Автоматическое связывание документов

© А.А. Князева © И.Ю. Турчановский © О.С. Колобов
Томский филиал Института вычислительных технологий СО РАН Томск
aknjazeva@ict.nsc.ru tur@hcei.tsc.ru
Институт сильноточной электроники СО РАН Томск
okolobov@hcei.tsc.ru

Аннотация

В работе рассматривается задача автоматического связывания документов, относящихся к одному и тому же объекту реального мира. Предлагается алгоритм, основанный на классификации с использованием расстояния Махаланобиса. Работа алгоритма иллюстрируется на примере связывания библиографических записей и авторитетных записей имен авторов в формате машиночитаемой каталогизационной записи (MARC).

1 Введение

В рамках работы рассматривается задача восстановления отсутствующих или утраченных связей между документами в контексте библиотечных данных. В качестве документа может выступать как запись из электронного каталога библиотеки, так и полнотекстовый документ с внедренными метаданными. Главное требование к документу - он должен содержать информацию о свойствах объекта в виде набора атрибутов с определенной структурой. В качестве примера работы алгоритма в данной работе приводятся результаты эксперимента по связыванию библиографических записей и авторитетных записей имен авторов. Тем не менее, подход является достаточно общим и может быть перенесен на связывание полнотекстовых документов с авторитетными записями. Под связыванием документов в рамках работы понимают сравнение информации из различных источников данных с целью определения, какие пары документов представляют один и тот же объект реального мира [4, 17]. Таким объектом может быть, например, некоторый документ, автор или организация. Эта задача также известна под названием связывания записей, идентификации сущностей и т.п.

Труды 14-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» — RCDL-2012, Переславль-Залесский, Россия, 15–18 октября 2012 г.

В более общей формулировке задача связывания может быть поставлена для документов разных типов, имеющих различную структуру. В то же время, задача связывания документов одного типа, то есть выявления дублирующихся документов в одном или нескольких источниках является частным случаем рассматриваемой задачи. Разумеется, при этом речь идет о нечетких дубликатах, поскольку нередки ситуации, когда дублирующиеся документы имеют различные значения в одном или нескольких полях [4]. Причинами такого несоответствия могут быть опечатки, перестановки слов и символов, пропуски данных, а также привычки и традиции каталогизаторов.

Безусловно, самым простым подходом к задаче связывания является принятие решения о соответствии записей на основе некоторых правил. Эти правила могут быть относительно простыми или достаточно сложными, в зависимости от конкретной системы. Такой подход к установлению связей можно назвать детерминистическим. Однако на практике далеко не всегда есть возможность выработать исчерпывающий набор правил, особенно в условиях, когда часть информации отсутствует.

2 Связанные работы

Впервые задача автоматического связывания без применения фиксированных правил была сформулирована Ньюкомби [14] в контексте сопоставления записей о рождениях с записями о регистрации брака. Суть предложенного решения заключается в подсчете количества совпавших полей. Если это количество превышает некоторый заданный заранее порог, то записи признаются соответствующими, в противном случае - несоответствующими. В дальнейшем для идей Ньюкомби была разработана формальная математическая модель, получившая название вероятностной модели связывания Fellegi-Sunter (FS) [5], на которой в настоящее время основано целое семейство вероятностных моделей, например, модели основанные на штрафах или использующие EM-алгоритм [4]. Описанный

подход основан на явной оценке условных вероятностей соответствия записей, он предполагает знание распределения признаков соответствия или их взаимную независимость [1].

Альтернативой является более прямой подход, основанный на методиках машинного обучения [2]. Это может быть обучение с учителем или без него. Основная идея заключается в том, чтобы относить пару документов к классу соответствующих или несоответствующих пар на основании ее схожести с остальными парами класса. Применение такого подхода не требует независимости признаков, что существенно расширяет область применения.

Диапазон существующих систем связывания документов достаточно широк: техники для установления RDF-ссылок в Веб, базы данных с адресами клиентов и организаций, системы для связывания демографической и медицинской информации о персонах и др.

В отдельную группу можно выделить методы, предназначенные для связывания данных в Веб с помощью установления RDF-ссылок, реализованные, напримерм в системе Silk [8]. Хотя в основе таких методик лежат те же общие предположения, что и во всех системах связывания, специфичность области применения не позволяет непосредственно использовать данные системы для решения рассматриваемой задачи.

