Восстановление связей между библиографическими записями

А.А.Князева
Институт вычислительных технологий СО РАН
amili@mail.ru

О.С. Колобов
okolobov@hcei.tsc.ru

Институт сильноточной электроники СО РАН

В работе рассматривается задача восстановления отсутствующих или утраченных связей между структурированными документами (с известной структурой) в ситуациях, когда между соответствующими записями могут встречаться расхождения на уровне полей. Причинами расхождений могут быть типографические ошибки, неполнота информации, различия в традициях и привычках каталогизаторов. Частным случаем задачи является выявление, связывание и слияние дублирующихся записей, особенно актуальное при создании сводных каталогов. Предлагается алгоритм автоматического сопоставления пар записей и вынесения решения об их соответствии друг другу. Работа алгоритма иллюстрируется на примере связывания библиографических записей с авторитетными записями имен авторов в формате RUSMARC.

1. Введение

В рамках работы рассматривается задача восстановления отсутствующих или утраченных связей между записями. Под связыванием записей понимают сравнение информации из различных источников данных с целью определения, какие пары записей представляют один и тот же объект реального мира [1, 2]. Таким объектом может быть, например, некоторый документ, автор или организация.

Частным случаем связывания является задача выявления дублирующихся записей в одном или нескольких источниках. В этом случае речь идет о нечетких дубликатах, поскольку нередки ситуации, когда дублирующиеся записи имеют различные значения в одном или нескольких полях [1]. Причинами такого несоответствия могут быть опечатки, транспозиции символов, измененный порядок слов, использование сокращений и аббревиатур, разночтения в зарубежных транскрипциях, неполнота данных и т.п. [3].

Впервые задача автоматического связывания была сформулирована Ньюкомби [4] в контексте сопоставления записей о рождениях с записями о регистрации брака. В дальнейшем для идей Ньюкомби была разработана формальная математическая модель, получившая название вероятностной модели связывания, основанной на ошибках [5], на которой в настоящее время основано целое семейство вероятностных моделей, например, модели основанные на штрафах или использующие EM-алгоритм [1]. Важной особенностью вероятностных моделей связывания является необходимость принятия предположений о распределении вероятностей признаков соответствия записей. Альтернативный подход,

позволяющий не строить таких предположений, основан на методиках обучения с учителем.

В настоящее время существуют системы автоматического связывания, такие как MARLIN [11], TAILOR [1], Febrl [9], VIAF [10] и другие. За исключением проекта VIAF приведенные системы настроены на работу с относительно несложными структурами данных, в том числе с данными содержащими неразмеченный текст. Задачей проекта VIAF, созданного в рамках Международной федерации библиотечных ассоциаций (IFLA), является связывание авторитетных записей имен авторов в формате UNIMARC.

Таким образом, основные отличия задачи, поставленной в данной работе заключаются в в том, что связываются структурированные, при этом записи принадлежат к разным типам, и имеется возможность обучения на уже имеющихся в системе данных.

2. Модель системы связывания

Рассмотрим задачу связывания применительно к библиографическим данным. В качестве объекта реального мира выступает некоторый автор, записи, имеющие к нему отношение, разделяются на два типа - авторитетная запись содержит информацию о самом авторе, библиографическая - о его произведении. При этом в библиографической записи также содержится некоторая (хотя возможно неполная) информация об авторе.

В работе предлагается модель системы автоматического связывания записей, состоящая из следующих блоков:

