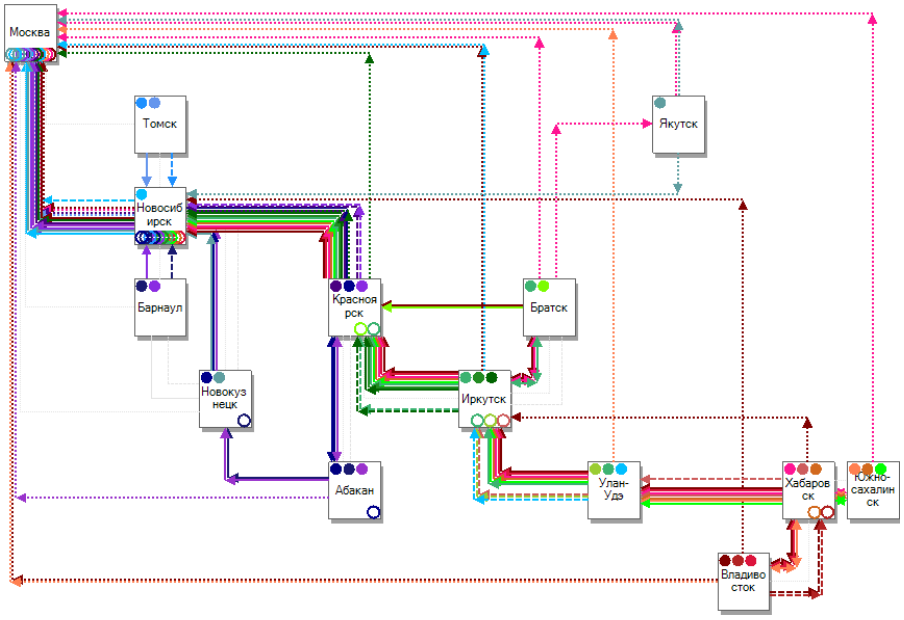


Рис. 4. Базовый вариант

Здесь сплошными линиями отображается железнодорожный транспорт, штриховыми — автомобильный, и пунктирными — авиационный.

Если пропускная способность возрастает в 1.4 раза (фиолетовый кластер), то использование авиационного транспорта существенно сокращается, как показано на рис 5.

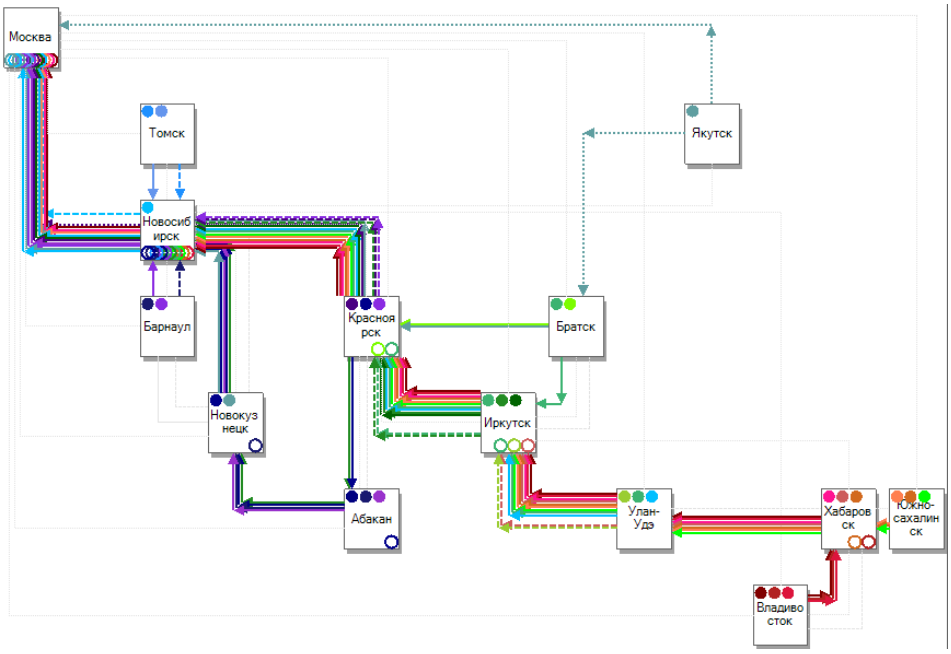


Рис. 5. Увеличение пропускной способности ЖД в 1.4 раза

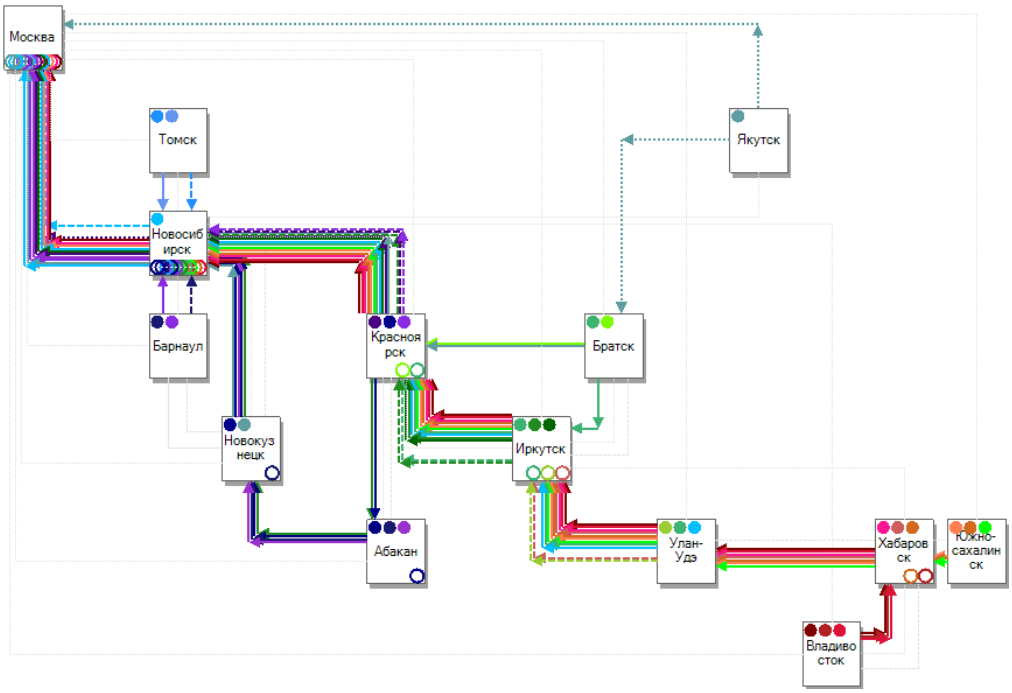


Рис. 6. Увеличение пропускной способности ЖД в 1.6 раз

Использование авиаперевозок исчезает практически полностью при увеличении пропускной способности железной дороги в 1.6 раз. Исключение составляет Якутск, до которого практически нет другого сообщения, кроме авиационного (рис. 6).

