Известия СПбГЭТУ «ЛЭТИ». 2023. Т. 16, № 5. С. 59–67
LETI Transactions on Electrical Engineering & Computer Science. 2023. Vol. 16, no. 5. P. 59–67

УДК 004.622 Научная статья

https://doi.org/10.32603/2071-8985-2023-16-5-59-67

Методика иерархической разметки текстовых данных на основе онтологического представления сценариев обработки персональных данных

М. Д. Кузнецов

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина), Санкт-Петербург, Россия

mkuznetsov7991@gmail.com

Аннотация. В связи с интенсивной цифровизацией практически всех сфер человеческой деятельности с каждым годом растут объемы собираемых и обрабатываемых персональных данных, которые используются для предоставления различных услуг. Необходимо автоматизировать процесс формализации и структуризации пользовательских соглашений, написанных на естественном языке, так как большинство пользователей соглашаются с их условиями, не осознавая потенциальных последствий ввиду сложности данных документов. В статье предложена методика разметки текстовых данных, которая учитывает возможные семантические связи между элементами разметки и позволяет аннотировать обучающие выборки для текстовых классификаторов. Разработан и апробирован программный инструмент, реализующий предложенную методику. Разработанный инструмент планируется использовать для дальнейших исследований в области формализации пользовательских соглашений.

Ключевые слова: соглашения об использовании персональных данных, методика аннотирования, аннотирование текстовых данных

Для цитирования: Кузнецов М. Д. Методика иерархической разметки текстовых данных на основе онтологического представления сценариев обработки персональных данных // Изв. СПбГЭТУ «ЛЭТИ». 2023. Т. 16, № 5. С. 59–67. doi: 10.32603/2071-8985-2023-16-5-59-67.

Original article

Technique for Hierarchical Markup of Text Data Based on the Ontological Representation of Personal Data Processing Scenarios

M. D. Kuznetsov

Saint Petersburg Electrotechnical University, Saint Petersburg, Russia

mkuznetsov7991@gmail.com

Abstract. Intensive digitalization in all spheres of human activity constantly increases the amount of personal data collected and processed for various services. It is necessary to automate the process of formalization and structuring of user agreements written in natural language, because most users agree with their terms without realizing the potential consequences due to the complexity of these documents. This paper proposes a text data markup technique that takes into account possible semantic links between markup elements and allows annotating training samples for text classifiers. The development and testing of a software tool that implements the proposed methodology has been performed. The developed tool is planned to be used for further research in the field of formalization of user agreements.

Keywords: personal data agreements, annotation technique, text data annotation

For citation: Kuznetsov M. D. Technique for Hierarchical Markup of Text Data Based on the Ontological Representation of Personal Data Processing Scenarios // LETI Transactions on Electrical Engineering & Computer Science. 2023. Vol. 16, no. 5. P. 59–67. doi: 10.32603/2071-8985-2023-16-5-59-67.

Введение. Интернет проникает во все сферы деятельности, повсеместно проводится цифровизация и автоматизация бизнес-процессов, что в свою очередь вызывает колоссальный рост объема обрабатываемых персональных данных пользователей информационных систем. Оборот и использование персональных данных проводится в соответствии с пользовательскими соглашениями, учитывающими требования федерального законодательства, в частности Федерального закона № 152-ФЗ. Они обязывают поставщика цифровых услуг явно указывать аспекты использования, хранения и распространения персональных данных. В то же время, такие юридические документы крайне сложны для понимания и потому редко читаются пользователями, которые, по сути, соглашаются с условиями, не всегда осознавая последствия.

На данный момент ведутся активные исследования, направленные на повышение доступности пользовательских соглашений. Для структуризации и формализации пользовательских соглашений используются методы их онтологического моделирования [1]–[6], применяются нейронные сети для автоматического выявления аспектов использования персональных данных [3], [7] и другие технологии. В наиболее известных публикациях [2], [3] авторы собрали выборку пользовательских соглашений и обучили классификатор, с помощью которого возможно наполнение онтологического представления сценариев использования персональных данных. Однако данное исследование проводилось в 2016 г., до изменения основных принципов обработки персональных данных, закрепленных в Общем регламенте по защите данных от 25.05.2018, действующем на территории стран Евросоюза и ФЗ РФ «О персональных данных» № 152-ФЗ с изменениями и дополнениями от 14 июля 2022.

Результаты другого важного исследования представлены в [8], в нем авторы показали эффективность методов глубокого обучения для выявления различных аспектов использования персональных данных. Однако в этой статье не применяется онтологическое моделирование, что снижает гибкость разработанного подхода.

Применение классификаторов осложняется отсутствием качественных и размеченных обучающих выборок. Кроме того, принимая во внимание, что один из лучших способов формализации – онтологическое моделирование (как указывается в [3], [4]), задача по созданию обучающих выборок становится еще сложнее из-за того, что нет достаточно гибких инструментов, позволяющих описать такой формат разметки.

