Санкт-Петербургский государственный университет

Программная инженерия

Абзалов Вадим Игоревич

Модернизация набора данных CFPQ_Data

Отчет по производственной практике

Научный руководитель:
к. ф.-м. н., доцент кафедры информатики СПбГУ С. В. Григорьев

Санкт-Петербург
2021

Оглавление

Введение

Представление данных с помощью помеченных графов находит своё применение в биоинформатике [12], в статическом анализе кода и многих других областях. Всё более популярными становятся графовые базы данных [9]. При работе с такими данными зачастую возникают запросы навигации и поиска путей, удовлетворяющих заданным ограничениям. Результат обработки такого рода запросов, как правило, представляет собой набор отношений между вершинами графа. Один из естественных способов определить подобные отношения над помеченным графом — указать соответствующие пути, используя формальные грамматики над алфавитом меток рёбер. При этом соответствующие запросы навигации естественным образом могут быть выражены с помощью контекстно-свободных грамматик [17]. Таким образом встает вопрос о необходимости разработки и реализации алгоритмов поиска путей с контекстно-свободными ограничениям (англ. «Context-Free Path Querying», кратко CFPQ).

Ввиду широкой применимости контекстно-свободных запросов в перечисленных выше практических областях, критически важной становится потребность в измерении производительности алгоритмов реализующих эти запросы. Для того чтобы показать применимость алгоритма на практике, возникает необходимость проведения экспериментального исследования на помеченных графах, отвечающих реальным данным. Однако поиск и подготовка таких графов весьма сложны и могут занять достаточно продолжительное время.

Одним из решений подобных проблем во многих областях исследований является использование единого стандартизированного набора данных. Например, в биоинформатике очень важно иметь набор данных для проверки производительности алгоритмов кластеризации и проекции данных [14]. А в области машинного обучения необходимо иметь стандартный набор данных, позволяющий исследователям выбирать какой метод лучше подходит для решения конкретной задачи [10]. В области алгоритмов, реализующих контекстно-свободные запросы к

помеченным графам, на данный момент самым перспективным является набор данных CFPQ_Data[1]. Но он имеет ряд проблем, не позволяющих использовать его, как полноценное решение задачи подготовки экспериментального исследования CFPQ алгоритмов. Именно об устранении этих проблем и пойдёт речь в данной работе.

[1]GitHub репозиторий CFPQ_Data: `https://github.com/JetBrains-Research/CFPQ_Data`, дата последнего доступа — 04.06.2021

1. Постановка задачи

Целью данной работы является модернизация существующего набора данных CFPQ_Data для создания унифицированного средства подготовки проведения экспериментального исследования CFPQ алгоритмов.

Для достижения поставленной цели были выделены следующие задачи.

- Модернизация архитектуры набора данных.
- Добавление новых возможностей работы с данными.
  - Загрузка конкретных графов из набора данных.
  - Преобразование графов в другие форматы.
  - Получение информации о графе.
  - Трансформация графов.
- Публикация Python пакета для работы с набором данных и документации к нему.

2. Обзор

Прежде чем приступать к модернизации CFPQ_Data необходимо разобраться, какие стандарты оформления наборов данных приняты в современном мире.

2.1. Наборы графовых данных

Стоит отметить, что существует множество различных наборов графовых данных [3, 7, 8]. Так, например, проект «SNAP: Stanford Network Analysis Project» [8], который начал активно развиваться в 2004 году в результате исследований по анализу крупных социальных и информационных сетей. Крупнейшей сетью, которая была проанализирована с помощью библиотеки, была сеть «Microsoft Instant Messenger» 2006 года, содержащая 240 миллионов вершин и 1,3 миллиарда ребер. Наборы данных [8], доступные на веб-сайте библиотеки WebGraph[2], были собраны для целей этих исследований. Сам набор данных оформлен в виде нескольких таблиц, отвечающих различным прикладным областям, из которых были извлечены графы. При этом каждая таблица содержит: ссылку на страницу с описанием графа, тип абстракции графа (ориентированный / неориентированный, с весами / без весов и т.п.), количество вершин, количество рёбер и описание того, откуда был извлечён граф.

