

Применение семантической сети для хранения слабоструктурированных данных

*С.В. Клименков, В.В. Николаев, А.Е. Харитонов, А.В. Гаврилов,
А.Е. Письмак, А.В. Покид*

Национальный исследовательский университет ИТМО, Санкт-Петербург

Аннотация: В данной статье рассматривается применение семантической сети для хранения и извлечения информации, полученной из слабоструктурированных источников. Приводится описание модели семантической сети на основе помеченного ориентированного графа, рассматриваются основные элементы и основные типы связей между элементами, а также определяются основные одиночные и групповые операции с сетью. Кроме того, описывается способ хранения экземпляров концептов и связанных с ними словоформ и словарных определений для автоматического извлечения информации из слабоструктурированных источников. Предложенная семантическая сеть декомпозирована на атомарные концепты, что позволяет обеспечить отсутствие избыточности хранимой информации без применения процедуры нормализации. Предложенная модель была опробована в нескольких практических проектах, на основе опыта применения модель была дополнена и уточнена.

Ключевые слова: семантическая сеть, граф, модель данных, концепт, отношение, атрибут, экземпляр, ассоциация, роль, слабоструктурированные данные.

Введение

С повышением производительности средств вычислительной техники и увеличения объема хранимой информации появилась возможность хранить и обрабатывать гигантский объем накопленной человечеством информации в электронном виде. Часто эти данные представлены как неструктурированные или слабоструктурированные данные в виде текстов. Важным средством представления и хранения знаний стали семантические сети. Для хранения структурированной информации широко используются разнообразные базы данных. Подходы к проектированию баз данных и реляционная алгебра хорошо изучены многочисленными исследователями. Однако сама задача извлечения информации из текстов и осуществление ее долгосрочного и эффективного хранения в виде семантической базы данных является все более широко востребованной [1].

Задача накопления и хранения знаний была актуальна задолго до появления первых компьютеров. С развитием вычислительной техники

большое распространение получили электронные текстовые документы, наиболее привычные для восприятия человеком. Стремление к упорядочению информации привело к созданию структурированных систем хранения данных, особое место среди которых заняли семантические сети.

В статье описывается уточненная и обновленная модель семантической сети, доработанная в процессе решения практических задач по обработке слабоструктурированных данных, рассматривается структура сети, определяются основные элементы сети и связи между ними, а также определяются основные операции над элементами сети и подсетями.

Существует несколько способов построения моделей данных, среди которых наиболее распространены следующие: реляционная, объектно-ориентированная, модель “ключ-значение”, триплетная. Рассмотрим основные ограничения, присущие данным моделям с точки зрения хранения слабоструктурированных данных.

Реляционная модель базируется на понятии отношения, имеет развитый математический аппарат, основанный на теории множеств, и широко используется для построения баз данных [2]. Однако реляционная модель имеет следующие ограничения:

- 1) сложность представления данных с иерархической структурой;
- 2) сложность работы со сложно-структурированными объектами;
- 3) недостаточная универсальность и привязка к предметной области;
- 4) полное устранение избыточности данных в реляционной модели приводит к резкому снижению производительности базы данных;
- 5) допускается существование промежуточных состояний, в которых целостность данных может быть нарушена.

Объектно-ориентированная и объектно-реляционная модель данных позволяет хранить в базе данных объекты, обеспечивая средства доступа к

ним и развитые возможности их обработки [3]. Объектно-ориентированная модель имеет следующие ограничения:

- 1) отсутствие общей математической теории, что приводит к разнообразию вариантов реализации объектно-ориентированной модели;
- 2) отсутствие стандартного языка манипулирования данными;
- 3) жесткая привязка к конкретному объектно-ориентированному языку;
- 4) сложность выявления и устранения избыточности данных.

Модель “ключ-значение” является базовой для NoSQL баз данных, документо-ориентированных хранилищ данных. Ее основным достоинством является простота, производительность, поддержка распределенного хранения [4]. Однако, эта модель имеет следующие ограничения:

- 1) отсутствие явных связей между записями данных и недостаточная структурированность данных;
- 2) слабые механизмы поддержки целостности;
- 3) отсутствие общей математической теории и сложность перехода к другим NoSQL моделям, сложность индексирования по значению.

Триплетная модель по сравнению с моделью “ключ-значение” более универсальна, позволяет задавать связи между данными, но также имеет некоторые ограничения [5]:

- 1) сложность описания объектов;
- 2) слабые механизмы контроля целостности и непротиворечивости данных;
- 3) сложность задания свойств связей (предикатов);
- 4) ограниченное количество типов значений элементов.