Онтологическое представление есть граф, однако его можно развернуть в виде дерева, разделив граф на некоторых вершинах и объединив снова на более позднем этапе. Отсюда возникает нетривиальная проблема формирования иерархической разметки текстов.

Статья имеет следующую структуру. Во втором разделе представлена постановка задачи, в третьем обсуждаются особенности и проблемы, которые могут возникнуть при разметке текстовых данных, предлагается методика разметки. Четвертый раздел посвящен проектированию базы данных для инструмента разметки, в пятом приводится описание основных компонентов инструмента и их взаимодействия. В последнем разделе обсуждаются полученные результаты и направления дальнейших исследований в данной области.

Постановка задачи. Для того чтобы автоматизировать процесс формализации и структуризации пользовательских соглашений, написанных на естественном языке, в виде онтологических моделей, необходимы обучающие выборки, сформированные с учетом структуры онтологического метапредставления пользовательского соглашения. Таких обучающих выборок для документов на русском языке нет. В [9] было разработано онтологическое метапредставление пользовательского соглашения, проведено его ручное и автоматическое наполнение данными из набора OPP-115 [2] и валидация. В настоящий момент автор участвует в работе по формированию размеченного корпуса пользовательских соглашений на русском языке на основе результатов, полученных в [8], учитывающего последние требования законодательства РФ. Размеченный набор пользовательских соглашений позволит формировать онтологии для пользовательских соглашений в автоматическом режиме для их дальнейшей обработки, например оценки рисков, связанных с использованием персональных данных. Однако для его получения необходимо разработать методику разметки текстовых данных, удовлетворяющую определенным условиям, а также ее программную реализацию.

Методика иерархической разметки текстовых данных. Базой в предлагаемой методике разметки текста служит онтологическое представление пользовательских соглашений на обра-

ботку персональных данных. Кроме того, разметка текста – процесс интуитивный и организованный по принципу «что вижу, то получаю». Эти условия определяют следующие задачи, которые необходимо решить при проектировании методики:

• онтологическое представление сложно организовать на месте, прямо в тексте;

• разметка текста ограничена с точки зрения информативности; отображение текста таким образом, чтобы были видны и понятны все метки, присвоенные фрагментам текста, сложно;

• разметка текста не должна нарушать его целостное восприятие (или его фрагмента), в противном случае чтение и понимание будет затруднено;

• допустимо пересечение маркированных фрагментов текста.

Онтологическое представление – это прежде всего, графовое представление, и при наложении нескольких базовых слоев разметки с сущностями, которые могут относиться к обоим этим слоям, может возникнуть неоднозначность. Для ее разрешения необходимы дополнительные усложнения интерфейсной части. Такое усложнение может плохо сказаться на восприятии информации пользователем. Кроме того, это неоправданное усложнение еще и программного кода. Решение этой проблемы можно найти на уровне проектирования – предлагается представлять онтологию в виде совокупности непересекающихся иерархических структур. Такое представление более естественно при определении формата разметки. По завершении аннотирования можно будет обратным образом объединить иерархии, полученные в ходе аннотирования, в единую онтологию, тем самым выполнив требование по онтологическому представлению предметной области.

Многослойное аннотирование сложно представить каким-либо образом, отличным от представленного на рис. 1. На данном рисунке показан макет фрагмента аннотации. При таком подходе информация о метке кодируется цветом, она не отделена от текста и представляет с ним одно целое. Использование всплывающих окон и подсказок нецелесообразно, так как они своим появлением будут перекрывать текст, мешая его восприятию. Вместо этого предлагается более статичный вариант отображения в виде наложения нескольких слоев.

Язык гипертекстовой разметки обладает рядом особенностей, которые препятствуют простому решению проблемы пересечения разметки. Ключевой момент в этом – древовидное представление документа, DOM-дерево. Любое пересечение в рамках данной структуры невалидно и, соответственно, не будет работоспособным. Поэтому предлагается в местах начала и окончания аннотированных фрагментов применять разбиение на 3 фрагмента: текст до выделения, текст самого выделения и текст после выделения. При этом элемент документа будет иметь глубину вложенности не более 1 уровня, что фактически означает разворот иерархии в ширину на уровне языка гипертекстовой разметки. Однако построение иерархической структуры разметки невозможно при использовании всего лишь 1 уровня вложенности.

Решение представлено на рис. 2. Расширения глубины иерархии разметки можно добиться с помощью других средств. Поскольку гипертекстовая разметка в данном случае не может быть адаптирована, то иерархия разметки может храниться во вспомогательных структурах данных – стеках. Ассоциировав с каждым элементом разметки такой стек, можно манипулировать уровнями разметки текста без нарушения гипертекстовой разметки. Такой стек можно хранить и использовать по-разному, например встроить непосредственно в html-тег в виде вспомогательного поля.

