УДК 510.64: 004.82

Е. А. Хала, г. Киев

НЕЧЕТКИЕ ОНТОЛОГИЧЕСКИЕ ЗНАНИЯ И МЕТОДОЛОГИЯ КЛАСТЕРИЗАЦИИ ДОКУМЕНТОВ

This article describes a hierarchical approach ontological document clustering. Used fuzzy logic control approach for finding suitable instrument cluster. Cased are giving to verify the approach.

Ключевые слова: нечеткий вывод, иерархическая кластеризация, структура онтологии, анализ текста.

Введение. В настоящее время перед компаниями разработчиками встает проблема непостоянности и ограниченности времени при разработке продукта, и все это из-за высокой конкуренции, сложности проектирования и короткого жизненного цикла изделия. Такое положение заставляет компании организовывать исследования с целью получить новые знания, что обогатит и расширит свой стратегический портфель интеллектуальной собственности. В области управления знаниями, кластеризация играет важную роль, она помогает определить будущие исследования и направления развития. Однако современные исследования по кластеризации текстовых документов зависят от статистических методологий, что используют ключевые слова и фразы, которые не обязательно представляют знания, что содержатся в документах. Для обеспечения лучшего решения кластеризации знаний, используется метод онтологического представления знаний и нечеткой логики управления. Онтологическое представление знаний позволяет экспертам в этой области определить знания на постоянной основе, и в целях повышения эффективности обмениваться знаниями с использованием стандартного формата (например, XML, описания ресурсов (RDF), или OWL). Затем нечеткая логика используется на языковых выражениях, для получения мер сходства между текстовыми документами для кластеризации. При поддержке этих двух методов можно получить не только новые шаблоны, но расширить уже существующие.

Обзор литературы

Несколько областей исследования часто упоминаются при выводе методов кластеризации документов. Эти области исследований включают анализ текста, создание онтологии, нечеткую логику и математическую кластеризацию. Глубокий анализ текста выводит структуру документа от последовательностей в тексте на естественном языке, и определяется как процесс анализа текста, чтобы извлечь метаданные или информацию более высокого уровня [1, 2]. Есть много известных исследований в области текстовых данных, например, для получения информации и обработки

естественного языка.

Информационный поиск рассматривает проблему поиска нужной информации из больших источников аккумулирования данных, таких как World Wide Web, интранет и электронные библиотеки. Информационные подходы извлечения часто используют ключевые фразы для индексирования и поиска документов. Например, методология для извлечения ключевых фраз документа, а затем вычисление частоты и получение соотношения между фразами [3]. Интерактивный способ для вывода иерархической структуры документа, где пользователи выбирают слова из больших фраз, в которых они появляются [4]. Алгоритм Sequitur извлекает иерархическую структуру фразы из текста [5]. Алгоритм использует наивную байесовскую статистику, частоту текстового термина, обратную частоту документа и расстояние размещения, чтобы определить ключевые фразы-последовательности, которые в свою очередь, позволяют узнать структуру документа. Алгоритм интеллектуального анализа данных, называемый «Одно значение за один раз», для классификации текстовых документов в непересекающиеся классы [6]. Использование модели Маркова и алгоритма Витерби для извлечения фразы, показывает, что этот подход более эффективен, чем метод, который использует теговые части речи [7].

Для обработки естественного языка используется компьютер, чтобы изучить, как люди обрабатывают и понимают язык. Общий подход заключается в анализе естественного языка, используя грамматику и семантику. Компьютерные программы разбирают текст естественного языка на предложения, используя правила грамматики. Однако, определение значения предложения является трудной и сложной задачей, обусловленой областью и определенным языком. Таким образом, исследователи начинают комбинировать различные подходы и создают онтологии, которые представляют общую структуру знаний для улучшения анализа текста и обработки естественного языка.