Наиболее многочисленной является группа систем, настроенная на поиск дубликатов в одном или нескольких текстовых файлах (либо в реляционых БД), содержащих сведения об именах, почтовых адресах, телефонах, номерах страховки и т.п. Чаще всего, для принятия решения о соответствии записей используется набор правил или классическая вероятностная модель F-S. В первом случае система предоставляет пользователю возможность определения того, насколько важно совпадение по тому или иному признаку. На практике это может быть достаточно трудно сделать, поскольку не всегда в распоряжении пользователя есть такая информация. Во втором случае вес того или иного признака вычисляется автоматически, на основании функции правдоподобия (например, система AutoMatch [10]). При этом принимается предположение о функции распределения признаков, а также их взаимной независимости. Также, многие системы предлагают использование одного из описанных механизмов на выбор пользователя (Febrl [3], FRIL [11] и другие). Системы этой группы отличаются друг от друга гибкостью настройки, инструментами для нормализации данных и сравнения отдельных полей, возможностями визуалицации и т.п. К недостаткам систем этой группы с точки зрения решаемой задачи можно отнести то, что они не поддерживают данные сложной структуры, требуют установления правил связывания в явном виде или принятия предположений о функции распределения признаков и их взаимной независимости, которые часто не выполняются на практике.

Группу систем, работающих с библиографическими данными, в свою очередь можно разделить на две части: системы для «простого» формата библиографической ссылки (такого как BiBTEX или неструктурированная библиографическая запись) и системы для работы с «профессиональными» форматами (семейство MARC-форматов). Первая группа систем вынуждена больше внимания уделять такой частной задаче, как автоматическая разметка неразмеченного текста (чтобы выделять из текстовой строки элементы библиографического описания). Во второй группе такая необходимость отпадает благодаря сложной структуре формата, но в то же время появляется необходимость учета этой структуры, в которой одна и та же информация может быть внесена по-разному, в зависимости от предпочтений каталогизаторов. К первой группе можно отнести системы DIFWICS [7] и MARLIN [2], а ко второй проект VIAF [16]. Однако, хотя в проекте VIAF и реализована работа с данными в MARC-формате, он не предлагает механизма для автоматической оценки весов признаков, поскольку основан на использовании эмпирических правил. Кроме того, проект нацелен на поск дублирующихся записей, а не связывание записей различных типов. Таким образом, несмотря на некоторое сходство, не представляется возможным заимствовать подход, использованный в проекте VIAF для решения поставленной задачи.

3 Модель системы связывания

Для процедуры связывания необходимо определить несколько основных моментов.

Так, необходимо задать правила для определения того, достаточно ли информации для связывания содержится в записи. Кроме того, необходимо определить правила для нормализации данных, которые бы позволили стандартизировать значения (например, с помощью словаря допустимых значений).

Далее следует предусмотреть варианты сокращения перебора при поиске записей-кандидатов для связывания, поскольку в крупных хранилищах затраты на подробный анализ всех возможных пар записей могут быть неприемлимо большими.

Еще одним важным моментом является выбор способа для сравнения значений на уровне

полей. Даже при проведении нормализации, использование строгого сравнения может быть необоснованным. Зачастую необходимо оценить степень соответствия полей записей, чтобы учесть и частичное соответствие информации.

Когда оценивается степень подобия между записями, состоящими из множества полей, возникает необходимость комбинировать оценки подобия для отдельных полей. Поскольку соответствие между общим подобием записей и подобием в отдельных полям может сильно варьироваться, то необходимо взвешивать поля и каким-либо образом оценивать вклад каждого из них в соответствие на уровне записи [2].

Перечисленные задачи, реализованные в большинстве систем связывания [4], можно рассматривать как этапы процедуры связывания:

1. Нормализация;
2. Составление пар;
3. Сравнение отдельных полей в парах записей;
4. Вынесение решения о соответствии.

Кроме данных четырех этапов, непосредственно участвующих в процедуре связывания, необходимо наличие еще двух: настройка системы и проверка качества связывания. 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.

Рассмотрим подробнее описанные выше этапы.

3.1 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.

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 производится только проверка записи на соответствие профилю.

3.2 Составление пар

Сравнение входящего документа с каждым из авторитетных документов, может оказаться необоснованно трудоемким процессом. В частности, при работе «на лету» может потребоваться сократить количество авторитетных документов, которые будут сопоставляться с входящим. Существует множество способов ограничить круг записей для сопоставления. Приведем некоторые из них.

