Министерство образования и науки Российской Федерации
Московский физико-технический институт
(государственный университет)
Факультет управления и прикладной математики
Вычислительный центр им. А. А. Дородницына РАН
Кафедра «Интеллектуальные системы»

Спирин Никита Валерьевич

Структурированный поиск с числовыми и логическими ограничениями в неструктурированных Веб-коллекциях

511656 - Математические и информационные технологии

Выпускная квалификационная работа магистра

Научный руководитель:
с.н.с. ВЦ РАН, д.ф.-м.н.
Воронцов Константин Вячеславович

Москва

2012

Содержание

Аннотация

В работе рассматривается решение новой задачи вертикального поиска объектных страниц с числовыми и логическими ограничениями. Предлагаемое решение объединяет в себе идеи из информационного поиска и извлечения информации, что позволяет избавиться от недостатков, присущих существующим подходам к построению вертикальных поисковых систем. В частности на основе наблюдений, выявленных с помощью численного эксперимента, был предложен двухуровненвый подход к предсказанию выполнимости ограничений, основанный на методах машинного обучения. В работе описываются решения задач распознавания «настоящих» позиций, классификации «объектных» страниц, и проверки выполнимости ограничений. Предложенные алгоритмы протестированы на специально созданной в рамках данной работы коллекции и согласно вычислительным экспериментам показывают превосходное качество решения поставленной задачи. Также для соответствия требованиям, налагаемым поисковыми системами на время работы алгоритмов поиска, предложена модифицированная структура инвертированного индекса. Процесс добавления новой объектной вертикали формализован и алгоритмизирован, и может быть перенесен на новые объектные вертикали.

Введение

С появлением интернета главным ресурсом стала информация. Помимо сопутствующих социальных явлений, это породило новое направление в исследованиях. Одна за другой стали появляться информационные поисковые системы, а также алгоритмы, решающие те или иные задачи, возникающие входе их функционирования. За последнее десятилетие поисковые системы значительно эволюционировали и интегрировались очень тесно в повседневную жизнь так, что эффективная деятельность без них более не представляется возможной. Люди используют поисковые системы для поиска ссылок на известные им сайты и веб-страницы, как средство навигации, для получения новой информации, фактов и поиска ответов на вопросы сформулированные на естественном языке (Q&A), для поиска объектов, реальных сущностей и их атрибутов (личности, организации, продукты, и тд.). Единственное объединяющее все эти сценарии использования поисковых систем свойство – это то, что на вход поисковой системе подаются ключевые слова, а выходом является ранжированный список тематически связанных документов, как это было 50 лет назад на заре зарождения информационного поиска. Однако, очевидно, что в некоторых случаях слово-ориентированный вход и релевантность, основанная на тематической схожести документа и запроса, не являются подходящими, что даже может привести к субоптимальным результатам. Актуальность критической переоценки данного поискового протокола возникла в последние годы, когда Веб помимо множества страниц, связанных гиперссылками, стал универсальным хранилищем реальных объектов, сгенерированных пользователями с использованием Web 2.0 технологий. Например, если пользователь хочет купить продукт онлайн, который удовлетворяет его бюджетным огра-

ничениям и имеет соответствующие характеристики, может быть более подходящим предоставить пользователю специальный структурированный интерфейс построения запросов, где он может указать эти ограничения и в результате получить список страниц, содержащих продукты, удовлетворяющие его потребностям. Согласно многочисленным исследованиям поисковых логов подобные объектно-ориентированные поисковые запросы составляют от 10% до 58% поискового трафика [1, 20, 27, 2]. Более того, простейшее подтверждение реальной потребности в подобного рода технологии, позволяющей искать страницы с объектами, удовлетворяющими определенным ограничениям, есть интерфейс подсказок к запросам Google(рис. 1), построенный на основе логов пользователей.

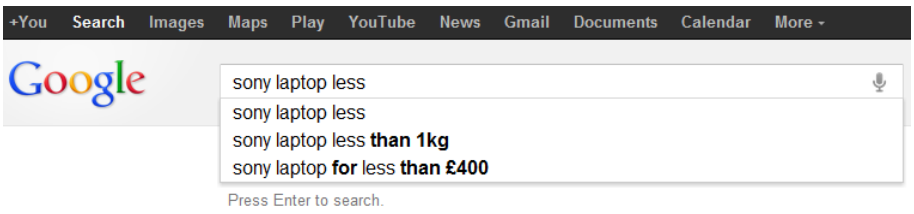

Рис. 1: Интерфейс подсказки запросов Google.

Для наглядности давайте проанализируем типичную ситуацию, когда пользователь ищет *цифровую камеру Canon, которая стоит дешевле, чем 300$ и имеет больше, чем 10 мегапикселей разрешение*. На данный момент сущетсвует несколько принципиальных способов удовлетворить данную информационную потребность.

