Интеграция публично доступных архивов списков рассылки[1]

Д. Барашев, А.Высоцкий С. Кукс, Е. Михайлова, И. Некрестьянов, Б. Новиков, Е. Павлова
Санкт-Петербургский Государственный Университет
E-mail: mas-project@meta.math.spbu.ru

Аннотация

В работе рассматривается задача создания системы, предоставляющей единый доступ к находящимся в Интернет архивам списков почтовой рассылки. Эта практическая задача используется в качестве полигона для применения передовых методов работы со слабоструктурированной информацией.

Введение

В настоящее время большое количество архивов списков рассылки доступно в Интернет в виде HTML страниц. Такие страницы обладают рядом черт, отличающих их от прочих HTML страниц, находящихся в Интернет:

- **Архив определяет "тематический" контекст**
  Вероятность того, что множество случайно выбранных HTML-страниц из одного и того же архива содержит информацию по одной и той же тематике практически единица, что далеко не так в случае произвольно выбранных страниц [7].
- **Схожая "семантическая структура"**
  Структура сообщения электронной почты строго определена соответствующим RFC. При публикации архива в Интернет обычно используется только часть доступной в сообщении информации. Тем не менее, даже такая неполная информация о семантическом содержании страниц недоступна для случайно выбранной страницы в Интернет.
- **Архив — источник слабоструктурированной информации**
  Информация доступная в архивах не имеет известной заранее схемы, а также может быть неполна и противоречива. Например, несмотря на то, что при создании каждого архива сообщения преобразуются в html одним и тем же способом, полученные страницы, зачастую, имеют не полностью идентичную структуру[2].

Эти особенности могут быть использованы при организации поиска информации в рамках архивов списков рассылки, например:

- Информационный запрос пользователя может быть использован для обнаружения архивов, содержащих информацию на близкую тему. Кроме того, если дальнейший поиск и/или навигация по выбранным архивам не помогли найти релевантный ответ, то пользователь имеет возможность задать свой вопрос в соответствующий список рассылки.
- При организации поиска по архивам можно учитывать структуру сообщений. Это позволяет проводить поиск не во всей HTML-странице с письмом, а только в некоторых частях этого письма - например,

[1] Работа выполнена при частичной поддержке РФФИ (грант 01-01-00935).
[2] Например, это может происходить из-за получения части информации при генерации страницы динамически, например, html-фрагмента с рекламным баннером из базы данных, или из-за изменений, внесенных в процедуру публикации в процессе ее эволюции.

только в поле “Тема” и самом теле сообщения. Это также позволяет исключить из рассмотрения те части писем, которые не относятся к содержанию письма (например, подпись автора, которая иногда бывает довольно длинной и содержит большое количество мусора)

- Автоматическое определение тематической направленности списка рассылки позволяет выделять письма, не соответствующие тематике и исключать их из поисковой базы (например, рекламный спам)

В данной работе мы рассматриваем задачу создания системы, предоставляющей единый доступ к множеству архивов списков рассылки, доступных в Интернет. Нашей практической целью является создание прототипа, выполняющего следующие задачи:

- (Полу) автоматическое обнаружение архивов списков рассылки в Интернет
- Автоматическая верификация того, что обнаруженная страница является частью архива списка рассылки
- Автоматическое извлечение индивидуальных писем из архива (с восстановлением их структуры)

Создание этого прототипа имеет также ряд научных целей:

- Изучение применимости передовых методов для работы со слабоструктурированной информацией к решению практической задачи
- Исследование возможности автоматической генерации адаптеров, извлекающих информацию из таких слабоструктурированных источников
- Разработка эвристик идентификации информации
- Разработка автоматических методов классификации информации по структуре

Статья организована следующим образом: в следующем разделе дан краткий обзор близких задач и систем; в разделе 3 описывается предлагаемый подход к решению рассматриваемой задачи, а в разделе 4 представлены результаты наших практических экспериментов.

Близкие темы

Традиционные поисковые системы, такие как Google и Altavista, помимо прочих страниц, индексируют и страницы архивов списков рассылки и таким образом предоставляют возможность поиска по имеющейся в них информации. Однако, среди результатов поиска письма из архивов будут встречаться вперемешку с обычными страницами, доступными в Web. Эта проблема может быть до некоторой степени нивелирована использованием методов поиска по категориям [13], но остаются и другие слабые места, например, информация о структуре писем не используется, как и информация о тематической направленности архивов.