1. Метод стандартных блоков выделяет записи в один блок в том случае, если они содержат идентичный блочный ключ [9]. 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 [6] 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, поскольку в такой ситуации будут сравниваться не все нужные записи.
3. Метод Bigram-индексирования [3] 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 (например, выбираются все записи, в которых встречается 80% биграм).

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 без использования сложных вычислений.

Одной из важных черт предлагаемого подхода, является использование расширенного авторитетного документа для сравнения с входящим документом, аналогичный подход используется в проекте VIAF [16]. Расширенная авторитетная запись кроме самой найденной авторитетной записи включает информацию из библиографических записей, уже хранящахся в системе и связанных с ней. Такой подход позволяет увеличить объем информации, задействованной в анализе, и получать более точные результаты.

3.3 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 [2].

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 использовалась комбинация точного сравнения и сравнения с выделением основы слова по методу Snowball [15] для русского языка. Такой выбор был обусловлен тем, что механизм стеммирования уже был реализован на момент разработки алгоритма.

3.4 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 документов (авторитетного и библиографического) и принять одно из двух возможных решений: соответствуют или не соответствуют эти записи друг другу.

Методы, используемые для решения задачи связывания документов разделяются на две обширные категории. Детерминистические методы в которых устанавливаются часто очень сложные правила и вероятностные методы, в которых для классификации пар документов используются статистические модели [3]. Вероятностные методы могут быть в свою очередь разделены на методы, основанные на классической вероятностной теории связывания [5], и более поздние подходы, использующие различные техники машинного обучения [2].

Основное отличие классической модели от методик машинного обучения заключается в том, что она основана на предположении того, что сравнительный вектор является случайным вектором, чья функция плотности различается для каждого из двух классов (совпадающих и несовпадающих пар документов). При этом предполагается, что классы этих плотностей известны заранее и их параметры можно оценить. Затем, задача сводится к вычислению вероятности принадлежности сравнительного вектора к каждому из классов при условии его конкретной реализации и выбора наиболее вероятного класса. Использование этого подход осложняется тем, что на практике, как правило, классы плотностей заранее неизвестны.

Альтернативный подход заключается в использовании методик машинного обучения, позволяющих не делать предположений о функциях плотности соответствующих и несоответствующих пар, а классифицировать пару документов на основе ее подобия одному из классов. Такой подход возможен, например, если выбрать некоторый центроид класса, а затем вычислить расстояние до центроидов обоих классов и выбрать наименьшее.

Рассмотрим три подхода к построению решающей функции [4].

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 индукционная модель. Классификация пары документов к классу соответствующих, либо к классу несоответствующих пар производится с помощью расстояния Махалонибиса [13], выбранного благодаря тому, что оно учитывает коррелированность признаков и инвариантно к масштабу. Учет коррелированности позволяет отказаться от предположения о взаимной независимости переменных, которое часто принимается при классификации, и работать с зависимыми признаками. В рамках решаемой задачи это является важным моментом, поскольку некоторые переменные достаточно сильно коррелированны.

Квадрат расстояния Махаланобиса между двумя точками X и Y, определенными в p-мерном пространстве можно записать в виде:

$$D^2(X,Y) = (X-Y)C^{-1}(X-Y)^T, \qquad (1)$$

где X и Y - векторы координат размерности p;

C^{-1} - матрица, обратная ковариационной матрице.

Заменив один или оба вектора в формуле (1) на вектор координат центроида первого класса μ_1 или второго μ_2, получим расстояние от точки до класса или расстояние между классами. На практике оценить расстояние Махаланобиса можно подставив соответствующие оценки средних значений и матрицы ковариации. В качестве оценки матрицы ковариации в формуле (1) будем использовать внутригрупповую матрицу ковариации W, элементы которой находится по формуле:

$$W_{ij} = \frac{1}{n. - 2} \sum_{k=1}^{g} \sum_{m=1}^{n_k} (X_{ikm} - X_{ik.})(X_{jkm} - X_{jk.}), \qquad (2)$$

где

g - число классов;

n_k - число наблюдений в k-м классе;

$n.$ - общее число наблюдений по всем классам;

X_{ikm} - величина переменной i для m-го наблюдения в k-м классе;

$X_{ik.}$ - средняя величина переменной i в k-м классе.

