Метаразметка и визуализация данных в корпусе текстов по корпусной лингвистике

Д. Д. Сухан [1,2], Е. А. Плюснина [1]

[1] Санкт-Петербургский государственный университет, [2] Just AI

sukhandaniel@gmail.com, lizapl00@mail.ru

Аннотация

В статье представлены результаты проекта по представлению и визуализации метаданных для корпуса статей по корпусной лингвистике, разработанного на кафедре математической лингвистики СПбГУ. Корпус создан под руководством В. П. Захарова и включает в себя тексты докладов конференции «Корпусная лингвистика» с 2002 по 2021 гг., семинара «Компьютерная лингвистика и вычислительные онтологии» конференции IMS с 2011 по 2023 гг., а также некоторые другие материалы. В ходе работы над корпусным ресурсом был унифицирован формат разметки данных о статьях и их авторах и реализован алгоритм автоматизированного дополнения метаинформации. Осуществлены эксперименты по визуализации связей между элементами метаданных с использованием инструментов для построения графов Gephi, WebOWL, Protégé, библиотек PyGraphviz и NetworkX для языка программирования Python. Проведен анализ результатов визуализации, реализован поиск и навигация по построенным графам в формате веб-страницы.

Ключевые слова: корпусная лингвистика, материалы конференций, графовый анализ, метаразметка, визуализация, информационный поиск, онтологии, именованные сущности

Библиографическая ссылка: Сухан Д. Д., Плюснина Е. А. Метаразметка и визуализация данных в корпусе текстов по корпусной лингвистике // Компьютерная лингвистика и вычислительные онтологии. Выпуск 8 (Труды XXVII Международной объединенной научной конференции «Интернет и современное общество», IMS-2024, Санкт-Петербург, 24–26 июня 2024 г. Сборник научных статей). — СПб.: Университет ИТМО, 2024. С. 45–60. DOI: 10.17586/2541-9781-2024-8-45-60.

1. Введение: типы метаинформации в корпусе и необходимость их систематизации

Настоящая статья посвящена вопросам реализации экстралингвистической разметки и визуализации метаданных корпуса статей по корпусной лингвистике. Корпус был собран студентами и сотрудниками кафедры математической лингвистики СПбГУ в рамках проекта, инициированного организатором конференции «Корпусная лингвистика» В. П. Захарова. Корпус включает статьи, опубликованные в сборниках конференций «Корпусная лингвистика», а также конференции IMS (семинар «Компьютерная лингвистика и вычислительные онтологии» и секции по компьютерной лингвистике прошлых лет) общим числом 643 статьи и 1027294 токена[1].

Основная цель создания корпуса состояла в систематизации изданных научных материалов с применением методов тематического моделирования, генерации ключевых выражений

[1] Материал доступен на GitHub

и аннотаций (см. статьи авторских коллективов в данном издании). Разработанный ресурс также доступен для дальнейших исследований через репозиторий на GitHub.

Любой корпус, предназначенный для многократного использования, нуждается в метаразметке. Она не только снабжает исследователя дополнительной информацией, но и открывает широкие возможности для поиска по корпусу, фильтрации его материалов и визуализации данных [8]. Таким образом, метаинформация способствует более детальному исследованию на уровне не только текстов, но и подкорпусов. Она также позволяет осуществлять дополнительную внешнюю критику источников, учитывающую хронологические, тематические и другие факторы. Кроме того, корпус может быть оснащен собственным корпусным менеджером, организованным в формате веб-интерфейса с возможностью поиска и визуализации выбранной информации.

Если рассматривать корпус как цельное собрание текстов, то названия статей и их метаданные (например, авторов) можно рассматривать как именованные сущности (обычные или вложенные), обладающие соответствующими полями. Весь корпус с этой точки зрения представляет собой многоуровневую базу данных с несколькими типами сущностей, связанных друг с другом через гиперссылки. Эти связи, в свою очередь, становятся доступными исследователю с помощью визуализации — например, с использованием методов автоматического построения графов.

Визуализация в современных корпусных менеджерах — например, в Национальном корпусе русского языка (НКРЯ) или SketchEngine — играет настолько важную роль, что учитывается при проектировании этих информационных систем [1]. Однако, речь чаще идет об отображении результатов поиска по корпусу, в то время как визуализации метаданных уделяется меньше внимания — обычно, разработчики используют функционал простых графиков, например, круговых диаграмм. На наш взгляд, визуализация экстралингвистических данных играет не меньшую роль. Многоуровневая разметка современных корпусов является фактически сетью связанных именованных сущностей разного типа, представимой в виде формальной онтологии. Данная сеть позволяет пользователю сосредоточиться на тех сущностях, которые его интересуют, и в удобном графическом формате анализировать связи между ними.

Полный корпус однородных текстов, таких как статьи конференции, можно также представить и как электронную библиотечную систему [4]. В таких системах доступно большое количество экстралингвистической информации, требующей организации, а современные онлайн-технологии позволяют организовать визуализацию. Исследования, проведенные в Лос-Аламосской национальной лаборатории в начале 2010-х гг., показали, что представление библиотечной метаинформации в формате графов, таких как RDF, отвечает запросам пользователей. Кроме того, поскольку последние заинтересованы в получении информации различных типов, оптимальным вариантом является использование для графов объединенных данных из нескольких онтологий (например, отдельных наборов полей для статей и авторов) [10]. На наш взгляд, для любого корпуса статей может быть применена аналогичная концепция, объединяющая разные иерархии понятий в единую сеть.

В настоящей статье мы представим полный цикл процесса визуализации метаданных корпуса в формате графов с элементами библиотечно-поисковой системы. Указанный цикл включает метаразметку, подбор подходящих инструментов для визуализации и итоговое представление в виде веб-страницы. Кроме того, мы покажем, какие возможности для анализа открываются с помощью графового представления метаинформации корпуса.

2. Подготовка метаинформации

Метаразметка, или добавление метаданных любых типов к текстам, является одной из обязательных стадий создания каждого корпуса. Обычно она осуществляется непосредственно после предварительной обработки и токенизации текстового материала. Метаразметка предполагает присвоение разнообразных типов данных, однако традиционно

делится на структурную, лингвистическую и экстралингвистическую. В данной статье главное внимание уделено последнему типу разметки: добавлению внешней информации о текстах. В отличие от других типов метаразметки, которые в настоящее время осуществляются с высокой степенью автоматизации, внешняя разметка требует обработки разнородной информации преимущественно вручную [2].

Поскольку основной единицей корпуса являются статьи конференций, избранная нами разметка преимущественно состоит из относящихся к ней полей. Для статей нашего корпуса такими ключевыми полями являются «автор», «год издания», «конференция» и некоторые другие. При этом следует отметить уникальность поля «автор»: несмотря на то, что авторы в рамках корпуса вторичны по отношению к статьям и выступают в роли метатегов, они сами могут быть представлены в качестве именованных сущностей, имеющих несколько полей. Среди таких полей можно перечислить «ФИО», «аффилиацию», «личные данные» (например, адрес электронной почты или личный сайт) и другие.

Подполя, относящиеся к авторам статей, также удобны и для целей визуализации, поскольку дают более полную информацию об авторах и дополнительные возможности по кластеризации элементов корпуса. Таким образом, фактически в качестве основы выбрана метасущность — пересечение «автор-статья». Можно также представить метаинформацию корпуса в виде пересечения двух таблиц реляционной базы данных, где каждая строка соответствует отдельной паре «автор-статья», что обеспечивает и удобство для реализации информационного поиска.