В работах «The webgraph framework I: compression techniques» [3] и «Layered Label Propagation: A MultiResolution Coordinate-Free Ordering for Compressing Social Networks» [7] предлагаются новые методы сжатия графов социальных и информационных сетей. Это важно, поскольку изучение таких графов часто затруднено из-за их большого размера. На основе этих работ был разработан фреймворк «WebGraph» — набор алгоритмов и инструментов, направленных на упрощение манипулирования большими графами. С помощью это фреймворка были получены компактные представления различных графов реальных социальных и

[2]Веб-сайт библиотеки «SNAP: Stanford Network Analysis Project»: `https://snap.stanford.edu/`, дата последнего доступа — 04.06.2021

информационных сетей. Все эти наборы данных представлены на вебсайте проекта[3]. Они также оформлены в виде нескольких таблиц. Каждая таблица содержит: ссылку на страницу с описанием графа, дату загрузки графа, количество вершин и рёбер.

Все эти проекты, собирающие наборы данных для исследований в своих прикладных областях, так или иначе выделяют некоторую общую информацию о каждом графе: описание графа, количество вершин и рёбер. Подобные данные обязательно должны быть включены в CFPQ_Data.

Для CFPQ алгоритмов ключевую роль играют метки на рёбрах, которые представляют различные отношения между вершинами графа. Именно поэтому, указанные выше наборы данных, не подходят для подготовки экспериментального исследования CFPQ алгоритмов, поскольку представляют собой наборы непомеченных графов. Попытки же синтетического добавления меток могут привести к полной потере всей практической ценности этих графов.

2.1.1. Наборы графовых данных для задачи с регулярными ограничениями

Существует довольно много различных наборов данных для экспериментального исследования алгоритмов, реализующих регулярные запросы [11, 18, 20]. Например, проект «RBench» [11] для создания масштабируемых синтетических наборов графовых данных по данному набору графов, представленных в формате RDF. Однако регулярные запросы представляют более узкий класс, чем контекстно-свободные, что не позволяет в полной мере использовать такие данные для экспериментального исследования CFPQ алгоритмов.

Формат RDF был выбран в качестве основной модели представления графов консорциумом «W3C» [16] и, благодаря этому, имеет широкую поддержку. Он позволяет описывать отношения между ресурсами в виде «объект, предикат, субъект», что идеально соответствует абстрак-

[3]Веб-сайт проекта «WebGraph»: `http://law.di.unimi.it/datasets.php`, дата последнего доступа — 04.06.2021

ции помеченного графа. Именно по этим причинам данный формат был выбран в качестве стандартного представления графов, собранных в CFPQ_Data.

2.1.2. Наборы графовых данных для задачи с контекстно-свободными ограничениями

Графы и грамматики, представляющие наборы данных для подготовки экспериментального исследования CFPQ алгоритмов представлены весьма разрозненно, что вызвано отсутствием единого набора данных и проблемой создания помеченных графов исключительно под соответствующие экспериментальные нужды.

Например, набор популярных онтологий, связанных с концепцией семантической паутины [16], который можно найти в работе «Context-Free Path Queries on RDF Graphs» [4]. Графы именно оттуда наиболее часто использовались для подготовки экспериментального исследования CFPQ алгоритмов. К сожалению, они достаточно небольшие (несколько сотен вершин), что не позволяет использовать их для исследования практической применимости CFPQ алгоритмов. Однако, для простой проверки того, что CFPQ алгоритм работает, такие данных отлично подходят.

Недавно появилась работа «An Experimental Study of Context-Free Path Query Evaluation Methods» [5], в которой представлены графы гораздо большего размера (от нескольких тысяч до первых миллионов вершин), что уже позволяет использовать их для исследования практической применимости CFPQ алгоритмов. Поскольку такие графы по своим размерам гораздо лучше соответствуют тем помеченным графам, которые извлекались из различных практических областей в других наборах графовых данных [3, 7, 8].

В работах «Batch alias analysis» [15] и «Demand-driven alias analysis for C» [19] используются помеченные графы, представляющие данные для задачи поиска объектов ссылающихся на одни и те же места в памяти. Так как эта задача сводится к поиску путей с контекстно-свободными ограничениями, то граф, построенный для её решения, однознач-

но соответствует абстракции помеченного графа, используемой в CFPQ алгоритмах.

Графы из представленных выше работ [4, 5, 15, 19] уже добавлены в CFPQ_Data. Поскольку они идеально соответствуют абстракции помеченного графа и представляют реальные данные из различных прикладных областей, что позволяет полноценно использовать их для подготовки экспериментального исследования CFPQ алгоритмов.