Таким образом, из всех рассмотренных современных моделей данных, только реляционная модель имеет развитые средства контроля целостности и избыточности данных, однако и в ней использование указанных возможностей требует дополнительных средств, а именно, проведения процедуры нормализации модели предметной области. Дополнительным

фактором является то, что даже при изменении модели данных нормализованная форма представления данных может быть нарушена, и потребуются повторная нормализация. Кроме того, реляционная модель имеет и другие недостатки, например, отсутствие универсальных средств работы со сложными объектами и иерархическими структурами [6].

Основной проблемой нереляционных моделей является то, что они не имеют развитых средств задания структуры информации. Это не позволяет в полной мере устранить избыточность данных.

При обработке слабоструктурированной информации желательно использовать и семантические характеристики, и онтологические знания о мире, чтобы упорядочить информацию и обеспечить ее размещение в соответствующих элементах семантической сети [7]. Проблему сохранения средств контроля целостности и исключения избыточности данных, можно решить с помощью использования семантической сети в виде графа для реализации структуры данных и хранения экземпляров данных в виде информационных кортежей с привязкой к узлам графа. Эту же сеть можно применить для описания и хранения базовой онтологии, содержащей универсальные понятия, чтобы предоставить общую структуру для интеграции моделей данных для отдельных предметных областей.

Семантическая сеть

Рассмотрим структуру семантической сети. В общем смысле семантическая сеть состоит из множества смысловых понятий или концептов, связанных между собой семантическими отношениями или связями. Концепты и связи могут иметь свойства или атрибуты, характеризующие их. Концепты, связи и атрибуты образуют структурный компонент семантической сети. Кроме этого, концепты, связи и атрибуты могут иметь экземпляры, образующие информационный компонент семантической сети. К информационному компоненту также относятся

связанные с концептами леммы (канонические формы лексем), предназначенные для образования множества словоформ, и глоссы (текстовые словарные определения соответствующего концепту понятия).

Разработанная семантическая сеть может быть представлена в виде помеченного ориентированного графа $N = (S, R)$ (рис. 1а). Множество вершин графа S – это множество узлов смысловых значений или концептов, а **множество ребер R** – множество связей между концептами, являющееся подмножеством декартова произведения: $R \subseteq S \times S \times T$, где T – множество типов связей между концептами, используемое в семантической сети. Связь между концептами представляет собой ребро $r_n = (s, d, t) | s \in S, d \in S, t \in T, n \in \mathbb{N}$, направленное из s в d , помеченное типом t .

Каждому узлу s ставится в соответствие множество его экземпляров I_s . Каждый концепт связан со множеством слов на естественном языке. Для сопоставления концепта с множеством обозначающих его слов естественного языка, каждому узлу s ставится в соответствие множество словоформ W_s , которые являются заранее известными опорными элементами для разрабатываемых алгоритмов. В данное множество также записываются типичные аббревиатуры и сокращения словоформ. Кроме этого, каждому узлу s ставится в соответствие множество словарных определений G_s (рис. 1а).

Вершины графа соединены помеченными ориентированными ребрами. Каждое ребро соответствует связи между концептами, выражающей некоторое семантическое отношение. Связи помечаются типом и в свою

очередь могут иметь атрибуты, характеризующие связь. Рассмотрим возможные типы отношений между понятиями.

