УДК 004.912

А.А. Мамчич

СИСТЕМА ПОИСКА ИНФОРМАЦИИ НА ОСНОВЕ ДИНАМИЧЕСКИХ КОРПУСОВ ТЕКСТОВ

Предлагается архитектура программного комплекса индексирования и поиска текстовой информации в глобальных и локальных компьютерных сетях на основе динамических корпусов текстов. Рассматриваются состав и структура лингвистических словарей базы знаний, проводится оценка производительности подсистемы индексирования текстовых документов. Разработанная методика может применяться для решения широкого круга задач информационного поиска текстовой информации.

Введение

В настоящее время в связи со значительным увеличением объемов текстовой информации становится актуальной проблема разработки интеллектуальных систем, обеспечивающих решение широкого круга задач информационного поиска: поиска с адаптацией к информационным потребностям пользователей, тематического поиска, кластеризации и классификации текстовых документов и т. д. Помимо этого наблюдается тенденция интеграции таких систем с различными информационными источниками (например, ресурсами глобальной сети Интернет, ресурсами локальной сети, документами на жестком диске), что существенно усложняет получение высоких значений релевантности и пертинентности поиска (степени соответствия полученной информации информационной потребности пользователя). Традиционные подходы по увеличению данных характеристик основаны главным образом на различных лингвистических методах, семантическом и статистическом анализе [1, 2], что, однако, не позволяет достичь приемлемого уровня этих параметров, а тем более достаточной степени интеллектуализации системы в целом.

В настоящей статье для решения перечисленных выше задач предлагается подход, основанный на использовании динамических корпусов текстов (наборов текстов, релевантных каждому конкретному тексту или запросу на поиск информации), что позволяет индексировать и искать не только полнотекстовые документы, но также и краткие сообщения. Проблема обеспечения высоких значений релевантности и пертинентности решается путем вычисления весов ключевых слов на основе динамических корпусов текстов, что дает возможность учитывать статистическую информацию не только самого документа, но и соответствующего ему корпуса текстов (предметной области поиска). Данная методика, используемая при поиске информации, существенно повышает интеллектуальность системы.

В рамках исследуемой проблемы нерешенными являются следующие задачи:

– разработка принципов работы программного комплекса индексирования и поиска текстовых документов на основе динамических корпусов текстов;

– разработка архитектуры программного комплекса индексирования и поиска текстовой информации в глобальных и локальных компьютерных сетях на основе динамических корпусов текстов;

– определение состава и структуры базы знаний (комплекса словарей) системы на основе динамических корпусов текстов;

– определение состава и структуры базы данных проиндексированных документов;

– оценка производительности подсистемы индексирования и разработка путей по ее увеличению.

Решение этих задач является основной целью данной статьи.

1. Принципы работы программного комплекса индексирования и поиска документов на основе динамических корпусов текстов

Проблема в достижении высоких значений релевантности и пертинентности заключается в том, что часто пользователь перед началом поиска плохо осознает свою информационную

потребность и имеет лишь общее представление о ней. В связи с этим он не в состоянии сформулировать точный запрос на поиск информации. По статистике [1, 3] большинство запросов популярных поисковых систем в сети Интернет имеют длину не более 2–4 ключевых слов. В результате обработки такого поискового предписания будет сформирован значительный массив документов, из которых релевантными и пертинентными окажутся немногие. Для повышения данных параметров предлагается использовать знания, полученные из динамических корпусов текстов, что позволяет расширять первоначальный запрос пользователя и тем самым динамически изменять пространство поиска.

Определим понятие динамического корпуса текстов. Пусть имеется некоторое непустое множество текстов (совокупность текстов по различным предметным областям). Будем обозначать их t_i (i – номер текста). Объединение $Cf = \bigcup_{i=1}^{n} t_i$ всех текстов назовем *полным корпусом* текстов. Тогда корпус текстов $\pi(t)$, который состоит из конкатенации документов корпуса *Cf*, релевантных первоначальному запросу пользователя на поиск информации, будем называть *динамическим корпусом текстов*.

Процесс поиска информации заключается в сравнении запросов пользователей с поисковыми образами документов, полученными посредством индексирования, и создании списка текстов, отвечающих запросу, на основании применяемого в системе критерия выдачи (формального правила, согласно которому определяется степень соответствия поискового образа документа поисковому предписанию и принимается решение о выдаче или невыдаче того или иного документа).