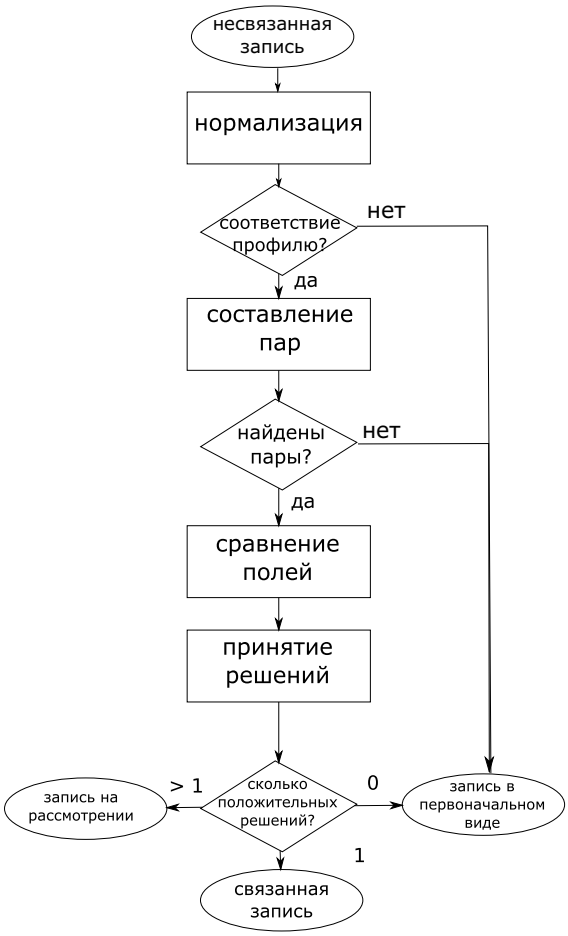
1. Нормализация;
2. Составление пар;
3. Сравнение отдельных полей в парах записей;
4. Вынесение решения для каждой из пар;
5. Обучение решающей функции;
6. Оценка качества связывания.

Первые четыре блока непосредственно участвуют в процедуре связывания (рис. 1), последние два включаются в работу периодически при расширении базы данных. Принцип работы у них общий: для записи, относительно которой уже известно правильное решение (с какой из авторитетных записей ее нужно связать) проводится процедура связывания и в первом случае уточняются параметры системы, а во втором оценивается, насколько успешно система справилась с задачей.

2.1. Нормализация

Блок нормализации решает две задачи: проверка записи на соответствие профилю и анализ ее отдельных полей. Проверка на соответствие профилю позволяет определить достаточно ли информации, содержащейся в записи, для связывания. Анализ отдельных полей предназначен для очистки и нормализации данных.

Рис. 1. Модель процедуры связывания



На практике большинство коллекций данных содержат засоренную, неполную, неправильно форматированную информацию. Очистка данных и их нормализация - необходимые этапы подготовки данных перед их загрузкой в хранилище и использованием для дальнейшего анализа. Особенно важно решение этих задач в распределенных системах. Цель нормализации данных - избавиться от вариаций в написании, возникающих из-за сокращений, перестановки слов и т.п.

Существует множество подходов к нормализации данных. Это может быть использование конечного словаря для значений поля, автоматическая разметка текста на естественном языке для определения о каком объекте идет речь и т.п.

В данной работе рассматриваются записи в формате RUSMARC, созданные профессиональными каталогизаторами, поэтому в блоке нормализации достаточно только проверять запись на соответствие профилю.

2.2. Составление пар

Сравнение входящего документа с каждым из авторитетных документов, как правило, очень трудоемкий процесс. Поэтому необходимо сократить количество авторитетных документов, которые будут сопоставляться с входящим. Существует множество способов ограничить круг записей для сопоставления [6]. Приведем некоторые из них.

1. Метод стандартных блоков выделяет записи в один блок в том случае, если они содержат идентичный блочный ключ [7]. Блочные ключи формируются на основе атрибутов записей, например, первые 4 символа фамилии. Кроме того, блочный ключ может быть и составным, например, атрибут «индекс» может сочетаться с атрибутом «возраст». Ключи должны быть выбраны таким образом, чтобы блоки не были ни слишком большими, ни слишком мелкими.

2. Метод ближайших соседей [8] сортирует записи на основе сортирующего ключа и затем двигает окно фиксированного размера ω последовательно по всем записям. Записи внутри окна составляют пары друг с другом и включаются в список пар-кандидатов. Использование окна ограничивает число возможных сравнений для каждой записи до $2\omega-1$. Метод может некорректно работать в том случае, если количество записей с одним значением ключа превышает размер окна, в такой ситуации не все записи будут сравниваться.