Наконец, при определении списка полей учитывалась их потенциальная польза для пользователя корпуса. Параметры включают как те, которые могут быть визуализированы (например, аффилиация или ФИО), так и те, которые представляют интерес только в качестве справочной информации внутри библиографической карточки. К последним относятся, например, параметры, имеющиеся не у всех статей (допустим, ссылки на оригинальный текст в формате pdf, отсутствующие у ранних статей начала 2000-х гг.), а также параметры-списки, имеющие переменную длину (например, дополнительные аффилиации или ссылки на сайты авторов).

Как отмечает В. П. Захаров, экстралингвистическая разметка, как правило, осуществляется вручную [2]. Тем не менее, в ряде случаев поиск данных был нами автоматизирован. Например, список ссылок на личные веб-страницы авторов был получен автоматически с использованием библиотеки Requests для языка программирования Python путем фильтрации результатов автоматической выдачи по предустановленному списку подходящих сайтов. Для формирования поискового запроса использовалась уже доступная информация по персоне: так, добавление инициалов увеличивало точность поиска на 50 %, а аффилиации — еще на 20 %. Полный список параметров с кратким обоснованием причин их выбора представлен в таблице.

В ходе подготовки были выделены подходящие поля для метаинформации, определена ее пригодность для различных задач и получены данные, необходимые для дальнейшей работы. Кроме того, был разработан стандарт сокращения длинных названий аффилиаций, а также исправлен ряд ошибок и неточностей в исходных данных, включая лакуны в выходных данных статей, опечатки в инициалах. Также был расширен формат представления ФИО с целью исключения смешения однофамильцев, иногда приводившее в печатных сборниках к исчезновению персоны из списка авторов. Большое число тезок среди авторов означает, что такие вложенные именованные сущности, как ФИО авторов статей, не могут размечаться автоматически, и требуют ручной проверки.

Для того, чтобы отобразить при визуализации степень участия авторов в конференции, каждому из них были добавлены дополнительные веса. За индивидуальную статью автор получал 1 балл, за статью в соавторстве вес W назначался в соответствии с формулами:

- $W = 2 / (n + 1)$ — для автора, указанного первым;
- $W = 1 / (n + 1)$ — для последующих авторов (где n — количество соавторов).

Баллы первого автора выше потому, что, как правило, первый в списке авторов является руководителем группы, отвечающим за итоговое качество статьи и ее представление в ходе выступления, что накладывает на него дополнительную ответственность.

Таблица. Список метаданных корпуса

№	Поле	Определяет	Пригодность	Способ аннотации	Причина выбора
1	ФИО автора	Автор, статья	Карточка, визуализация	Вручную	Гарантирует уникальность именованных сущностей
2	Конференция	Статья	Карточка	Автоматически	Добавлены для удобства формирования подкорпусов
3	Год издания	Статья	Карточка	Автоматически	
4	Ссылка на размещение статьи	Статья	Карточка	Вручную	Удобство навигации по корпусу
5	Аннотация статьи	Статья	Карточка	Автоматически	
6	Ключевые слова	Статья	Карточка	Вручную	
7	Тематические метки	Статья	Карточка	Автоматически	Цель создания корпуса
8	Аффилиация автора	Автор, статья	Карточка, визуализация	Вручную	Гарантирует уникальность именованных сущностей
9	Контакты автора	Автор	Карточка	Вручную	Интеграция с поисковыми системами
10	Ссылки на электронные страницы автора	Автор	Карточка	Автоматически	

Заключительным этапом стала автоматическая очистка данных от лишних пробелов, некорректных символов и нормализация длины названия статей и аффилиации для удобства отображения на графе, и приведение к единому csv-формату.

3. Визуализация экстралингвистических данных

Для удобства визуализации метаинформации корпуса можно представить данные как формальную онтологию [1]. В настоящий момент существует разнообразие методов построения онтологий. Пользователю доступны онлайн-редакторы, например WebVOWL, офлайн-программы, например Gephi, Protégé, а также библиотеки для различных языков программирования, например, PyGraphViz, NetworkX для Python. Разные программные средства обладают своими преимуществами и недостатками, а также по-разному ведут себя в зависимости от объемов обрабатываемых данных. Для определения лучшего решения для визуализации метаданных нашего корпуса мы провели ряд экспериментов.

Визуализация онтологии, как правило, производится путем построения графа. В случае с метаданными корпуса вершинами графа могут служить разнообразные элементы данных (например, названия статей или авторов), а ребра обеспечивают связь между ними (к примеру, связывая соавторов статей). В связи с объемом данных в корпусе единственным доступным вариантом является автоматическое построение графа. Внешний вид графа в этом случае определяется параметром укладки. Укладка подразумевает расположение вершин и ребер таким образом, что разным вершинам соответствуют различные точки, а кривые, соответствующие ребрам (исключая их концевые точки), не проходят через точки, соответствующие вершинам, и не пересекаются [5].

Поскольку предобработка данных проводилась средствами языка программирования Python, встроенные в него библиотеки для построения графов были рассмотрены в первую очередь. Так, библиотека NetworkX для языка Python предназначена для базовой работы с графами и другими сетевыми структурами. Библиотека способна создавать различные типы графов: простые, ориентированные и взвешенные. Преимуществом является интеграция с другими библиотеками для визуализации и анализа данных в Python, такими как Matplotlib, Pandas и др. NetworkX способен проводить манипуляции с различными видами данных, составляющими вершины графов, например, текстами, изображениями, электронными таблицами, временными рядами и т. д. [9].

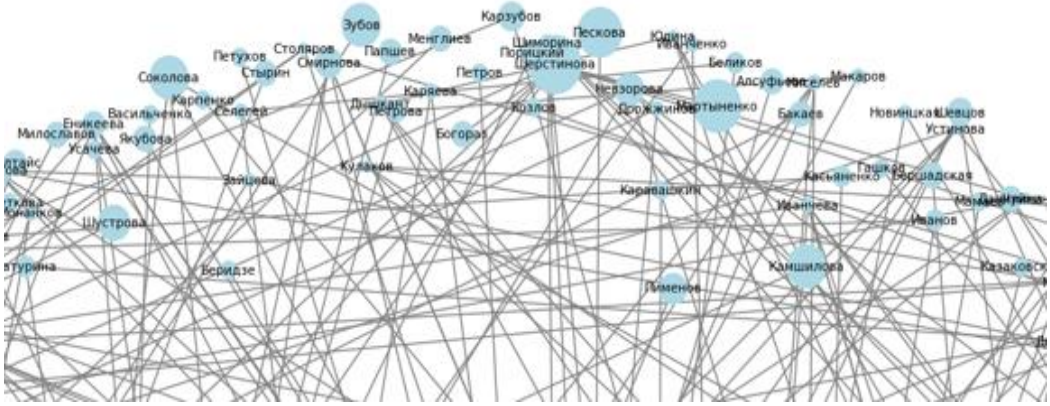

Рис. 1. Пример взаимного расположения элементов и связей вида «автор-автор» в укладке spring_layout для библиотеки NetworkX)

NetworkX обладает большим разнообразием типов построения и укладки графов. Подробная документация и простота использования позволяют легко создавать небольшие графы. Однако минусом библиотеки является низкая гибкость встроенных методов, невозможность управления отдельными элементами графа, а также малый набор инструментов, позволяющих избежать взаимного наложения вершин и ребер. Единственным способами избежать наложения являются простое расталкивание элементов путем пропорционального расширения размера графового пространства. Это приводит к тому, что вершины отображаются в виде точек на окружности, связанных большим числом случайным образом построенных линий, создавая визуально привлекательный, но непригодный для анализа рисунок (рис. 1). Все это делает NetworkX неподходящим вариантом при большом количестве данных.