Под индексированием в информационно-поисковой системе на основе динамических корпусов текстов будем понимать процесс представления документов в виде некоторого множества ключевых слов, отражающих их смысловое содержание, с определенными числовыми коэффициентами (*весами* ключевых слов, которые определяют их информативность). По результатам индексирования создается специальная служебная структура (*поисковый образ*), использующаяся при поиске, и формируется база данных, в которой хранятся поисковые образы документов и дополнительная информация о них (адреса расположения документов, краткое описание и т. п.).

При определении весов ключевых слов современные системы используют различные методики и подходы, которые объединяет одно – все они ориентированы на исследование структуры и статистических характеристик самих документов без привлечения дополнительной информации о предметной области поиска, которая существенным образом повышает эффективность функционирования системы за счет ее интеллектуализации [4]. В работе [5] авторами была получена формула для определения информативности (веса) словоформы α в индексируемом документе в виде

$$I_{Th}^{\alpha} = \frac{n_{Th}}{n_{Cf}}, \tag{1}$$

где n_{Th} – абсолютная частота встречаемости словоформы α в индексируемом документе (или тематическом корпусе текстов Th_i); n_{Cf} – абсолютная частота встречаемости в полном корпусе текстов *Cf*.

Обобщая формулу (1) на случай динамического корпуса текстов $\pi(t)$, можно получить выражение вида [6, 7]

$$I_{\pi(t)}^{\alpha} = \frac{n_{\pi(t)}}{n_{Cf}}, \tag{2}$$

где $n_{\pi(t)}$ и n_{Cf} – абсолютные частоты встречаемости словоформы α соответственно в индексируемом документе (или релевантном ему корпусе текстов $\pi(t)$) и *Cf*.

С учетом словоизменения и синонимии формула (2) примет вид

$$I_{\alpha} = \frac{n_1 + n_2 + n_3}{N + (n_1 + n_2 + n_3)}, \tag{3}$$

где n_1 – количество вхождений словоформы α в индексируемом документе (или корпусе текстов $\pi(t)$); n_2 – количество словоизменений данной словоформы в документе ($\pi(t)$); n_3 – суммарное количество вхождений словоформ, которые являются синонимами α в документе ($\pi(t)$); N – суммарное количество вхождений словоформы, ее словоизменений и синонимов в полном корпусе текстов *Cf*.

В случае если текстовый документ включен в полный корпус *Cf*, слагаемые n_1, n_2 и n_3 в знаменателе формулы (3) отсутствуют, так как учитываются при вычислении N.

В качестве индексируемого документа могут выступать как полнотекстовые документы, так и краткие сообщения, объем которых небольшой и не позволяет выявить их статистические характеристики. В случае индексирования краткого сообщения при формировании поискового образа в формуле (3) используются абсолютные частоты словоформ в соответствующем сообщению динамическом корпусе текстов $\pi(t)$, а в случае полнотекстового документа – абсолютные частоты в индексируемом документе.

Представим первоначальный запрос пользователя на поиск информации в виде множества пар $z = \{(a_1, 1), \ldots, (a_l, 1)\}$, где $a_1, a_2, \ldots, a_l$ – ключевые слова запроса пользователя. В качестве весов ключевым словам в первоначальном запросе приписывается значение, равное единице.

Расширенным поисковым запросом пользователя $Z_{\pi(t)}$, учитывающим информацию из динамического корпуса текстов $\pi(t)$, который соответствует запросу z, будем называть множество $Z_{\pi(t)} = \{(b_i,\ I_{b_i}) \mid i = \overline{1, n}\}$, где $b_1, b_2, \ldots, b_n$ – ключевые слова динамического корпуса текстов $\pi(t)$, I_{b_i} – веса соответствующих слов в $\pi(t)$, вычисленные по формуле (3), такое, что все ключевые слова первоначального запроса z принадлежат $Z_{\pi(t)}$ (т. е. $z \subset Z_{\pi(t)}$). Количество пар n в множестве $Z_{\pi(t)}$ подбирается эмпирически и включает, помимо ключевых слов z, слова, информативность которых больше, чем информативность остальных слов в $\pi(t)$.