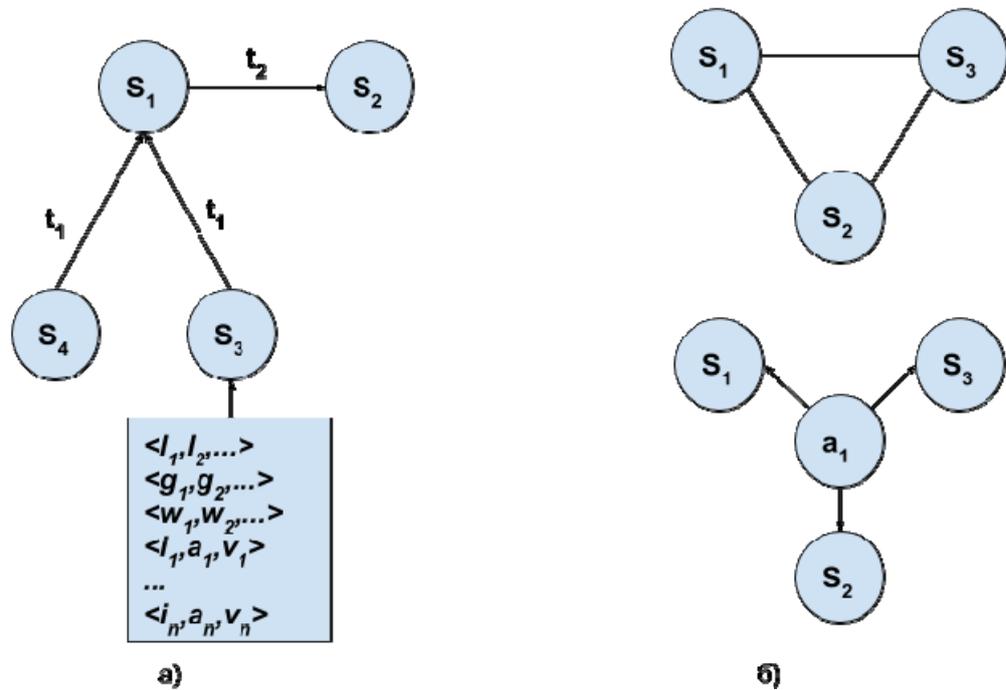


Рис. 1. Элементы сети

Таксономические или гипонимические связи определяют иерархические отношения между концептами, а именно представляют из себя связь, характеризующую отношение типа “is-a”, определяющее принадлежность концепта некоторому классу. Концепт более общего типа, лежащий выше по иерархии, называется гиперонимом, концепт, лежащий ниже по иерархии – гипонимом.

Пусть $h \in R$ – таксономическая связь. Тогда она обладает следующими свойствами:

- 1) Транзитивность. Если y является гипонимом концепта x , а z - гипонимом концепта y , то z также является гипонимом концепта x .

$$x, y, z \in S, e_1(x, y, h) \in KR, e_2(y, z, h) \in R \rightarrow e(x, z, h) \in R$$

2) Антирефлексивность. Концепт не является ни гипонимом, ни гиперонимом самого себя.

$$e(x, x, h) \notin R$$

3) Антисимметричность. Если y является гипонимом концепта x , то x не может являться гипонимом y .

$$x, y \in S, e_1(x, y, h) \in R, e_2(y, x, h) \notin R$$

Иерархические отношения, заданные гипонимическими связями, на каждом уровне иерархии характеризуются двумя множествами признаков - общими и классифицирующими, отличающие их друг от друга. Гипонимическая связь помечается классифицирующими признаками. Тогда для любого концепта путем прохождения по связям, соединяющим его с гиперонимами, мы получим множество классифицирующих признаков. А проходя по пути от начальной вершины графа до гиперонима концепта, мы получим множество общих признаков.

В эту же иерархию встраиваются концепты, связанные отношениями синонимии и антонимии. Полные синонимы, имеющие одинаковые значения во всех смыслах, имеют один и тот же представляющий их концепт. Частичные синонимы, у которых есть различия в концептах, можно считать гипонимами, классифицирующими признаками которых будут различия в их значениях. Антонимы – это понятия с общим гиперонимом, различающиеся по признаку, обычно принимающему противоположные значения.

Меронимические связи определяют отношения между концептами типа “has-a”. Такие связи обладают свойствами конъюнктивности и антисимметричности. Концепт типа, представляющий целое, называется холонимом, концепт, представляющий часть, называется меронимом. Меронимические связи, в свою очередь, подразделяются на следующие подтипы - связь “коллекция-элемент”, связь “агрегат-элемент”, связь

“композит-элемент”. Транзитивность и рефлексивность меронимических связей соблюдается только для связей того же подтипа.

Концепты семантической сети соответствуют обобщенным абстрактным понятиям или типам. Экземпляры соответствуют конкретным объектам и характеризуются уникальным идентификатором. Обычно идентификатором является наименование экземпляра. Экземпляры связаны с соответствующими им концептами классифицирующей связью типа “instance-of”.

Связи между экземплярами связанных концептов представляются в виде множества кортежей, каждый из которых содержит общие идентификаторы экземпляров. В случае, если связь имеет собственный атрибут, то этот атрибут также включается в кортеж.

Атрибут представляет собой внутреннее свойство концепта, то есть свойство, присущее концепту вне зависимости от других концептов. Так как атрибуты концептов в свою очередь также являются концептами, то атрибут a концепта s может быть представлен бинарной связью между концептом s и концептом a . Например, у концепта *Автомобиль* может быть свойство *Цвет*, которое также является концептом сети. Для хранения значений свойств у экземпляров необходимо хранить отдельное множество A_{i_s} , которое, например для экземпляра автомобиля “ХТА21214021686733” (VIN номер) содержит экземпляр цвета “Синий” в виде пары $a_{i_s} = \{i_s, i_p\} | \forall i_s \in I_s, i_p \in I_p, \exists ref(s, p, attr)$.

