А. В. Жданова, П. В. Манкевич

СТАТИСТИЧЕСКИЙ ПОДХОД К СЕМАНТИЧЕСКОМУ СТРУКТУРИРОВАНИЮ ПРЕДМЕТНЫХ ОБЛАСТЕЙ ДЛЯ ЗАПРОСОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

1. ВВЕДЕНИЕ

В настоящее время можно выделить два качественно различных подхода к поиску ответов на вопросы в глобальной сети Интернет и локальных Интранетах. В первом, ориентированном на базы данных, подходе фрагменты Сети рассматриваются как базы данных, а во втором подходе, типичном для многих поисковых систем, данные Интернета рассматриваются как множество неупорядоченных естественно-языковых текстов, плохо структурированных и содержащих избыточную информацию.

Исторически, в решениях задачи удовлетворения естественно-языковых запросов первый подход стал применяться ранее второго. Это обусловлено тем, что базы данных получили широкое распространение еще до повсеместного появления Интернета. В решении задачи удовлетворения запросов на естественном языке в контексте первого подхода были проведены теоретические разработки и построены работающие прототипы информационных систем, в частности основанные на реляционных базах данных [2, 3]. Однако, далеко не всю информацию, имеющуюся в распоряжении пользователей на сегодняшний день, можно наиболее выгодно представить в доступных типах баз данных. Например, тогда как перечень результатов футбольного матча структурно соответствует реляционной базе данных, отображение произвольного естественно-языкового текста в реляционную базу данных нетривиально и неэффективно.

В рамках второго подхода к обработке запросов было продемонстрировано, что упорядочивание множества естественно-языковых документов конкретной предметной области в иерархию способствует повышению эффективности в решении задач классификации сообщений и создания автоответчиков для электронной почты [7, 8]. (Для обобщения понятий следует заметить, что задача автоматического ответа на e-mail сводится к задаче информационного поиска в локальном Интранете, так как для автоматического ответа на сообщение требуется произвести поиск ответа во множестве заданных ответов.) И если подойти к практике, то на сегодняшний день большинство популярных поисковых систем (Google, Yahoo, Yandex…)

создают свои собственные древовидные иерархии веб-ресурсов и используют их в поиске информации. Проблема в том, что даже в системах-рекордсменах по величине проиндексированных ресурсов иерархии составляются вручную. Учитывая прогнозируемый рост и развитие Интернета, существующий неавтоматизированный подход оказывается неприемлемым.

Разработки, ведущиеся в направлении автоматического структурирования предметных областей, включают в себя реализацию алгоритмов, построенных на основе Self-Organizing Maps (SOM – это разновидность нейронных сетей) [4], вероятностных моделей, использующих адаптированный для построения иерархии probabilistic latent semantic analysis [7].

К недостаткам использования SOM для автоматического структурирования предметной области относится отсутствие логического обоснования моделей, использующих SOM. Как следствие, задача определения пороговых значений (например, ограничивающих разрастание SOM в определенном направлении) и выбора параметров становится достаточно сложной. Более того, пороговые значения и параметры будут зависеть от конкретного типа набора кластеризуемых документов. К недостаткам работ по использованию SOM для иерархической кластеризации документов относится также отсутствие оценок корректности получившихся древовидных структур и их пригодности к использованию в информационных системах. Оценки корректности можно получить сравнением автоматически сгенерированной структуры со структурой, составленной экспертом вручную на тех же данных. Оценки пригодности автоматически образованных древовидных структур можно получить после использования получившихся структур в приложениях, выполняющих конкретные задания, качество выполнения которых относительно легко проконтролировать. К таким приложениям, например, относится поисковая система-автоответчик, удовлетворяющая запросы пользователей.

В работе Винокурова и Джиролами [7] оценки корректности, полученные в рамках решения задачи классификации документов, показывают, что использование предложенных вероятностных моделей оправдывает себя в решении данной задачи. Однако вопрос пригодности данных моделей для удовлетворения запросов на естественном языке не рассматривается, у построенной системы нет естественно-языкового интерфейса, алгоритмы для информационного поиска по конкретному запросу не предложены и не опробованы.