Отметим, что существуют и специализированные Web-каталоги списков рассылки, такие как Geocrawler [28] или Mail-Archive [29]. В таких каталогах архивы списков рассылки классифицированы по категориям, присутствуют и некоторые средства навигации и поиска. Однако, и в этом случае информация о структуре индивидуальных писем обычно не учитывается. Существенным недостатком доступных каталогов является необходимость значительного объема ручной работы по классификации и предоставлению доступа к архивам. Автоматизация этой задачи, безусловно, может помочь значительно[3] улучшить охват и актуальность каталогов.

Наша работа основывается на целом ряде исследований, посвященных извлечению информации из HTML страниц. Так, для получения содержательных частей писем из страниц, на которых они опубликованы, требуется создание специальных адаптеров, что является частным случаем задачи построения адаптеров для

[3] Например, Google Web Directory заметно актуальнее Yahoo!. Тем не менее, задача автоматического создания каталогов не проста и не является решенной.

извлечения информации из слабоструктурированных источников данных, которая активно исследуется в течение нескольких последних лет [15,6,10,22].

Поскольку каждый адаптер ориентирован на работу с источниками информации с некоторой определенной структурой разметки, то для большого количества различных источников, вероятно, необходимо множество разных адаптеров (что верно в приложении к архивам списков рассылки). Поэтому, был предпринят ряд попыток автоматизации процесса создания таких адаптеров [19,4,23,27,11,14,9].

Некоторые из этих подходов [4] основаны на эвристиках, но большинство основано на использовании различных методов добычи знаний [12,11,27,16]. В работе [14] был предложен метод восстановления схемы реляционных таблиц. К сожалению, в основе этого метода лежит предположение о наличии множества записей одинаковой структуры на одной странице, что не верно в случае архивов списков рассылки. Отметим также иерархический метод, предложенный в работе [23]. Мы также придерживаемся идеи об использовании иерархического подхода при создании адаптеров.

Важно отметить, что все эти методы, как минимум[4], получают на вход серию примеров HTML-страниц с одинаковой структурой.

Использование информации о структуре и/или тематике при поиске информации также активно обсуждалось в научной литературе. Так, вопросы повышения качества за счет информации о тематике и тематическом контексте были предметом исследования работ [3,25,20,7]. Использование же информации о структуре изучалось в работах [18,21,2].

Наш подход

Для создания системы, предоставляющей интегрированный доступ к разнообразным архивам списков рассылки, доступным в Интернет, необходимо последовательно решить несколько задач.

1. Обнаружить потенциальные ссылки на архивы в Интернет.
2. Выделить среди потенциальных ссылок на архивы те, которые действительно ведут к архивам, отбросив “мусор”.
3. Выделить страницы, содержащие индивидуальные письма и отделить содержимое писем от окружающей разметки и возможного “мусора”.

Для построения масштабируемой системы необходимо минимизировать объем ручной работы, т.е. максимально автоматизировать решение всех задач. Наиболее важно минимизировать долю ручной работы в решениях второй и третьей задач, поскольку общий объем работы для решения этих задач велик.

Общая архитектура системы проиллюстрирована на рисунках 1 (задачи 1 и 2) и 2 (задача 3).

В следующих разделах мы подробнее остановимся на каждой из задач, а также обсудим некоторые возможные варианты использования собранной информации.

[4] Многие из них также требуют дополнительных данных и даже некоторого участия пользователя.

Обнаружение потенциальных ссылок на архивы

Для того, чтобы обеспечить универсальный доступ к архивам списков рассылки, прежде всего необходимо отыскать их в Интернет. Существуют несколько возможных подходов к поиску архивов.

Можно воспользоваться технологиями, используемыми поисковыми системами при построении индексов, т.е. использовать интеллектуальных роботов для сбора информации о ресурсах в Интернет [26]. Этот путь предоставляет большую гибкость и свободу при поиске информации. Даже в том случае, если робот используется для поиска не во всей доступной в Интернет информации, а только в некоторой ее части[5], можно использовать специализированные стратегии обхода сети, максимизирующие количество обнаруженных подходящих системе ресурсов [17,24,1]. Однако, реализация этого подхода требует значительных затрат на программную реализацию собственного робота, а кроме того, из-за огромного объема информации, доступной в Интернет, требуется наличие значительных вычислительных/сетевых и временных ресурсов на этапе практического сбора информации.