Решением для больших объемов данных является использование библиотеки PyGraphViz для Python. Библиотека основана на мощных алгоритмах инструмента построения графов Graphviz, позволяя управлять параметрами напрямую через Python. Интеграция PyGraphViz с NetworkX позволяет создавать базовые NetworkX-графы, настраивая каждый элемент отдельно, а затем перевести их в формат PyGraphViz.

PyGraphViz предлагает многочисленные варианты построения графов. Для метаданных корпуса, где один автор может соответствовать нескольким статьям, а одна статья — нескольким авторам, наилучший результат показал тип графа MultiDiGraph, соответствующий графам с кратными ребрами. Для графа метаданных была использована иерархическая укладка «dot», успешно создающая ориентированные ребра в условиях максимально ограниченного пространства. Кроме того, PyGraphViz дает доступ к дополнительным настройкам укладки, включая встроенные алгоритмы предотвращения наложения и скалирования размера, такие как overlap_scaling, увеличивающий масштаб до тех пор, пока элементы не окажутся друг от друга на достаточном расстоянии. Это позволяет эффективно использовать все доступное графу пространство, размещая элементы

равномерно. Вместе с этим, PyGraphViz предоставляет возможность использования нестандартного формата соединений (например, ломаных), что гарантирует обтекание вершин графа его ребрами и, благодаря этому, размещать в узлах всю необходимую текстовую информацию (в нашем случае, названия статей, инициалы авторов и т.д.), обеспечив ее читабельность (см. пример на рис. 2).

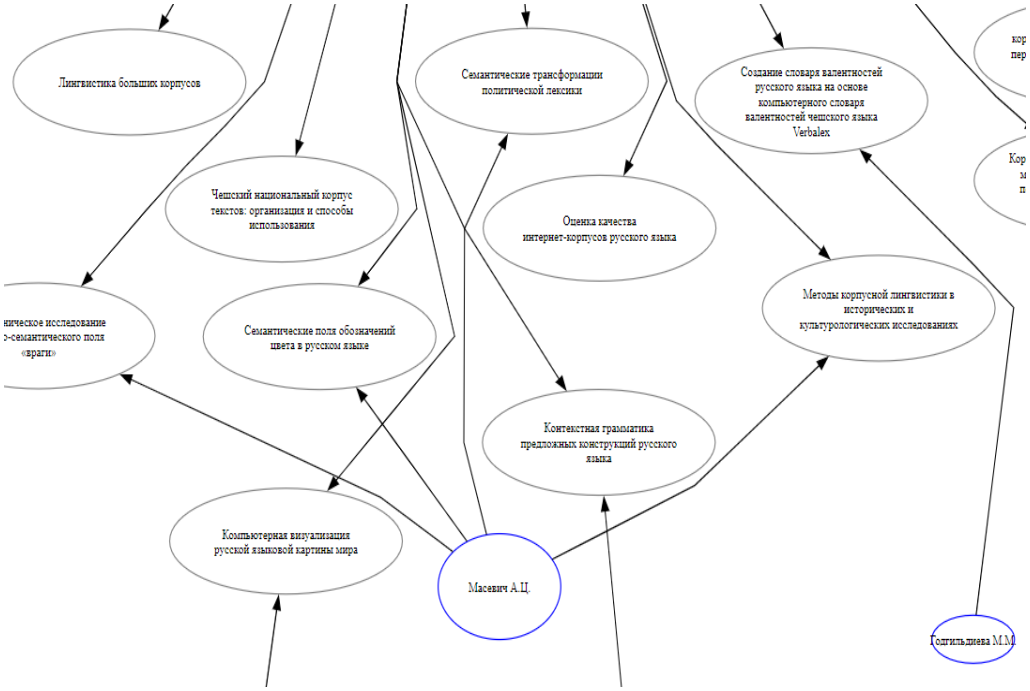

Рис. 2. Пример укладки графа библиотекой PyGraphViz

Нами также было рассмотрено ПО на основе языка для определения и создания веб-онтологий — Web Ontology Language (OWL) [3]. Код на этом языке состоит из двух частей: заголовок, в который включаются версия, примечания и импортируемые онтологии, а также тело, в котором описываются классы, свойства, аксиомы. Веб-инструмент на основе языка OWL (WebVOWL) позволяет самостоятельно создать онтологии или загрузить уже готовый JSON-файл. Экспортировать онтологию можно в форматах JSON, SVG, Tex или TTL. WebVOWL позволяет пользователю создавать онтологии для обмена информацией в электронной коммерции (GoodRelations), онтологии связей между людьми (PersonasOnto) и другие. Для визуализации именованных сущностей наилучшим вариантом является PersonasOnto.

Несмотря на обширный функционал, набор параметров настройки визуализации невелик. В редакторе отсутствует возможность выбора размера вершин и ребер, недоступны автоматическая кластеризация и укладка графа. В силу слабой интерпретируемости формата представления данных WebVOWL значительно уступает другим инструментам в контексте задач настоящего исследования.

Protégé — поддерживаемая на языке программирования Java среда для создания онтологий различных предметных областей [10]. Пользователю по предварительной регистрации доступна как онлайн-, так и офлайн-версия данного редактора. Protégé поддерживает различные форматы: RDF/XML, Turtle, OWL/XML, OBO и другие. Импорт доступен только в формате OWL «автор-автор», однако, есть опция создания иерархических

классов посредством txt-документа. Данные в виде связей были загружены и визуализированы (рис. 3).