Воспользовавшись расстоянием Махалонобиса можно произвести отбор наиболее информативных признаков (то есть признаков, позволяющих наиболее четко разделить классы), а также спрогнозировать принадлежность к классу для новых наблюдений (вычислив расстояния до обоих классов и выбрав наиболее близкий класс).

4 Описание эксперимента

При проведении эксперимента ставилась цель проанализировать пригодность предлагаемого алгоритма для решения задачи автоматического

связывания библиографических записей с авторитетными записями имен авторов. Данные для эксперимента были предоставлены НП МедАрт [18].

Эксперимент проводился на системе, включающей:

1. Библиографическую базу данных (ББД), около 300 000 записей в формате RUSMARC;
2. Авторитетный файл имен авторов (АФА), около 10 000 записей в формате RUSMARC AUTHORITY.

На основе АФА был составлен список фамилий с инициалами, соответствующих сразу двум и более авторитетным записям (42 фамилии с инициалами). Для каждой из фамилий были составлены пары из авторитетной записи (АЗ) и библиографической записи (БЗ), всего 1215 пар. Полученное множество было случайным образом разделено на обучающую и тестовую выборки.

Кроме наличия однофамильцев к авторитетным записям предъявлялось требование полноты: наличие информации о дате рождения, географических и профессиональных дополнений, аннотации (наличие полей 001, 200 ($a, $b, $c, $f, $y), 830$a).

В рамках эксперимента намеренно игнорировалась информация о расшифровке инициалов (200 $g) для увеличения области совпадения авторитетных и библиографических записей и, следовательно, объема обучающей выборки. Разумеется, рабочий алгоритм не будет игнорировать эту информацию, что позволит повысить его точность.

В свою очередь, библиографическая запись обязательно должна была содержать указание на авторитетную запись (наличие поля 701$3) для того, чтобы можно было ответить на вопрос о ее принадлежности. В рамках эксперимента требование полноты к БЗ не предъявлялось, хотя очевидно, отсутствие информации сразу по нескольким переменным существенно повышает вероятность ошибки.

```
001 AIvanovVladV2004042963480700
200 1$a Иванов $b В. В. $c биохимия
$f 19530130 $g Владимир Владимирович $y Томск
830  $a Образование: в 1975 г. окончил
Томский университет, биолого-почвенный
факультет, аспирантуру в Томском медицинском
институте.
    $a Ученая степень: в 1975 г. защитил
кандидатскую диссертацию. Кандидат
биологических наук.
```

Рис. 1: Фрагмент авторитетной записи

```
00161/H340-682478
700 1$a Шилов $b Б. В.$gБорис
 Владимирович$сцитолог$f19710323
 $3AShilov\_BoriB2003100663480700
701 1 $aИванов$bB. В. $gВладимир
 Владимирович $сбиохимик $f19530130
 $3AIvanovVladV2004042963480700
 $ркафедра биохимии и молекулярной
 биологии СГМУ
701 1 $aКазанский $bB. Е.
71202 $aСибирский медицинский
 университет $cТомск
```

Рис. 2: Фрагмент библиографической записи

4.1 Факторы

Рассмотрим подробнее переменные, по которым осуществляется связывание записей (таблица 1).

Результирующая переменная out отвечает за принадлежность библиографической записи данному автору (другими словами за принадлежность наблюдения к одной из двух групп). Остальные переменные в нашем эксперименте являются факторными и указывают на степень соответствия информации о годах жизни автора,

Таблица 1: Основная группа переменных

Переменная	Сравнение	АЗ	БЗ
out (соответствие)	точное совпадение	001	701$3
birth (дата рождения)	совпадение с точностью до года	200$f	701$f
death (дата смерти)	совпадение с точностью до года	200$f	701$f
addition (профессиональное дополнение)	совпадение усеченных форм	200$c	701$c
place1 (географическое)	совпадение усеченных форм	200$y	712$c
place2 (географическое)	вхождение усеченных форм	200$y	712$a
work 1 (место работы)	совпадение усеченных форм	830$a	701$p
work2 (место работы коллектива)	вхождение усеченных форм	830$a	712$a