С другой стороны, поисковые системы общего назначения уже проиндексировали значительную часть[6] ресурсов Интернет[7] [5,8] и поэтому заманчивой кажется идея воспользоваться результатами их труда. Иными словами, можно осуществлять поиск архивов списков рассылки, посылая запросы на публично доступные поисковые сервера. Остается только научиться формулировать соответствующие запросы.

Поскольку целью этого шага является обнаружение не одного конкретного архива, а как можно большего их числа, то рассылаемые на поисковые сервера запросы должны учитывать общие черты, присущие многим архивам. В то же время запросы должны быть по возможности не очень “расплывчатыми” и не захватывать огромное множество не релевантных страниц. Запрос, который удовлетворяет таким (расплывчатым) пожеланиям, мы далее будем называть сильным запросом.

[5] Например, посвященной определенной тематике [25].

[6] Конечно, обычно они не индексируют динамические ресурсы или ресурсы, которые запрещены для доступа роботами при помощи механизмов наподобие robots.txt. Но эти правила редко затрагивают архивы списков рассылки.

[7] `http://www.searchenginewatch.com/reports/sizes.html`

Например, интуитивно формулируемый запрос “mailing list archives” относительно “сильный”, так как более сотни первых ссылок возвращенных для этого запроса системой Altavista соответствуют архивам списков рассылки. В то же время, запрос “+subject +message –forum +thread” еще “сильней”, хотя и не настолько интуитивно очевиден.

В Интернет доступны архивы десятков тысяч списков рассылки и мы хотим обнаружить их как можно больше. Часть архивов можно обнаружить при помощи составленных вручную запросов. Значительное количество ссылок на архивы можно почерпнуть также и из существующих каталогов архивов списков рассылки, таких как GeoCrawler [28] или MailArchive [29]. Тем не менее таким образом нельзя обнаружить все архивы, в частности, архивы сообщений на других языках.

Поэтому перспективной идеей кажется проведение автоматического анализа уже обнаруженных архивов с целью выявления общих черт и автоматического построения “сильных” запросов. Для этого мы планируем адаптировать идеи предложенные в работе [13] и применить методы добычи знаний [16].

Метод верификации ссылок

Вне зависимости от того, каким образом мы получили ссылку на архив (от поисковой системы, из существующего каталога или как-то еще), перед попыткой включить этот архив в систему необходимо убедиться, что эта ссылка действительно указывает на архив. Или, другими словами, верифицировать ссылку.

Отметим, что нас в первую очередь интересует обнаружение так называемой “точки входа” в архив, т.е. такой страницы, от которой доступно[8] все содержимое архива, и не существует другой подобной страницы с более короткими путями доступа к содержимому. Типичной точкой входа является страница, содержащая упорядоченный список тем всех писем, опубликованных в архиве.

Очевидно, что проверяемая ссылка может не быть точкой входа, а указывать на страницу внутри архива. Поэтому, если рассматриваемая ссылка не верифицирована положительно, то мы также рассматриваем страницы в окрестности заданной и проверяем их на то, что они являются точкой входа. При этом предпочтение отдается проверке страниц, которые ссылаются на рассматриваемую.

Процедура верификации основана на выполнении ряда тестов, проверяющих соблюдение различных (эвристических или нет) свойств, присущих архивам списков рассылки. Примерами таких тестов являются:

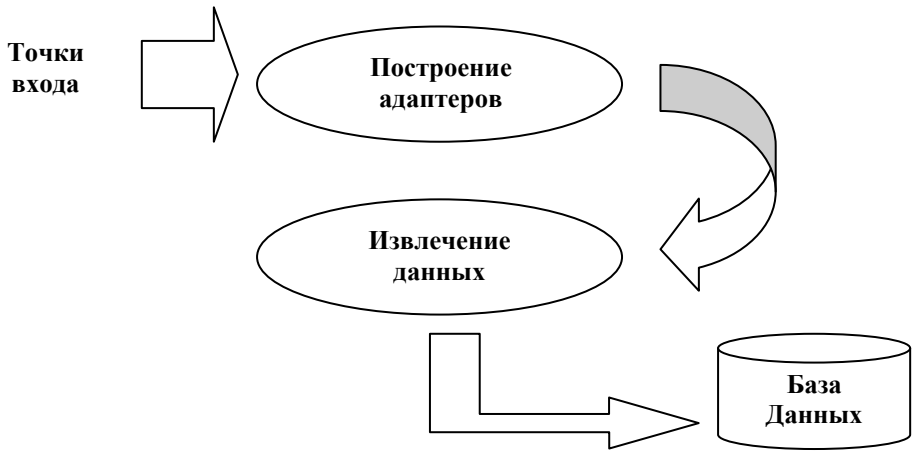
- Наличие либо непосредственно по ссылке, либо на странице второго уровня информации, состоящей из большого количества однотипных записей (например, перечисления или таблицы). С некоторой вероятностью таковые страницы могут быть точками входа (назовем их возможными точками входа).
- Наличие на втором или третьем уровне дерева, с корнем в возможной точке входа, страниц с большим количеством текстовой информации. Это могут быть отдельные письма (более точно это будет выяснено на следующем шаге) и их присутствие повышает вероятность того, что возможная точка входа в действительности таковой является, и, как следствие, вероятность того, что исходная ссылка ведет в архив.
- Наличие непосредственно по ссылке и на страницах нескольких нижележащих уровней большого количества текстовой информации (цепочка из индивидуальных писем).

После выполнения процедуры верификации у нас имеется список актуальных ссылок и соответствующих им точек входа в архив (см. 1).

[8] Страница A доступна со страницы B если из B можно попасть на страницу A при помощи навигации по гиперссылкам за конечное число шагов.

Извлечение индивидуальных писем

После обработки и верификации ссылок и фиксации точек входа мы переходим к извлечению отдельных писем. Эта процедура разбивается на несколько этапов.



Для начала необходимо идентифицировать страницы с отдельными письмами. Поскольку мы предполагаем, что такие архивы создаются автоматически, то естественно считать, что страницы с письмами имеют примерно одинаковую структуру HTML-разметки. Точнее сказать, структуры таких страниц значительно сильнее похожи, чем структура страницы с письмом и другой случайно выбранной страницы.

Поэтому для выявления страниц с письмами мы анализируем структуру всех страниц, доступных из точки входа по пути длины не более, чем L, и выделяем (путем кластеризации по структуре) наибольшую группу страниц со схожей структурой, страницы из которой считаются содержащими письма. Длина L выбирается эвристически и пока лучшие практические результаты получаются при использовании значения L=3. Еще одним возможным критерием отбора является наличие относительно длинных кусков текстовой информации (т.е. тело письма) в приблизительно одном и том же месте в большинстве страниц полученной группы.

Далее необходимо проанализировать структуру добытых страниц с письмами. Она может довольно сильно варьироваться даже внутри одного архива, а мелкие отличия (наличие или отсутствие закрывающих тегов, баннеров и т.п.) встречаются очень часто. Нам же необходимо универсальным образом извлечь информацию из всех добытых писем.

Для решения этой задачи мы используем результаты иерархической кластеризации [16] документов. Однако, в нашем случае представляющий документ вектор описывает не текстовое содержимое документа, а структуру используемой в документе разметки.

Далее, мы используем полученное дерево кластеров для построения адаптеров. Документы в кластерах самого нижнего уровня имеют довольно схожую (и главное “регулярную”) структуру и к ним можно попробовать применить один из известных методов автоматического построения адаптеров [9,27,19,4].

Построив адаптер для документов какого-нибудь кластера, его можно попытаться применить к документам из кластера уровнем выше (в построенном ранее дереве кластеров). Даже если этот адаптер не подошел сразу для некоторых страниц, его все еще можно попытаться автоматически исправить, попробовав использовать механизм исключений [6]. Такой подход позволяет уменьшить общее число адаптеров, хотя бывает полезно использовать более одного адаптера для одного архива.