В формализованном виде структуру разметки можно представить следующим образом. Имеется множество $L = \{l_1, l_2, ..., l_n\}$ всех возможных ме-

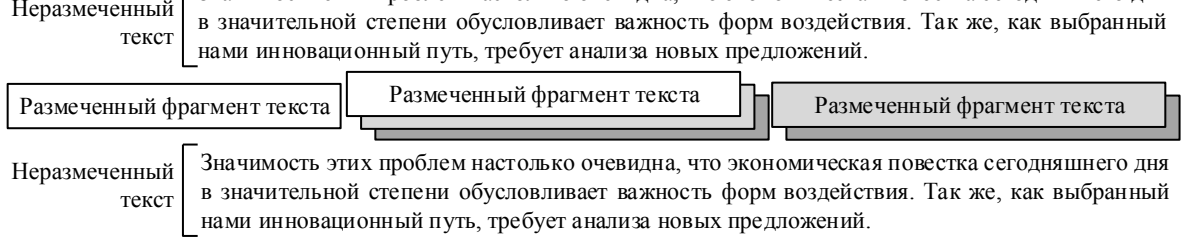

Рис. 1. Макет фрагмента разметки
Fig. 1. Markup fragment layout

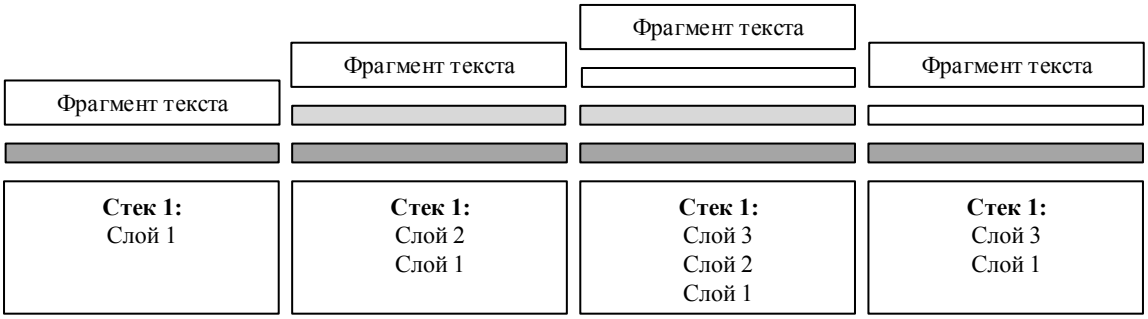

Рис. 2. Схема решения с учетом пересечения разметки
Fig. 2. Scheme of the solution, taking the intersection of the markup into account

ток, обусловленных онтологическим метапредставлением. $L_i'' \in L_j'$ тогда и только тогда, когда L_i'' характеризует атрибуты сущности, описанной меткой L_j' в соответствии с онтологическим метапредставлением. Тогда разметка текста S может быть определена следующим образом: $\text{LabelText} = \{s_i, \{L_i\}\}$, где $s_i \in S$, при этом множество фрагментов текста таково, что $s_i \cap s_j = \varnothing, i \neq j$, а $\{L_i\}$ – множество наборов меток, ассоциированных с фрагментом текста s_i. Таким образом, каждый из наборов $\{L_i\}$ задает список меток из L, относящихся к конкретному фрагменту аннотируемого текста из S, с учетом структуры онтологического метапредставления.

На уровне пользователя предлагаемую методику разметки можно представить в виде последовательности шагов:

• пользователь получает текст для выполнения аннотирования, который передается его клиентской части от сервера;

• пользователь осуществляет аннотирование:

– добавляет слои аннотаций к тексту,

– убирает слои аннотаций с текста;

• пользователь завершает процесс аннотирования;

• клиентская часть приложения формирует структуру данных, отражающую полученный результат разметки и отправляет ее на сервер;

• серверная часть получает структуру данных и производит ее валидацию с точки зрения соответствия заданной структуре;

• по завершении валидации, если структура разметки не повреждена, она сохраняется в базу данных.

Проектирование базы данных инструмента аннотирования. Перед реализацией инструмента разметки было проведено моделирование на разных уровнях – объектном и реляционном. Объектная модель инструмента разметки представлена на рис. 3.

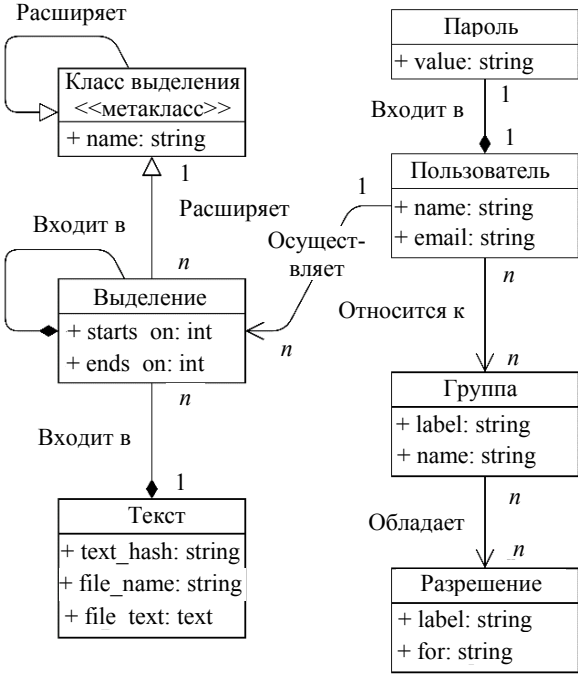

Рис. 3. Объектная модель
Fig. 3. Object model

В соответствии с полученной объектной моделью, ключевыми для процесса аннотирования являются 3 сущности:

1) «Текст» – текст пользовательских соглашений, подлежащих аннотированию;

2) «Выделение» – фрагмент пользовательского аннотирования;

3) «Класс выделения» – классификатор фрагмента аннотирования.