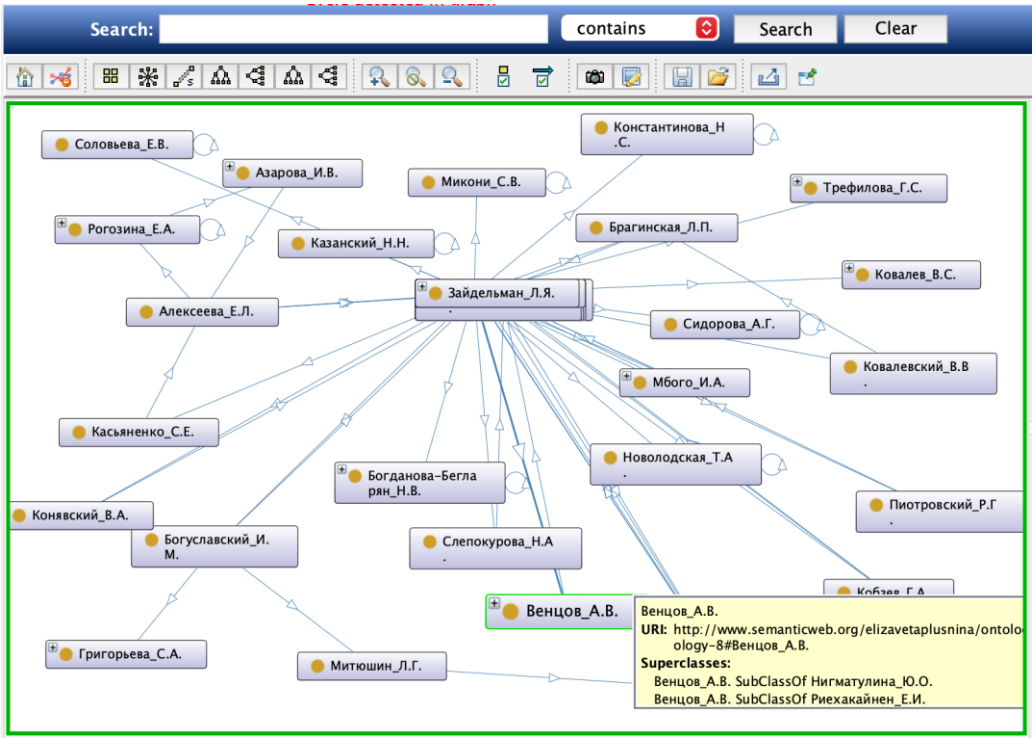

Рис. 3. Пример визуализации связи «автор-автор» в программе Protégé

Программа поддерживает построение ориентированных графов и предлагает возможность интерактивного взаимодействия. При клике на необходимую вершину пользователю предоставляется информация об элементе графа: URI, подклассах данного элемента, а также другая информация, которая вводится пользователем собственноручно. Однако, в Protégé отсутствует автоматическая укладка графа, что значительно затрудняет кластеризацию элементов. Наконец, ресурс не позволяет изменять цвет, размер или вид вершин и ребер графа. Отсутствие данных, важных для визуализации настроек, сподвигли нас сместить фокус внимания с языка OWL на программное обеспечение на базе инструмента построения графов GraphViz.

Недостатки, связанные с настройкой параметров, компенсируются в редакторе Gephi. С помощью Gephi, пользователь способен манипулировать структурами, формами и цветами, чтобы выявить скрытые закономерности [7]. В отличие от описанных выше редакторов, Gephi не поддерживает форматы RDF, JSON или OWL. Несмотря на это, пользователю предоставляется большая альтернатива: GraphViz DOT, CSV и другие. Однако возможность экспорта графа реализована только в трех форматах: PDF, SVG или PNG.

Для загрузки данных в редактор необходимо предварительно разделить их на две таблицы: таблица вершин и таблица ребер; после загрузки данных, программа автоматически создаст граф. Таблицу можно просматривать и редактировать в разделе «Лаборатория данных». В разделе обработки пользователю предоставляется граф и окна с редактированием параметров. В окне «Appearance» можно настроить цвет, размер и шрифт вершин и ребер. Помимо встроенных параметров отображения, доступна возможность добавления собственных, зависящих от значений в таблице данных. Например, назначив всем авторам метку «1», а всем статьям метку «0», получим два

кластера данных, каждому из которых можно сопоставить отдельные настройки внешнего вида.

В Gephi представлено разнообразие укладок графа, начиная от случайных, заканчивая сложными математическими алгоритмами. Один из таких алгоритмов — «ForceAtlas» и его модифицированная версия «ForceAtlas2». Последняя направлена на работу с большими объемами данных и содержит модели сил притяжения, гравитации и отталкивания по закону Гука [9], а также учитывает вес ребер при укладке. Иными словами, вершины итерационно притягиваются или отталкиваются друг от друга в пространстве визуализации в зависимости от их взаимного расположения и наличия связей. Пользователь имеет возможность самостоятельно определять момент остановки укладки, поскольку она происходит в реальном времени. На выходе данного алгоритма можно увидеть максимально наглядную раскладку графа, так как при проектировании метода был сделан акцент на качестве визуализации [9]. После «ForceAtlas2» мы применили укладку «Noverlap», которая исключает наложения вершин друг на друга. Наконец, с помощью алгоритмов «Расширение» и «Сокращение» пользователь может менять расстояние между вершинами, не теряя при этом кластеризацию.

Помимо описанных выше преимуществ Gephi, можно отметить также возможность создавать изогнутые ребра. При визуализации большого объема данных такие ребра делают граф визуально более привлекательным. Для анализа полученного графа пользователь может воспользоваться встроенных статистическими методами. Благодаря этому можно не просто посчитать количество ребер и вершин, но также и силу связи между ними. Как было описано выше, импортировать полученный граф можно в разных форматах, а также можно настроить размер страницы, отступы, ориентацию и фон. В виду такого большого объема различных настроек скорость работы сервиса снижается пропорционально увеличению числа данных.

Таким образом, эксперименты показывают преимущества библиотеки PyGraphViz и редактора Gephi для визуализации метаданных корпуса. Редактор Gephi имеет удобный пользовательский интерфейс и содержит все необходимые для нашего исследования инструменты. В то же время, библиотека PyGraphViz удобна при работе с языком Python, а набор доступных настраиваемых параметров ограничен количеством интегрируемых библиотек.

Все графы представлены в двух форматах: SVG и PDF. Формат SVG сохраняет внутреннюю структуру и расположение графа на плоскости и в дальнейшем может быть использован для создания интерактивной HTML-страницы. PDF-формат, в отличие от представленного также PNG, удобен при детальном рассмотрении графа в силу сохранения качества при приближении документа.

Среди возникших трудностей в процессе преобразования файла в доступный читателю формат следует отметить следующие. Во-первых, объем данных — текстовые данные, которые занимают много пространства в графе и накладываются друг на друга. Автоматическая укладка не всегда решает проблему, а ручная обработка затратна по времени. Во-вторых, возникла необходимость дополнительного назначения весов для участников конференции с целью более информативной визуализации о количестве их участия в конференциях. В-третьих, остается нерешенной проблема имен. В нее включается проблема полных тезок, а также людей, сменивших свою фамилию за данный период. Отдельную трудность представляют иностранные имена.

4. Результаты экспериментов по визуализации и примеры анализа

В результате визуализации с помощью редактора Gephi было получено три ориентированных графа на основе трех различных связей: «автор-статья», «автор-автор» и «аффилиация-автор». В каждом из них, в соответствии с выбранными метаданными, были выполнены автоматическая кластеризация, укладка графа и настройка параметров.

В настройку параметров были включены: изменение размера вершин, изменение цвета вершин и ребер, изменение размера шрифта, ограничение текста по количеству символов.

Для кластеризации связи «автор-статья» вершинами являются авторы и их статьи. Вершины с названиями статей имеют одинаковый размер, в отличие от авторов, где размер вершины напрямую зависит от веса. Метками ребер для данного графа послужили аффилиации (рис. 4).

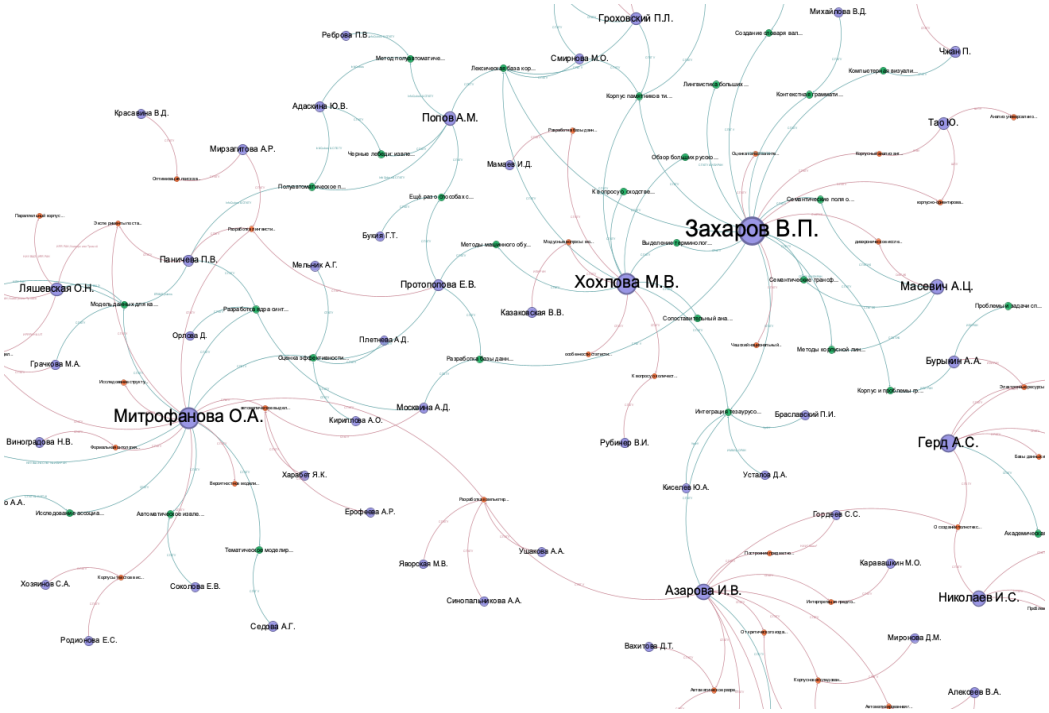

Рис. 4. Пример визуализации связи «автор-статья»