5. Транспортная доступность

Система МИКС-ПРОСТОР получила развитие в части визуализации транспортной доступности. В качестве определяющего показателя была выбрана средняя стоимость перевозки пассажиров (ССПП) для пары <пункт отправления, пункт назначения>. Несложно показать, что она может быть вычислена для каждого полученного решения как средневзвешенная сумма перевозки пассажиров этого типа по всем транспортным плечам.

Для графического отображения транспортной доступности и конурбаций были рассмотрены несколько вариантов и проведены соответствующие эксперименты. Первая попытка состояла в использовании анаморфических карт, основная идея которых состоит в смещении узлов так, чтобы расстояние между ними соответствовало ССПП, по возможности сохраняя взаимное расположение. Поскольку ССПП не является метрикой, и для него не выполняется неравенство треугольника, то такое отображение может быть только приблизительным. Для решения этой задачи мы использовали силовые методы размещения графов, но результаты оказались неудовлетворительными для поставленных выше целей.

Вторая попытка состояла в том, чтобы показывать транспортную доступность с точки зрения жителя конкретного узла. Метод отображения состоит в следующем:

- рассматриваемый узел помещается в центр;
- все остальные узлы размещаются относительно него в направлении, соответствующем реальному географическому;
- расстояние до смежных узлов пропорционально ССПП, а до остальных – длине кратчайшего относительно ССПП пути;
- объём перевозки отображается размером вершины – площадь соответствующего кружочка пропорциональна объёму погрузки/разгрузки.

Это хорошо работает в случае, когда величины ССПП не сильно разбросаны. К сожалению, в нашем случае это не так. В некоторых экспериментах из Новосибирска в Москву можно доехать за 700 руб., а из Москвы в Якутск за 60 000 руб. В результате при отображении Новосибирск и Москва находятся вплотную друг к другу, а Якутск — где-то у противоположной стороны диаграммы.

Для решения этой проблемы была использована технология «рыбьего глаза» (fish-eye view), при которой расстояния на изображении увеличиваются в центре и сжимаются по краям. Существуют множество разных способов сделать это. Мы остановились на методе, описанном в [9], где расстояние d трансформируется по следующей формуле:

$$T(d) = d_{max} * \frac{dist + 1}{dist + \frac{d_{max}}{d}}$$

где

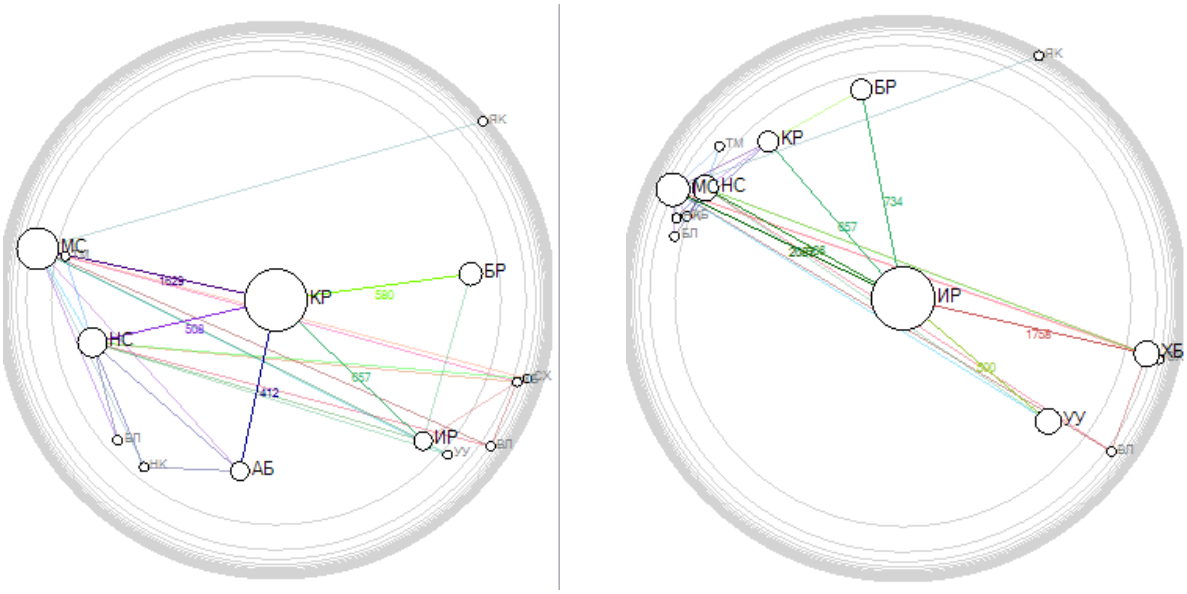
d_{max} – максимальное расстояние между узлами;

dist – коэффициент искажения.

При *dist* = 0 отображение тождественно, то есть картинка оказывается неискажённой. Мы позволяем варьировать параметр *dist* интерактивно.

На рис. 7 серые концентрические окружности отображают ССПП, и в реальности идут с равным шагом. То есть, то, что находится в серой зоне на краю окружности для жителя, расположенного в центре узла, означает «очень далеко и дорого».

Такое отображение может быть полезно для экспертной оценки конурбаций, с точки зрения транспортной доступности. Например, на рис. 7(б) явно выделяется конурбация Иркутск – Братск – Улан-Удэ. С другой стороны, Красноярск принадлежит как Иркутской, так и Новосибирской конурбациям.



(а) (б)

Рис. 7. Отображение транспортной доступности: (а) Красноярск, (б) Иркутск

Система МИКС-ПРОСТОР позволяет прогнозировать изменение состава конурбаций при варьировании параметров транспортной системы. Например, в описываемом ниже эксперименте одновременно варьировались стоимость и пропускная способность ЖД. На рис. 8 показаны результаты эксперимента, где в качестве центрального узла выбран Иркутск.

