ИПМ им.М.В.Келдыша РАН

Абрау-2016 • Труды конференции

С.А. Афонин, А.С. Козицын, Д.А. Шачнев

Использование онтологического представления структуры реляционной базы для агрегации наукометрических данных

Рекомендуемая форма библиографической ссылки

Афонин С.А., Козицын А.С., Шачнев Д.А. Использование онтологического представления структуры реляционной базы для агрегации наукометрических данных // Научный сервис в сети Интернет: труды XVIII Всероссийской научной конференции (19-24 сентября 2016 г., г. Новороссийск). — М.: ИПМ им. М.В.Келдыша, 2016. — С. 58-63. — doi:10.20948/abrau-2016-1

Размещена также презентация к докладу

Использование онтологического представления структуры реляционной базы для агрегации наукометрических данных

С.А. Афонин, А.С. Козицын, Д.А. Шачнев

МГУ им. М.В. Ломоносова

Аннотация. Описывается математическая модель агрегатора наукометрических данных, представляющих результаты деятельности отдельных субъектов (персон). Модель основана на использовании онтологической структуры категорий данных и на суммировании значений различных индикаторов, определяющих отдельные составляющие оценки деятельности, для каждой из этих категорий. На основе формул расчёта строятся запросы к реляционной базе данных, на основе которых заполняется таблица с соответствующими формуле результатами деятельности. Модель может быть использована для поиска авторов или коллективов, показавших наиболее эффективные в какой-либо области результаты, а также для сбора статистики о деятельности подразделений. В качестве примера структуры базы данных используется структура информационно-аналитической системы «ИСТИНА» (Информационной Системы Тематического Исследования НАукометрических данных), которая разработана и в течение трёх лет успешно эксплуатируется в МГУ имени М.В. Ломоносова. Механизмы агрегации данных прошли успешные испытания на этой системе.

Ключевые слова: онтологии, семантическая паутина, наукометрические данные, генерация SQL-запросов, агрегация данных.

Актуальность задачи

Существующие системы наукометрии для оценки деятельности субъекта используют, как правило, количество цитирований статей или индекс Хирша. К таким системам можно отнести Elsevier Pure, Thomson Reuters Converis или информационно-аналитические системы некоторых университетов России. В настоящей работе предлагается способ, позволяющий комбинировать *любые* имеющиеся в системе показатели деятельности при помощи формулы, определяющей возможные классы работ, весовые коэффициенты для этих классов, и ограничения на параметры подлежащих учёту работ.

Общая структура онтологии

Информационно-аналитические системы имеют дело с большим количеством данных, в рассматриваемом случае — наукометрических данных. Основная онтологическая связь, представляющая интерес в контексте настоящей статьи, — это отношение *authorOf*, субъектом которой является автор, а объектом — результат деятельности этого автора, например, опубликованная статья или доклад на конференции. Разумеется, каждый объект может быть связан такими отношениями с несколькими субъектами. У некоторых объектов есть родительские объекты, например, для статьи это журнал или сборник, в котором она опубликована, а для доклада на конференции это сама конференция. Физически каждому типу связи автора и объекта соответствует таблица в реляционной базе данных.

Пример множества объектов и связей между ними:

Каждый объект относится к своему классу. Все классы организованы в соответствующее их отношениям дерево классов. Некоторые классы, как правило «верхние», соответствуют отдельным таблицам в реляционной базе данных, а их подклассы определяются на основе правил: например, верхним классом может являться класс «книга», а на основе проставленных пометок конкретная книга может принадлежать к классам «монография» или «учебник». Возможно и обратное, а именно объединение некоторых физических классов в один класс онтологии.

У объекта могут также быть некоторые свойства, тип которых, как правило, либо действительные числа, либо булевы значения. Свойства могут как храниться непосредственно в таблице, содержащей сам объект, так и вычисляться на основе более сложных запросов (например, наличие или количество связей с другими объектами в базе).

Постановка задачи

Пусть даны несколько наборов объектов $\{O^k = O_1^k, \dots O_m^k\}$. Например, *k*-ым набором может являться множество результатов деятельности (далее — работ) некоторого автора. Требуется создать структуру, позволяющую получить для каждого набора некоторый числовой показатель, учитывающий количество объектов в наборе, их свойства и заданные весовые коэффициенты

для каждого класса объектов. Такую структуру будем называть формулой расчёта.

Структура формулы расчёта

Формула состоит из множества строк, каждая из которых определяет некоторое множество объектов и задаёт функцию, сопоставляющую подходящему объекту некоторое действительное число.

Каждая строка формулы представляет собой набор $L = (K^L, R^L, P^L, M^L, \omega^L)$, где

- K^L — класс онтологии;
- R^L — массив ограничений (каждое ограничение R_i^L это функция, сопоставляющая каждому объекту булево значение: 0 или 1);
- P^L — функция-свойство, сопоставляет объекту значение некоторого параметра, являющееся действительным числом;
- M^L — массив модификаторов, функций, принимающих на вход объект и старое значение, и возвращающих новое значение (примеры модификаторов: «разделить на число соавторов» или «ограничить сверху величиной *N*»);
- ω^L — весовой коэффициент, действительное число.

Определение 1. Будем считать, что строка *L подходит* для объекта *O*, если *O* является экземпляром класса K^L, и для каждого ограничения R_i^L выполнено $R_i^L(O) = 1$.