Значения атрибута для каждого экземпляра представляются в виде множества кортежей $\langle i, a, v \rangle$, каждый из которых содержит идентификатор концепта, идентификатор атрибута и значение (рис. 1а).

Кроме атрибутов, которые характеризуют единичный концепт и фактически являются бинарными связями между концептом и его атрибутом,

существуют n -арные ассоциативные связи между несколькими концептами. Эти связи представляют собой внешние свойства или роли. Концепт обладает внешними атрибутами исключительно по отношению к некоторому внешнему концепту. Например, концепт *Здание* имеет атрибут *Номер* по отношению к концепту *Улица*. Такие связи можно обобщить как ролевые или функциональные характеристики, которые концепты имеют в определенной ситуации. Так как n -арные связи усложняют граф, их можно преобразовывать в множество из n бинарных связей. Тогда в сеть вводится дополнительный концепт, представляющий собой псевдоконцепт-ассоциацию a , связанный с концептами-ролями s бинарными отношениями (рис. 1б).

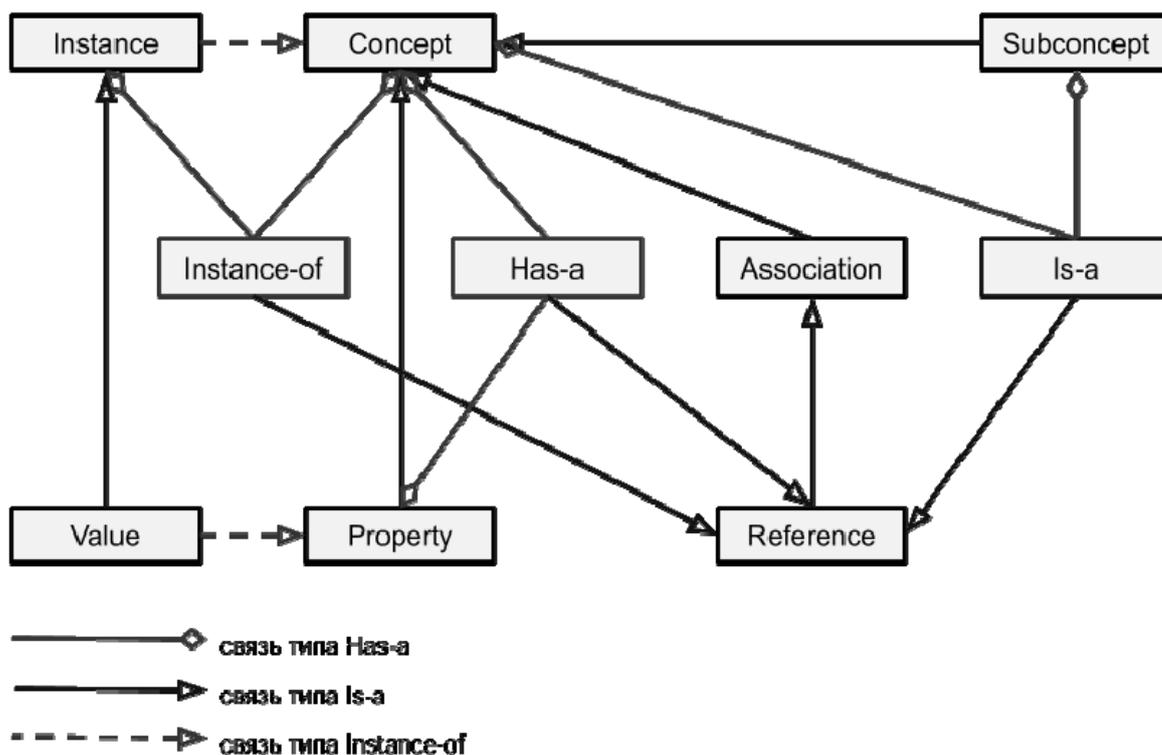
Метамодел ь сети и принципы построения

Метамодел ь семантической сети представляет основные элементы семантической сети, используемые для представления самой сети (рис. 2). Основным элементом является *Концепт* (*Concept*), представляющий произвольный концепт. Данный элемент имеет три потомка (гипонима) - *Ассоциация* (*Association*), *Субконцепт* (*Subconcept*) и *Свойство* (*Property*). *Ассоциация* представляет собой n -арное отношение, подтипом которого является элемент *Связь* (*Reference*), представляющий собой бинарное отношение. Так как в разработанной сети n -арные отношения для упрощения построения зависимостей и реализации операций преобразуются в множество бинарных отношений, то напрямую для формирования сети ассоциации не используются, они заменяются множествами бинарных отношений с псевдоконцептом, представляющим ассоциацию.