В нашей статье мы представляем статистический подход к построению структуры (иерархии) предметной области. Данный подход ориентирован на использование получившейся структуры для поиска информации на ес-

тественном языке. В разд. 2 нашей статьи мы обсуждаем понятие «иерархия» и описываем принципы её построения, в разд. 3 приводим алгоритм построения структуры предметной области, в разд. 4 характеризуем используемый естественно-языковой интерфейс, в разд. 5 приводим результаты тестирования алгоритма и делаем заключение.

2. ИЕРАРХИЯ И ЕЁ ПРЕИМУЩЕСТВА

Во всех относящихся к рассматриваемой проблеме известных нам работах вообще и в нашей работе в частности текстовые документы упорядочиваются в иерархию (конкретнее, в дерево) на основе частичного порядка «категория-подкатегория», введенного на множестве документов, где каждый документ представляет собой категорию, т.е. характеризует некое идентифицируемое понятие, объект или явление. Примерами отношения частичного порядка «категория-подкатегория» являются отношения «is-a» и «part-of». Очевидно, что значительное преимущество в использовании иерархически упорядоченной предметной области в задачах информационного поиска состоит в том, что в самой иерархии заложена информация, которая может выгодно использоваться в поисковых алгоритмах. Например, из иерархии можно узнать, что категория «кошачьи» является подкатегорией категории «животные», и после этого значительное количество времени будет сэкономлено на том, что мы будем искать требуемую нам категорию «львы», сразу направившись по правильной ветке иерархии «животные/кошачьи/…», а не перебирая вместо этого все подкатегории прочих разделов: «птицы», «рептилии» и т.д.

Мы предлагаем статистический алгоритм для автоматического структурирования предметных областей, основанный на понятии *веса текстового документа*. Вес текстового документа в конкретной предметной области определяется количеством и частотностью значимых слов этой области. Чем больше значимых понятий встречается в документе, тем тяжелее для понимания этот документ [9]. Соответственно, легкие для понимания документы описывают общеизвестные концепции с помощью общеупотребительной лексики. Таким образом, после структурирования предметная область представляется деревом категорий, в котором «легкие» документы расположены ближе к корню, а наиболее «тяжелые» находятся в листьях. В свою очередь, решение об установлении отношения «категория-подкатегория» между двумя документами осуществляется, исходя из величины *меры семантической близости* двух документов. Примеры того, как

можно определять вес текстового документа и вычислять величину меры близости двух документов, показаны на рис. 1 и 2.

ВЕС ТЕКСТОВОГО ДОКУМЕНТА

Правило Зипфа: трудность понимания слова $\boldsymbol{V} \sim \frac{1}{Fr(V)}$

Примеры функций ***w(X)***, отражающих «легкость» восприятия документа:

$$w(X) = \frac{1}{\prod x_i},\ w(X) = \frac{-\sum \ln(x_i)}{n},\ w(X) = \sum \frac{x_i x_i^*}{\sqrt{\sum (x_i x_i^*)^2}}$$

Рис. 1. Вес текстового документа *X* может выражаться через частоты встречающихся в нем слов $\{x_i\}$, общее количество известных слов *n*, и быть основанным на законе Зипфа: «трудность понимания слова обратно пропорциональна его частоте»

ПРИМЕРЫ МЕР БЛИЗОСТИ ТЕКСТОВЫХ ДОКУМЕНТОВ

Jaccard Association $\quad sim(X,Y) = \left(1 - \frac{a}{a+b+c}\right)^{-1}$,

где $a = \| X \cap Y \|$, $b = \| X \setminus Y \|$, $c = \| Y \setminus X \|$

Taxonomic Distance $\quad sim(X,Y) = \left(\sum_{i=1}^{\| F \|} (x_i - y_i)^2\right)^{-1/2}$

Cosine Measure $\quad sim(X,Y) = \frac{\sum 1 \le i \le \|F\| x_i y_i}{\sqrt{\sum 1 \le i \le \|F\| x_i^2 \sum 1 \le i \le \|F\| y_i^2}}$

Рис. 2. Меры семантической близости текстовых документов могут выражаться через частоты встречающихся в документах слов $\{x_i\}$, $\{y_i\}$, суммирование ведется по мощности множества *F* слов предметной области