Таблица 2: Расширенная группа: соавторы по фамилии

Переменная	Поля для сравнения БЗ и расширенной АЗ
coauthor1 (количество найденных фамилий соавторов)	701$a, 702$a
coauthor2 (доля найденных фамилий соавторов)	
coauthor3 (макс. число общих БЗ среди указанных фамилий)	

профессиональной деятельности, географическом положении и т.п. Факторные переменные можно разделить на две группы. Значения переменных основной группы вычисляются на основе непосредственного сравнения библиографической записи с авторитетной. Расширенная группа использует расширенную авторитетную запись, которая строится следующим образом: находятся все библиографические записи, уже связанные с авторитетной записью и оценивается степень их подобия рассматриваемой библиографической записи. Такой подход позволяет существенно расширить объем библиографических записей, которые можно связать с соответствующими авторитетными, поскольку информация по переменным основной группы часто отсутствует в библиографических записях. С другой стороны, подход не дает никакого выигрыша в случае, когда с авторитетной записью не связано ни одной библиографической.

В расширенной группе переменных анализируется информация по соавторам и предметным рубрикам, указанным в библиографической записи. При этом разделение на авторов и соавторов условное: автором будем называть персону, указанную в авторитетной записи (для которой производится связывание), а соавторами все остальные персоны, независимо от того, как они указаны в библиографических записях.

В таблице 2 приведены три переменные, расчитываемые для соавторов по фамилии, аналогично обрабатываются соавторы по кодам (группа переменных coauthorId поля 701$3, 702$3), предметные рубрики по наименованиям (переменные subject поле 606$a), предметные рубрики по кодам (переменные subjectId поле 606$3). Всего в расширенной группе вычисляется 12 переменных. Следует отметить, что эти переменные нельзя назвать независимыми. В случае, когда не указаны соавторы, например, все три переменные coauthor1, coauthor2 и coauthor3 будут равны 2. Выбор наиболее значимых переменных обсуждается ниже.

Вычисляя значения перечисленных переменных для записей, приведенных в примере, а затем проделав аналогичное сравнение для всех пар из обучающей выборки, получим исходные данные эксперимента, фрагмент которых приведен в таблице 3.

Таблица 3: Фрагмент исходных данных

out	Факторные переменные				
	addition	birth	death	place1	...
2	3	3	2	1	...
2	1	3	2	2	...
1	1	1	2	2	...

Введение переменной place2 было основано на том факте, что в библиографических записях при заполнении поля 712$a иногда в скобках указывают место расположения организации, кроме того, в названиях некоторых организаций содержится указание на географическое положение (например, можно найти подстроку «Томск» в названии «Томский государственный университет»). Переменная work1 отвечает за указание места работы автора в аннотации. В данном случае 701$p = «кафедра биохимии и молекулярной биологии СГМУ», тогда как 830$a содержит подстроку «кафедры биохимии и молекулярной биологии Сибирского медицинского университета». Сопоставить эти строки можно, если использовать словари. Если словарь сокращений не привлекать, то получим значение work1 равное 1, хотя очевидно, что налицо совпадение информации. В то же время пара записей относится к классу соответствующих и переменная out принимает значение 2.

4.2 Предварительный анализ

Факторные переменные, используемые в работе, не подчиняются нормальному распределению, что исключает применение параметрических критериев. Для проверки гипотезы значимости различия двух групп использовался ранговый коэффициент корреляции τ Кендалла [19]. Переменные, для которых принималась гипотеза об отсутствии различий (при уровне значимости 0,01), исключались из работы.

Как видно из таблицы 4, переменную place2, уровень значимости для которой больше 0,01, можно исключить из рассмотрения.

Таблица 4: Анализ переменных

Переменная	τ	p-value	Корреляция
addition	0,428	$<2{,}2*10^{-16}$	значима
birth	0,868	$<2{,}2*10^{-16}$	значима
death	0,296	$<2{,}2*10^{-16}$	значима
place1	0,257	$<2{,}2*10^{-16}$	значима
place2	0,041	0,149	незначима
work1	0,094	$1{,}1*10^{-3}$	мало значима
work2	0,137	$1{,}4*10^{-6}$	мало значима
coauthor1	0,587	$<2{,}2*10^{-16}$	значима
coauthor2	0,595	$<2{,}2*10^{-16}$	значима
coauthor3	0,588	$<2{,}2*10^{-16}$	значима
coauthorId1	0,513	$<2{,}2*10^{-16}$	значима
coauthorId2	0,513	$<2{,}2*10^{-16}$	значима
coauthorId3	0,508	$<2{,}2*10^{-16}$	значима
subject1	0,408	$<2{,}2*10^{-16}$	значима
subject2	0,489	$<2{,}2*10^{-16}$	значима
subject3	0,417	$<2{,}2*10^{-16}$	значима
subjectId1	0,405	$<2{,}2*10^{-16}$	значима
subjectId2	0,41	$<2{,}2*10^{-16}$	значима
subjectId3	0,405	$<2{,}2*10^{-16}$	значима