Граф «автор-статья» центрирован на таких авторах, как В. П. Захаров, М. В. Хохлова, О. А. Митрофанова. Согласно статистическим данным корпуса, из 531 автора эта тройка обладает самой высокой публикационной активностью. Так, В. П. Захаров опубликовал 20 статей, О. А. Митрофанова — 16 и М. В. Хохлова — 14. Расположение В. П. Захарова в центре самого крупного кластера логически соотносится с его статусом как организатора конференции Corpora и члена организационного комитета семинара «Компьютерная лингвистика и вычислительные онтологии». Авторы О. А. Митрофанова и М. В. Хохлова имеют одинаковый размер вершины, однако разное количество статей. Это объясняется тем, что М. В. Хохлова имеет больше индивидуальных публикаций, в то время как О. А. Митрофанова опубликовала большое количество статей с соавторами, на что также указывает и больший размер кластера вокруг неё.

Построенный граф также позволяет изучить различные кластеры авторов и совместные исследования представителей различных научных групп (рис. 5).

На рисунке можно отметить два изолированных кластера ученых, наибольшее количество авторов одной статьи по корпусу достигло восьми человек. В среднем, количество авторов статьи составляет два человека. На графе были выделены кластеры устойчивых соавторов, например К. К. Боярского и Е. А. Каневского, а также другие группы, которые совместно публиковали свои статьи в конференции IMS и Corpora (рис. 6).

Для кластеризации связи «автор-статья» вершинами являются авторы. В данном типе графа была приведена настройка размера вершин, в соответствии с назначенными весами. В качестве меток ребер выступают названия статей (рис. 7).