Следовательно, отношение «категория-подкатегория» в нашей древовидной иерархии можно определить как «от простого к сложному», «от общего к частному», «от известного к неизвестному». Логическое обоснование выбора этого отношения состоит в том, что оно наиболее верно отражает процесс обучения человека. Сначала человек «запрашивает» и узнает «простые» вещи (они быстро находятся на вершине иерархии). Потом, если он не потерял интерес к предметной области, его словарный запас в этой предметной области начинает расширяться, и он «запрашивает» более «сложные» вещи более «сложными» словами. «Сложные» вещи нужно искать в глубине иерархии, т.е. дольше, но и запросы, адресующиеся к «сложным» и редким документам из глубин иерархии, встречаются реже. Таким образом, наши принципы конструирования иерархии позволяют естественно-языковому интерфейсу, используемому в системе-автоответчике, эффективно выдавать в качестве ответа документы, соответствующие уровню осведомленности пользователя о предметной области, который, в свою очередь, определяется исходя из «сложности» запроса.

3. АЛГОРИТМ ПОСТРОЕНИЯ СТРУКТУРЫ ПРЕДМЕТНОЙ ОБЛАСТИ

Вход: множество документов.

Выход: иерархически структурированное множество документов.

1. Определить вес каждого документа согласно функции веса.
2. Выбрать самый легкий документ и разместить его в корень иерархии.
3. Пусть в иерархии размещено n документов. Выбрать самый «легкий» документ d из оставшихся неразмещенных. Вычислить значения функции семантической близости между d и $\{d_i\}$ — множеством размещенных документов.
4. Установить отношение «категория-подкатегория» между документом d_{max}, наиболее сходным с документом d, и документом d. Разместить документ d в иерархии как подкатегорию документа d_{max}.
5. Если остались неразмещенные документы, повторить шаг 4, иначе алгоритм завершен.

В данном алгоритме функции веса документа и близости двух документов являются параметрами, т. е. разработчик может их выбрать из известных (в том числе приведенных ранее) функций или построить свои собственные, исходя из своих обстоятельств (см. результаты тестирования в табл. 1).

Таблица 1

Результаты тестирования алгоритма структуризации в зависимости от функций веса документа и мер сходности

Веса	Меры близости							
	CD	CC	JA	SP	SI	SDA	TD	Cosine
entr	61.83 11.73	58.2 12.41	72.56 13.50	77.66 14.24	58.3 10.52	71.56 13.02	75.83 14.07	66.16 11.97
stat	64.2 12.28	57.1 15.95	64.13 11.84	78.60 14.39	59.23 12.05	79.8 14.58	77.86 14.44	66.26 13.15
atn	64.6 9.79	55.7 9.40	65.03 12.80	69.83 12.39	49.53 8.73	76.3 12.53	72.96 13.72	71.46 12.61
bnn	65.23 12.07	60.6 14.04	74.96 14.02	72.16 12.91	51.23 10.01	75.63 13.82	78.13 14.11	69.6 13.16
ntc	68.2 11.82	72.26 13.75	65.86 11.78	71.23 12.73	63.8 11.31	76.3 13.46	81.16 14.80	67.99 11.29
npn	65.2 17.60	61.0 11.90	53.53 10.74	68.13 13.13	58.16 15.66	75.13 13.86	68.26 13.11	62.7 12.53
nnn	61.7 13.32	77.8 15.15	68.0 12.73	61.99 13.53	53.2 13.90	70.36 15.07	61.1 12.49	56.73 11.76
nfn	64.93 12.23	64.1 15.20	67.8 12.45	77.63 14.59	56.16 11.01	80.0 14.90	78.76 14.70	69.33 13.90
ltn	68.33 19.23	59.2 26.79	62.4 24.96	62.3 19.37	65.23 19.22	64.63 18.58	64.56 18.80	65.03 19.31
ltc	68.83 19.47	57.1 25.77	65.43 26.82	65.00 22.45	66.33 19.16	72.1 19.98	65.23 17.87	69.0 21.07
lnc	72.76 19.85	59.2 27.38	68.0 27.00	63.13 21.10	66.5 19.21	71.03 20.17	65.56 18.66	68.16 20.7
dtc	51.33 10.10	56.50 10.93	65.2 13.39	54.39 11.00	61.56 13.49	62.16 11.76	59.66 12.64	54.23 11.51

Верхнее число в ячейке показывает $recall = \frac{\| A \cap B \|}{\| A \|} \times 100\%$, нижнее — $precision = \frac{\| A \cap B \|}{\| B \|} \times 100\%$, где A — множество правильных ответов, а B — множество ответов, полученных системой-автоответчиком на основе автоматически структурированной предметной области.