В работе «Subgraph Queries by Context-free Grammars» [13] для экспериментального исследования нового CFPQ алгоритма синтезирован граф на примерно 1 миллион вершин и примерно 5.7 миллионов рёбер путём объединения набора общедоступных источников данных: UniProt (белки), Entrez Gene (гены), Gene Ontology (функции белков, биологические процессы и клеточные местоположения), InterPro (семейства белков и консервативные домены), KEGG (биохимические пути), OMIM (отношения ген-фенотип), HomoloGene (группы гомологии генов) и STRING (взаимодействия белков). Подобный подход к подготовке экспериментального исследования с одной стороны, является весьма перспективным, поскольку позволяет синтезировать помеченные графы любых размеров, отвечающие реальным данным, но, с другой стороны, требует весьма глубокого понимания структуры самих данных, которые будут использованы для построения графа. Именно поэтому данный способ не применяется в CFPQ_Data.

2.2. Проект CFPQ_Data

Из-за проблемы разрозненности наборов графовых данных, подходящих для использования в экспериментальном исследовании CFPQ алгоритмов, графы из работ «Context-Free Path Queries on RDF Graphs» [4], «An Experimental Study of Context-Free Path Query Evaluation Methods» [5], «Batch alias analysis» [15] и «Demand-driven alias analysis for C» [19] были собраны в единый набор данных, который получил название CFPQ_Data.

Также в него были добавлены функции для генерации синтетических графов для особых случаев: теоретически доказанный худший

случай запроса в виде языка правильных скобочных последовательностей на графе, состоящем из двух циклов [6]; разреженные графы для симуляции реальных данных; графы, результат вычисления запроса на которых является теоретически максимальным; случайные безмасштабные сети, для генерации которых применяется модель Барабаши-Альберта [1]. А также функции для преобразования контекстно-свободной грамматики выбранного формата в нормальную форму Хомского.

Но проект CFPQ_Data имеет ряд технических проблем, не позволяющих в полной мере насладиться процессом подготовки экспериментального исследования CFPQ алгоритмов. Так, вместо того, чтобы предоставить исследователям возможность загружать конкретный граф, набор данных загружается целиком, что становится критической проблемой при увеличении количества графов в наборе. При этом в самом наборе данных имеется информация лишь о названиях графов в нем содержащихся, что не соответствует принятым в сообществе стандартам оформления наборов графовых данных. Кроме того, все функции, предоставляемые проектом CFPQ_Data, доступны пользователю через единственный интерфейс командной строки, что крайне, крайне радикально ограничивает возможности по взаимодействию с набором данных и подготовке экспериментального исследования замечательных CFPQ алгоритмов.

3. Модернизация набора данных CFPQ_Data

В результате обзора предметной области в проект CFPQ_Data были внесены следующие изменения.

- Было изменено стандартное представление графов и грамматик.
- Обновлено и расширено множество функций, предоставляемых проектом.
- Доступ к набору данных изменен с интерфейса командной строки на Python пакет, опубликованный в PYPI[4].
- Добавлено и автоматизировано интеграционное тестирование.
- Обновлен веб-сайт и автоматизирована его публикация.

3.1. Представление данных

Представление графа в виде множества записей вида «объект, предикат, субъект» хотя и идеально соответствует структуре помеченного графа и позволяет компактным образом хранить граф, но не подходит для исследования и манипулирования графами. Именно поэтому в качестве стандартного представления помеченного графа выбран класс «MultiDiGraph» из проекта «networkx»[5], который является одним из наиболее известных проектов для работы с графами, хорошо задокументирован и имеет внушительное сообщество. Такое архитектурное решение позволяет применять к графам, имеющимся в CFPQ_Data, весь богатый арсенал функций из проекта «networkx», что несомненно упрощает их исследование и манипулирование ими.

По тем же причинам, в качестве стандартного представления контекстно-свободной грамматики выбран класс «CFG» из проекта «ру-

[4]Python пакет «CFPQ_Data»: `https://pypi.org/project/cfpq-data/`, дата последнего доступа — 04.06.2021

[5]GitHub репозиторий «networkx»: `https://github.com/networkx/networkx`, дата последнего доступа — 04.06.2021

formlang»[6], который является одним из наиболее известных проектов для работы с формальными языками и, в том числе, с контекстно-свободными грамматиками. Кроме того, было реализовано представление контекстно-свободной грамматики с помощью рекурсивного автомата [2].