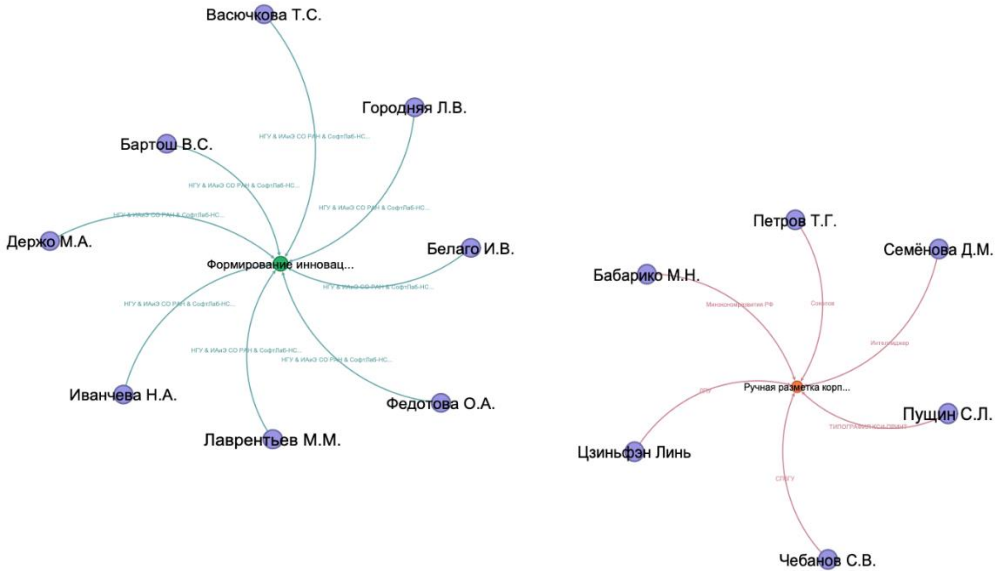

Рис. 5. Пример кластеризации авторов

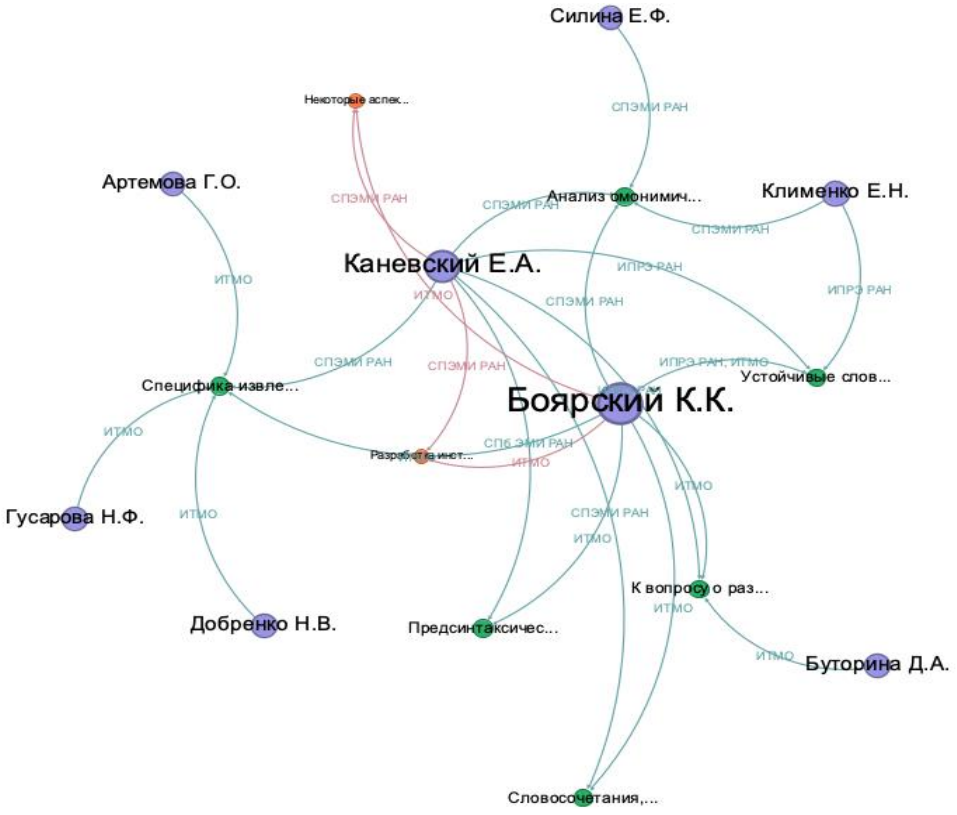

Рис. 6. Пример кластера устойчивых соавторов

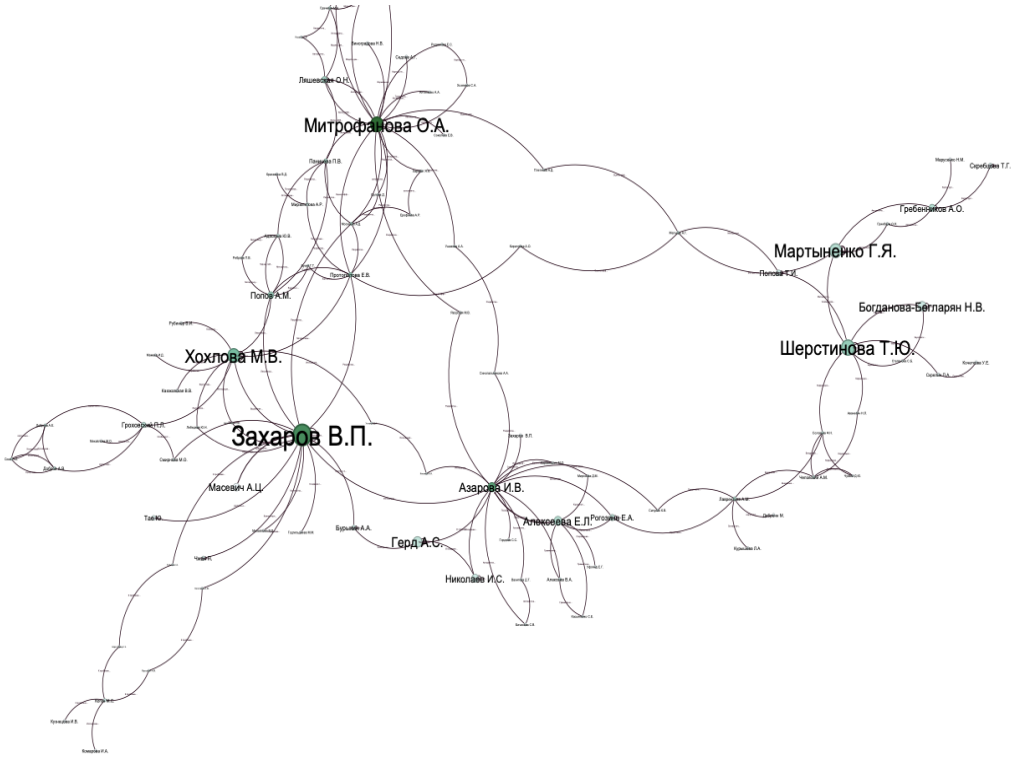

Рис. 7. Пример визуализации связи «автор-автор»

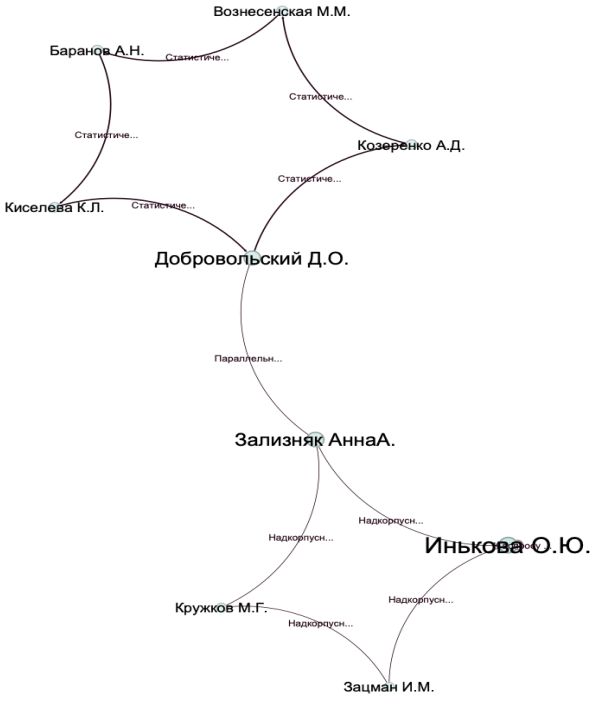

Рис. 8. Пример графа с петлей

Граф связей «автор-автор» повторяет эту структуру: фокус внимания вновь на авторах статей. В отличие от графа «автор-статья», в данной визуализации допустимо наличие графов с петлями, если статья была написана одним ученым без соавторов, но такая визуализация трудночитаема, так как название ребра и названия вершины накладываются друг для друга (рис. 8).

Данная визуализация помогает проследить, как могут быть связаны между собой представители различных научных групп.

Заключительным экспериментом стала визуализация связи «аффилиация-автор». Чтобы вершины можно было отличить от вершин авторов, они были увеличены в размере для наглядности (рис. 9).

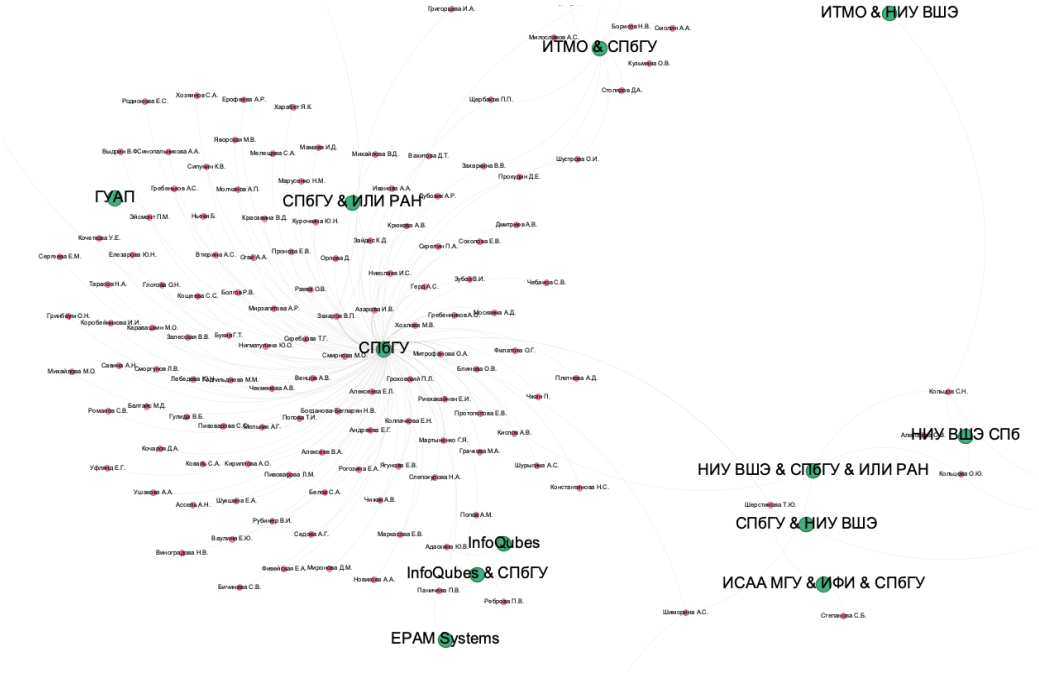

Рис. 9. Пример визуализации связи «аффилиация-автор»

Граф аффилиации центрирован на ведущих вузах Санкт-Петербурга: СПбГУ, ИТМО и ВШЭ. С помощью данного графа можно определить не только ученых, относящихся к той или иной организации, но также и взаимодействие организаций между собой. Некоторые исследователи публиковали статьи с разными аффилиациями, в зависимости от требований конкретной конференции или в связи со сменой места работы. Согласно статистике, из СПбГУ были опубликованы 274 автора, что превышает половину участников. Авторов из ИТМО оказалось 36, на третьем месте Институт русского языка РАН с 31 публикацией и на одну меньше набрала ВШЭ. Однако, почти столько же статей исследователи из ВШЭ опубликовали совместно с авторами из других организаций. Более 80 авторов имели две и более аффилиаций, вплоть до четырех.