Анализ результатов эксперимента позволяет сделать следующие выводы:

1. Верхняя левая картинка (малая пропускная способность – большая цена) фактически изолирует данную конурбацию от остальной части страны.
2. Верхняя правая (большая пропускная способность – большая цена) сближает её с Дальним Востоком.
3. Нижняя левая (малая пропускная способность – малая цена) пододвигает конурбацию к европейской части.
4. Наконец, нижняя правая, когда совсем всё хорошо, сближает всю страну, за исключением Якутска.

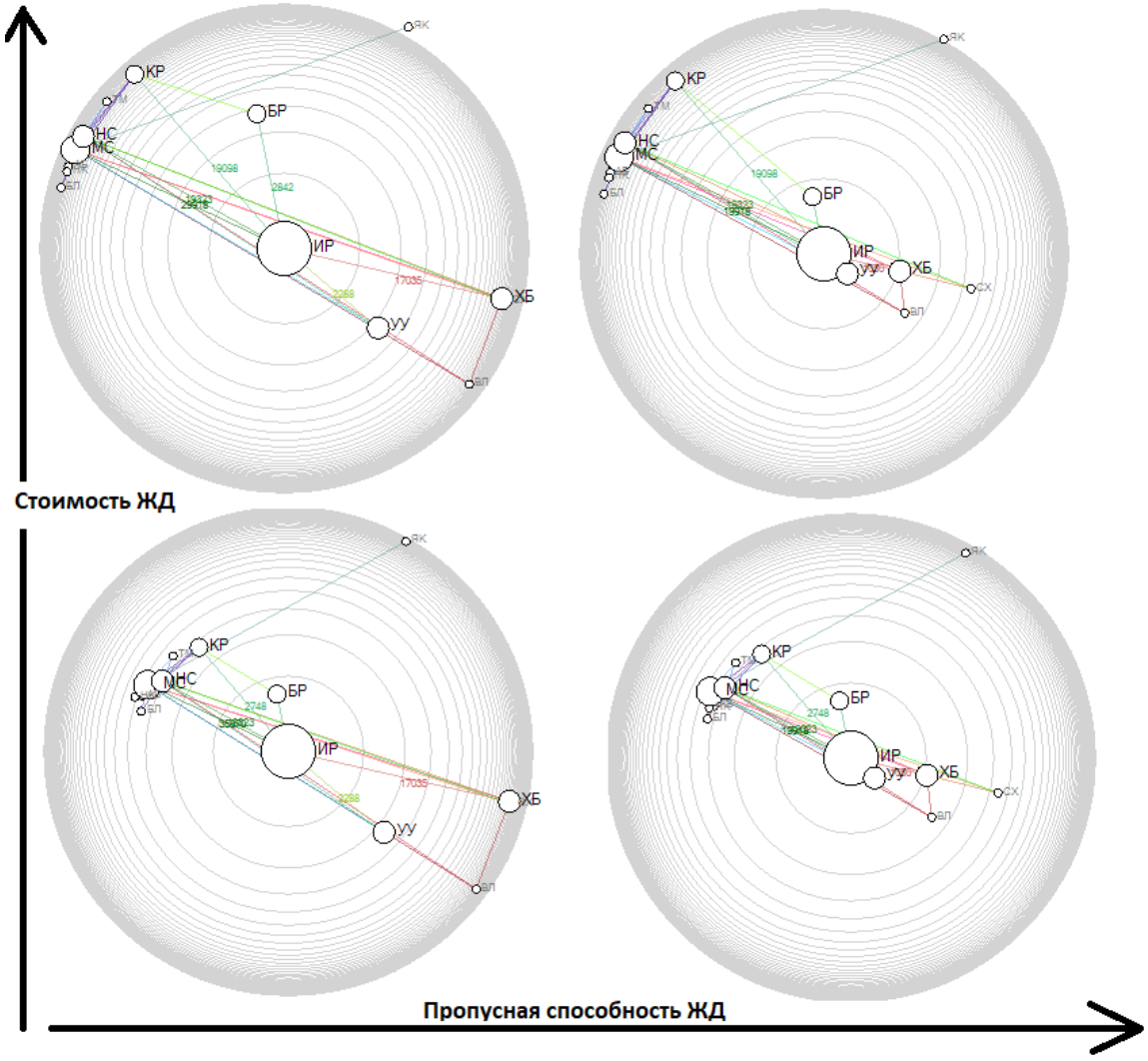


Рис. 8. Прогнозирование изменения конурбаций

Заключение

В данной работе рассмотрены некоторые вопросы прогнозирования развития опорной транспортной сети в Азиатской части России в части пассажирских перевозок. Используемая

система МИКС-ПРОСТОР показала себя как мощный инструмент [2, 4, 7], позволяющий выделить и сконцентрировать эксперта на наиболее значимых аспектах. Дальнейшие исследования могут быть связаны как с интегрированным исследованием пассажирских и грузовых перевозок, так и с расширением транспортной сети и увеличением номенклатуры перевозимых продуктов.