4. ИНДЕКСИРОВАНИЕ ДОКУМЕНТОВ И РАБОТА ЕСТЕСТВЕННО-ЯЗЫКОВОГО ИНТЕРФЕЙСА

При индексировании документов и разборе естественно-языковых запросов мы различаем три типа слов: значимые, незначимые и вспомогательные. Под незначимыми словами мы понимаем множество слов, которое может встретиться в любой предметной области, и таким образом, не несет конкретной фактической информации. К этим словам могут относиться вспомогательные глаголы, артикли, союзы, предлоги, местоимения, такие как «is», «and», «to», «yours». Вспомогательные слова — это те слова, которые сами по себе не несут конкретной информации, но помогают установить значительные для понимания смысла запроса связи между значимыми словами предметной области. К ним относятся, например, слова-отрицания, меняющие смысл сказанного с точностью до наоборот, такие как «no», «don't», «never». В некоторых случаях к важным вспомогательным словам могут относиться некоторые местоимения и предлоги (например, когда системе принципиально понять «кто кому что дал»). Все остальные слова, лежащие в предметной области, являются значимыми, и именно ими производится индексирование документов.

На рис. 3 естественно-языковой интерфейс показан в работе: пример запроса (query), извлеченная из него значимая лексика (retrieved items), ответ (answer) и узлы древесной иерархии, содержащие ответ.

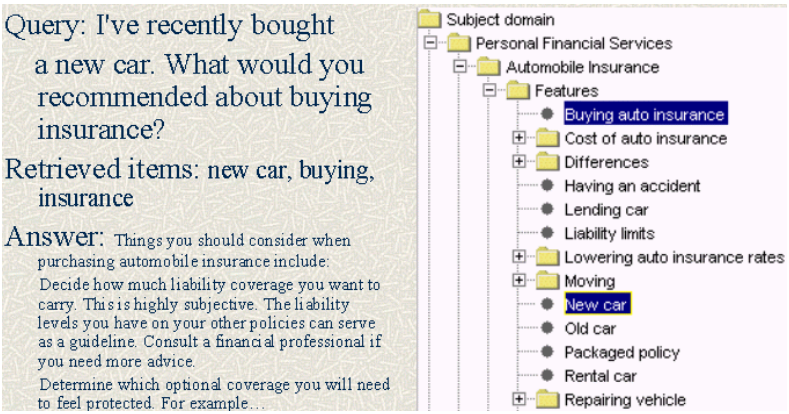

Рис. 3. Функциональность естественно-языкового интерфейса

В данной реализации системы-автоответчика индексирование также производится автоматически. Следует заметить, что присутствующие в индексах категорий значимые слова не записываются физически в индексы всех подкатегорий этих категорий по глубине иерархии, но наследование содержания индексов от категории к её подкатегории учитывается алгоритмом информационного поиска естественно-языкового интерфейса. Например, категории «страхование транспортных средств» и «страхование жизни» будут непосредственными подкатегориями категории «страхование». Так как все подкатегории категории «страхование» наследуют индекс категории «страхование» по глубине иерархии, нет необходимости хранить в них индекс категории «страхование». Таким образом, индексы на уровне категорий «страхование транспортных средств» и «страхование жизни» будут содержать слова близкие к таким, как «автомобиль», «жизнь» и прочим терминам, характеризующим именно эти категории. Используемые правила для алгоритма нахождения наиболее соответствующих запросу категорий дерева были подробно описаны в предыдущей работе [8].