Поисковый образ текстового документа t представим в виде $O_t = \{(c_1, J_1), (c_2, J_2), \ldots, (c_k, J_k)\}$, где $c_1, c_2, \ldots, c_k$ – ключевые слова документа t; $J_1, J_2, \ldots, J_k$ – веса соответствующих слов, вычисленные по формуле (3).

Сформируем множество вида $O^+ = \{(d_m,\ I_{d_m},\ J_{d_m}) \mid m = \overline{1, s},\ s \le \min(n, k)\}$, где $\{d_m \mid m = \overline{1, s}\} = \{b_i \mid i = \overline{1, n}\} \cap \cap \{c_j \mid j = \overline{1, k}\}$, I_{d_m} – вес слова d_m в расширенном поисковом запросе пользователя $Z_{\pi(t)}$; J_{d_m} – вес этого же слова в поисковом образе текстового документа t. Другими словами, O^+ – это множество ключевых слов, входящих как в поисковый образ документа, так и в $Z_{\pi(t)}$, с соответствующими весовыми коэффициентами.

Введем в рассмотрение n-мерное евклидово пространство E. Запрос $Z_{\pi(t)}$ и поисковый образ O_t можно представить как векторы в пространстве E, в качестве координат которых будут выступать соответствующие веса ключевых слов. С учетом такого векторного представления в рассматриваемой системе наиболее целесообразно использовать критерий выдачи в виде косинуса угла между векторами запроса $Z_{\pi(t)}$ и поискового образа O_t. Формула для вычисления критерия выдачи с учетом принятых обозначений примет вид

$$\cos\varphi = \frac{\sum_{m=1}^{s} I_{d_m} J_{d_m}}{\sqrt{\sum_{i=1}^{n} I_{b_i}^{\,2}}\ \sqrt{\sum_{j=1}^{k} J_{c_j}^{\,2}}}. \qquad (4)$$

Документы, значение критерия (4) для которых выше, считаются наиболее удовлетворяющими запросу пользователя (найденные документы ранжируются по убыванию значения $\cos\varphi$).

2. Архитектура программного комплекса индексирования и поиска документов на основе динамических корпусов текстов

Функциональными компонентами программного комплекса индексирования и поиска текстовых документов на основе динамических корпусов текстов являются (рис. 1):

– подсистема индексирования текстовых документов;
– подсистема поиска текстовых документов;
– интерфейс администратора;
– интерфейс пользователя.

Рис. 1. Архитектура программного комплекса индексирования и поиска текстовых документов на основе динамических корпусов текстов

В программном комплексе индексирования и поиска текстовых документов предусмотрены два интерфейса: администратора и пользователя. Интерфейс администратора – это программно-информационный инструментарий, позволяющий получить доступ к программным средствам, посредством которых возможно создание и ведение словарей базы знаний, накопление корпусов текстов по различным предметным областям. Под интерфейсом пользователя понимается программный инструментарий, с помощью которого пользователь формирует первоначальный запрос на поиск информации и указывает область поиска (Интернет, локальная сеть, жесткий диск компьютера и т. д.).

Первоначальный запрос пользователя передается в подсистему поиска текстовых документов, которая осуществляет поиск и выдает массив найденных документов пользователю.

Опишем принципиальный алгоритм работы подсистемы поиска текстовых документов.

Алгоритм 1. На входе подсистемы – первоначальный запрос пользователя z, на выходе – множество $T_{\text{вых.}}$ найденных текстовых документов. Алгоритм работы включает в себя следующие шаги:

Шаг 1. Пользователь через соответствующий интерфейс программного комплекса указывает область поиска текстовых документов (например, «Поиск в сети Интернет», «Поиск в локальной сети» и т. д.) и вводит запрос на поиск информации на естественном языке.

Шаг 2. Запрос пользователя обрабатывается (из него удаляются предлоги, союзы, междометия и т. п.) и определяются ключевые слова. Формируется первоначальный запрос пользователя *z*.

Шаг 3. Из полного корпуса текстов *Cf* выбираются тексты, в каждом из которых содержатся все ключевые слова запроса *z*. Из найденных текстов путем их конкатенации формируется динамический корпус $\pi(t)$.

Шаг 4. Из динамического корпуса текстов $\pi(t)$ формируется расширенный запрос пользователя $Z_{\pi(t)}$.