Сущность «Текст» содержит исходные данные для аннотирования – текст пользовательского соглашения. Пользователь, аннотируя соглашение, выделяет фрагменты текста («Выделение») и отмечает их как фрагменты, принадлежащие определенному классу («Класс выделения»). «Класс выделения», в свою очередь, позволяет

сформировать дерево классификации разметки; таким образом, имея координаты фрагмента в тексте и дерево классификации разметки, можно проводить эффективный поиск и анализ размеченных текстов пользовательских соглашений.

Сущности «Пароль», «Пользователь», «Группа» и «Разрешение» также необходимы. Они не относятся непосредственно к аннотированию текстов соглашений, но позволяют идентифицировать лица, выполняющие разметку текста, и разграничивать доступ к тем или иным функциям инструмента аннотирования.

Далее на основе результатов объектного моделирования инструмента аннотирования была построена реляционная модель (рис. 4). Здесь закономерны рекуррентные связи в отношениях «Класс выделения» («selection_class») и «Выделение» («selection»), таким образом в реляционной модели обеспечивается построение иерархических структур, в данном случае – иерархии разметки текста.

Программные компоненты инструмента разметки и их взаимодействие. Приложение было реализовано на основе шаблона проектирования MVC (Model–View–Controller), которое реализует серверную логику инструмента разметки, тем самым связывая все программные части в единую информационную систему.