Список литературы

1. Азиатская часть России: моделирование экономического развития в контексте опыта истории : сб. науч. тр. / отв. ред. В.А. Ламин, В.Ю. Малов; РАН, Сиб. Отд-е, ИЭОПП, Ин-т истории, Ин-т геогр. им. В.Б. Сочавы, Ин-т систем энергетики им. Л.А. Мелентьева, Ин-т динамики систем и теории упр-я. – Новосибирск: Изд-во СО РАН, 2012. – 463 с. (Интеграционные проекты; Вып. 34).
2. Бульонков М.А., Нестеренко Т.В. Автоматизация исследований развития опорной транспортной сети // Вестник СибГУТИ №3(47) 2019. - С.45-54.
3. Бульонков М.А., Карпан В.В., Малов В.Ю., Марусин В.В., Радченко В.В. Концептуальные вопросы построения Модельно-Информационно-Картографической Системы (МИКС) // Моделирование производственных и региональных систем на основе ГИС и информационных технологий: сб. науч. тр. / под ред. Ю.Ш. Блама, В.В. Радченко. - Новосибирск: ИЭОПП СО РАН, 2011. - С. 5-28.
4. Бульонков М.А., Филаткина Н.Н. Ситуационный анализ в системе транспортного прогнозирования МИКС-ПРОСТОР // Информационные технологии – 2013 - №8. - С. 43 – 52.
5. Воробьёва В.В., Малов В.Ю., Радченко В.В., Поттер М.В., Серебрянников И.Е. Модель прогнозирования развития опорной транспортной сети // Моделирование производственных и региональных систем на основе ГИС и информационных технологий: сб. науч. тр. / под ред. Ю.Ш. Блама, В.В. Радченко. - Новосибирск: ИЭОПП СО РАН, 2011. - С. 68-96.
6. Проблемные регионы ресурсного типа: Азиатская Россия / / отв. ред. В.А. Ламин, В.Ю. Малов – Новосибирск: Изд-во СО РАН, 2005.
7. Ситуационная комната как элемент организации экспертного сообщества: задачи планирования и прогнозирования / под ред. Г.А. Унтура; – Новосибирск: ИЭОПП СО РАН, 2018.
8. Google OR-Tools. [Электронный ресурс]. URL: https://developers.google.com/optimization/ (дата обращения 03.03.2022).
9. Sarkar, Manojit, Brown, Marc (1992): Graphical Fisheye Views of Graphs. In: Bauersfeld, Penny, Bennett, John, Lynch, Gene (eds.). Proceedings of the ACM CHI 92 Human Factors in Computing Systems Conference June 3-7, 1992, Monterey, California. pp. 83-91. [Электронный ресурс]. URL: https://www.acm.org/pubs/articles/proceedings/chi/142750/p83-sarkar/p83-sarkar.pdf

УДК 004.912 + 004.8

Извлечение информации из научных текстов на русском языке

Батура Т.В. (Институт систем информатики им. А.П. Ершова СО РАН),

Бручес Е.П. (Институт систем информатики им. А.П. Ершова СО РАН, Новосибирский государственный университет),

Мезенцева А.А. (Институт систем информатики им. А.П. Ершова СО РАН, Новосибирский государственный университет)

В статье описаны методы автоматического извлечения терминов и связывания их с Викиданными. Преимуществом предложенных методов является потенциальная возможность их применения к любым областям знаний при наличии только неразмеченных текстов и начальных словарей терминов небольшого размера. Для проведения экспериментов был собран и размечен корпус научных текстов RuSERRC. Корпус и модели находятся в открытом доступе и могут быть полезны для дальнейших исследований другими научными коллективами.

Ключевые слова: *извлечение информации, машинное обучение, компьютерная лингвистика,* обработка текстов, *извлечение терминов, связывание сущностей.*

1. Введение

С распространением Интернета количество информации растет чрезвычайно быстро. По данным журнала “Nature” [15] во всем мире ежегодное количество научных публикаций с 2008 по 2018 г. выросло с 1.8 миллиона до 2.6 миллионов статей только по биомедицинской тематике. Однако эффективная обработка и извлечение наиболее важной информации из текстов является трудоемкой задачей. Тексты разных жанров отличаются по структуре и содержанию. Например, научные отчеты и публикации содержат ценные сведения о передовых достижениях в разных областях знаний, а правительственные документы (распоряжения, отчеты, постановления) описывают проделанную работу и запланированные мероприятия по развитию регионов. Очевидно, что для более эффективного анализа большого потока информации, необходимо создавать новые автоматические методы и инструменты.

Одной из фундаментальных задач извлечения информации из текстов является распознавание именованных сущностей (Named Entity Recognition, NER), которая в

настоящее время не является полностью решенной. Под сущностями понимаются слова или группы слов, отражающие основной смысл текста. Для того, чтобы решить обозначенную задачу, необходимо найти и классифицировать упоминания именованных сущностей в тексте по заранее определенным категориям, таким как имена людей, организации, местоположения, медицинские коды, научные термины, выражения времени, денежные значения и т.д.

Не менее важной является задача связывания сущностей (Entity Linking, EL), которая состоит в том, чтобы алгоритм мог автоматически соотнести упоминание сущности в тексте с сущностью в структурированной базе знаний, такой как Wikidata, DBPedia и др. Информация из базы знаний повышает качество автоматической системы, помогая разрешать лексическую неоднозначность слов и понятий, точнее определять их значение в текстах. Особую сложность представляет работа с информацией из узких предметных областей, когда подходящей терминологией владеют только специалисты. Поэтому для качественного автоматического извлечения информации важно, чтобы в системе присутствовал компонент связывания элементов текста с базой знаний.

Существующие на сегодняшний день автоматические методы, как правило, относительно неплохо решают обозначенные задачи для английского языка, но качество обработки текстов на русском языке оставляет желать лучшего. Для построения современных языковых моделей, которые используют алгоритмы машинного обучения, требуется большое количество обучающих данных. Разметка таких данных выполняется вручную и зависит от конкретной предметной области, поэтому существует проблема доступности специально подготовленных обучающих данных для большинства языков, в том числе для русского эта проблема стоит довольно остро.

В данной статье исследуются методы автоматического извлечения сущностей и связывания их с базой знаний. Для экспериментов собран и размечен корпус научных текстов RuSERRC, который также описан в данной статье. Исходный код и данные опубликованы в открытом доступе[1].

2. Обзор существующих методов

В настоящее время существует некоторое количество готовых решений, работающих с английским языком: OpenTapioca [8], OpenNRE [12], spaCy[2], Stazna[3]; есть открытые

[1] https://github.com/iis-research-team/terminator

[2] https://spacy.io

размеченные корпуса большого размера: TACRED [31], DocRED [27], SciERC [17], NNE [21], DWIE [29] и др. Для русского языка данных и исследований по извлечению информации из текстов значительно меньше: для извлечения сущностей из новостных текстов могут использоваться библиотеки DeepPavlov[4] и Natasha[5]; открытые коллекции данных содержат разметку только сущностей и отношений: FactRuEval [24], NEREL [16], RURED [11].

В данной работе под сущностью понимается слово или словосочетание, являющееся названием некоторого понятия из области науки, техники, искусства и др. В научных текстах (научных статьях, отчетах, диссертациях, монографиях) в роли сущностей выступают термины. Общая идея, которая лежит в основе традиционных подходов, состоит в том, что автоматическое извлечение терминов происходит в два этапа: на первом этапе из текстов извлекаются *n*-граммы слов, которые потенциально могут быть терминами, а на втором этапе выполняется классификация, в результате которой принимается решение, является ли данная фраза термином. Алгоритмы, архитектура которых соответствует этой идее, можно условно разделить на несколько групп.