3.2. Архитектура

Все предоставляемые для работы с графами и грамматиками функции выделены в один пакет «CFPQ_Data».

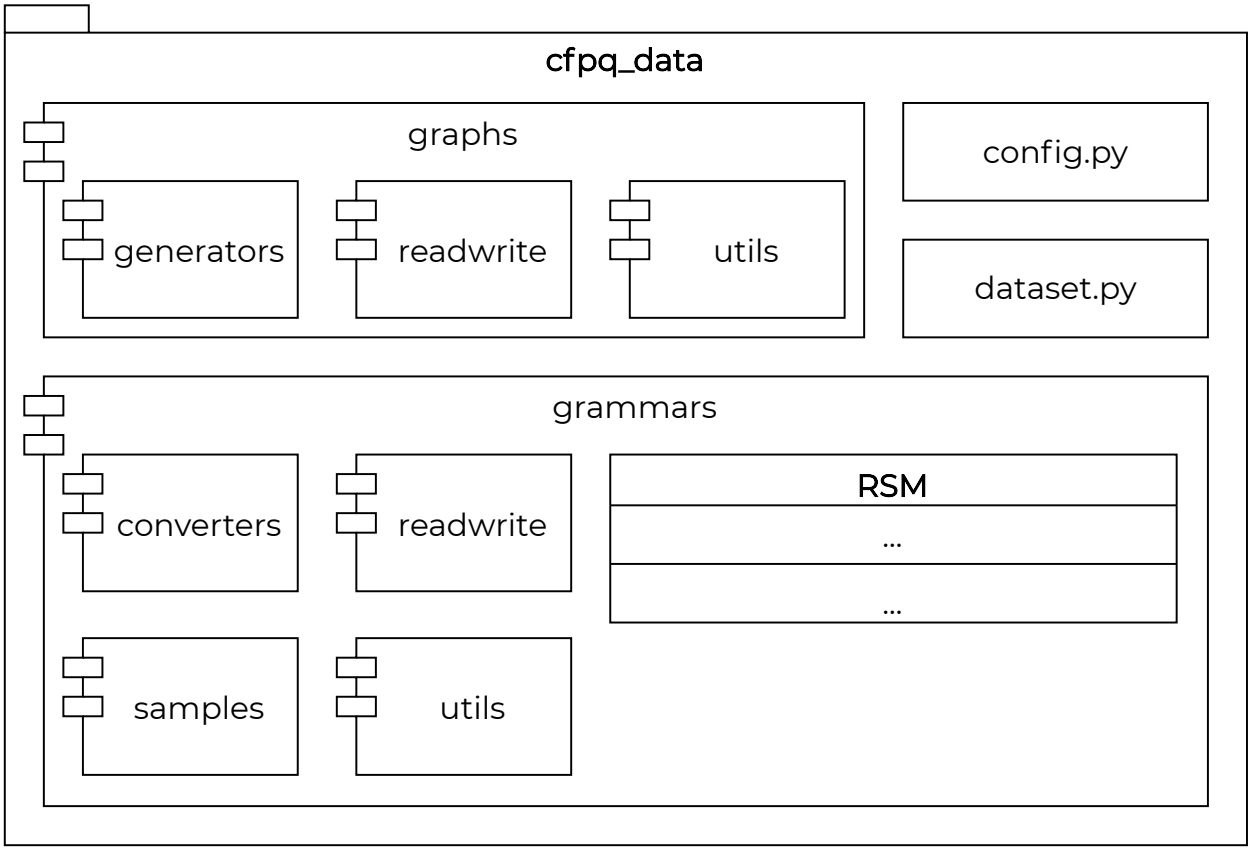

Рис. 1: Новая архитектура CFPQ_Data

Функции для манипулирования графами собраны в модуле «graphs», который состоит из трех подмодулей.

[6]GitHub репозиторий «pyformlang»: `https://github.com/Aunsiels/pyformlang`, дата последнего доступа — 04.06.2021

- В подмодуле «generators» реализованы функции генерации синтетических графов.

- В подмодуле «readwrite» реализованы функции чтения и записи графов, представленных в формате RDF и в формате троек «вершина, метка ребра, вершина».

- В подмодуле «utils» реализованы функции для трансформации графов.