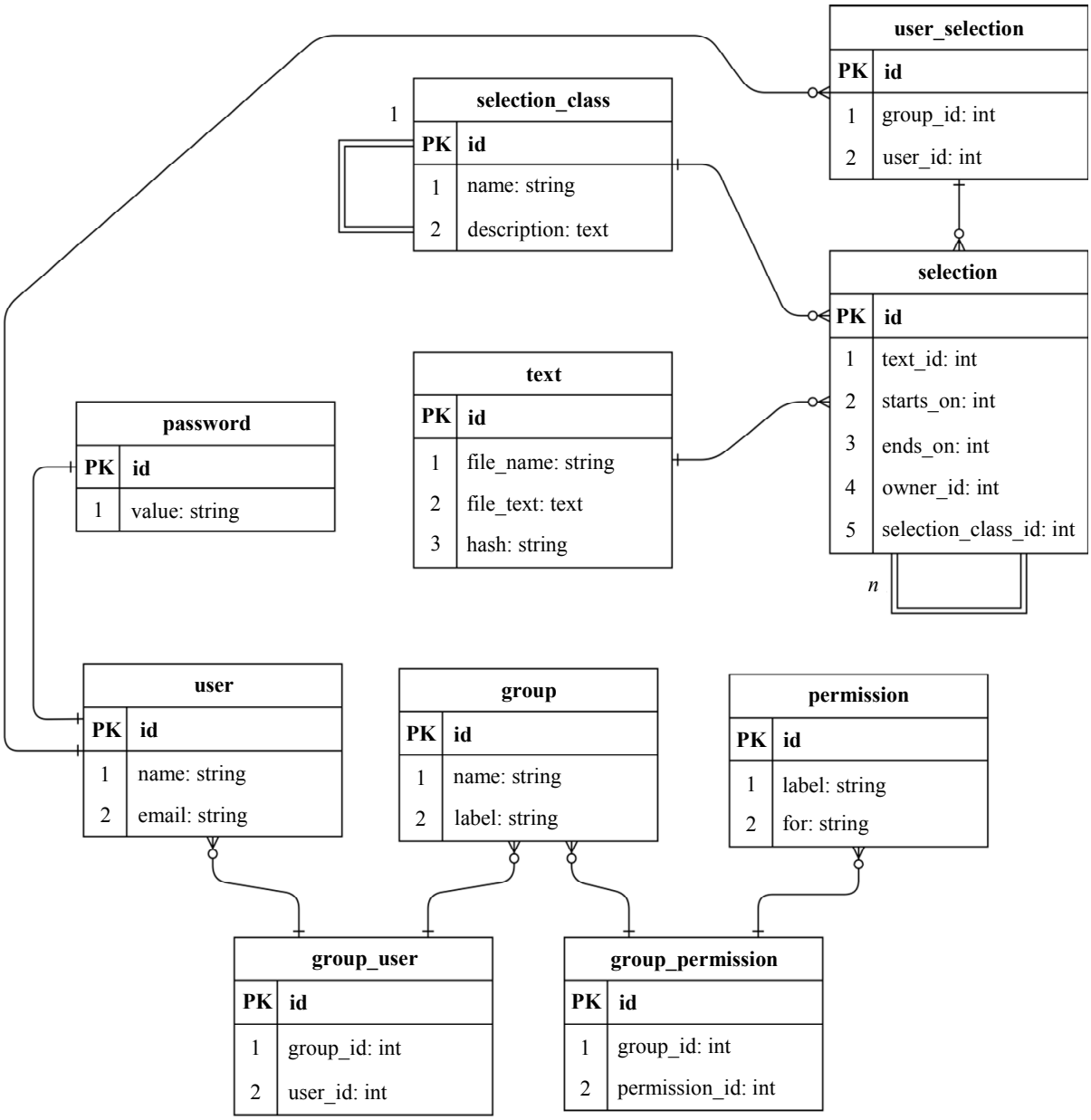

Рис. 4. Реляционная модель
Fig. 4. Relational model

Инструмент разметки набора данных реализован с помощью веб-технологий. Серверная часть полагается на приложение, написанное на языке PHP, которое регулирует порядок выдачи текста для его разметки. Процесс разметки – высокодинамичный, поэтому невозможно избежать использования клиентской части приложения, написанной на языке javascript. Это позволяет сделать работу аналитиков максимально эффективной, т. е. в одну сессию (страница не будет перезагружаться).

В целом схема взаимодействия компонентов приложения непосредственно в процессе аннотирования текста представлена на рис. 5. Клиентская часть приложения для разметки состоит из трех основных элементов: поверхность аннотирования, контейнер слоев разметки и панели управления слоями. Поверхность аннотирования ведет учет пользовательских выделений текста. Контейнер слоев регистрирует новые слои и удаляет старые по запросу, также он предоставляет информацию о слое по его идентификатору. Панель управления слоями предоставляет пользователю возможность добавлять и удалять слои разметки, а также предоставляет информацию о слоях, наложенных на те или иные фрагменты текста. Таким образом, в результате действий пользователя происходит обновление содержимого контейнера слоев, который описывает текущее состояние аннотированных данных. Каждое изменение в контейнере отправляется на сторону сервера и после проверки прав пользователя фиксируется в базе данных. Затем серверная часть отправляет контейнеру со слоями ответ, была ли операция с разметкой проведена успешно. Получив положительный ответ сервера, контейнер со слоями посылает его вниз по цепочке, в результате чего изменение отображается на панели управления слоями и поверхности разметки.

В серверной части отчетливо видна область с реализацией паттерна MVC – виден этап, на котором проводится вызов контроллера, контроллер посредством моделей взаимодействует с базой данных, затем выдает данные на отрисовку в представление. Кроме того, в серверной части используются многочисленные сервисы – маленькие программные пакеты, решающие узкие задачи, например переадресация, контроль доступа и т. д. Все сервисы работают внутри специального контейнера, обратившись к которому, можно получить к ним доступ. Также приложение включает в себя так называемых посредников. Они обеспечивают последовательную обработку запросов вплоть до отправки ответа клиенту.

Пользовательский интерфейс инструмента разметки – один из его ключевых компонентов. Аннотирование – сложный, выматывающий процесс, поэтому важно создать комфортные условия для пользователя. Как можно видеть из рис. 6 и 7, основная идея заключается в разделении материала на 2 колонки: основная колонка содержит в себе текст пользовательского соглашения, а слева – инструмент добавления, просмотра и удаления слоев разметки. На рис. 6 представлен пример добавления слоя разметки посредством выделения фрагмента текста, а слева – инструмент управления