Во-первых, подходя со стороны классического информационного поиска, пользователь может отправить запрос поисковой машине общего назначения (GWSE), например, Google или Yandex, и далее просматривать страницы поисковой выдачи одну за другой, выбирая страницы, которые действительно ассоциированны с цифровой камерой, и прове-

ряя ограничения на аттрибуты камеры (бренд, цена, разрешение). На момент написания данной работы запрос «canon digital camera [price: 0..300][resolution: 10..100]» возвращает только 2 релевантные страницы с цифровой камерой, удовлетворяющие ограничениям, на первой странице (*Top*-10 ссылок). Данный эксперимент демонстрирует, что на данный момент поисковые системы общего назначения предоставляют лишь удовлетворительные результаты для информационных потребностей такого типа – точность очень низкая. Мы видим основополагающей причиной данного недостатка то, что GWSE работают с неструктурированными линейными запросами, составленными из ключевых слов, и осуществляют лишь поверхностный анализ страницы, подсчитывая частотности отдельных терминов, вместо того, чтобы работать со структурированными запросами и анализировать структуру и семантику страницы детально.

Во-вторых, вручную пользователь может посетить каждый отдельный веб-магазин и использовать соответствующие поисковые и навигационные возможности на сайте. Хотя в этом случае все результаты будут удовлятворять поисковым ограничениям пользователя, поскольку они берутся из «чистой» базы данных магазина, недостатком очевидно является то, что это подход очень утомительный и не масштабируемый, так как пользователь должен повторить ту же самую процедуру на сайте каждого Веб-магазина. Следовательно, для данного подхода точность является высокой, а полнота низкой.

В-третьих, подходя со стороны методов, основанных на извлечении информации, пользователь может обратится к вертикальной поисковой системе по шоппингу, которая в качестве результатов поиска показывает записи из интегрированной базы данных, собранные со множества сайтов (веб-магазинов). Вертикальный поисковик, будучи построенным на

основе структурированного хранилища данных, уже работает со структурированным запросами и адресует объектную семантику. Данный вариант является наиболее привлекательным с точки зрения пользователя, так как он имеет высокую точность и полноту поиска. Однако, подход на основе извлечения информации все же имеет 4 серьезных недостатка.

1. В отличие от информационного поиска, когда коллекция документов индексируется только один раз и единственная ранжирующая функция строится для всей коллекции, вертикальный поиск на основе извлечения информации не расширяем на новые вертикали с легкостью и наоборот требует дополнительную обработку коллекции при добавлении новой вертикали. В основном, это вызвано тем, что страницы из разных вертикалей – разные, и технология извлечения информации, осуществляющая глубинный анализ содержания, структуры и визуального представления страницы, использует эвристики, правила, и признаки, специфичные для данной вертикали, что ограничивает обобщение и применение алгоритмов на новые вертикали. С другой стороны, в информационном поиске используются генеральные предметно-независимые признаки, доступные через инвертированный индекс, и осуществляется поверхностный анализ страницы на основе ключевых слов из запроса. Более того, даже внутри одной вертикали при добавлении нового атрибута мы должны дополнительно обработать коллекцию и обновить алгоритм извлечения информации, так как атрибуты объектов и соответствующие признаки могут быть зависимыми.

2. Так как стадия извлечении информации при построении вертикального поисковика происходит оффлайн, запросы, сформированные пользователями, не используются в процессе извлечения. В то же

самое время запросы могут быть полезны при извлечении информации. Например, если рассмотреть вертикальный поиск по профессорам и, в частности, поиск по их научным интересам, то модуль извлечения информации должен извлечь все возможные строки, потенциально связанные с научными интересами. Но так как научные интересы могут быть почти чем угодно, например, *марковские поля* или *океанография нео-триасского периода*, то эффективность оффлайного метода извлечения информации может быть очень низкой, и следовательно, пользователи конечной поисковой системы получат посредственный опыт взаимодействия. С другой стороны, поисковый запрос, содержащий правильные ключевые слова, выбранные пользователем, может указывать модулю извлечения на потенциально полезные страницы и позиции отдельных терминов на этих страницах.

3. Подход на основе извлечения информации сталкивается с проблемами интеграции информации (например, сопоставление схем баз данных и источников), так как объектные записи извлекаются с множества источников с гетерогенной структурой и способом представления информации об атрибутах, в то время как SQL запросы, используемые при формировании поисковой выдачи, ориентированы на предопределенную центральную схему вертикали.

4. Создание и поддержание модулей извлечения информации в течении времени также требует значительных ресурсов [?, ?]. Поэтому, лишь несколько объектных вертикалей поиска представлены на данный момент – шоппинг и академический поиск[1].