Первая группа предполагает использование правил для выделения из текстов фраз, которые являются терминами. Например, в работе [23] предлагается использование словарей и информации о синтаксической структуре предложения для извлечения многословных терминов. Однако составление терминологических словарей вручную требует привлечения специалистов и затратно по времени.

Вторую группу представляют методы, в основе которых лежат алгоритмы машинного обучения с вручную извлеченными признаками. Например, в статье [6] авторы используют несколько групп признаков для извлечения терминов: лингвистические (части речи, главное слово фразы, количество имен существительных во фразе и др.), статистические (длина фразы, TF, IDF, TF-IDF и др.) и гибридные признаки (например, частота встречаемости фразы в корпусах обычных и научных текстов). Также было исследовано применение алгоритма PageRank для более точной классификации [32]. В работе [1] предлагается использовать признаки, основанные на информации из Викиданных. Главным недостатком таких методов является необходимость извлечения признаков вручную.

В третью группу входят методы глубокого обучения. В работе [26] исследуется проблема отсутствия достаточного количества данных. Для этого авторы на ограниченном количестве данных обучают две модели (CNN и LSTM), которые на вход принимают векторные

[3] https://stanfordnlp.github.io/stanza
[4] https://github.com/deepmipt/DeepPavlov/blob/master/docs/features/models/ner.rst
[5] https://github.com/natasha/natasha

представления слов фразы, а на выходе определяют, является данная фраза термином или нет. Затем этими моделями размечается новая порция данных, которая добавляется в обучающую выборку, и процесс обучения повторяется еще раз.

Четвертая группа опирается на методы тематического моделирования. В статье [2] описывается попытка применения различных методов тематического моделирования для улучшения нахождения однословных терминов: невероятностные (разные методы кластеризации – K-means, NFM и др.) и вероятностные (в качестве метода такой группы был выбран алгоритм LDA).

К пятой группе можно отнести методы, рассматривающие задачу извлечения терминов как задачу сопоставления последовательностей входных токенов с последовательностями меток из заранее определенного множества (sequence labelling task), т. е. для каждого токена в тексте требуется определить его класс (является он термином или нет). Таким образом, решение задачи осуществляется в один этап. Так, в работе [14] исследуются различные архитектуры и векторные представления слов при решении задачи sequence labelling. Большим преимуществом данного подхода является то, что во внимание принимается контекст (как синтаксический, так и семантический) употребления конкретной фразы, что составляет один из ключевых признаков для нахождения терминов в тексте.

Далее автоматическое связывание сущностей с элементами базы знаний выполняется в два этапа: генерация кандидатов и их ранжирование.

Для генерации множества кандидатов применяются различные подходы: сопоставление словоформ с заранее построенным индексом, методы нормализации строки и меры схожести триграмм [33]; страницы разрешения неоднозначности и редиректов Wikipedia, которые в том или ином виде содержат омонимичные и синонимичные слова и фразы [10]; априорную вероятность совместной встречаемости сущности и упоминания в различных источниках [5].

При ранжировании кандидатов происходит оценка того, насколько хорошо объект-кандидат соответствует контексту. Здесь можно выделить три основных подхода. Первый подход основан на вычислении схожести контекстов, которые представляются в виде векторных представлений на основании как вручную сформированных признаков [4], так и полученных из языковых моделей [28]. При другом подходе задача ранжирования трансформируется в задачу бинарной классификации, в которой целью является определить, относится ли данное упоминание к сущности. В качестве классификатора могут использоваться наивный байесовский классификатор [25], SVM классификатор [30], глубокие нейронные сети [13]. В последнее время широкое распространение получили подходы, которые использую векторные представления, полученные из графов знаний. Такая

информация помогает понять, какое положение сущность занимает в графе, какими отношениями она связана с другими сущностями и др. Например, в статье [19] авторы строят векторные представления ребер графа, полученного из DBpedia, с помощью алгоритма DeepWalk [20]. В работе [18] авторы используют алгоритм TransE [3] для векторизации сущностей в графе.

Как правило, большинство упомянутых алгоритмов требуют для обучения большого количества специально подготовленных данных. Отличительной особенностью предлагаемых нами методов является возможность использования их, когда вручную размеченных данных имеется совсем немного.

3. Подготовка данных

Коллекции научных текстов существуют для английского языка и активно используются научным сообществом для обучения и оценки качества алгоритмов извлечения информации, однако в настоящее время на русском языке такие корпуса в открытом доступе не представлены. Поэтому было решено подготовить подобный корпус самостоятельно.

В собранную коллекцию RuSERRC[6] вошли тексты аннотаций научных статей по теме информационные технологии на основе данных, находящихся в открытом доступе, из журналов “Вестник НГУ. Серия: Информационные технологии”[7], “Программные продукты и системы”[8]. Объем корпуса 1600 неразмеченных документов и 80 текстов, вручную размеченных сущностями.

Разметка сущностей выполнялась в формате BIO (каждой единице текста присваивается значение тега B-TERM, если она является начальной для сущности, I-TERM, если она находится внутри термина или O, если она находится вне любой сущности). В рамках такой разметки предполагается, что именованные объекты не являются рекурсивными и не перекрываются. Всего в 80 размеченных текстах содержатся 11 157 токена и 2 027 терминов. Средняя длина термина – 2.43 слова. В качестве терминов рассматривались существительные и именные группы. Самый длинный термин состоит из 11 токенов.

Каждый документ был размечен двумя аннотаторами независимо, разногласия были разрешены модератором. Для аннотаторов была написана подробная инструкция с примерами. Процент согласия аннотаторов в задаче выделения сущностей составил 51.77%. Значение было вычислено как отношение пересечения выделенных терминов к объединению

[6] https://github.com/iis-research-team/ruserrc-dataset

[7] https://journals.nsu.ru/jit/archive/

[8] http://www.swsys.ru/

выделенных терминов. Полученное значение показывает высокую степень субъективности при нахождении слов и фраз, являющихся терминами, и при определении точных границ сущностей, что свидетельствует о сложности решаемой задачи.

Для поиска терминов в Викиданных были допущены следующие видоизменения сущностей.