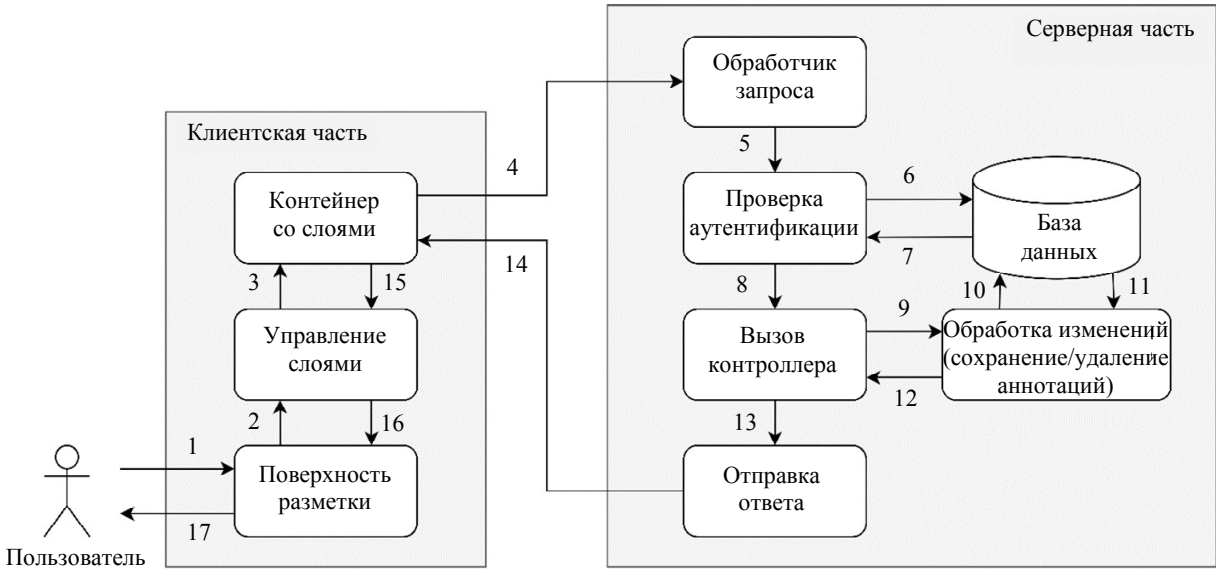

Рис. 5. Компоненты и их взаимодействие
Fig. 5. Components and their interaction

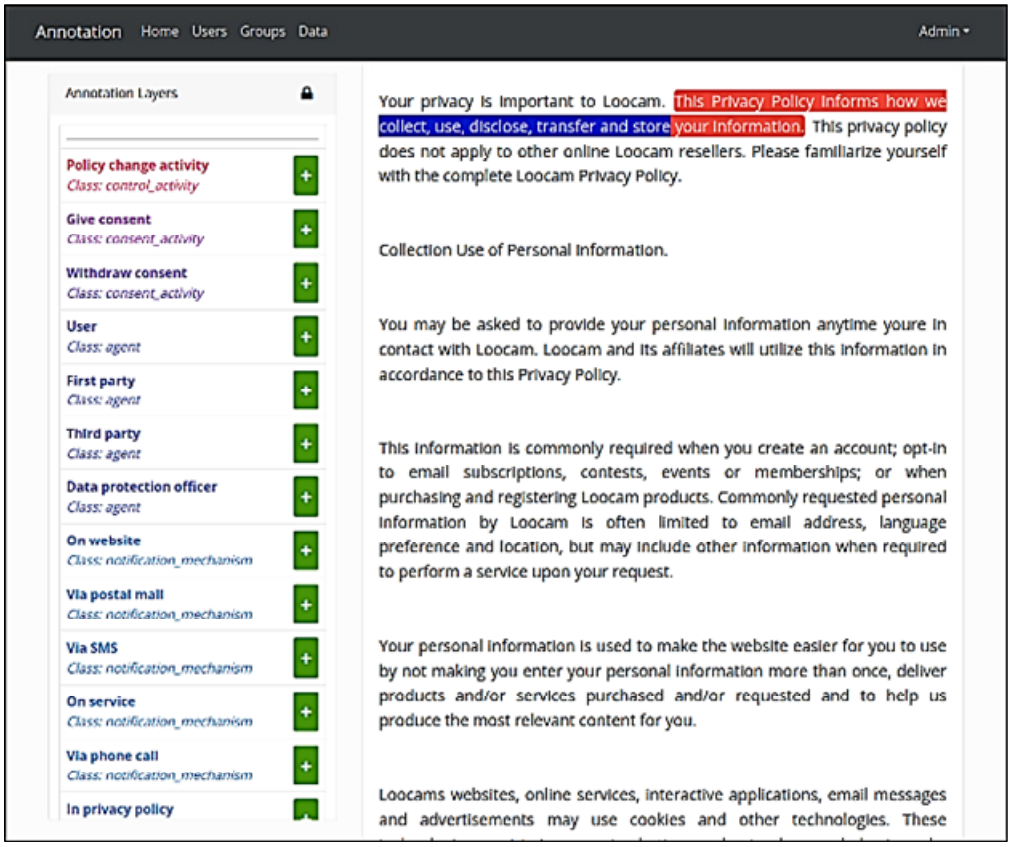

Рис. 6. Добавление слоя разметки
Fig. 6. Adding a new labeling layer

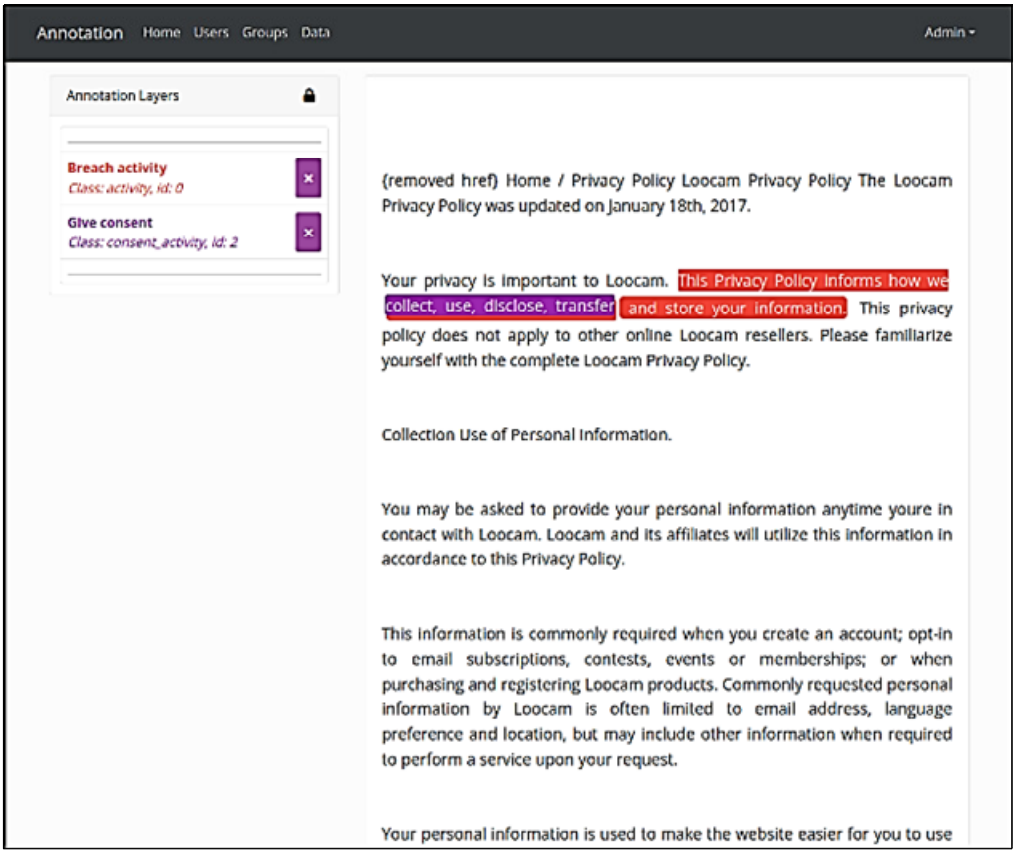

Рис. 7. Удаление слоя разметки
Fig. 7. Removing a labeling layer

слоями разметки, который делает предложение по нанесению какого-либо слоя, в рамках заданной иерархии аннотирования. На рис. 7 представлен пример удаления слоя разметки; слева на рисунке – инструмент управления слоями разметки, который предоставляет возможность снять метку с фрагмента текста.

В инструменте разметки также предусмотрены функции контроля доступа в соответствии с объектной моделью. В приложении предусмотрена глобальная навигация с помощью верхней панели, которая всегда присутствует на экране. В ней же кроме ссылок на страницы приложения присутствует кнопка выхода из учетной записи.

Выводы и заключение. В статье решается задача формализации пользовательских соглашений об использовании персональных данных. Было показано, что в настоящее время отсутствуют размеченные корпуса пользовательских соглашений, которые могут быть использованы в качестве выборок для обучения аналитических моделей анализа текстов. Также было выявлено, что отсутствуют инструменты, позволяющие задавать иерархическую разметку на основе онтологического представления предметной области.