5. РЕЗУЛЬТАТЫ

В рамках предложенного нами статистического подхода к построению иерархического представления предметной области были проведены эксперименты по структурированию предметной области страхования транспортных средств и страхования жизни (83 категории). Установлено, что автоматически сгенерированные иерархии с отношением «от простого к сложному» не совпадают в смысле совмещения наложением с иерархиями с отношением «is-a» и «part-of», построенными вручную (что вполне естественно в силу различия отношений упорядочивания). Как правило, в автоматически сгенерированных иерархиях мы наблюдали меньшее количество уровней, но структура такой иерархии существенно зависит от качества используемых текстовых документов, выбора функции веса и меры близости. Ниже приведен фрагмент автоматически сгенерированного дерева для предметной области Life Insurance (LI). Здесь номера (0, 1, 2,…) являются уровнями иерархии, а текстовые значения (Decreasing TLI, Graded-premium WLI,…) – названиями категорий. Расположение записей типа номер-название отображают отношения зависимости между категориями (зависимое множество подкатегорий находится непосредственно под своей категорией).

```
0 : Decreasing TLI
 1 : Graded-premium WLI
     2 : Rising premiums
     2 : Modified-premium WLI
 1 : Increasing TLI
 1 : Funeral insurance
     2 : Medical Payments
 1 : Continuous-premium WLI
     2 : Limited-payment WLI
 1 : Single-premium WLI
     2 : Indeterminate premium WLI
     2 : Current assumption WLI
         3 : Universal WLI
```

Мы оцениваем качество автоматического структурирования предметной области, исходя из результатов работы системы-автоответчика, построенной на основе этой области. Иными словами, на каждый переданный запрос системой-автоответчиком возвращалось два множества документов-ответов A и B. Первое из них получалось в результате работы системы с использованием области, структурированной вручную лингвистом. Это множество принималось за множество правильных ответов. Множество ответов B — результат работы системы с использованием автоматически структурированной области. Оценки работы системы-автоответчика, в зависимости от функций веса документа и мер сходности двух документов, усредненные по множеству тестовых запросов и полученные в широко известных терминах precision и recall [5], приведены в табл. 1. Использованные в тестировании функции веса документа [6] и меры семантической близости документов [1] показаны на рис. 4 и 5 соответственно.

Одной из основных проблем систем поиска и автоматического ответа является низкое значение precision, поэтому относительно небольшое значение precision в наших экспериментах является «нормальным». Результаты показывают, что при использовании отдельных функций веса и мер близости, значение recall превышает 75%. Принимая во внимание временные и умственные затраты на ручное структурирование информации, мы полагаем, что наш подход очень перспективен.

(CD) Camberra Distance $sim(X,Y) = \sum_{1 \le i \le \|F\|} \frac{|x_i - y_i|}{(x_i + y_i)}$

(CC) Correlation Coefficient $sim(X,Y) = \left(\frac{1-r}{2}\right)^{-\frac{1}{2}}, r = \frac{\sum x_i y_i - \frac{\sum x_i \sum y_i}{n}}{\sqrt{(\sum x_i^2 - \frac{(\sum x_i)^2}{n})(\sum y_i^2 - \frac{(\sum y_i)^2}{n})}}$

(JA) Jaccard Association $sim(X,Y) = \left(1 - \frac{a}{a+b+c}\right)^{-1}$,

где $a = \| X \cap Y \|, b = \| X \setminus Y \|, c = \| Y \setminus X \|$

(DP) Dot Product $sim(X,Y) = \sum_{1 \le i \le \|F\|} x_i y_i$

(SP) Simple Intersection $sim(X,Y) = a$

(SDA) Sorensen Dice Association $sim(X,Y) = \left(1 - \frac{a+d}{2a(2a+b+c)}\right)^{-1}$, где $d = \| F \setminus (X \cup Y) \|$

(TD) Taxonomic Distance $sim(X,Y) = \left(\sum_{1 \le i \le \|F\|} (x_i - y_i)^2\right)^{-\frac{1}{2}}$

(Cosine) Cosine Measure $sim(X,Y) = \frac{\sum_{1 \le i \le \|F\|} x_i y_i}{\sqrt{\sum_{1 \le i \le \|F\|} x_i^2 \sum_{1 \le i \le \|F\|} y_i^2}}$