Бинарное отношение может быть трех видов:

- 1) Отношение Is-A – отношение между концептом и его гипонимом.
- 2) Отношение Has-A – отношение между концептом и свойством.
- 3) Отношение Instance-Of – отношение между концептом и экземпляром.

Элемент *Экземпляр* (*Instance*) представляет собой экземпляр концепта *Concept*. Подтипом элемента *Instance* является *Значение* (*Value*), который в



свою очередь является экземпляром концепта *Property*.

Рис. 2. Мета модель семантической сети

Каждый концепт имеет связанное с ним множество экземпляров, представленных в виде множества кортежей. Каждый кортеж содержит идентификатор концепта, идентификатор экземпляра, значение экземпляра (обычно наименование). Для удобства представления идентификаторы концепта и экземпляра объединяются в один общий идентификатор, содержащий и идентификатор концепта, и идентификатор экземпляра, разделенные символом "точка".

При построении семантической сети все связи между концептами могут быть сведены к множеству бинарных отношений между парами

концептов. Отношения между концептами в этой паре содержат единственную функциональную зависимость, Так как других функциональных зависимостей в данном бинарном отношении нет, можно считать, что отношение автоматически нормализовано.

Если для некоторого экземпляра существует более одного атрибута, и эти атрибуты характеризуются одним множеством экземпляров, но при этом часть значений атрибутов отсутствует, то атрибут, имеющий для непустого множества экземпляров пустое значение, должен быть выделен в отдельный концепт.

Атрибуты и меронимические связи наследуются гипонимами. Таким образом, для исключения избыточности представляемой информации, атрибуты и все связи типа “has-a” должны присваиваться концепту, максимально высоко расположенному в иерархии, все еще обладающему этими атрибутами и связями.

Операции с сетью

Для работы с семантической сетью определен набор операций, аналогичных тем, которые применяются при работе с графами и множествами. Однако эти операции имеют свои особенности в связи с вышеописанной структурой семантической сети. Ниже приводится высокоуровневое описание некоторых операций.

Сначала рассмотрим понятие эквивалентности концептов. Пусть имеются два несвязанных графа X и Y , и два концепта: s_x , являющийся узлом графа X , и s_y , являющийся узлом графа Y . Пусть функция $sense(c)$ определяет понятие, которое обозначает концепт c . Будем считать, что концепты s_x и s_y эквивалентны, если значение $sense(x)$ равно значению $sense(y)$:

$$s_x \sim s_y, s_x \in X, s_y \in Y: sense(x) = sense(y)$$

Поиск метода определения эквивалентности концептов является отдельной задачей. В литературе предлагаются разные способы определения критериев эквивалентности узлов семантической сети. Среди них есть методы, основанные на сингулярном разложении матриц смежности графа [8], на векторном представлении понятий [9], на основе лексической эквивалентности словарных определений понятий [10]. В процессе реализации практических проектов на основе разработанной модели семантической сети авторами был опробован метод определения эквивалентности концептов на основе сравнения топологий, при котором концепты считаются эквивалентными, если они имеют идентичный набор связей с другими концептами [11]. Кроме того, был исследован метод определения эквивалентности концептов, основанный на точном совпадении словоформ концептов в разных языках [12]. Все вышеупомянутые методы обеспечивают приемлемые результаты и позволяют использовать их в практических целях. В данной статье не уточняется конкретный метод определения эквивалентности концептов.

Добавление нового концепта в семантическую сеть производится в несколько этапов. Сначала создается новая вершина сети, соответствующая вводимому смысловому понятию. Далее в соответствии с семантическими критериями осуществляется поиск смысловых вершин, связанных с добавляемым концептом, которые есть в семантической сети, и строятся связи соответствующих семантическому критерию типов. Семантический критерий определяется исходя из предметной области и знаний о вводимом понятии. При добавлении концепта может потребоваться реконструкция связей, если добавляемый смысл должен быть включен в уже существующие отношения между концептами, то есть, если концепт является транзитивным по отношению к уже существующим – в таком случае связь между

исходными концептами заменяется парой связей между исходными концептами и добавляемым.

При удалении существующего концепта в семантической сети производится поиск вершины, отображающей требуемый смысл. Когда вершина найдена, осуществляется поиск инцидентных вершин и удаление соответствующих связей. При этом собирается информация о смыслах, имеющих транзитивные связи через удаляемый концепт, чтобы после удаления понятия заменить пару связей на одну. Заключительным этапом операции является удаление вершины вместе со словоформами и реконструкция связей между оставшимися вершинами.