Совокупность знаний в интересуемой области представлена объектами, понятиями, сущностями и отношениями между ними. Всемирная паутина постоянно расширяет объем знаний, которые требуют четкой структуры, то есть онтологии, чтобы описать их и сделать доступным для использования. Таким образом, был создан RDF для моделирования метаданных о веб-ресурсах и для формирования онтологии. RDF состоит из модели RDF, ее фундаментального синтаксиса, семантических аспектов, понятий и соответствующего словаря. Основной элемент RDF - тройка: ресурс (предмет) связанный с другим ресурсом (объект) через дугу, маркированную ресурсом (предикат). Это означает, что (субъект) имеет свойство (предикат) ценностью (объект). Компьютеры могут легко делиться знаниями через RDF, и некоторые исследователи используют эту онтологию в качестве подхода улучшить методы анализ текста. Палмер [8] представил алгоритм, основанный на расстоянии, что вычисляет ценности подобия попарных ключевых слов в онтологии. Так же был разработан алгоритм, который

автоматически генерирует онтологию и классифицирует информацию, используя нечеткие нейронные сети; и методология классификации документов, использующая не только автоматически построенную онтологию, но и частоты ключевого понятия документа для классификации. Нечеткая логика обеспечивает исследователей средствами, для подражания правилам классификации экспертов. Грюнингер и Фокс предложили методологию, для облегчения конструкции онтологии и оценки, реализовав его через TOVE (TOronto Virtual Enterprise) проект моделирования.

Знания представлены и хранятся с использованием языка, который регулируется правилами и договоренностями. Эксперты могут находить и обрабатывать знания, благодаря тому что, что они понимают язык и знают правила и договоренности. Поскольку эксперты не всегда последовательны в интерпретации знаний, то они обрабатывают их с разным уровнем точности. Тем не менее, если правила и договоренности экспертов превращаются в математику, то компьютер может быть запрограммирован, чтобы подражать экспертам, последовательно накапливая знания. Например, использовать предопределенную онтологию, чтобы извлечь содержание новостей и применить нечеткую модель вывода, чтобы получить подобие новостей и генерировать сводки новостей [9].

Кластеризация – общий метод, чтобы создать наборы, которые являются довольно гомогенными в пределах групп, но значительно гетерогенными между группами. Математический принцип объединения в кластеры максимизирует различие между группами и минимизирует различие в пределах групп. Методы кластеризации были успешно применены к текстовой обработке. Ранклер и Бездек сгруппировали тексты веб-страниц и последовательности веб-страниц, которые посещают пользователи (блоги). Алгоритм расстояния Левенштейна и нечеткий алгоритм с-средних были совместно применены для создания кластеров. Другой пример кластеризации для анализа и синтеза текста, продемонстрировал Сюем, который использовал подход К-средних для того, чтобы сгруппировать доступные документы.

Исходя из выше сказанного, фразы, извлеченные из документов, часто используются, чтобы установить отношения подобия между текстами документа, и эти отношения подобия используются в качестве основы, для группирования документов. Однако, статистический анализ ключевых фраз не может полностью представлять основу знаний. Эта работа представляет метод проанализировать и сгруппировать документацию, используя схему онтологии заданной области, а не подход имитирования текста ключевой фразой. Для такой методологии необходимо, чтобы эксперты построил онтологическую схему, то есть, структуру знаний для области, и затем обучили систему, используя заданный набор шаблонов. Обработка естественного языка приспособлена, чтобы вывести онтологию доступных документов, а нечеткая логика используется, чтобы получить онтологическое сходство между документами для группирования.

Системная методология

Методология для нечеткой онтологической кластеризации документа (НОКД) описывается следующим образом. Первоначально, эксперты по области определяют онтологию области, используя базу знаний онтологии и инструменты редактирования RDF под названием Protege [5], а также слова и фразы (например, речь, куски и аннотации) сопоставляются с соответствующими понятиями онтологии области. Эксперты также создают учебный набор шаблонов, используя простую и удобную в работе обработку естественного языка и помечая инструмент под названием MontyLingua [6]. После этого, вычисляются вероятности понятий в заданных кусках документа. Вероятности понятия рассчитываются в каждом конкретном документе, а затем используются для группирования шаблонов с нечеткими логическими выводами. Следовательно, иерархический алгоритм кластеризации уточняется, адаптируя нечеткую логику к процессу вывода онтологического понятия.

Строительство доступной онтологии

Первым шагом методологии требуется использовать основанный на базе знаний инструмент редактирования RDF под названием Protge. Инструмент помогает экспертам по области в определении структуры онтологии, используя графическое чередование. Нов и Макгинесс были одними из первых, предложивших использование методологии основанной на технических знаниях для построения онтологии. Protge - структура, в которую другие различные расширения программного обеспечения могут быть легко добавлены и связаны между собой [8]. Благодаря этим особенностям Protge считается подходящим автоматизированным инструментом для развития онтологии. Онтологическая сеть может быть автоматически преобразована в стандартные форматы данных (XML, RDF или OWL) для дальнейшей манипуляции и интерпретации, для анализа знаний и синтеза.