Шаг 5. Согласно области поиска, заданной пользователем, в базе данных проиндексированных документов проводится поиск документов, соответствующих запросу $Z_{\pi(t)}$, по критерию (4).

Шаг 6. Найденные документы ранжируются по убыванию значения $\cos\varphi$. Низкорелевантные документы отбрасываются. Формируется множество $T_{\text{вых.}}$. Конец.

Для решения задач информационного поиска, таких, как поиск с адаптацией к информационным потребностям пользователя, тематический поиск, кластеризация и классификация текстовых документов и т. д., принципиальный алгоритм работы подсистемы не меняется. В качестве первоначального поискового предписания при решении данных задач будет выступать не запрос пользователя на естественном языке, а текстовый документ-образец, удовлетворяющий информационной потребности пользователя или отражающий тематику поиска (поиск с адаптацией к информационной потребности пользователя и тематический поиск), или документ-классификатор, определяющий конкретный класс текстовых документов (кластеризация и классификация текстовых документов).

Подсистема индексирования осуществляет непрерывную работу по формированию и обновлению базы данных проиндексированных документов, в которой хранятся поисковые образы текстов из различных информационных источников. Программная архитектура подсистемы индексирования (рис. 2) включает в себя следующие структурные компоненты:

– интерфейс управления;
– программу-планировщик;
– программу, осуществляющую загрузку документов;
– программу предварительной обработки текстовых документов;
– программу статистического анализа текстов.

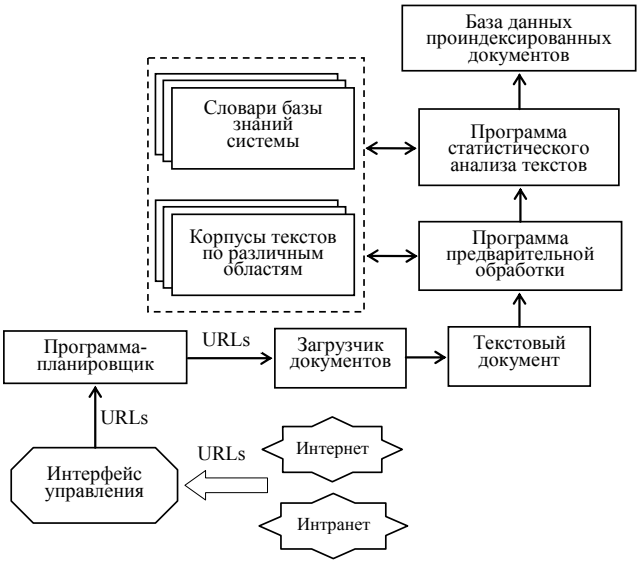

Рис. 2. Программная архитектура подсистемы индексирования текстовых документов на основе динамических корпусов текстов

Посредством интерфейса управления создается массив уникальных URL-адресов ресурсов сети Интернет (локальной сети), предназначенных для индексирования.

Программа-планировщик формирует список адресов ресурсов по принципу простой очереди FIFO (First IN First OUT) и передает их на вход программы, осуществляющей загрузку текстовых документов.

Программа-загрузчик текстовых документов на основе URL-адреса, полученного от программы-планировщика, загружает текстовый документ во временное хранилище.

Программа предварительной обработки текстовых документов осуществляет структурное разбиение исходного документа, которое заключается в выделении заголовка документа, ключевой информации о документе и избавлении от информации, не использующейся на этапе статистического анализа (формул, рисунков, стоп-слов, тэгов языков разметки (например, HTML, XML) и т. д.).

Программа статистического анализа текстов осуществляет подсчет статистической информации текстового документа. Под статистической информацией понимается абсолютное число вхождений в текстовый документ каждой словоформы (т. е. суммарное число вхождений в документ словоформы, ее словоизменений и синонимов). На основании полученной информации осуществляется процесс индексирования документа (построения служебной структуры «ключевое слово – вес») и сохранения результатов работы в базу данных проиндексированных документов.

Опишем принципиальный алгоритм работы подсистемы индексирования текстовых документов.

Алгоритм 2. На входе подсистемы – множество текстовых документов $T_{вх}$, на выходе – база данных проиндексированных документов. Алгоритм работы включает в себя следующие шаги:

Шаг 1. Посредством загрузчика документов загружается очередной документ t из множества $T_{вх}$.