Если концепты эквивалентны друг другу, то они склеиваются в один концепт, представляющий изначальные концепты. Определим функцию *equiv* для склеивания концептов:

$$equiv(s_1, s_2) = s, \text{ если } s_1 \sim s_2$$

Объединение двух непересекающихся подсетей – это подсеть, для которой выполняются следующие утверждения:

1. множество концептов объединения подсетей S является объединением множеств концептов исходных подсетей с применением операции отождествления вершин для эквивалентных концептов:

$$S = S_1 \cup S_2 = \{equiv(s_1, s_2) \mid s_1 \sim s_2 \wedge s_1 \in S_1 \wedge s_2 \in S_2\} \\ \cup \{s_1 \mid s_1 \in S_1 \wedge (\forall s'_1 \in S_2: s'_1 \neq s_1)\} \cup \{s_2 \mid s_2 \in S_2 \wedge (\forall s'_1 \in S_1: s'_1 \neq s_2)\}$$

2. множество связей объединения подсетей R является объединением множеств связей исходных подсетей:

$$R = R_1 \cup R_2$$

Объединение концептов производится следующим образом. Пусть есть семантическая сеть и единичный концепт, не принадлежащий данной сети. Для концепта определен эквивалентный концепт, принадлежащий сети.

Тогда объединением сети и концепта будет сама сеть. Пусть имеется множество концептов, не принадлежащих данной сети, соединенных связями. Для каждого концепта имеется эквивалентный концепт, принадлежащий сети. Часть связей между концептами отсутствует в семантической сети. Тогда объединением сети и множества концептов будет семантическая сеть, в которую добавлены недостающие связи. Пусть в объединяемом множестве концептов имеются такие, для которых нет соответствующих эквивалентных концептов в сети. Тогда объединением сети и множества концептов будет сеть, в которую добавлены недостающие концепты.

Пересечением двух подсетей является сеть, для которой выполняются следующие утверждения:

1. множество концептов S пересечения подсетей является пересечением множеств концептов исходных подсетей с применением операции отождествления вершин для эквивалентных концептов:

$$S = S_1 \cap S_2 = \{equiv(s_1, s_2) \mid s_1 \sim s_2 \wedge s_1 \in S_1 \wedge s_2 \in S_2\}$$

2. множество связей R пересечения подсетей является пересечением множеств связей исходных подсетей:

$$R = R_1 \cap R_2$$

В результате операции пересечения семантических подсетей получается сеть, содержащая концепты, общие для исходных подсетей. Соответствие концептов определяется на основе критерия эквивалентности. Пересечение семантических сетей осуществляется путем поиска эквивалентных концептов в подсетях. Найденные концепты образуют вершины новой семантической сети – результата операции пересечения. Для каждого эквивалентного понятия создаётся множество словоформ, которое образуется путем добавления словоформ из соответствующих концептов

изначальных сетей. Далее осуществляется построение семантических связей между концептами.

Разность двух семантических сетей является сетью, в которую входят концепты и связи, встречающиеся в первой семантической сети и не встречающиеся во второй. Соответствие концептов определяется на основе критерия эквивалентности. Разность двух семантических сетей выполняется путем поиска эквивалентных концептов в подсетях. В результирующую сеть войдут концепты, для которых не найдены эквивалентные понятия во второй семантической сети:

$$S = S_1 \setminus S_2$$

Для концептов, образующих вершины разности двух семантических сетей, сохраняются связи, присутствовавшие в первой семантической сети между исходными смысловыми понятиями (для которых не найдены эквиваленты во второй сети), а также словоформы. Связи, для которых присутствуют эквиваленты во второй сети, удаляются:

$$R = R_1 \setminus R_2$$

Симметрическая разность двух сетей представляется новой сетью, состоящей из уникальных концептов и связей, присутствующих в одной из двух исходных сетей. Эквивалентные концепты двух исходных сетей и инцидентные связи удаляются:

$$S = S_1 \Delta S_2; R = R_1 \Delta R_2$$

Заключение

Разработанная сеть в основном необходима для создания универсальной модели на основе базовой онтологии, и дальнейшей интеграции в единое целое семантических сетей из различных источников, имеющих разные форматы представления информации. Базовая сеть при этом должна быть заведомо корректной, прошедшей ручную валидацию.