4.3 Отбор факторных переменных

Воспользовавшись расстоянием Махалонобиса можно произвести отбор наиболее информативных признаков. Для этого на каждом шаге отбираем по одной переменной, дающей наибольшее расстояние между центроидами классов в сочетании с уже выбранными (таблица 5).

В качестве первой включаемой переменной выберем birth, дающую наибольший коэффициент корреляции с результирующей переменной out. Так, на первом шаге в дополнение к birth будет выбрана переменная coauthor2 (отмечена *), а на втором переменные birth, coauthor2, subjectId1, и так далее. Чем раньше включаются переменные - тем больше информации в них содержится.

В результате процедуры отбора был получен список факторных переменных в порядке их значимости для дискриминации, приведенной в таблице 6. Пользуясь этой информацией можно исключить наименее информативные переменные и сократить время работы алгоритма.

Следует отметить, что ранжирование учитывает не только вклад отдельной переменной, но и ее взаимодействие с остальными, это достигается за счет учета корреляции в расстоянии Махаланобиса. Таким образом, на каждом этапе включение менее информативной, но при этом менее коррелированной переменной может оказаться полезнее включения более информативной переменной, если она слишком тесно коррелирована с уже включенными переменными.

Таблица 5: Первый этап ранжирования факторных переменных

Шаг отбора	Переменная	Расстояние между классами $D^2(\mu_1\|\mu_2)$
шаг 1 (выбрана birth)	addition	19,63
	coauthor1	20,77
	coauthor2 *	21,39
	coauthor3	20,27
	coauthorId1	20,06
	coauthorId2	20,85
	coauthorId3	19,77
	death	19,44
	place1	19,31
	subject1	20,51
	subject2	20,58
	subject3	19,63
	subjectId1	20,57
	subjectId2	20,47
	subjectId3	19,51
	work1	19,32
	work2	19,35

Таблица 6: Результат ранжирования переменных

Место	Переменная	Место	Переменная
1	birth	10	place1
2	coauthor2	11	coauthor1
3	subjectId1	12	coauthorId1
4	coauthorId2	13	coauthorId3
5	death	14	subject2
6	addition	15	coauthor3
7	subjectId2	16	work2
8	work1	17	subject3
9	subject1	18	subjectId3

4.4 Проверка качества дискриминации

С помощью расстояния Махаланобиса можно прогнозировать принадлежность наблюдения к одной из групп. Для этого достаточно рассчитать расстояния до центроидов обоих классов, подставив в формулу (1) координаты этого наблюдения, координаты центроида класса и матрицу внутригрупповой ковариации, рассчитанную по формуле (2). После чего следует выбрать в качестве прогноза тот класс, расстояние до которого наименьшее.

Проведя расчеты для тестовой выборки, наблюдения которой не использовались при вычислении параметров алгоритма, получим так

называемую матрицу классификации, на основе которой можно рассчитать долю правильно классифицированных объектов и оценить точность прогноза.

Поскольку количество ошибок в тестовой выборке зависит от того, какие именно пары попали в нее, было проведено 100 прогонов с разными выборками. Средний процент ошибок составил 2,36%.

5 Заключение

В данной статье представлен алгоритм автоматического авторитетного контроля, позволяющий делать заключение о связи библиографической и авторитетной записей без участия человека; а также процедура статистического анализа признаков и отбора наиболее информативных из них. Подход, описанный в работе, является достаточно общим и не накладывает ограничений на используемые переменные и информацию, которую они отражают.

Важной особенностью предлагаемого подхода является возможность обучения на конкретных данных. С одной стороны, это является ограничением, поскольку такие данные не всегда доступны. С другой стороны, возможность обучения позволяет настроить алгоритм на работу с базой данных и, тем самым, учесть ее особенности.