Определение 2. *Функция аппликации* $\alpha(L, O)$ определена как $\alpha(L, O) = \left(M_1^L \circ \cdots \circ M_q^L\right)\left(O, \omega^L \cdot P^L(O)\right)$, если L подходит для O, и $\alpha(L, O) = 0$ иначе.

Пусть теперь $L_1, \dots L_n$ — строки, $O_1, \dots O_m$ — объекты. При подсчёте для каждого из объектов должна выбираться строка, для которой функция аппликации возвращает наибольшее значение. В качестве альтернативного варианта возможен выбор первой по порядку строки, которая подходит для объекта.

Определение 3. *Результатом выполнения формулы* называется число

$$S = \sum_{j=1}^{m} \max_{1 \le i \le n} \alpha\left(L_i, O_j\right).$$

Машинным представлением формулы является массив в формате JSON. Каждым элементом массива является JSON-объект со следующими атрибутами:

- *category* — название класса K^L онтологии в формате *корневой_класс. класс1…классN* — от самого общего, корневого, к самому узкому классу;

- *label* — заголовок, демонстрируемый человеку, на русском языке;
- *restrictions* — массив с ограничениями R^L; каждое ограничение является объектом и содержит название параметра *parameter*, для численных ограничений — значения верхней (*upper_bound*) и/или нижней (*lower_bound*) границы допустимых значений параметра, для булевых ограничений — их требуемое значение *bool_value;*
- *parameter* — название свойства P^L, которое будет учитываться в системе;
- *weight* — значение весового коэффициента ω^L;
- *modifiers* — массив модификаторов M^L.

  Каждый из возможных модификаторов задан отдельным классом на языке JavaScript, который предоставляет название, интерфейс для ввода данных и их текстовое представление. В формуле хранится название класса и массив его параметров.

В формуле также хранятся её название, год начала и год окончания подсчёта. Формула связана также отношениями с пользователем, создавшим её, и с одним или несколькими подразделениями.

Привязки параметров к полям в базе данных бывают нескольких типов, перечисленных в следующем списке.

- Прямое соответствие полю в основной таблице.
- Соответствие комбинации значений полей, например сумме или произведению.
- Соответствие полю в другой таблице.
- Количество записей, удовлетворяющих некоторому условию.

Генерация запросов к базе данных

При выполнении каждой строки формулы генерируется запрос на языке SQL. Каждый запрос возвращает 8 полей:

- ID автора;
- ID объекта работы;
- результат применения функции аппликации для объекта;
- название работы;
- некоторое свойство и его значение: например, «тип курса»: «лекции»;
- количество авторов работы;
- год, к которому относится работа.

Генерация запросов основана на использовании шаблонов запросов и подстановке в них необходимых данных и ограничений. В шаблонах определены названия таблиц в базе данных, соответствия параметров полям и шаблоны для ограничений.

После выполнения запросов для каждой строки их результаты складываются во временную таблицу, затем происходит применение модификаторов и суммирование баллов.

Возможности улучшения генератора

Рассматривается возможность использовать отображение между реляционной базой данных и онтологиями для генерации запросов без использования шаблонов. Для этого планируется создать описания базы на языке R2RML, который является рекомендацией консорциума World Wide Web для задания соответствий между реляционной и онтологической базами данных.

Другой задачей, решению которой могла бы поспособствовать автоматическая генерация запросов, является поиск похожих строк и их объединение в одну строку.

Результаты апробации модели

Формулы расчёта рейтинговых показателей, основанные на предложенной модели, используются в системе «ИСТИНА» с 2015 года. В рамках системы был разработан графический редактор формул, в котором доступны для использования все категории данных, имеющиеся в системе. Категории разбиты на несколько блоков, таких как «Научная работа» или «Учебная работа».

Подсистема разделена на несколько частей в зависимости от статуса пользователя.

- Для ответственных за сопровождение данных доступен интерфейс, позволяющий создавать формулы, редактировать их, привязывать их к подразделениям и должностям, а также публиковать (делать общедоступными). Ответственные могут также смотреть результаты выполнения формулы для каждого сотрудника подразделения.
- Каждый пользователь системы может просмотреть любую опубликованную формулу и посмотреть результат её выполнения для своих работ.

Большинство страниц подсистемы генерируются на стороне сервера при помощи фреймворка Django. Генерацию содержимого на стороне клиента использует только редактор формул, написанный на языке JavaScript.

Система показала свою актуальность и востребованность в рамках МГУ имени М.В. Ломоносова, а также в ряде других организаций, которые приступили к её использованию в своих интересах.

Литература

1. Технология создания тематических интеллектуальных научных интернет-ресурсов, базирующаяся на онтологии / Загорулько Ю.А., Загорулько Г.Б., Боровикова О.И. // Программная инженерия. — 2016. — № 2. — С. 51–60.
2. Интеллектуальная система тематического исследования научно-технической информации (ИСТИНА) / Садовничий В.А., Афонин С.А, Бахтин А.В. и др. // Издательство Московского университета Москва, 2014. — 262 с.
3. Использование онтологий в поисковых системах / Афонин С.А., Козицын А.С. // Материалы Всероссийской конференции «Знания-Онтологии-Теории», г. Новосибирск, т. 2. — 2009. — С. 47–52.