[1]Microsoft Academic Search.

В данной работе мы предлагаем новый взгляд на проблему поиска со структурированными ограничениями посредством объединения идей из информационнго поиска и извлечения информации в одном подходе. Мы переопределяем традиционный для информационного поиска протокол, рассматривая структурированные запросы, задаваемые парами оператор/значение для различных атрибутов объекта. Мы также переопределяем концепцию того, как мы осуществляем извлечение информации в контексте построения вертикального поиска. Вместо того, чтобы осуществлять извлечение информации оффлайн с целью извлечения значений атрибутов, мы фокусируемся на конечную поисковую задачу и осуществляем запросо-ориентированную верификацию ограничений на атрибуты объектов, представленных на страницах, в режиме онлайн[2]. В этом случае также, как и в информационном поиске запрос способствует нахождению страниц и позиций на этих страницах, которые содержат значения (ключевые слова). Также как при извлечении информации мы анализируем контекст каждой указанной запросом позиции на предмет того, что она связана с соответствующим атрибутом объекта и удовлетворяет соответствующему ограничению. Так же как в сущностном поиске надежность и точность решения о том, что ограничения выполнены, увеличивается путем агрегирования предсказаний отдельных позиций, то есть, выражаясь в терминах алгебраического подхода к построению алгебраических композиций [43] – осуществляется голосование индивидуальных позиций, применяется корректирующая операция.

Резюмируя, ниже мы приводим список ключевых идей и решений,

[2]Данная задача в свою очередь может быть сформулирована, как задача распознования образов в классической постановке, что позволяет применять различные алгоритмы машинного обучения.

предложенных в данной работе.

- Объединяя идеи из информационного поиска и извлечения информации в контексте новой задачи – вертикального поиска объектных страниц с ограничениями на атрибуты объектов, мы предлагаем подход, который использует инвертированный индекс для нахождения потенциально релевантных страниц и позиций и осуществляет запросо-зависимое извлечение информации для определения того, что ограничения выполнены.

- Для преодаления специфичных требований с точки зрения эффективности вычислений и времени отклика потенциальной поисковой системы, которая может быть построена на основе предложенных алгоритмов, мы расширили структуру инвертированного индекса, добавив понятие контекстного блока.

- Предлагаемое решение следует последним трендам в предметно-независимом извлечении информации и разработано с соображением быть легко расширяемым на новые вертикали поиска без дополнительной обработки коллекции. В частности, в рамках данной работы мы решили задачи распознавания локальных «настоящих» позиций, предсказания выполнимости ограничений, и классификации «объектных» страниц.

- Мы протестировали предложенное решение на специально созданной тестовой коллекции. Мы также дополнительно создали плагин для разметки данных, который упрощает добавление новых вертикалей, и коллекцию размеченных страниц для дальнейшего исследования научным сообществом.

Работа по главам организованна следующим образом. В главе 1 мы формализуем задачу и приводим анализ релевантной литературы. В главе 2 мы представляем наше решение. В частности, в секции 2.1 мы описываем общий подход, основанный на вероятностном принципе ранжирования. Далее, мы рассматриваем предпосылки и наблюдения 2.2.1, способствовавшие предложенному решению, и описываем алгоритмы проверки структурированных ограничений на текстовые и числовые атрибуты 2.2. Так как рассматриваемая проблема поиска имеет критически важные ограничения на отклик по времени, в главе 3 мы представляем специально модифицированную для решения задачи структуру инвертированного индекса. В главе 4 мы описываем вычислительный эксперимент и в главе 5 мы приводим план будущих работ и заключение.

1 Постановка задачи и описание ключевых научных работ

1.1 Задача структурированного поиска с логическими и числовыми ограничениями в неструктурированных Веб-коллекциях

В данной главе мы более детально опишем проблему и формализуем поисковый протокол в строгой математической нотации.

Представим пользователя, желающего купить онлайн *цифровую камеру, произведенную Canon с ценой дешевле 300$ и разрешением больше 10 мегапикселей*. Эта информационная потребность может быть формализована следующим образом. Во-первых, пользователь ищет объекты в определенной категории/вертикали – цифровые камеры, и поэтому, мы вертикализируем Веб на основе категорий объектов для того, чтобы учесть при построении алгоритмов поиска соответствующие детали, присущие отдельной вертикали. В то же самое время, это накладывает ограничения на наш подход и предлагаемое решение – *требуется простота расширения и добавления новых вертикалей без значительных затрат на адаптацию*. Во-вторых, пользователь выражает несколько структурированных ограничений на атрибуты рассматриваемого объекта – бренд, цена, разрешение. Каждое ограничение может быть представлено в виде пары – оператор и значение. Например, в случае вышеописанной информационной потребности для цены оператором является «<», а значением «300»; для текстового атрибута бренд оператором является непосредственно «строковое равенство», а значение «Canon». Следует заметить, что в зависимости от оператора семантика соответ-

ствующего значения может изменится. Так на примере вертикального поиска профессоров можно сравнить различные операторы «преподаетВ» и «выпускник» некоторого университета; или же при поиске фильмов – «содержитсяВ» или «совпадаетС» названием фильма.