Обработка естественного языка и обучение терминологии

Чтобы измерить знание, содержавшееся в документах относительно определенной структуры онтологии, система обучается, используя ряд шаблонных документов. Предложения из учебных документов помечаются для извлечения частей речи, фрагментов и аннотаций, с помощью инструмента обработки естественного языка MontyLingua. Позже, в таблице сопоставляются извлеченные слова к понятиям онтологии. Система делает запись вероятностей понятий, определяя, что слово подразумевает в шаблоне. Условная вероятность, P (доступное понятие | Слово C в блоке Б корпусов), получена во время учебной сессии.

Чтобы поддержать полноту системы, исследование также включает повторяющийся механизм «переучивания», чтобы включать новые слова, которые не являются частью текущей базы терминологии, когда новый термин обнаружен, он сначала сохраняется в базе данных терминологии. После этого, системный администратор назначает соответствующее

онтологическое понятие этому термину, что позволить системе автоматически повторно вычислить и обновить онтологическую терминологию понятий базы знаний.

Анализатор терминологии

После обработки естественного языка и обучения терминологии, все выведенные понятия предложений случайные. Следовательно, вероятности понятий для каждого блока вычислены.

Извлечение знаний

После анализа терминологии мы вычисляем вероятности понятия для каждого блока. Блоки, подразумевающие понятия как предикаты, первыми входят в онтологию. Рис. 1 показывает, что блок 5 подразумевает два понятия (кандидаты) как предикаты в онтологии.

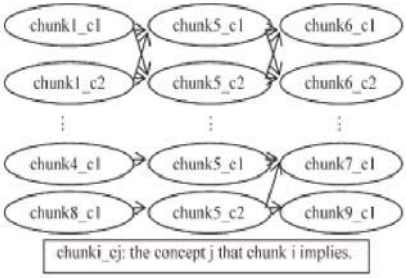

Рис. 1 - Фильтрование значений онтологии

Следующий шаг должен выбрать блоки, которые подразумевают понятия, как предмет в онтологии от предыдущего предложения до следующего предложения. Тот же самый процесс используется, чтобы определить кандидатов объекта. Если есть десять кандидатов на предмет, два кандидата на предикат и десять кандидатов на объект, то есть 200 (10*2*10) заявленных кандидатов. Заявления, которые не существуют в онтологии, устраняются. Наконец, выход генерируется с использованием вероятности, полученной из следующего уравнения:

$$Max_{\text{для всех значечий блока 5}} \times \frac{\text{кандидат}(\textit{субъект})+\text{кандидат}(\textit{педкат})+\text{кандидат}(\textit{объект})}{3} \quad (1)$$

Процесс, описанный ранее, используется для блоков, которые подразумевают понятия предиката в документе онтологии. Таким образом, документ превращается в набор операторов в онтологии. Эти заявления рассматриваются как индексы документа и являются основой сходства при сравнении с другими документами.

Подобные совпадения

Для того, чтобы вычислить подобие между доступными документами, нечеткая логика используется для получения меры сходства. Во-первых, содержание шаблонных документов разделено на набор главных понятий и набор деталей, Табл. 1.

Перед вводом к модели вывода, документы переводят на онтологический формат, включая главные понятия и детали. Главные понятия состоят из троек сверху, а детали состоят из нижних троек

$$S = \frac{\text{TT}}{\text{ST}} \tag{2}$$

где S - мера сходства документа 1 и 2 документа; TT - схожие тройки в документе 1 и 2 документа; ST - суммы троек в документе 1 и 2 документа.

Нечеткие логические представления “многие совпадения”, ”некоторые совпадения ” и “немногие совпадения” определены функциями предледности.

Нечеткая модель вывода Мамдани применяет наследование «если-то» правила к нечеткости входа и выхода. Непринужденность формулировки модели, простого вычисления и ясности в представлении человеческой лингвистики поддерживает выбор этого подхода. Таким образом, нечеткая модель вывода Мамдани использует мин-мин-макс операцию, рассматривая два правила принят и изменен. Изначальная мин-мин-макс операция Мамдани рассматривает подход с двумя правилами, но эта корреспонденция рассматривает девять правил одновременно. Этапы процедуры заключаются в следующем.

1) Вычисляют сходство документов, удовлетворяющих основные понятия (X_{mc}) и сходство документов, удовлетворяющих подробные описания (X_{dd}).

2) Оценка X_{mc} и X_{dd}, используя правила (Табл. 1), чтобы получить соответствующую принадлежность.