В данной статье автором была предложена методика формирования иерархических аннотаций текстовых данных и выполнена разработка и апробация программного инструмента, реализующего предложенную методику. На данный момент инструмент разметки позволяет задавать иерархию меток любой конфигурации и глубины вложенности.

Разработанный инструмент планируется использовать для дальнейших исследований в области формализации пользовательских соглашений. Ведется работа по формированию корпуса пользовательских соглашений на русском языке, учитывающих последние требования законодательства Российской Федерации. С помощью разработанной методики аннотирования и выборки пользовательских соглашений будет создана обучающая выборка для решения задачи оценки рисков использования персональных данных на основе анализа документов, написанных на естественном языке.

Список литературы

1. Novikova E., Doynikova E., Kotenko I. P2Onto: Making Privacy Policies Transparent, CyberICPS 2020, SECPRE 2020, ADIoT 2020 // Lecture Notes in Comp. Sci. 2020. Vol. 12501. P. 235–252. doi: 10.1007/978-3-030-64330-0_15.

2. The creation and analysis of a website privacy policy corpus / S. Wilson, F. Schaub, A. A. Dara, F. Liu, S. Cherivirala, P. G. Leon, M. S. Andersen, S. Zimmeck, K. M. Sathyendra, N. C. Russell, T. B. Norton, E. Hovy, J. Reidenberg, N. Sadeh // Proc. of the 54th Ann. Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016. P. 1330–1340. doi: 10.18653/v1/P16-1126.

3. PrivOnto: a semantic framework for the analysis of privacy policies / A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, Th. B. Norton, N. C. Russell, P. Story, J. Reidenberg, N. Sadeh // Semantic Web. 2018. Vol. 9(6). P. 185–203. doi: 10.3233/SW-170283.