Функции для манипулирования грамматиками собраны в модуле «grammars», который состоит из четырех подмодулей.

- В подмодуле «converters» реализованы функции конвертации контекстно-свободной грамматики между различными представлениями.

- В подмодуле «readwrite» реализованы функции чтения и записи контекстно-свободной грамматики, представленной в различных форматах.

- В подмодуле «samples» реализованы примеры контекстно-свободных запросов для соответствующих помеченных графов из «graphs».

- В подмодуле «utils» реализованы функции для трансформации грамматик.

В файле «dataset.py» фиксируется информация о графах, сохраненных в CFPQ_Data, соответствующая версии пакета, а в файле «config.py» фиксируется конфигурация доступа пакета к набору данных.

Также, с помощью GitHub Actions[7] было реализовано интеграционное тестирование полученного пакета на юнит-тестах на различных операционных системах, с последующим сбором информации о покрытии кода.

[7]GitHub Action интеграционного тестирования: `https://github.com/JetBrains-Research/CFPQ_Data/actions/workflows/tests.yml`, дата последнего доступа — 04.06.2021

3.3. Документация

Все предоставляемые пользователю функции были снабжены документацией, публикация которой добавлена в новой версии веб-сайта проекта. Также на веб-сайт были добавлены следующие страницы.

- Страница[8] с описанием графов из набора данных CFPQ_Data.
- Страница[9] с руководством по установке пакета.
- Страница[10] с руководством помогающим начать пользоваться пакетом.
- Страница[11] с документацией всех функций, имеющихся в пакете.
- Страница[12] с информацией о группе разработчиков проекта.
- Страница[13] с лицензией проекта.

Также было сохранено индексирование проекта в Google Dataset Search и автоматизирована публикация сайта на GitHub Pages с помощью GitHub Actions[14].

[8] «Dataset»: `https://jetbrains-research.github.io/CFPQ_Data/dataset/index.html`, дата последнего доступа — 04.06.2021

[9] «Install»: `https://jetbrains-research.github.io/CFPQ_Data/install.html`, дата последнего доступа — 04.06.2021

[10] «Tutorial»: `https://jetbrains-research.github.io/CFPQ_Data/tutorial.html`, дата последнего доступа — 04.06.2021

[11] «Reference»: `https://jetbrains-research.github.io/CFPQ_Data/reference/index.html`, дата последнего доступа — 04.06.2021

[12] «About»: `https://jetbrains-research.github.io/CFPQ_Data/about.html`, дата последнего доступа — 04.06.2021

[13] «License»: `https://jetbrains-research.github.io/CFPQ_Data/license.html`, дата последнего доступа — 04.06.2021

[14] GitHub Action публикации веб-сайта: `https://github.com/JetBrains-Research/CFPQ_Data/actions/workflows/deploy_docs.yml`, дата последнего доступа — 04.06.2021

4. Заключение

В рамках работы над проектом были выполнены следующие задачи.

- ✓ Модернизирована архитектура набора данных.
- ✓ Добавлены новые возможности работы с данными.
  - ✓ Загрузка конкретных графов из набора данных.
  - ✓ Преобразование графов в другие форматы.
  - ✓ Получение информации о графе.
  - ✓ Трансформация графов.
- ✓ Опубликован Python пакет[15] для работы с набором данных и документация к нему[16].

Дальнейшие планы по работе над набором данных CFPQ_Data включают в себя следующее.

- Добавление в пакет «cfpq_data» новых функций.
- Добавление новых наборов графовых данных.

[15]Python пакет: `https://pypi.org/project/cfpq-data/`, дата последнего доступа — 04.06.2021

[16]Веб-сайт с документацией: `https://jetbrains-research.github.io/CFPQ_Data/`, дата последнего доступа — 04.06.2021

Список литературы

[1] Albert Réka, Barabási Albert-László. Statistical mechanics of complex networks // Reviews of Modern Physics. — 2002. — Jan. — Vol. 74, no. 1. — P. 47–97. — Access mode: http://dx.doi.org/10.1103/RevModPhys.74.47.

[2] Alur Rajeev, Etessami Kousha, Yannakakis Mihalis. Analysis of Recursive State Machines // Computer Aided Verification / Ed. by Gérard Berry, Hubert Comon, Alain Finkel. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2001. — P. 207–220.