3) Сравнить принадлежности и выбрать минимальное значение из двух наборов, соответствующего понятия (высокое сходство, среднее сходство и низкое сходство) для каждого правила.

4) Собрать принадлежности, которые представляют то же самое понятие в одном наборе.

5) Получить максимальную принадлежность для каждого набора и вычислить заключительный результат вывода.

Дефаззификация и объединение в кластеры

Процедуры вывода генерируют принадлежность в представлении различных уровней сходства. Однако, эти значения - все еще нечеткие, и они требуют, выделение процесса дефаззификации, чтобы помочь сгенерировать значения, представляющие сходство документов. Процессы дефаззификации состоят из двух шагов. Первый этап должен решить, какие сходства (“высокое сходство”, “среднее сходство”, и “низкое сходство”) лучше всего представляет отношения между этими двумя документами. На втором этапе основное внимание уделяется преобразованию значения принадлежности из сходства. Подробное преобразование значения схожести изложено в следующих трех случаях.

Таблица 1 - Нечеткие правила для вывода подобия документа

No	Если два документа, состоящие из предложений (полученные как заявления онтологии) с … .	тогда, полное подобие этих двух документов
1	Many matches of main concepts and Many matches of detailed descriptions	High
2	Many matches of main concepts and Some matches of detailed descriptions	High
3	Many matches of main concepts and Few matches of detailed descriptions	Medium
4	Some matches of main concepts and Many matches of detailed descriptions	High
5	Some matches of main concepts and Some matches of detailed descriptions	Medium
6	Some matches of main concepts and Few matches of detailed descriptions	Medium
7	Few matches of main concepts and Many matches of detailed descriptions	Medium
	Few matches of main consents and Some matches of detailed descriptions	Low
9	Few matches of main concepts and Few matches of detailed descriptions	Low

Случай 1 – высокое сходство ($U_H > U_L$ и $U_H > U_M$): Если значение, вычисленное из вышеупомянутой процедуры (нечеткого вывода Мамдани) происходит от понятия "высокого сходства", следующее уравнение используется для определения сходства значений документов *i* и *j*:

$$r_{ij}(U_H) = \left\{\frac{2+U_H}{3}\right\} \quad (3)$$

где U_H - функция принадлежности для высокое сходство; U_M функция принадлежности для среднее сходство; U_L функция принадлежности для низкое сходство, с

$$0 \leq U_H, U_M, U_L \leq 1 .$$

Случай 2 - среднее сходство ($U_M > U_H$ и $U_M > U_L$): Если значение, рассчитанное по вышеупомянутой процедуре происходит от “среднего сходства”, следующее уравнение используется, чтобы определить значение сходства. Определяя значение сходства для “ среднего сходства ”, взаимосвязь между “высоким сходством” и “низким сходством” влияет на сдвиг значения дефаззификации. В результате используются три уравнения для соответствия различным отношениям между “высоким сходством” и “низким сходством”.

$$r_{ij}(U_M) = \begin{cases} \frac{2+U_M}{6} & U_L > U_H \\ \frac{4-U_M}{6} & U_H > U_L \\ \frac{1}{2} & U_H = U_L \end{cases} \quad (4)$$

Случай 3 - низкое сходство ($U_L > U_H$ и $U_L > U_M$): Если значение, рассчитанное по вышеупомянутой процедуре, происходит от понятия "низкого сходства", используется следующее уравнение:

$$r_{ij}(U_L) = \left\{\frac{1-U_L}{3}\right\} \quad (5)$$

После того, как все меры сходства вычислены, генерируется матрица подобия. Иерархический алгоритм кластеризации затем используется, чтобы последовательно искать различные кластеры в соответствии с различной степенью вязи между объектами, как выражено в матрице

$$\begin{bmatrix} 1 & \cdots & r_{1i} \\ \vdots & \ddots & \vdots \\ r_{i1} & \cdots & 1 \end{bmatrix} \quad (6)$$

где r_{ij} - сходство документа i и документа j: следовательно, значение r_{ij} равно r_{ji}.

Применение иерархического алгоритма кластеризации заключается в следующем.

1) Найти максимальное (r_{ij}) в матрице и сгруппируйте документы i и j в новую группу.

2) Вычислить отношение между новыми кластерами и другими документами при помощи метода средней связи.

3) Перейти в Шаг 1) до тех пор, пока не останется один кластер слева.