Шаг 2. Документ t передается в программу предварительной обработки.

Шаг 3. Обработанный документ поступает на вход программы статистического анализа текстов.

Шаг 4. На основании статистической информации, полученной на шаге 3, определяются веса соответствующих ключевых слов документа t по формуле (3).

Шаг 5. Формируется и сохраняется в базу данных проиндексированных документов служебная структура «ключевое слово – вес» для документа t. Если все документы из множества $T_{вх}$ исчерпаны, то конец (база данных проиндексированных документов сформирована), иначе перейти к шагу 1.

В случае если текстовый документ t из $T_{вх}$ является кратким сообщением, после его загрузки формируется динамический корпус текстов $\pi(t)$, релевантный t. На основании этого корпуса в дальнейшем производится подсчет статистической информации ключевых слов краткого сообщения.

3. Состав и структура лингвистических словарей и базы данных проиндексированных документов

В процессе индексирования и поиска на основе динамических корпусов текстов используются следующие словари и базы данных:

– база данных корпусов текстов по различным предметным областям;
– частотный словарь словоформ;
– словарь словоизменительных парадигм;
– словарь синонимов;
– словарь стоп-слов.

На основе базы данных корпусов текстов по различным предметным областям в автоматическом режиме формируется частотный словарь словоформ с помощью программ актуализации словарей базы знаний. Пусть ω_k – некоторая словоформа из текстов полного корпуса Cf, а f_i – ее частота в тексте t_i полного корпуса Cf. Структура частотного словаря словоформ представлена в табл. 1.

Таблица 1

Состав и структура частотного словаря словоформ

Код словоформы	Словоформа	Частота в *Cf*	f_1	...	f_n	Код парадигмы
...						
5068	агент	0000234	0000004	...	0000007	00000414
5069	агента	0000129	0000002	...	0000005	00000414
...						
1456258	убавить	0004055	0000001	...	0000002	00077446
1456279	убавьте	0001657	0000003	...	0000001	00077446
...						

На этапе вычисления весов необходимо учитывать словоизменения и синонимию ключевых слов (см. формулу (3), разд. 1). Для этих целей в базу знаний системы включены словари словоизменительных парадигм и синонимов.

Словарь парадигм (табл. 2) служит для поиска всех словоформ парадигмы после нахождения словоформы и ее кода в словаре словоформ. Словарь парадигм создается и актуализируется в человеко-машинном режиме посредством соответствующего инструментария.

Таблица 2

Состав и структура словаря парадигм

Код парадигмы	Парадигма
...	
000000414	агент
	агента
...	
00077446	убавить
	убавлен
	убавьте
...	

Словарь синонимов (табл. 3) позволяет по коду парадигмы для определенной словоформы найти ее синонимы.

Таблица 3

Состав и структура словаря синонимов

Код парадигмы	Слово	Синонимы
...		
000000101	абсурд	бессмыслица
		абракадабра
		вздор
		ерунда
		чепуха
		чушь
...		
000050171	пласт	слой
		ряд
...		

Словари парадигм и синонимов формируются «вручную», однако на первоначальном этапе программный комплекс может функционировать без данных словарей, что приведет к некоторому ухудшению качества поиска, но не скажется на работе системы в целом.

Словарь стоп-слов содержит информацию о словах, которые не учитываются при индексировании текстовых документов ввиду их малой информативности (например, предлоги, частицы, междометия и т. д.). Помимо этого словарь стоп-слов используется для приведения к оптимальному виду запроса пользователя путем удаления из него неинформативных слов.

База данных проиндексированных документов формируется по результатам индексирования и включает в себя информацию о ключевых словах, их весах, расположении текстовых доку-

ментов, а также их краткое описание. С учетом большого объема данных, хранящихся в базе, основными требованиями, которые предъявляются к ее организации, являются низкая избыточность и отсутствие дублирования информации. Архитектуру базы данных проиндексированных документов наиболее удобно сформировать на основе инверсного индекса. Данный способ организации хранения информации с успехом применяется в поисковой системе Google и позволяет производить быстрый и эффективный поиск [8]. В инверсном индексе любому слову из запроса пользователя соответствует набор документов, в которых это слово встречается, что обеспечивает удовлетворение представленных выше требований к архитектуре базы данных (рис. 3).