Последним этапом является сопоставление семантических понятий определенных в RFC, таких как “Subject”, “From”, “Message body”, и выделенных элементов структуры. Для этого мы используем эвристический подход, основанный на наблюдениях вида:

- тело письма – это самый длинный текстовый фрагмент письма в среднем (по всем письмам)
- поля отправитель и получатель обязательно содержат хотя бы один адрес электронной почты, в частности, символ “@”
- тема письма, тело письма и отправитель присутствуют всегда

Таким образом мы пытаемся построить отображение, для которого выполняются все заданные эвристические правила.

Обработка полученной информации

Интеграция архивов в единую систему сама по себе не представляет законченного решения и необходимо подумать о способах организации доступа к собранной информации.

Вот только некоторые возможные варианты:

- **Навигация по архивам**
  Это классический вид сервиса, предоставляемый, например, GeoCrawler [28]
- **Улучшенный поиск**
  Учитывая структуру страниц, не рассматривать статические части документов при проведении поиска и, тем самым, повысить качество поиска. Так, например, по слову “Subject” будут возвращены не все собранные системой сообщения, а только некоторые из ник, которые используют это слово в теле письма или заголовке.
- **Поиск по полям**
  Используя знание о структуре сообщений, можно проводить поиск с учетом точного указания поля, где это слово может встречаться Так, например, можно искать только сообщения с заданным словом в поле “From”.
- **Построение рубрикатора**
  Поскольку письма из одного архива имеют общий тематический контекст, то можно пытаться автоматически построить некоторое описание тематики архива, а также классифицировать архив, используя некоторый существующий тематический рубрикатор типа Yahoo! или List.Ru.
- **Поиск архивов по описанию тематики**
  Запрос пользователя может использоваться не для поиска индивидуальных писем, а для того чтобы обнаружить список рассылки, в котором происходит обсуждение вопросов на интересующую пользователя тему. Отметим, что это далеко не всегда возможно сделать, используя только тематический рубрикатор.

Мы планируем попробовать реализовать эти способы в рамках нашего прототипа.

Экспериментальные результаты

На данный момент система еще не реализована и проводятся эксперименты, оценивающие эффективность выбранного подхода на каждом из этапов. Результаты этих экспериментов будут представлены позже. Мы также надеемся, что к моменту публикации система будет в значительной степени реализована и доступна в Интернет.

Заключение

Целью этой работы является изучение применимости передовых разработок в области работы со слабоструктурированной информацией и информационного поиска к решению практических задач.

В статье описан подход к решению задачи предоставления единого доступа к находящимся в Интернет архивам списков рассылки. Наш опыт показывает, что несмотря на обилие исследовательских работ в близких областях множество вопросов все еще требуют дополнительных исследований.

Проект находится в стадии экспериментальной проверки работоспособности выбранных подходов к решению отдельных подзадач и созданию прототипа системы. Прототип доступен по адресу `http://meta.math.spbu.ru/MAS/`.

Библиография

1 Е.В. Романова, М.В. Романов, и И.С. Некрестьянов. Использование интеллектуальных сетевых роботов для построения тематических коллекций. Программирование, 3:63-71, 2000.

2 Е.Ю. Павлова и А.В. Томашевский. Использование информации о структуре HTML-документа при построении профайлов. Труды Всероссийской научно-методической конференции ``Интернет и современное сообщество'', Санкт-Петербург, декабрь 1998.

3 И. Некрестьянов. Тематико-ориентированные методы информационного поиска. Диссертация на соискание степени к.ф.-м.н., Санкт-Петербургский Государственный Университет, 2000.

4 N. Ashish and C. Knoblock. Wrapper generation for semi-structured internet sources. In Proc. of the Workshop on Management of Semistructured Data, Tucson, Arizona, 1997.

5 Krishna Bharat and Andrei Broder. A technique for measuring the relative size and overlap of public web search engines. May 1998.

6 V. Crescenzi and G. Mecca. Grammars have exceptions. Information Systems, Special Issue on Semistructured Data, 1998.

7 Brian D. Davison. Topical locality in the web. In Research and Development in Information Retrieval, pages 272-279, 2000.

8 Kemal Efe, Vijay Raghavan, Adrienne Chu, C. Henry L. Broadwater, Levent Bolelli, and Seyda Ertekin. The shape of the web and its implications for searching the web. In Proc. of the International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet, Rome, Italy, August 2000.

9 D. Freitag. Using grammatical inference to improve precision in information extraction. In Working Papers of the ICML-97 Workshop on Automata Induction, Grammatical Inference, and Language Acquisition., 1997.