4. Tang Y., Meersman R. Judicial support systems: Ideas for a Privacy Ontology-Based Case Analyzer // Lecture Notes in Comp. Sci. 2002. T. 3762. P. 800–807. doi: 10.1007/11575863_100.

5. Gharib M., Mylopoulos J., Giorgini P. COPri – A Core Ontology for Privacy Requirements Engineering // Research Challenges in Information Sci. 2020. Vol. 385. C. 472–489. doi: 10.1007/978-3-030-50316-1_28.

6. Gharib M., Giorgini P., Mylopoulos J. COPri v.2 – a Core Ontology For Privacy Requirements // Data & Knowledge Engin. 2021. Vol. 133. P. 101888. doi: 133.101888.10.1016/j.datak.2021.101888.

7. Polisis: automated analysis and presentation of privacy policies using deep learning / H. Harkous, K. Fawaz, R. Lebret, F. Schaub, K. G. Shin, K. Aberer // USENIX Security. 2018. P. 1–22. doi: 10.48550/arXiv.1802.02561.

8. Kuznetzov M., Novikova E., Kotenko I. An approach to formal description of the user notification scenarios in privacy policies // 30th Euromicro Intern. Conf. on Parallel, Distributed and Network-Based Processing (PDP); Special session 2. Valladolid, Spain. 2022. P. 275–282 doi: 10.1109/PDP55904.2023.00049.

9. Privacy Policies of IoT Devices: Collection and Analysis / M. Kuznetsov, E. Novikova, I. Kotenko, E. Doynikova // Sensors. 2022. Vol. 22, № 5. P. 1–23. doi: 10.3390/s22051838.

Информация об авторе

Кузнецов Михаил Дмитриевич – аспирант кафедры информационных систем СПбГЭТУ «ЛЭТИ».
E-mail: mkuznetsov7991@gmail.com
https://orcid.org/0000-0002-0970-8473

References

1. Novikova E., Doynikova E., Kotenko I. P2Onto: Making Privacy Policies Transparent, CyberICPS 2020, SECPRE 2020, ADIoT 2020, // Lecture Notes in Comp. Sci., Springer. 2020. Vol. 12501. P. 235–252. doi: 10.1007/978-3-030-64330-0_15.

2. The creation and analysis of a website privacy policy corpus / S. Wilson, F. Schaub, A. A. Dara, F. Liu, S. Cherivirala, P. G. Leon, M. S. Andersen, S. Zimmeck, K. M. Sathyendra, N. C. Russell, T. B. Norton, E. Hovy, J. Reidenberg, N. Sadeh // Proc. of the 54th Ann. Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016. P. 1330–1340. doi: 10.18653/v1/P16-1126.

3. PrivOnto: a semantic framework for the analysis of privacy policies / A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, Th. B. Norton, N. C. Russell, P. Story, J. Reidenberg, N. Sadeh // Semantic Web. 2018. Vol. 9(6). P. 185–203. doi: 10.3233/SW-170283.

4. Tang Y., Meersman R. Judicial support systems: Ideas for a Privacy Ontology-Based Case Analyzer // Lecture Notes in Comp. Sci. 2002. T. 3762. P. 800–807. doi: 10.1007/11575863_100.

5. Gharib M., Mylopoulos J., Giorgini P. COPri – A Core Ontology for Privacy Requirements Engineering // Research Challenges in Information Sci. 2020. Vol. 385. C. 472–489. doi: 10.1007/978-3-030-50316-1_28.

6. Gharib M., Giorgini P., Mylopoulos J. COPri v.2 – a Core Ontology For Privacy Requirements // Data & Knowledge Engineering. 2021. Vol. 133. P. 101888. doi: 133.101888.10.1016/j.datak.2021.101888.

7. Polisis: automated analysis and presentation of privacy policies using deep learning / H. Harkous, K. Fawaz, R. Lebret, F. Schaub, K. G. Shin, K. Aberer // USENIX Security. 2018. P. 1–22. doi: 10.48550/arXiv.1802.02561.

8. Kuznetzov M., Novikova E., Kotenko I. An approach to formal description of the user notification scenarios in privacy policies // 30th Euromicro Intern. Conf. on Parallel, Distributed and Network-Based Processing (PDP); Special session 2. Valladolid, Spain. 2022. P. 275–282 doi: 10.1109/PDP55904.2023.00049.

9. Privacy Policies of IoT Devices: Collection and Analysis / M. Kuznetsov, E. Novikova, I. Kotenko, E. Doynikova // Sensors. 2022. Vol. 22, № 5. P. 1–23. doi: 10.3390/s22051838.

Information about the author

Mikhail D. Kuznetsov – postgraduate student, Department of Information Systems, Saint Petersburg Electrotechnical University.
E-mail: mkuznetsov7991@gmail.com
https://orcid.org/0000-0002-0970-8473

Статья поступила в редакцию 17.03.2023; принята к публикации после рецензирования 08.04.2023; опубликована онлайн 25.05.2023.

Submitted 17.03.2023; accepted 08.04.2023; published online 25.05.2023.