Рис. 3. Структура базы данных проиндексированных документов

Структура базы данных проиндексированных документов на основе инверсного индекса может быть также дополнена информацией о номерах позиций каждого слова в пределах соответствующего документа (на практике реализуется посредством двух таблиц, связанных посредством внешних ключей). Данное усовершенствование позволяет автоматически восстанавливать копию текстового документа в случае его отсутствия (или недоступности) на удаленном ресурсе и улучшать качество поиска за счет учета прагматически полных синтагматических структур (например, *дистанционное зондирование Земли*, *синтаксический анализ предложения* и т. п.).

4. Оценка производительности подсистемы индексирования текстовых документов и пути ее увеличения

При проектировании и разработке информационно-поисковых систем необходимо проанализировать и оценить производительность работы подсистемы по созданию поисковых образов текстовых документов, так как от данной характеристики напрямую зависит возможность оперативного обновления информации в базе данных проиндексированных документов и количество информационных ресурсов, которые может охватить система.

Оценим время работы подсистемы индексирования текстовых документов на основе динамических корпусов текстов следующим образом. Представим общее время работы подсистемы в виде

$$T_{\text{общ}} = k \cdot t_{\text{index}},$$

где $k \cdot t_{\text{index}}$ – время, необходимое для обработки k текстовых документов и сохранения их поисковых образов в базу данных проиндексированных документов (t_{index} – среднее время индексирования и сохранения поискового образа в базу данных одного текстового документа).

Оценим среднее время индексирования одного текстового документа для случая работы с документами глобальной сети Интернет.

Для создания поискового образа текстового документа сети Интернет необходимо выполнить следующую последовательность операций:

1) соединение с удаленным сервером, на котором расположен текстовый документ;

2) передача серверу запроса на чтение текстового документа посредством протокола HTTP;

3) загрузка текстового документа;

4) обработка, индексирование текстового документа и сохранение результатов в базу данных проиндексированных документов.

Будем считать, что в момент соединения и передачи запроса удаленный сервер не перегружен и готов открыть соединение на чтение текстового документа. Тогда процесс соединения и передачи запроса происходит достаточно быстро – порядка 0,1 с.

Рассмотрим длительность выполнения операций 3) и 4). Средняя скорость доступа в сеть Интернет, по данным научно-технического отчета компании «Яндекс» за 2009 г., в России и странах СНГ составила примерно 410 кбит/с, или 50 КБ/с [9], а средний размер веб-документа Рунета ≈ 17 КБ [10]. Таким образом, время на загрузку текстового документа из сети Интернет составит ≈ 0,34 с.

Для оценки времени обработки, индексирования и сохранения результатов в базу данных проиндексированных документов был реализован алгоритм индексирования текстовых документов на основе динамических корпусов текстов, подробно описанный в [6, 7]. При разработке использовался язык программирования C++ с поддержкой библиотеки Qt 4.5, а в качестве системы управления базами данных (СУБД) для хранения лингвистических словарей и базы данных проиндексированных документов – СУБД MySQL 5.1. В ходе проведенных экспериментов для текстовых документов размером ≈ 17 КБ среднее время обработки, индексирования и сохранения результатов составило ≈ 0,61 с.

Таким образом, полное время, необходимое для создания поискового образа на основе динамических корпусов текстов для одного документа сети Интернет, составляет

$$t_{\text{index}} = 0{,}1 + 0{,}34 + 0{,}61 \approx 1{,}05 \text{ (с)}.$$

Учитывая это, производительность подсистемы индексирования на основе динамических корпусов текстов за 1 сутки составит ≈ 82 000 текстовых документов.

В случае если производительность подсистемы недостаточна для оперативного обновления информации и количества информационных ресурсов, по которым необходимо проводить поиск, возможны следующие пути по ее увеличению:

– организация многопоточного режима работы, т. е. организация приема и предварительной обработки текстовых документов одновременно в несколько потоков, что позволит уменьшить время, необходимое на загрузку и обработку текстовых документов;

– использование в работе подсистемы параллельных вычислений, что позволит, оставляя неизменной сложность алгоритма индексирования текстовых документов на основе динамических корпусов текстов, значительно снизить время его выполнения.