На наш взгляд, графы, фиксирующие соотношение элементов метаразметки статей корпуса, могут функционировать не только как способы визуализация, но и как разновидность формальной онтологии. Кроме того, было запланировано размещение корпуса на веб-странице проекта. По этим причинам все графы, созданные при помощи Gephi и PyGraphViz, были выгружены в формате SVG — подвиде XML для графовых файлов, сохраняющем информацию о типе и взаимном расположении элементов. Каждая

вершина графа была снабжена собственной невидимой HTML-ссылкой, уникальной для элемента.

После этого граф был включен в состав веб-страницы, написанной на HTML и CSS. Пользователю доступны: выбор графа для отображения, навигация по графу, поиск любого элемента по названию. Кроме того, при клике мышью на элемент (статью или автора) средствами JavaScript подгружается библиотечная карточка, содержащая дополнительную метаинформацию из числа той, которую невозможно разместить на основном графе. Для статей карточка содержит выходные данные, аннотацию, информацию об авторах, тематические метки статей и т. д. Карточки авторов демонстрируют список статей и соавторов исследователя, его аффилиацию и предоставляют доступ к ссылкам на внешние интернет-страницы с публикациями и информацией об авторе. При клике на элементы карточки, соответствующие узлам графа, пользователь переносится к соответствующему узлу (рис. 10).

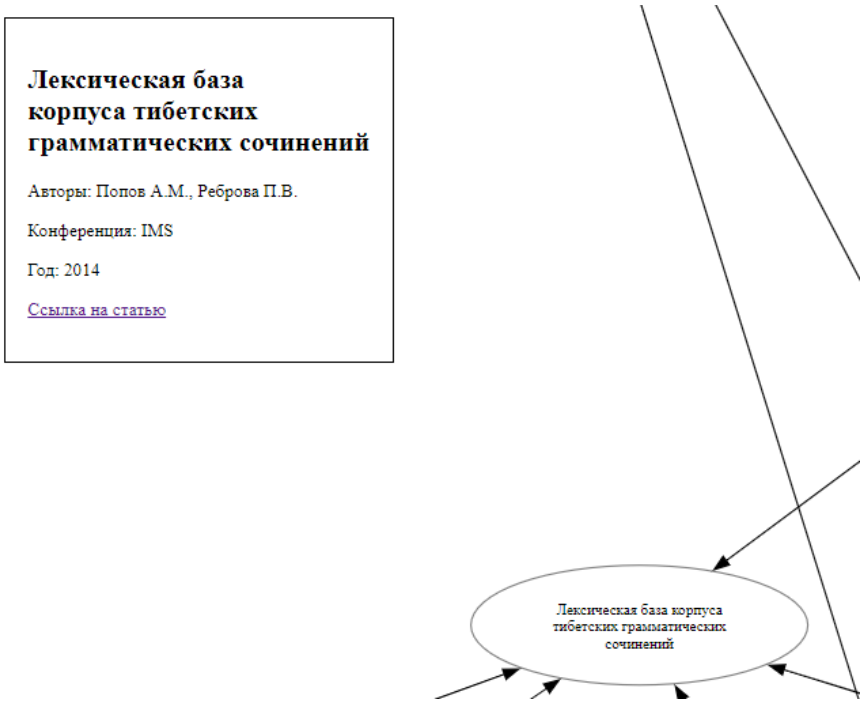

Рис. 10. Пример библиографической карточки статьи на веб-странице метаданных корпуса

Веб-страницы подобного формата представляют собой вид библиотечной онтологии, в которой для навигации используется визуальная составляющая в виде графа, а также ссылочные элементы. Визуализация, на наш взгляд, играет существенную роль в поиске данных, позволяя, например, находить похожие на целевую статьи без дополнительных сложных SQL-запросов, давая возможность избегать громоздких форм. Кроме того, данные об авторах доступны в сжатом виде, что избавляет читателя от необходимости поиска персоналии в интернете. Автоматическое обновление и графа, и HTML-страницы обеспечивают бесперебойное функционирование системы.

5. Заключение

Экстралингвистические метаданные корпуса играют важную роль в любом корпусном менеджере. Они обеспечивают пользователя дополнительной информацией, позволяют

обрабатывать большие данные, а также структурируют корпус и обеспечивают возможность информационного поиска. Работа с метаданными любого корпуса, вне зависимости от его размера, представляет собой трудоемкую задачу, состоящую из нескольких самостоятельных этапов.

В рамках настоящего исследования нами был применен комплексный подход к сбору и применению метаданных для нового корпуса статей по корпусной лингвистике. Основной акцент в исследовании был сделан на пользе метаданных для визуализации экстралингвистической информации о корпусе. Был проведен сравнительный анализ ряда программных средств визуализации формальных онтологий в виде графов. Программы для визуализации формальных онтологий и отношений между сущностями показали свою эффективность для визуализации внешней разметки корпуса, однако конкретный результат значительно зависит от размера базы данных. Несмотря на то, что большинство алгоритмов построения графов становятся менее эффективными с ростом объема данных, некоторые из них способны строить системы отношений между многочисленными сущностями. Среди них выделяются схемы, построенные на алгоритме силового отталкивания, поскольку они одновременно обеспечивают равномерное заполнение пространства и препятствуют наложению элементов друг на друга. Наилучший результат с точки зрения внешнего вида показала программа Gephi. Экстраполируя применение этих методов на еще большие данные, доступные, например, в крупных корпусах русского языка, можно говорить об их практической эффективности.

Визуализация отношений между сущностями экстралингвистической разметки корпуса обеспечивает пользователя еще одним способом навигации по нему. Кроме того, в доступном формате демонстрируются базовые кластеры элементов метаданных корпуса, что дает дополнительную статистическую информацию для любого исследования. Единственным недостатком применения графового метода является большой размер итогового изображения. В полном виде и на максимальном отдалении она становится нечитабельной. В настоящей работе эта проблема решается за счет двух факторов. Во-первых, графовые файлы могут масштабироваться, что позволяет сосредоточиться на отдельном участке данных. Во-вторых, чтобы облегчить переход к нужному участку, граф содержит интерактивные возможности поиска, позволяющие легко найти нужную статью или автора и моментально перейти к ним. Кроме того, важным преимуществом графовых библиотек (прежде всего, PyGraphViz) является возможность автоматизации построения визуализации. Данные для построения схемы подгружаются из базы автоматически и могут быть отфильтрованы образом, удобным пользователю — например, по определенным полям, по части данных или пользовательскому подкорпусу.

Практическим результатом исследования представлены в виде веб-страницы, составленной с помощью средств HTML, CSS и скриптов на Python и JavaScript, обеспечивающая онлайн-доступ к результатам визуализации, интегрированная с репозиторием хранения корпуса. При этом метаинформация, признанная неподходящей для использования внутри графов, доступна в интерактивном режиме в виде библиографических карточек для каждой статьи и справочных материалов для каждого автора непосредственно на веб-странице, где размещен граф. Это позволяет пользователям получать доступ к максимальном объему экстралингвистической информации онлайн и в удобном формате. В перспективе планируется размещение материалов в открытом доступе на сайте СПбГУ.

Исследование показало применимость графового метода визуализации корпусной метаинформации в корпусах небольшого размера. В практической плоскости результаты графовой визуализации показали потенциал для встраивания в более крупные структуры — такие, как корпусные менеджеры.