Информацию, на основе которой производится связывание, можно разделить на основную (годы жизни, профессиональное дополнение, место работы) и косвенную (наименование коллективного автора, географическая отметка, информация о соавторах и тематических рубриках). При этом не требуется взаимной независимости признаков, по которым производится сравнение, а также допускается возможность отсутствия информации в части полей. Еще одна особенность подхода заключается в использовании расширенных авторитетных записей, позволяющих увеличить объем информации для сравнения. Конечно, информация, привлеченная из библиографических записей, уже связанных с расматриваемой авторитетной, является косвенной и потому необходимо тщательно анализировать ее с точки зрения достаточности для принятия решений. Однако с другой стороны, «основная» информация, такая как годы жизни, профессиональное дополнение и место работы автора, часто отсутствует в библиографической записи. В результате необходимо делать выбор: использовать косвенную информацию или отказываться от связывания вовсе.

Как уже упоминалось, к авторитетным записям в рамках работы предъявляются достаточно жесткие требования полноты, в отличие от библиографических записей. На практике это приводит к попытке найти соответствующую авторитетную запись даже и для той библиографической, которая содержит недостаточно информации. Поэтому к алгоритму предъявляется требование возможности работы в условиях частично пропущенных данных. Требования полноты библиографической записи можно варьировать в зависимости от степени надежности принятия решения, которой требуется достичь.

В целом можно утверждать, что предлагаемый алгоритм способен улучшить качество данных библиотеки за счет установления недостающих связей между библиографическими записями, полнотекстовыми документами и авторитетными документами. При дальнейшей разработке алгоритма планируется подключить дополнительные методы сопоставления строк, улучшить анализ соответствия по предметным рубрикам, а также дополнить блок нормализации специальными словарями, отсутствующими на данный момент в библиотечной системе.

Список литературы

[1] Belin T. R., and Rubin D. B. (1995), "A method for Calibrating False-Match Rates in Record Linkage Journal of the American Statistical Association, 90, 694-707.

[2] 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.

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

[4] Elfeky M. G., Elmagarmid A. K., Verykios V. S. "TAILOR: A Record Linkage Tool Box". In Proceedings of the 18th International Conference on Data Engineering (ICDE 02). IEEE Computer Society, Washington, DC, USA, 17 - 28, 2002.

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

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

[7] Hylton J. A. Identifying and merging related bibliographic records. M. S. thesis,

MIT, 1996. Published as MIT Laboratory for Computer Science Technical Report 678.

[8] Isele R., Jentzsch A., Bizer C., & Volz J. (2010). Silk - A Link Discovery Framework for the Web of Data, User manual and link language specification 2.0. Language.

[9] Jaro M. A. 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.

[10] Jaro M. A. Probabilistic linkage of large public health data files // Statistics in Medicine 1995; 14: P. 491-498.

[11] Jurczyk P., Lu J., Xiong L., Cragan J., Adolfo Correa, FRIL: A Tool for Comparative Record Linkage, American Medical Informatics associations (AMIA) 2008 annual Symposium.

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

[13] Mahalanobis P. C. (1936). «On the generalised distance in statistics». Proceedings of the National Institute of Sciences of India 2 (1): 49-55.

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

[15] Сайт: Russian stemming algorithm http://snowball.tartarus.org/algorithms/russian/stemmer.html

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

[17] 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.

[18] Мешечак Н. А. Web-справочник «Медики России» [Электронный ресурс] / Н. А. Мешечак, О. С. Колобов, Ф. Е. Татарский // Информационные технологии, компьютерные системы и издательская продукция для библиотек: материалы конф. LIBCOM-2007. - Электрон. текстовые дан. - М. : ГПНТБ России, 2007. – 1 электрон. опт. диск (CD-ROM). – Загл. с этикетки диска. – ISBN 978-5-85638-120-6. - гос. регистрации 0320702219.

[19] Закс Л. Статистическое оценивание. Пер. с нем. В.Н. Варыгина. Под ред. Ю.П. Адлера, В.Г. Горского. М.: Статистика, 1976. – 599 с.: ил.

Automatic document linking

Anna Knyazeva, Igor Turchanovsky, Oleg Kolobov

The problem of automatic linking of documents relating to the same real world object is considered. An algorithm based on classification using the Mahalanobis distance is proposed. The algorithm is illustrated by linking between bibliographic and authority records of author names in Machine-Readable Cataloging format (MARC).