– Все извлечённые сущности ищутся в базе знаний в нормализованной форме с учётом согласования и без учёта регистра, например: “*Линейных уравнений*” -> “*линейное уравнение*”.

– Если две и более сущности представлены как набор однородных членов с одним общим элементом, то каждый однородный член с общим элементом рассматривается как сущность, например: “*спутниковая и мобильная связь*” -> “*спутниковая связь*”, “*мобильная связь*”.

– Разного рода кореференции также связываются с одной сущностью, например: если в начале текста упоминается “*метод k-means*”, а затем в тексте “*предложенный [метод]*”, то эти две сущности следует связать одним идентификатором.

– Также мы считаем синонимами термины “*подход*” и “*метод*”.

– Если из текста была извлечена сущность, подходящая по шаблону “общее понятие + название” (например, “*язык программирования Python*”, “*операционная система Windows*”), при этом в базе знаний находится только сущность с названием (например, “*Python*” (Q28865)), то такие две сущности связываются.

– Если в тексте сущность написана с опечаткой, то в графе знаний мы ищем сущность без опечатки, например: “*3Dреконструкцию*” -> “*3d реконструкция*”.

– Допускаются трансформации вида “*архитектура системы*” -> “*системная архитектура*”.

– Расшифрованные аббревиатуры, например “*wps*” -> “*Wi-Fi Protected Setup*”.

– Допускается поиск синонима сущности в базе знаний (проверяется запросом в поисковую систему или Википедию), например: “*статистическая зависимость*” -> “*корреляция*”, “*генетическая последовательность*” -> “*нуклеотидная последовательность*”, также допускается поиск перевода сущности, например, на английском языке.

Каждая сущность была размечена двумя асессорами. Мера согласованности была рассчитана как отношение количества сущностей без конфликта в разметке к общему количеству сущностей в корпусе и составила 82,33 %. Всего в корпусе выделено 3386

терминов, 1337 из которых удалось связать с сущностями в Викиданных. Средняя длина связанной сущности – 1,55 токен, минимальная длина – 1 токен, максимальная – 8 токенов.

4. Описание предлагаемых методов

4.1. Извлечение терминов

Для извлечения терминов были реализованы: словарный метод, статистический метод и методы на основе архитектуры BERT.

В качестве базового алгоритма был реализован метод на основе словаря. Его идея состоит в том, чтобы собрать конечный словарь фраз, которые являются терминами, а затем искать их во входном тексте. Как правило, метод такого типа обладает высокой точностью, но низкой полнотой, т.к. учесть разнообразие всех форм терминов, а также появление новых, невозможно. В рамках работы был собран словарь из 17 252 терминов длиной от 1 до 12 токенов. При реализации словарного подхода были использованы библиотеки NLTK[9] для токенизации, pymorphy2[10] для лемматизации и ahocorapy[11] для построения префиксного дерева и работы с ним.

Для сравнения был рассмотрен статистический метод RAKE (Rapid automatic keyword extraction) [22], который кратко может быть описан следующим образом. Сначала применяется список стоп-слов и разделителей для выделения многословных терминов. После чего используется статистическая информация: для каждого слова из ключевых фраз-кандидатов оценивается частота, с которой оно встречалось, и количество связей между этим словом и остальными. На основании этих двух величин вычисляется вес ключевой фразы, и все фразы сортируются по весам, наиболее вероятные ключевые фразы получают максимальный вес. Этот метод хорошо применим к динамическим корпусам документов и к абсолютно новым областям знаний, при этом не зависит от языка и его особенностей. Было замечено, что данный алгоритм среди результатов часто выдает словосочетания, содержащие глагольные формы. Так как в качестве терминов мы рассматриваем существительные или именные группы, было решено оптимизировать эксперимент и выполнить предобработку текстов, убрав глаголы и их формы перед применением RAKE.

Кроме этого, была проведена серия экспериментов с использованием методов машинного обучения. Сложность проведения экспериментов с использованием различных алгоритмов

[9] https://pypi.org/project/nltk/
[10] https://pypi.org/project/pymorphy2/
[11] https://pypi.org/project/ahocorapy/

машинного обучения заключается в отсутствии размеченных данных. Эта проблема была решена следующим образом. Были взяты 1 118 полных текстов научных статей (включая, аннотацию и основную часть), которые предварительно были очищены от формул, таблиц, схем и пр., и автоматически размечены терминами из словаря, описанного выше. Таким образом, у нас получился размеченный набор данных, общим объёмом 1 992 498 токенов и содержащий 177 050 терминов.

Была поставлена гипотеза, что обобщающая способность модели позволит находить термины в текстах, где, предположительно, концентрация терминов выше, в то время как, модель была обучена на полных текстах статей, в которых концентрация терминов ниже. Также, таким способом, будут находиться термины в текстах, которые отсутствовали в исходном словаре.

Для проверки этой гипотезы были проведены эксперименты с посимвольной нейронной сетью, а также предложен итеративный метод на основе слабо контролируемого обучения к извлечению терминов. Для получения векторных представлений слов была использована предобученная модель BERT bert-base-multilingual-cased [9]. На вход модели подаётся токенизированный текст (входные тексты никак не преодобрабатываются). Выход модели представляет собой последовательность предсказанных классов для соответствующих токенов. Были проведены эксперименты с двумя архитектурами моделей: BERT-LSTM: полученные векторные представления подавали на вход двунаправленной LSTM, за которой идут два полносвязных слоя, и BertForTokenClassification: после векторных представлений идёт один полносвязный слой. Идея предложенного подхода заключается в том, чтобы обучить модель на небольшом количестве размеченных данных, а затем разметить полученной моделью некоторое количество новых текстов, добавить их к обучающему множеству и обучить вторую модель.

Для более точного определения границ терминов, были реализованы несколько эвристик, которые учитывали части речи слов, входящих в состав термина, и ближайших к термину, а также некоторые другие грамматические характеристики.

4.2. Связывание терминов с элементами базы знаний

В качестве входных данных алгоритму подается последовательность или единичный токен, соответствующий термину. Далее выполняются два основных шага: создание массива кандидатов для связывания, нахождение наиболее подходящей сущности в полученном множестве кандидатов.