Заключение. Традиционно, методологии обработки документов с использованием знаний ключевых фраз. Тем не менее, фраза может представлять множество значений, и много различных фраз могут иметь те же значения. Приведенный метод анализирует грамматику предложения и строит онтологию документов. Затем отношения между документами выводятся, и документы сходства и различия сравниваются. Представленная методология нечеткой онтологической кластеризации документов, удобнее, по сравнению с часто используемым подходом К-средних ключевых фраз.

1. *M. Fallon. G. Pcdrazzi. and R. Tuna*. "Text mining applied lo patent mapping: Л practical business case." *World Pa. Inf.*. vol. 25. no. 4.pp. 335-342. Dec. 2003.

2. *R. N. Kostoft". D. R. Toochman. H. J. Ebcrhart. and J. A. Humcnik*. "Text mining using database tomography and biblionKtrics: A review." *Technol. Forecast. Soc. Change*, vol. 68. no. 3. pp. 223-253. Nov. 2001.

3. *J. L. Hou and C. A. Chan*. "A document content extraction model using keyword correlation analysis."//n../. *Electron. Bus. Manag.*. vol. I. no. I. pp. 54-62.2003.

4. *C. G. Nevill-Manning. I. H. Widen, and G. W. Paynicr*. "Lcxically-generaled subject hierarchies for browsing large collections.- *Int. J. Digit. Libr.*. vol. 2. no. 2/3. pp. 111-123. Sep. 1999.

5. *I. H. Witten.* "Adaptive text mining: Interring structure from sequences."/ *Discret Algorithms*, vol. 2. no. 2. pp. 137-159. Jun. 2004.

6. *S. N. Sanchez. E. Triantaphyllou. and O. Kraft.* "A feature mining based approach for the classification of text documents into disjoint classes." *Inf. Process. Manag.*. vol. 38. no. 4. pp. 283-604. Jul. 2002.

7. *F. Feng and B. W. Croft*. "Probabilistic techniques for phrase extraction." *Inf. Process. Manag.*. vol. 37. no. 2. pp. 199-220. Mar. 2001.
8. *Z. Wu and M. Palmer*. "Verb semantics and lexical selection.- in *Proc. 12nd Anna. Meeting Assoc. Comput. Linguist.*. Las Cruccs. NM. Jun.27-30. 1994. pp. 133-138.
9. *C. C. Kung*. "Personalized XML information service system with automatic object-oriented ontology construction." M.S. thesis. Dept. Comput. Sci. Inform. Eng.. Nat. Cheng Kung Univ.. Tainan. Taiwan. 2000.

Поступила 16.9.2013р.

УДК 623.746-519

Б.В. Дурняк, О.Ю-Ю. Коростіль

ЗАГАЛЬНА ОРГАНІЗАЦІЯ ФУНКЦІОНУВАННЯ СИСТЕМИ ТЕКСТОВИХ МОДЕЛЕЙ

Аннотация. Рассматривается общая организация функционирования системы текстовых моделей. Проводится анализ всех задач, которые решаются на основе использования системы текстовых моделей. К таким задачам относятся задачи управления, задачи мониторинга, задачи тестирования социальных объектов и задачи обеспечения их эволюционного развития.

Ключевые слова: модель, тестирование, мониторинг, анализ, текстовые модели, социальные объекты.

Загальна організація функціонування системи моделювання соціальних об'єктів є багатоплановою і суттєво залежить від характеру задач, які планується з її допомогою розв'язувати. Тому, необхідно більш детально визначитися з окремими задачами, які планується з її допомогою розв'язувати і на прикладі однієї з них реалізувати схему функціонування такої системи. До таких з адач можна віднести наступні задачі, які на загальному рівні полягають у наступному, або вони допускають слідуючі інтерпретації:

- задачі моніторування деякого соціального середовища,
- задачі управління сукупністю SO_i,
- задачі формування сукупності визначених типів SO_i,
- задачі діагностики соціальних процесів, що проявляються в середовищі деякої сукупності SO_i,
- задачі забезпечення еволюційного розвитку деякої сукупності SO_i.

Задачі моніторування сукупності SO_i, що позначється, як деяка система SSO_i, можуть представлятися не коректними при використанні засобів моделювання типу TM_i для опису SO_i у зв'язку з тим, що текстові моделі TM_i та їх сукупність STM_i, не мають безпосереднього фізичного зв'язку з об'єктами моделювання $SO_i \in SSO_i$. Тим не менше, використання STM_i в