Рис. 4. Функции меры семантической близости документов. $\{x_i\}$, $\{y_i\}$ — частоты слов документов X и Y соответственно. F — множество слов предметной области

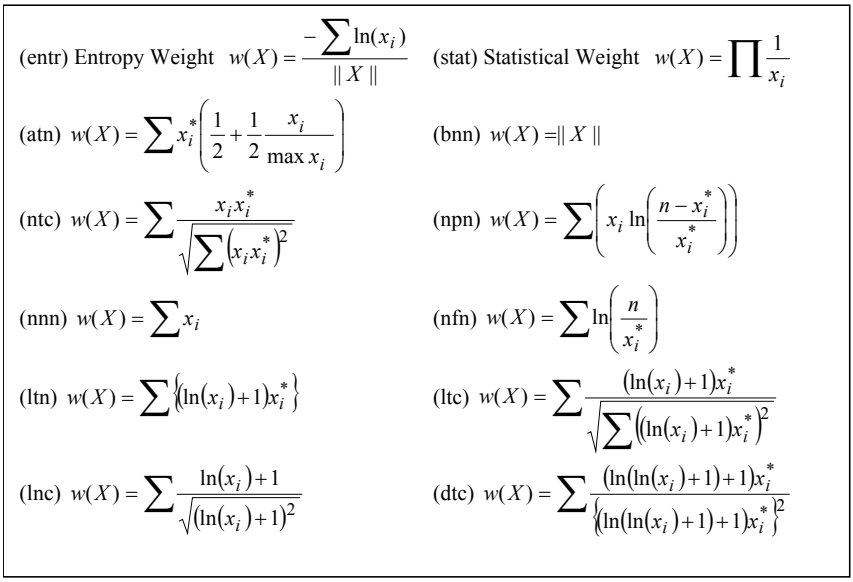

Рис. 5. Функции веса документа. Здесь x_i обозначает частоту i-го слова внутри документа X, а x_i^* — частоту этого слова во всей предметной области

СПИСОК ЛИТЕРАТУРЫ

1. **Anquetil N., Fourrier C., Lethbridge T**. Experiments with Hierarchical Clustering Algorithms as Software Remodularization Methods — Ottawa, 1999. — (Tech. Rep. / Department of Computer Science. University of Ottawa; N 99-1).
2. Approach to Development of a System for Speech Interaction with an Intelligent Robot / **Cheblakov, G.B., Dinenberg, F.G., Levin, D.Ya., Popov, I.G., Zagorulko, Yu.A**. // Perspectives of System Informatics: Proc / Conf. held in Novosibirsk, Russia, 1999 / Ed. by D. Bjørner *et al.* — Berlin etc.: Springer, 1999. — P. 517–529. — (Lect. Notes Comput. Sci.; 1755*)*.
3. **Dinenberg, F. G., Levin, D. Ya.** Natural Language Interfaces for Environmental Data Bases // Applications of Natural Language to Information Systems. — Amsterdam etc.: IOS Press, 1996.
4. **Freeman, R., Yin, H**. Self-Organising Maps for Hierarchical Tree View Document Clustering Using Contextual Information // Lect. Notes Comput. Sci.— Berlin etc., 2002.— Vol. 2412.— P. 123–128.

5. **Manning, C.D., Schütze, H.** Foundations of Statistical Natural Language Processing — Cambridge: The MIT Press, 2001.
6. **Savoy J.** Cross-language information retrieval: experiments based on CLEF 2000 corpora // Information Processing and Management. — 2003. — Vol. 39. — P. 75–115
7. **Vinokourov, A., Girolami, M. A.** Probabilistic Framework for the Hierarchic Organisation and Classification of Document Collections // Intellig. Inform. Systems. — 2002. — Vol. 18. — P. 153–172
8. **Zhdanova, A.V., Shishkin, D.V.** Classification of E-mail Queries by Topic: Approach Based on Hierarchically Structured Subject Domain // Lect. Notes. Comput. Sci. — Berlin etc., 2002. —Vol. 2412. — P. 99–104.
9. **Zipf, G. K.** Human Behavior and the Principle of Least Effort — Cambridge: Addison-Wesley, 1949.