Базовая онтология верхнего уровня представляет собой граф $U = (S_0, R_0)$, где S_0 - множество вершин графа, представляющих элементарные концепты, R_0 - множество ребер графа, представляющих элементарные связи. Далее к базовой сети добавляются подсети, представляющие собой множества концептов предметных областей, которые объединяются с базовой на основе критерия эквивалентности концептов. Экспериментально показано, что таким образом можно добиться повышения связности сети примерно на 20% [11].

Таким образом, разработана семантическая модель базы данных, в качестве структурного компонента которой используется семантическая сеть, а в качестве информационного компонента – множества экземпляров, словоформ и словарных определений, связанные с элементами сети; определены основные операции с отдельными элементами семантической сети, а также операции с подсетями. На базе данной модели разработано программное обеспечение на языке C++, которое было использовано для получения ряда практических результатов.

Для решения задачи валидации адресной информации был предложен алгоритм автозаполнения на основе семантической модели базы данных, который позволяет определять по строке ввода и требуемому концепту множество экземпляров, которые полностью или частично подпадают под описание [13]. Алгоритм использует топологию подсети и определяет множество экземпляров, подходящих для описания, по мере спуска по иерархии. На каждом уровне иерархии определяются ссылки на словоформы в строке ввода. Упорядочение позиций найденных понятий в тексте позволяет уменьшить размер подстроки, в которой ищется экземпляр концепта.

Для решения задачи автоматического извлечения информации из объявлений о недвижимости был разработан программный модуль, производящий автоматическую обработку текста с отображением его частей на концепты семантической сети, с выделением и сохранением значимой информации [14]. Это позволило агрегировать информацию об объектах недвижимости на основе содержащегося в ней смысла.

Семантическая сеть была использована для решения задачи извлечения ценовой информации из электронных таблиц [15]. При этом сеть применяется для описания структуры объекта, в то время как экземпляры предоставляют шаблоны для данных. Разработанная на основе метода программа просматривает каждую строку в электронной таблице, сопоставляет свойства с концептами, и создает объекты в семантической сети. Подход был опробован на корпусе типичных для дистрибуции техники прайс-листов.

Практические результаты показывают, что предложенная семантическая модель базы данных может быть успешно использована для решения задач, связанных с автоматизированным извлечением неструктурированной и слабоструктурированной информации из источников, относящихся к разнообразным предметным областям.

Литература

1. Карташов О.О., Бутакова М.А., Чернов А.В., Костюков А.В., Жарков Ю.И. Средства представления знаний и извлечения данных для интеллектуального анализа ситуаций // Инженерный вестник Дона. 2018. №4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5421
2. Date C.J. Database Design and Relational Theory. — Sebastopol, CA : O'Reilly Media, 2012. 260 p.

3. Ogunlere S.O., Idowu S.A. Comparison Analysis of Object-Based Databases, Object- Oriented Databases, and Object Relational Databases. Asian Journal of Computer and Information Systems. Volume 03, Issue 02, April 2015. pp 52-57.
4. Makris A., Tserpes K., Andronikou V., Anagnostopoulos D. A Classification of NoSQL Data Stores Based on Key Design Characteristics. Procedia Computer Science, vol 97, pp 94-103.
5. Абрамский М.М., Тимерханов Т.И. Сравнительный анализ использования реляционных и графовых баз данных в разработке цифровых образовательных систем // Вестник НГУ. Серия: Информационные технологии. 2018. №4. С. 5–12.
6. Лебедев, С.В. Модель-ориентированный подход к построению связанных данных на основе разнородных источников // Онтология проектирования. 2019. Т.9, №1(31). С. 101-116.
7. Загорулько Ю.А. Современные средства формализации семантики областей знаний на основе онтологий // Информационные и математические технологии в науке и управлении. 2018. № 3 (11). С. 27-36.
8. Мизюков Г.С. Нахождение подобия между неструктурированными объектами данных на основе метода сингулярного разложения спектра графа // Инженерный вестник Дона. 2018. №1. URL: ivdon.ru/ru/magazine/archive/n1y2018/4651
9. Paradis R.D., Guo J.K., Moulton J., Cameron D., Kanerva P. Finding semantic equivalence of text using random index vectors, Procedia Computer Science, vol. 20, pp. 454-459, 2013.
10. Bijaksana M.A., Permadi R.I. WordNet Gloss for Semantic Concept Relatedness. In: Herawan T., Ghazali R., Nawi N., Deris M. (eds) Recent