3. Метод Bigram-индексирования [9] предназначен для нечеткого разбиения на блоки. Основная идея заключается в том, что значения блочных ключей конвертируются в лист биграм (подстрок, состоящих из двух символов) и затем из этих биграм формируются списки на основе заданного порога.

В рамках данной работы принят метод поиска по составному ключу, состоящему из двух значений: фамилия и инициалы автора. Значение ключа определяется по входящему документу, а поиск производится в авторитетной базе данных. При этом используется точное сопоставление. Такой механизм позволяет существенно снизить трудоемкость.

Кроме того, для сравнения с входящей записью предлагается использовать расширенную авторитетную запись [10]. Расширенная авторитетная запись кроме самой найденной авторитетной записи включает информацию из библиографических записей, уже хранящихся в системе и связанных с ней. Такой подход позволяет увеличить объем информации для сравнения и получать более точные результаты.

2.3. Сравнение отдельных полей в парах записей

Цель блока сравнения отдельных полей заключается в оценке того, насколько записи совпадают по различным параметрам. Результатом работы блока является вектор, составленный из оценок близости двух строк, которые являются значениями соответствующих полей. Существует огромное разнообразие методов сопоставления строк, учитывающих различные аспекты сходства. Множество методов можно классифицировать в соответствии с тем, на чем они базируются, как определяются их параметры и в каком виде представляются результаты сопоставления.

В основе метода сопоставления строк могут быть символы (как отдельные символы, так и q-граммы, наборы подстрок длины q) или токены. В качестве примеров методов, базирующихся на токенах, можно привести Метрику Джаккарда или косинусную меру

сходства в векторном пространстве. Методы, работающие с набором подстрок определенной длины позволяют сравнивать не целые слова, а комбинации в них, что может быть полезно при наличии орфографических ошибок. Символьные метрики, такие как расстояние Левенштейна и его различные варианты, вычисляют подобие между строками, оценивая минимальное количество изменений, которые достаточны для перевода одной строки в другую. В случае, когда данные представлены относительно короткими строками, которые содержат одинаковые, хотя и орфографически различно записанные слова, символьные меры предпочтительнее, поскольку они могут оценить разницу между строками более детально [11].

Далее методы можно классифицировать по тому, как вычисляются их параметры, например «стоимость» операции редактирования или вес токена. Параметры могут быть фиксированными, вручную подобранными исследователем (контекстно-независимые методы), вычисленными на основе характеристик БД или полученными в результате обучения с учителем (контекстно-зависимые). В случае, если используются контекстно-зависимые методы, включающие обучение с учителем, необходимо определить обучающую выборку.

Результаты сопоставления строк также могут варьироваться от одного метода к другому, они могут быть записаны в виде бинарных, категориальных, порядковых или непрерывных величин.

Например, классический метод Левенштейна [12], определяющий расстояние как минимальное число вставок, удалений или замен, необходимых для перевода одной строки в другую, относится к символьным методам с фиксированными параметрами (следовательно, контекстно-независимый) и непрерывной переменной результата.

В рамках настоящей работы используется комбинация точного сравнения и сравнения с усечением по методу Snowball для русского языка.

2.4. Вынесение решения для каждой из пар

Соответствие на уровне записей необязательно означает однозначное соответствие на уровне полей.

Блок вынесения решения призван провести анализ сравнительного вектора, полученного для пары записи (авторитетного и библиографического) и принять одно из двух возможных решений: соответствуют или не соответствуют эти записи друг другу. Рассмотрим три подхода к построению решающей функции [1].

1. Индукционная модель связывания записей: в основе лежит машинное обучение с учителем, предполагаем, что есть обучающая выборка, в которой для каждого образца точно известен класс. Эта выборка используется для построения классификатора, приванного относить любой новый образец к определенному классу.

2. Кластерная модель связывания записей — это модель обучения без учителя, она не требует обучающей выборки. Принцип таков: разбиваем все пары на три кластера с помощью некоторого алгоритма кластеризации, затем определяем какой кластер относится к какому статусу: соответствие, несоответствие или возможное соответствие.