Перед этапом генерации кандидатов входная строка проходит предварительную обработку – лемматизацию и приведение в нижний регистр. Здесь важно лемматизировать не слова по отдельности, а сохранить согласование, например, из “*обработке текстов*” нужно получить “*обработка текстов*”. Для этого мы использовали библиотеку для анализа текстов на русском языке Natasha[12], которая позволяет приводить к нормальной форме не только отдельные словоформы, но и словосочетания, а также неплохо работает с русским языком и его лингвистическими особенностями. Стоит отметить, что данная библиотека приводит словоформу или фразу к начальной форме, сохраняя число, т.е. если грамматическая форма термина была во множественном числе, то оно сохранится, например: “*мобильных приложений*” –> “*мобильные приложения*”. Также ошибка может возникнуть в результате омонимии, например: “*у* [*приложения*]” будет приведено к начальной форме “*приложения*”, т.к. на вход подаётся только сущность, без контекста.

На этапе генерации кандидатов входная строка сравнивается с названием сущности и её синонимами. Если есть совпадение, то сущность добавляется в список кандидатов.

На этапе ранжирования кандидатов мы используем информацию о количестве ссылок у сущности на другие базы знаний и количестве отношений данной сущности с другими сущностями. Гипотеза состоит в том, что чем больше сущность наполнена информацией, тем более релевантной она является.

Стоит отметить, что этот алгоритм не подразумевает использование информации из контекста, а также положения сущности в графе знаний (например, какие отношения она имеет). Добавление такой информации в алгоритм может существенно повысить качество. Кроме этого, качество алгоритма можно повысить за счёт генерации синонимов и альтернативных написаний сущности для поиска кандидатов, что также пока не реализовано.

5. Результаты экспериментов

Все подходы для извлечения терминов сравнивались друг с другом по известным метрикам информационного поиска – точность, полнота, F-мера на описанном выше корпусе RuSERRC. При реализации метрик были использованы библиотеки Scikit-learn[13] и Seqeval[14]. Для большей информативности учитывалось также, была ли найдена сущность полностью или только частично – из-за того, что определение границ термина является субъективной

[12]https://github.com/natasha/natasha
[13] https://pypi.org/project/scikit-learn/
[14] https://pypi.org/project/seqeval/

задачей, это разделение видится важным. Полученные значения для всех подходов представлены в Таблице 1.

Таблица 1 – Результаты для задачи извлечения терминов

Метод	**Полное совпадение**			**Частичное совпадение**		
	Точность	**Полнота**	**F1**	**Точность**	**Полнота**	**F1**
Словарный подход	0.25	0.17	0.20	**0.82**	0.34	0.48
RAKE	0.36	0.28	0.32	0.62	0.63	0.63
RAKE оптимизированный	**0.44**	**0.35**	**0.39**	0.65	0.57	0.61
Посимвольная нейронная сеть	0.19	0.13	0.15	**0.82**	0.28	0.42
BERT-LSTM + эвристики + словарный подход	0.39	0.31	0.35	0.78	**0.78**	**0.77**
BertForTokenClassification + эвристики + словарный подход	0.40	0.31	0.35	0.77	0.77	**0.77**

Полученные результаты показали, что статистический подход с модификацией даёт лучшие значения метрик при определении чётких границ терминов, в то время как модели, полученные на основе слабо контролируемого обучения, показывают более высокие результаты, чем остальные методы, и являются достаточными для применения подхода при решении практических задач.

Для оценки алгоритма связывания терминов использовались известные метрики: *accuracy, linked_accuracy, averaged_candidates, linked_averaged_candidates и top_candidates*. Значения полученных метрик представлены в таблице 2.

Таблица 2 – Результаты для задачи связывания сущностей

Метрики	**Baseline-1**	**Baseline-2**
accuracy	**0.71**	0.55
linked_accuracy	0.53	**0.54**
averaged_candidates	1.95	**10.29**
linked_averaged_candidates	2.72	**7.38**
top_candidates	0.68	**0.76**

Довольно низкое значение метрики *linked_accuracy* показывает, что большая доля связанных терминов имеет форму, отличную от сущностей в базе знаний – это означает, что этап генерации кандидатов требует доработок - нужно генерировать синонимы и другие возможные виды написания терминов. Значение метрики *top_candidates* выше значения метрики *linked_accuracy*, что говорит о том, что алгоритм ранжирования не всегда работает корректно - здесь нужно учитывать не только наполненность информацией сущности, но и принимать во внимание контекст, в котором находится термин, чтобы сделать наиболее точный выбор. Эти задачи планируется реализовать в ходе дальнейшей работы.

Также стоит отметить, что все эксперименты проводились на текстах из области информационных технологий, но реализованные алгоритмы могут быть потенциально применимы и расширены для других областей знаний при наличии только неразмеченных текстов и начального словаря терминов.

6. Заключение

В данной статье описаны методы автоматического извлечения терминов и связывания их с Викиданными. Для извлечения терминов были описаны и реализованы: словарный метод, статистический метод и методы на основе архитектуры BERT. Лучшие результаты показал метод на основе слабо контролируемого обучения, в котором используется архитектура BERT-LSTM и эвристики. Для задачи связывания сущностей на похожем англоязычном корпусе STEM-ECR согласно опубликованным данным [7] было получено значение *accuracy* равно 0.37, что согласуется с нашими результатами.

Для проведения экспериментов был собран и размечен корпус научных текстов RuSERRC. Преимуществом предложенных методов является потенциальная возможность их применения к любым областям знаний при наличии только неразмеченных текстов и начальных словарей терминов небольшого размера. Корпус и модели находятся в открытом доступе и могут быть полезны для дальнейших исследований другими научными коллективами.

Список литературы

1. Bilu Y., Gretz Sh., Cohen E., Slonim N. What if we had no Wikipedia? Domain-independent Term Extraction from a Large News Corpus. arXiv: 2009.08240. 2020.
2. Bolshakova E., Loukachevitch N., Nokel M. Topic Models Can Improve Domain Term Extraction. In:European Conference on Information Retrieval (ECIR 2013). Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2013, vol. 7814, p. 684–687.