Заключение

В рамках данной статьи был предложен новый подход по интеллектуализации процесса поиска текстовой информации – использование релевантных динамических корпусов текстов при вычислении значений весов ключевых слов документов и расширении первоначальных запросов пользователей. На основе данного подхода были разработаны архитектура и лингвистическое обеспечение программного комплекса индексирования и поиска текстовой информации в локальных и глобальных компьютерных сетях, позволяющего решать широкий спектр задач информационного поиска. Проведен анализ производительности работы подсистемы индексирования текстовых документов и рассмотрены пути по ее увеличению.

Дальнейшие исследования в этой области могут быть продолжены в направлении разработки и совершенствования критерия оптимальности процесса поиска, согласно которому функционирование системы признается наилучшим из всех возможных вариантов. Кроме того, предполагается разработка методов реферирования текстовых документов на основе динамических корпусов текстов, что позволит в дальнейшем реализовать промышленный вариант информационно-аналитической системы интеллектуального поиска и обработки текстовой информации по различным предметным областям.

Список литературы

1. Ландэ, Д.В. Поиск знаний в Internet. Профессиональная работа / Д.В. Ландэ. – М. : Издательский дом «Вильямс», 2005. – 272 с.

2. Chakrabarti, S. Mining the Web. Discovery knowledge from hypertext data / S. Chakrabarti. – UK : Morgan Kaufmann, 2002. – 344 p.

3. Сравнение аудитории поисковых систем посредством анализа поисковых запросов // Маркетинг-журнал [Электронный ресурс]. – 10.10.2007. – Режим доступа : http://www.4p.ru/main/research/13138/. – Дата доступа : 8.12.2009.

4. Технологии извлечения знаний из текста / Н. Ильин [и др.] // Открытые системы [Электронный ресурс]. – 2006. – № 6. – Режим доступа : http://www.i-teco.ru/article104.html. – Дата доступа : 8.12.2009.

5. Липницкий, С.Ф. Веб-поиск и аннотирование научно-технической информации на основе тематических корпусов текстов / С.Ф. Липницкий, А.А. Мамчич, С.А. Сорудейкина // Информатика. – 2009. – № 2. – С. 114–125.

6. Липницкий, С.Ф. Создание поисковых образов текстовых документов на основе динамических корпусов текстов / С.Ф. Липницкий , А.А. Мамчич // Материалы Четвертого Белорусского космического конгресса. В 2 т. Минск, Республика Беларусь, 27–29 октября 2009 г. – Минск : ОИПИ НАН Беларуси, 2009. – Т. 2. – С. 180–184.

7. Липницкий, С.Ф. Индексирование и поиск текстовой информации на основе динамических корпусов текстов / С.Ф. Липницкий, А.А. Мамчич // Доклады VIII Междунар. конф. «Развитие информатизации и государственной системы научно-технической информации (РИНТИ-2009)», Минск, Республика Беларусь, 16 ноября 2009 г. – Минск : ОИПИ НАН Беларуси, 2009. – С. 244–249.

8. Brin, S. The Anatomy of a Large-Scale Hypertextual Web Search Engine / S. Brin, L. Page // [Electronic resource]. – Mode of access : http://infolab.stanford.edu/~backrub/google.html. – Date of access : 21.09.2009.

9. Средняя скорость доступа в Интернет. Новостной и аналитический портал «Время электроники» [Электронный ресурс]. – Режим доступа : http://www.russian-electronics.ru/leaderr/news/russianmarket/doc39972.phtml. – Дата доступа : 12.11.2009.

10. Статистика по веб-страницам Рунета / [Электронный ресурс]. – Режим доступа : http://promosite.ru/articles/stat-inet.php. – Дата доступа : 12.11.2009.

Поступила 17.12.09

Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: lexamam@newman.bas-net.by

A.A. Mamchich

THE CONCEPT OF SYSTEM OF INTELLECTUAL INFORMATION SEARCH ON THE BASIS OF DYNAMIC CORPORA

The architecture of the software for indexing and searching textual information in global and local computer networks on the basis of dynamic corpora is suggested. The composition and structure of linguistic dictionaries of the knowledge base are considered and the productivity of a subsystem indexing text documents is estimated. The described method can be used for the decision of a wide range of tasks on searching textual information.