10 D. Freitag. Information extraction from html: Application of a general learning approach. In Proc. of the Fifteenth Conference on Artificial Intelligence AAAI-98, pages 517-523, 1998.

11 D. Freitag and N. Kushmerick. Boosted wrapper induction. In Proc. of the AAAI-2000, 2000.

12 Dayne Freitag and Andrew McCallum. Information extraction with hmm structures learned by stochastic optimization. In Proc. of the AAAI'2000, 2000.

13 Eric Glover, Gary Flake, Steve Lawrence, William P. Birmingham, Andries Kruger, C. Lee Giles, and David Pennock. Improving category specific web search by learning query modifications. In Symposium on Applications and the Internet, SAINT, pages 8 -12, San Diego, CA, January 2001.

14 S. Grumbach and G. Mecca. In search of the lost schema. In Proc. of International Conference on Database Theory, 1999.

15 J. Hammer, H. Garcia-Molina, J. Cho, R. Aranha, and A. Crespo. Extracting semi-structured data from the web. In Proc. of Workshop on Management of Semi-structured Data, pages 18-25, 1997.

16 Jiawei Han and Micheline Kamber. Data Mining: Concept and Techniques. Morgan Kaufmann Publishers, 2001.
17 Cho Junghoo, Garcia-Molina Hector, and Page Lawrence. Efficient crawling through URL ordering. In Proceedings of the Seventh International World Wide Web Conference, 1998.
18 U. Kruschwitz. Exploiting Structure for Intelligent Web Search. In Proceedings of the 34th Hawaii International Conference on System Sciences (HICSS), Maui, Hawaii, 2001. IEEE.
19 N. Kushmerick, D. Weld, and R. Doorenbos. Wrapper induction for information extraction. In Proc. of the IJCAI-97, August 1997.
20 Steve Lawrence. Context in web search. IEEE Data Engineering Bulletin, 23(3):25-32, 2000.
21 L. Liu, C. Pu, and W. Han. Xwrap: An xml-enabled wrapper construction system for web information sources. In Proc. of the International Conference on Data Engineering, pages 611-621, 2000.
22 Ion Muslea. Extraction patterns for information extraction tasks: A survey. In Proc. of the AAAI-99 Workshop on Machine Learning for Information Extraction, 1999.
23 Ion Muslea, Steve Minton, and Craig. A. Knoblock. A hierarchical approach to wrapper induction. In Proc. of the 3rd International Conference on Autonomous Agents, Seattle, WA, 1999.
24 Igor Nekrestyanov, Tadhg O'Meara, and Ekaterina Romanova. Building topic-specific collections with intelligent agents. In Proc. of Sixth International Conference on Intelligence in Services and Networks (IS&N'99), volume 1597 of Lecture Notes in Computer Science, Barcelona, Spain, April 1999. Springer.
25 A. Patel, L. Petrosjan, and W. Rosenstiel, editors. OASIS: Distributed Search System in the Internet. St. Petersburg State University Published Press, St. Petersburg, 1999.
26 Brian Pincerton. Finding what people want: Experiences with the webcrawler. In Proc. of the second International World-Wide Web Conference, 1992.
27 S. Soderland. Learning information extraction rules for semi-structured and free text. Machine Learning, 34, 1999.
28 geocrawler.com. http://www.geocrawler.com
29 Mail-archive.com. http://www.mail-archive.com

Integrated access to the Web mailing list archives

D.Barashev, S.Coox, E.Michailova, I.Nekrestyanov, B.Novikov, E.Pavlova, A.Vysotsky.
University of Saint-Petersburg
E-mail: mas-project@meta.math.spbu.ru

Our work addresses the problem of providing a unified integrated access to the Web based mailing list archives which are considered as an example of semi-structured data. Due to specific structure of e-mail messages, general-purpose search engines are not effective for certain important classes of queries to such archives, for example for searching text in e-mail subject only. The proposed approach is based on combination of semistructured data processing and information retrieval techniques.

The major focus of the work is on automatic archive identification and wrappers generation. Several heuristics are explored and evaluated, such as searching links to archive pages, links verification, individual mails extraction and archives categorization. Possible ways of the practical usage of such system are described.