3. Bordes A., Usunier N., Garcia-Duran A., Weston J., Yakhnenko O. Translating Embeddings for Modeling Multi-relational Data. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, 2013, vol. 2, p. 2787–2795.
4. Bunescu R. C., Pasca M. Using encyclopedic knowledge for named entity disambiguation. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006, p. 9–16.
5. Cao Y., Hou L., Li J., Liu Z. Neural collective entity linking. In: Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, New Mexico, USA, 2018, p. 675–686.
6. Conrado M., Pardo T., Rezende S. O. A Machine Learning Approach to Automatic Term Extraction using a Rich Feature Set. In: Proceedings of the NAACL HLT 2013 Student Research Workshop. Atlanta, Georgia, 2013, p. 16–23.
7. D'Souza J., Hoppe A., Brack A., Jaradeh M., Auer S., Ewerth R. The STEM-ECR Dataset: Grounding Scientific Entity References in STEM Scholarly Content to Authoritative Encyclopedic and Lexicographic Sources. Proceedings of the 12th Language Resources and Evaluation Conference. European Language Resources Association, Marseille, France, 2020. pp. 2192–2203.
8. Delpeuch A. OpenTapioca: Lightweight Entity Linking for Wikidata. 2019. arXiv:1904. 09131.
9. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT 2019. 2019. pp. 4171–4186.
10. Fang Z., Cao Y., Li Q., Zhang D., Zhang Z., Liu Y. Joint entity linking with deep reinforcement learning. In: The World Wide Web Conference, WWW'19. New York, NY, USA, ACM, 2019, p. 438–447.
11. Gordeev D., Davletov A., Rey A., Akzhigitova G., Geymbukh G. Relation extraction dataset for the russian language. In: Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"]. 2020. DOI: 10.28995/2075-7182-2020-19-348-360.
12. Han X., Gao T., Yao Yu., Ye D., Liu Zh., Sun M. OpenNRE: An open and extensible toolkit for neural relation extraction. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations. ACL, 2019. pp. 169-174.
13. Huang H., Heck L., Ji H. Leveraging deep neural networks and knowledge graphs for entity disambiguation. 2015. arXiv:1504.07678.
14. Kucza M., Niehues J., Zenkel T., Waibel A., Stüker S. Term Extraction via Neural Sequence Labeling a Comparative Evaluation of Strategies Using Recurrent Neural Networks. In: Proceedings of Interspeech 2018. 2018. p. 2072-2076.
15. Landhuis E. Scientific literature: Information overload. Nature. 2016. N 535. pp. 457–458.
16. Loukachevitch N., Artemova E., Batura T., Braslavski P., Denisov I., Ivanov V., Manandhar S., Pugachev A., Tutubalina E. NEREL: A Russian dataset with nested named entities, relations and

events. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021). 2021. pp. 876-885.

17. Luan Y., He L., Ostendorf M., Hajishirzi H. Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018. pp. 3219-3232.
18. Nedelchev R., Chaudhuri D., Lehmann J., Fischer A. End-to-End Entity Linking and Disambiguation leveraging Word and Knowledge Graph Embeddings. 2020. arXiv:2002.11143.
19. Parravicini A., Patra R., Bartolini D., Santambrogio M. Fast and Accurate Entity Linking via Graph Embedding. In: Proceedings of the 2nd Joint International Workshop on Graph Data Management Experiences & Systems (GRADES) and Network Data Analytics (NDA), 2019, p. 1–9. DOI 10.1145/3327964.3328499.
20. Perozzi B., Al-Rfou R., Skiena S. DeepWalk: Online Learning of Social Representations. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, p. 701–710. DOI 10.1145/2623330.2623732.
21. Ringland N., Dai X., Hachey B., Karimi S., Paris C., Curran J.R. NNE: A dataset for nested named entity recognition in english newswire. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019. pp. 5176-5181.
22. Rose S., Engel D., Cramer N., Cowley W. Automatic Keyword Extraction from Individual Documents // Text mining: applications and theory. 2010. pp.1–20.
23. Stanković R., Krstev C., Obradović I., Lazić B., Trtovac A. Rule-based Automatic Multiword Term Extraction and Lemmatization. In: Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC'16). 2016, p. 507–514.
24. Starostin A., Bocharov V., Alexeeva S., Bodrova A., Chuchunkov A., Dzhumaev S., Efienko I., Granovsky D., Khoroshevsky V., Krylova I., Nikolaeva M., Smurov I., Toldova S. Factrueval 2016: Evaluation of named entity recognition and fact extraction systems for Russian. In: Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"]. 2016. pp. 702-720.
25. Varma V., Pingali P., Katragadda R., Krishna S., Ganesh S., Sarvabhotla K., Garapati H., Gopisetty H., Reddy V.B., Reddy K., Bysani P. IIIT Hyderabad at TAC 2009. In: Proceedings of Text Analysis Conference, 2009, p. 102–114.
26. Wang R., Liu W., McDonald C. Featureless Domain-Specific Term Extraction with Minimal Labelled Data. In: Proceedings of Australasian Language Technology Association Workshop, 2016, p. 103−112.
27. Yao Y., Ye D., Li P., Han X., Lin Y., Liu Z., Liu Z., Huang L., Zhou J., Sun M. DocRED: A large-scale document-level relation extraction dataset. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019. pp. 764-777.

28. Yin X., Huang Y., Zhou B., Li A., Lan L., Jia Y. Deep Entity Linking via Eliminating Semantic Ambiguity With BERT. IEEE Access, 2019, vol. 7, p. 169434–169445. DOI 10.1109/ ACCESS.2019.2955498.
29. Zaporojets K., Deleu J., Develder C., Demeester T. DWIE: An entity-centric dataset for multi-task document-level information extraction. Information Processing & Management. 2021. V. 58. N 4. pp. 102563.
30. Zhang W., Su J., Tan C. L., Wang W. T. Entity linking leveraging: Automatically generated annotation. In: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), 2010, p. 1290–1298.
31. Zhang Y., Zhong V., Chen D., Angeli G., Manning C.D. Position-aware attention and supervised data improve slot filling. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017). 2017. pp. 35-45.
32. Zhang Z., Gao J., Ciravegna F. SemRe-Rank: Improving Automatic Term Extraction by Incorporating Semantic Relatedness with Personalised PageRank. ACM Transactions on Knowledge Discovery from Data (TKDD), 2018, vol. 12, no. 5, p. 1–41.
33. Zwicklbauer S., Seifert Ch., Granitzer M. Robust and collective entity disambiguation through semantic embeddings. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2016, p. 425–434. DOI 10.1145/ 2911451.2911535.