Однако, следует отметить, что научная работа в этом направлении не завершена и может быть продолжена. Среди потенциальных направлений можно рассмотреть увеличения объема данных как за счет добавления новых метаданных, так и за счет работы с корпусами

большего размера, улучшение укладки графов, а также применение графового анализа для работы не только с внешней, но и с лингвистической разметкой материалов корпуса. Учитывая растущую роль интерактивных интернет-ресурсов в жизни ученых, мы вправе ожидать новых исследований в этом направлении.

Литература

[1] Гладилин С. А. и др. Прототип корпусной платформы нового поколения для НКРЯ // Сборник 28-й международной конференции по компьютерной лингвистике и интеллектуальным технологиям (15–18 июня 2022 г., Москва). М., 2022. С. 1043–1054.

[2] Захаров В. П., Богданова С. Ю. Корпусная лингвистика. СПб., 2020. 234 с.

[3] Лебедев С. В., Нгуен Н. Т., Баймуратов И. Р., Жукова Н. А. Анализ средств визуализации OWL-онтологий // Труды 5-ой Международной научной конференции «Технологическая перспектива в рамках евразийского пространства: новые рынки и точки экономического роста». Санкт-Петербург, 7–8 ноября 2019 г. СПб.: Центр научно-производственных технологий «Астерион», 2019. С. 273–274.

[4] Славута Т. А. Национальный корпус русского языка как информационно-библиографический ресурс // Материалы Всероссийской научно-практической конференции «Динамика библиотечно-информационного обеспечения образования, науки и культуры». Омск: Омский государственный технический университет, 2020. С. 175–181.

[5] Укладка графа // Большая российская энциклопедия. М.: Большая российская энциклопедия, 2017. URL: https://bigenc.ru/c/ukladka-grafa-09f2e3 (дата обращения: 14.04.2024).

[6] Хоай Л., Тузовский А. Ф. Использование онтологии в электронных библиотеках // Известия Томского политехнического университета. 2012. Т. 320, № 5. С. 36–41.

[7] Bastian M., Heymann S., Jacomy M. Gephi: An Open Source Software for Exploring and Manipulating Networks // Proceedings of the International AAAI Conference on Web and Social Media, 2009. 2009. Vol. 3 (1). P. 361–362.

[8] Dash N. Language Corpora Annotation and Processing. Singapore: Springer, 2021. P. 71–90.

[9] Jacomy M., Venturini T., Heymann S., Bastian M. ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software // PLOS One. 2014. Vol. 9 (6). URL: https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0098679&type=printable (дата обращения: 14.04.2024).

[10] Powell J., Collins L., Martinez M. Semantically Enhancing Collections of Library and Non-Library Content // D-Lib Magazine. 2010. Vol. 16 (7/8). URL: https://www.dlib.org/dlib/july10/powell/07powell.html (дата обращения: 14.04.2024).

[11] Sivakumar R., Arivoli P. Ontology Visualization Protégé Tools ― a Review // International Journal of Advanced Information Technology (IJAIT). 2011. Vol. 1 (4). URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3429010 (дата обращения: 14.04.2024).

Meta Tagging and Visualization for the Corpora Linguistics Texts Corpora

D. D. Sukhan [1,2], E. A. Pliusnina [1]

[1] Saint–Petersburg State University, [2] Just AI

This paper presents the project representation and visualization results of metadata for a corpora linguistics corpus of articles. The corpus, which was built at the Saint Petersburg State University Computer and Applied Linguistics department under V. P. Zakharov supervision, included papers

texts from the conference «Corpus Linguistics» which were published from 2002 to 2021, the IMS conference workshop «Computational Linguistics and Computational Ontologies» paper texts from 2011 to 2023, as well as some other materials. The author and article data markup format were unified, and an automated algorithm for adding information of metadata has been implemented. Experiments were carried out to visualize connections between metadata elements using graph tools such as Gephi, WebOWL, Protégé, as well as Python programming language libraries PyGraphviz and NetworkX. The visualisation results were analysed, moreover the created graphs search and navigation in the web page format was implemented.

Keywords: corpora linguistics, conferences materials, graph analysis, metatagging, visualization, informational search, ontologies, named entities

Reference for citation: Sukhan D. D., Pliusnina E. A. Meta Tagging and Visualization for the Corpora Linguistics Texts Corpora // Computational Linguistics and Computational Ontologies. Vol. 8 (Proceedings of the XXVII International Joint Scientific Conference «Internet and Modern Society», IMS-2024, St. Petersburg, June 24–26, 2024). — St. Petersburg: ITMO University, 2024. P. 45–60. DOI: 10.17586/2541-9781-2024-8-45-60.

Reference

[1] Gladilin S. A. i dr. Prototip korpusnoi platformy novogo pokoleniia dlia NKRIA // Sbornik 28-i mezhdunarodnoi konferentsii po komp'iuternoi lingvistike i intellektual'nym tekhnologiiam (15-18 iiunia 2022 g., Moskva). M., 2022. S. 1043–1054. (In Russian)
[2] Zakharov V. P., Bogdanova S. Yu. Korpusnaia lingvistika. SPb., 2013. 234 s. (in Russian)
[3] Lebedev S. V., Nguen N. T., Bajmuratov I. R., ZHukova N. A. Analiz sredstv vizualizacii OWL-ontologij // Trudy 5-oj Mezhdunarodnoj nauchnoj konferencii «Tekhnologicheskaya perspektiva v ramkah evrazijskogo prostranstva: novye rynki i tochki ekonomicheskogo rosta». Sankt-Peterburg, 7–8 noyabrya 2019 g. SPb.: Centr nauchno-proizvodstvennyh tekhnologij «Asterion», 2019. S. 273–274. (In Russian)
[4] Slavuta T. A. Natsional'nyi korpus russkogo iazyka kak informatsionno-bibliograficheskii resurs // Materialy Vserossiiskoi nauchno-prakticheskoi konferentsii «Dinamika bibliotechno-informatsionnogo obespecheniia obrazovaniia, nauki i kul'tury». Omsk: Omskii gosudarstvennyi tekhnicheskii universitet, 2020. S. 175–181. (In Russian)
[5] Ukladka grafa // Bol'shaia rossiiskaia entsiklopediia. Moskva: Bol'shaia rossiiskaia entsiklopediia, 2017. URL: https://bigenc.ru/c/ukladka-grafa-09f2e3 (access date: 14.04.2024). (In Russian)
[6] Khoai L., Tuzovskii A. F. Ispol'zovanie ontologii v elektronnykh bibliotekakh // Izvestiia Tomskogo politekhnicheskogo universiteta. 2012. T. 320, № 5. S. 36–41. (In Russian)
[7] Bastian M., Heymann S., Jacomy M. Gephi: An Open Source Software for Exploring and Manipulating Networks // Proceedings of the International AAAI Conference on Web and Social Media. 2009. Vol. 3 (1). P. 361–362.
[8] Dash N. Language Corpora Annotation and Processing. Singapore: Springer, 2021. P. 71–90.
[9] Jacomy M., Venturini T., Heymann S., Bastian M. ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software // PLOS One. 2014. Vol. 9 (6). URL: https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0098679&type=printable (access date: 14.04.2024).
[10] Powell J., Collins L., Martinez M. Semantically Enhancing Collections of Library and Non-Library Content // D-Lib Magazine. 2010. Vol. 16 (7/8). URL: https://www.dlib.org/dlib/july10/powell/07powell.html (access date: 14.04.2024).
[11] Sivakumar R., Arivoli P. Ontology Visualization Protégé Tools — a Review // International Journal of Advanced Information Technology (IJAIT). 2011. Vol. 1 (4). URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3429010 (access date: 14.04.2024).