3. Гибридная модель. На первом шаге используя кластерную модель для анализа некоторого количества пар, затем эти пары становятся обучающей выборкой для применения обучения с учителем.

В рамках данной работы используется классификация пары документов к классу соответствующих, либо к классу несоответствующих пар, которая производится с помощью расстояния Махалонибиса, известного тем, что оно учитывает коррелированность признаков и инвариантно к масштабу.

3. Заключение

Задача автоматического связывания структурированных документов актуальна и имеет множество приложений в различных областях. В том числе в электронных каталогах библиотек, в особенности распределенных. При разработке систем автоматического связывания необходимо так или иначе решать вопросы, упомянутые в докладе, такие как нормализация данных, сокращение перебора и так далее. Кроме того, как мне кажется, автоматическое связывание нужно налаживать с учетом сильных и слабых сторон каждой конкретной области. Так например, при работе с библиографическими записями в формате RUSMARC можно не так много внимания уделять автоматической разметке неразмеченного текста, как при анализе названий организаций, записанных в одну строку. Однако возникает проблема относительной сложности формата и это усиливает требования к решающей функции, используемой в системе.

Описанная в данной работе модель автоматического связывания была реализована в ходе эксперимента на основе данных предоставленных НП МедАрт. В результате связывания для тестовой выборки, состоящей из 624 пар записей 622 пары были связаны верно и 2 пары ошибочно. Полученные результаты в целом соответствуют ожидаемым, для уменьшения ошибок связывания необходимо дальнейшее развитие предлагаемого подхода.

Особенности предлагаемого подхода заключаются в использовании расширенных авторитетных записей и возможности настройки системы на конкретную базу данных. Такая настройка позволяет повысить точность решений за счет оценки веса совпадения для набора полей записей.

Список литературы

[1] Elfeky, M., Vassilios, V., and Elmagarmid, A.: TAILOR: A Record Linkage Tool Box. ICDE 2002: 17-28.

[2] Winkler, W. E. Overview of Record Linkage and Current Research Directions. Research Report Series, RRS: Statistics #2006-2. http://www.census.gov/srd/papers/pdf/rrs2006-02.pdf.

[3] Рубцов Д.Н., Барахнин В. Б. Выявление дубликатов в разнородных библиографических источниках // Вестник НГУ. Сер.; Информационные технологии. 2009. Т. 7. Вып. 3. С.86-93.

[4] H.B. Newcombe, J.M. Kennedy, S.J. Axford, and A.P. James. Automatic linkage of vital records. Science, 130:954-959, 1959.

[5] I.P. Fellegi and A.B. Sunter. A theory for record linkage. Journal of the American Statistical Association, 64: 1183-1210, 1969.

[6] Baxter, R., Christen, P., and Churches, T. (2003), "A Comparison of Fast Blocking Methods for Record Linkage,"Proceedings of the ACM Workshop on Data Cleaning, Record Linkage and Object Identification, Washington, DC, August 2003.

[7] M.A. Jaro. Advances in Record Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. Journal of the American Statistical Society, 84(406): 414-420, 1989.

[8] M.A. Hernandez and S.J. Stolfo. Real-world data is dirty: data cleansing and the merge/purge problem. Journal of Data Mining and Knowledge Discovery, 1(2), 1998.

[9] P. Christen and T. Churches. Febrl: Freely extensible biomedical record linkage Manual, release 0.2.2 edition, November 2003.

[10] Bennett, Rick, Christal Hengel-Dittrich, Edward T. O'Neill, and Barbara Tillett. 2007. "VIAF (Virtual International Authority File): Linking the Deutsche Nationalbibliothek and Library of Congress Name Authority Files."International Cataloging and Bibliographic Control 36,1: 12–19.

[11] Bilenko M. Learning to Combine Trained Distance Metrics for Duplicate Detection in Databases / M. Bilenko, R. Mooney. Technical Report AI-02-296, Artificial Intelligence Lab, University of Texas at Austin, 2002.

[12] V.I. Levenshtein. Binary codes capable of correcting insertions and reversals. Soviet Physics Doclady, 10(8): 707-710, Feb. 1966.