- Advances on Soft Computing and Data Mining. SCDM 2016. Advances in Intelligent Systems and Computing, vol 549, pp 406-413.
11. Письмак А.Е., Клименков С.В., Цопа Е.А., Слободкин А.Ю., Николаев В.В. Объединение семантических сетей на основе эквивалентности топологий // Известия высших учебных заведений. Приборостроение. 2019. № 1. С. 50-55.
 12. Клименков С.В., Цопа Е.А., Жмылёв С.А., Покид А.В., Ткешелашвили Н.М. Метод быстрого поиска узлов семантической сети по точному совпадению словоформы // Известия высших учебных заведений. Приборостроение. 2017. № 10. - С. 932-939
 13. Яркеев А.С. Использование семантической сети для автодополнения и валидации информации // Сборник тезисов докладов конгресса молодых ученых. Электронное издание. – СПб: Университет ИТМО, 2018. URL: openbooks.ifmo.ru/ru/file/7313/7313.pdf
 14. Пашнин А.Д. Использование семантической сети для извлечения информации о недвижимости // Сборник тезисов докладов конгресса молодых ученых. Электронное издание. – СПб: Университет ИТМО, 2018. URL: openbooks.ifmo.ru/ru/file/7327/7327.pdf
 15. Tkeshelashvili N.M., Klimenkov S.V. Spreadsheet data extraction using semantic network // 19th International Multidisciplinary Scientific GeoConference, SGEM 2019. Vol. 19. № 2.1. pp. 637-644

References

1. Kartashov O.O., Butakova M.A., Chernov A.V., Kostyukov A.V., Zharkov Yu.I. Inzenernyj vestnik Dona. 2018. №4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5421

2. Date C.J. Database Design and Relational Theory. Sebastopol, CA. O'Reilly Media, 2012. 260 p.
3. Ogunlere S.O., Idowu S.A. Comparison Analysis of Object-Based Databases, Object-Oriented Databases, and Object Relational Databases. Asian Journal of Computer and Information Systems. Volume 03, Issue 02, April 2015. pp 52-57.
4. Makris A., Tserpes K., Andronikou V., Anagnostopoulos D. A Classification of NoSQL Data Stores Based on Key Design Characteristics. Procedia Computer Science, vol 97, pp 94-103.
5. Abramskiy M.M., Timerkhanov T.I. Vestnik Novosibirskogo gosudarstvennogo universiteta. Seriâ: Informacionnye tehnologii. 2018. №4. pp. 5–12.
6. Lebedev, S.V. Ontologiâ proektirovaniâ. 2019. Vol 9, №1(31). pp. 101-116.
7. Zagorul'ko Yu.A. Informacionnye i matematiçeskie tehnologii v nauke i upravlenii. 2018. № 3 (11). pp. 27-36.
8. Mizyukov G.S. Inzenernyj vestnik Dona. 2018. №1. URL: ivdon.ru/ru/magazine/archive/n1y2018/4651
9. Paradis R.D., Guo J.K., Moulton J., Cameron D., Kanerva P. Finding semantic equivalence of text using random index vectors, Procedia Computer Science, 2013. vol. 20, pp. 454-459.
10. Bijaksana M.A., Permadi R.I. WordNet Gloss for Semantic Concept Relatedness. Soft Computing and Data Mining. 2016. Advances in Intelligent Systems and Computing, vol 549, pp. 406-413.
11. Pis'mak A.E., Klimenkov S.V., Tsopa E.A., Slobodkin A.Yu., Nikolaev V.V. Izvestiâ vysshih uchebnyh zavedenij. Priborostroenie.. 2019. № 1. pp. 50-55.
12. Klimenkov S.V., Tsopa E.A., Zhmylev S.A., Pokid A.V., Tkeshelashvili N.M. Izvestiâ vysshih uchebnyh zavedenij. Priborostroenie.. 2017. № 10. pp. 932-939.



13. Yarkeev A.S. Ispol'zovanie semanticheskoy seti dlya avtodopolneniya i validatsii informatsii. Sbornik tezisov dokladov kongressa molodykh uchenykh. SPb: Universitet ITMO, 2018. URL: openbooks.ifmo.ru/ru/file/7313/7313.pdf
14. Pashnin A.D. Ispol'zovanie semanticheskoy seti dlya izvlecheniya informatsii o nedvizhimosti. Sbornik tezisov dokladov kongressa molodykh uchenykh. SPb: Universitet ITMO, 2018. URL: openbooks.ifmo.ru/ru/file/7327/7327.pdf
15. Tkeshelashvili N.M., Klimenkov S.V. Spreadsheet data extraction using semantic network. 19th International Multidisciplinary Scientific GeoConference, SGEM 2019. Vol. 19. № 2.1. pp. 637-644.