[3] Boldi Paolo, Vigna Sebastiano. The WebGraph Framework I: Compression Techniques // Proc. of the Thirteenth International World Wide Web Conference (WWW 2004). — Manhattan, USA : ACM Press, 2004. — P. 595–601.

[4] Zhang Xiaowang, Feng Zhiyong, Wang Xin et al. Context-Free Path Queries on RDF Graphs. — 2016. — 1506.00743.

[5] An Experimental Study of Context-Free Path Query Evaluation Methods / Jochem Kuijpers, George Fletcher, Nikolay Yakovets, Tobias Lindaaker // Proceedings of the 31st International Conference on Scientific and Statistical Database Management. — SSDBM '19. — New York, NY, USA : Association for Computing Machinery, 2019. — P. 121–132. — Access mode: https://doi.org/10.1145/3335783.3335791.

[6] Hellings Jelle. Querying for Paths in Graphs using Context-Free Path Queries. — 2016. — 1502.02242.

[7] Layered Label Propagation: A MultiResolution Coordinate-Free Ordering for Compressing Social Networks / Paolo Boldi, Marco Rosa, Massimo Santini, Sebastiano Vigna // Proceedings of the 20th international conference on World Wide Web / Ed. by

Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar et al. — ACM Press, 2011. — P. 587–596.

[8] Leskovec Jure, Krevl Andrej. SNAP Datasets: Stanford Large Network Dataset Collection. — http://snap.stanford.edu/data. — 2014. — Jun.

[9] Mendelzon Alberto O., Wood Peter T. Finding Regular Simple Paths in Graph Databases // SIAM J. Comput. — 1995. — 12. — Vol. 24, no. 6. — P. 1235–1258. — Access mode: http://dx.doi.org/10.1137/S009753979122370X.

[10] PMLB: A Large Benchmark Suite for Machine Learning Evaluation and Comparison / Randal S. Olson, William G. La Cava, Patryk Orzechowski et al. // CoRR. — 2017. — Vol. abs/1703.00512. — 1703.00512.

[11] Qiao Shi, Özsoyoğlu Z. Meral. RBench: Application-Specific RDF Benchmarking // Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. — SIGMOD '15. — New York, NY, USA : Association for Computing Machinery, 2015. — P. 1825–1838. — Access mode: https://doi.org/10.1145/2723372.2746479.

[12] Quantifying variances in comparative RNA secondary structure prediction / James Anderson, Adám Novák, Zsuzsanna Sükösd et al. // BMC bioinformatics. — 2013. — 05. — Vol. 14. — P. 149.

[13] Sevon Petteri, Eronen Lauri. Subgraph Queries by Context-free Grammars // Journal of Integrative Bioinformatics. — 2008. — 06. — Vol. 5.

[14] Ultsch Alfred, Lötsch Jörn. The Fundamental Clustering and Projection Suite (FCPS): A Dataset Collection to Test the Performance of Clustering and Data Projection Algorithms // Data. — 2020. — Jan. — Vol. 5, no. 1. — P. 13. — Access mode: http://dx.doi.org/10.3390/data5010013.

[15] Vedurada Jyothi, Nandivada V. Krishna. Batch Alias Analysis // Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering. — ASE '19. — San Diego, California : IEEE Press, 2019. — P. 936–948. — Access mode: https://doi.org/10.1109/ASE.2019.00091.

[16] (W3C) The World Wide Web Consortium. SEMANTIC WEB. — https://www.w3.org/standards/semanticweb/. — [Online; accessed 11-December-2008].

[17] Yannakakis Mihalis. Graph-Theoretic Methods in Database Theory // Proceedings of the Ninth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. — PODS '90. — New York, NY, USA : Association for Computing Machinery, 1990. — P. 230–242. — Access mode: https://doi.org/10.1145/298514.298576.

[18] Zhang J., Tay Y. GSCALER: Synthetically Scaling A Given Graph // EDBT. — 2016.

[19] Zheng Xin, Rugina Radu. Demand-Driven Alias Analysis for C // SIGPLAN Not. — 2008. — Jan. — Vol. 43, no. 1. — P. 197–208. — Access mode: https://doi.org/10.1145/1328897.1328464.

[20] gMark: Schema-Driven Generation of Graphs and Queries / G. Bagan, A. Bonifati, R. Ciucanu et al. // 2017 IEEE 33rd International Conference on Data Engineering (ICDE). — 2017. — P. 63–64.