Как мы описали в введении для того, чтобы удовлетворить данную информационную потребность пользователь имеет три основные возможности: (a) использовать поисковик общего назначения, (b) посетить вручную множество отдельных интернет магазинов, (c) использовать вертикальный поисковик для шоппинга. Мы рассмотрим модель поведения пользователя в случае (c) – множество интернет-магазинов, так как это позволит нам описать ключевые понятия, используемые в работе наиболее интуитивным образом.

Сначала, пользователь выберет множество проверенных сайтов интернет-магазинов, таких как *amazon.com, newggg.com, ozon.ru и т.д.* На каждом сайте он попадет на страницу-каталог для категории цифровые камеры, используя поисковые возможности сайта. Далее, используя интерфейс фасетного поиска, пользователь сузит пространство поиска только до камер, удовлетворяющих его потребностям (бюджетные и функциональные ограничения на продукт): «*цена* $<$ *300$*» и «*разрешение* $>$ *10*». В заключение, пользователь начнет анализировать индивидуальные страницы объектов (продуктовые страницы) посредством переходов со страницы каталога. Закончив с одним веб-сайтом (магазином), пользователь перейдет к следующему сайту и повторит ту же процедуру.

Следовательно, что непосредственно нужно пользователю, так это множество страниц продуктов, содержащих объекты, удовлетворяющие ограничениям на атрибуты, указанные в запросе. Именно этот сценарий мы и рассматриваем в данной работе: ***вертикальный поиск объект-***

ных страниц с структурированными (числовыми и логическими) ограничениями на атрибуты объектов.

В отличие от классического информационного поиска, где запросы представляют собой лишь множество ключевых слов и релевантность оценивается на основе простейшей словесной функции близости, в нашем случае мы строим специальную ранжирующую функцию для каждого типа объектов (объектной вертикали) и предоставляем пользователю структурированный интерфейс запросов, позволяющий формулировать информационные потребности в декларативном стиле. Более того, мы детально анализируем структуру страницы и контекст каждой позиции, указанной значением из запроса, при принятии решения о том выполнены ли ограничения на атрибуты объекта на данной странице или нет.

Что касается определения понятия релевантности, в нашем случае мы определяем ее следующим образом. Мы говорим, что страница удовлетворяет запросу, если: (1) это объектная страница (представляющая лишь один ключевой объект интереса); (2) страница содержит информацию о всех атрибутах, на которые были выражены ограничения в запросе, и все ограничения выполнены. То есть логика сопряжения атрибутных ограничений соответствует логическому «И». Следует отметить, что если объектная страница не содержит информации о некоторых атрибутах, на которые наложены ограничения в запросе (например, атрибут отсутствует на странице), а следовательно, метка класса о выполнимости ограничения не может быть непосредственно установлена по этой странице, мы рассматриваем такую страницу как нерелевантную, чтобы минимизировать ошибки второго рода (нерелевантная страница детектированна, как релевантная). Это означает, что каждая страница рас-

сматривается в отдельности и не используется никакая база данных / база знаний с информацией о верных значениях атрибутов объектов.

Формально проблема, рассматриваемая в данной работе, имеет следующий поисковый протокол:

Дано: коллекция веб страниц $D = \{d_1, \ldots, d_N\}$.

Вход (запрос): *вертикаль объектов* V и *структурированный запрос* с ограничениями $Q = (ac_1, \ldots, ac_L)$, где $ac_i = (op_i, val_i)$, $i \in 1, \ldots, L$, составленные из пар *оператор и значение*, задают ограничения на атрибуты объектов интереса. Для каждого атрибута существует конечное непустое множество допустимых операторов.

Выход (поисковый результат): список *объектных страниц* $D_{sat} = \{d_{sat,1}, \ldots, d_{sat,l(D_{sat})}\}$, упорядоченный согласно вероятности того, что все ограничения на атрибуты объектов выполнены.

Стоит также отметить два дополнительных требования/условия, которые делают рассматриваемую проблему особенно интересной.

Во-первых, решение должно быть с легкостью расширяемым на новые типы объектов/вертикали. Мы учитываем это требование формально, констатируя, что признаки, используемые в алгоритмах поиска, должны быть индексируемые с той точки зрения, что они доступны через хеш-отображение {слово -> список документов}, называемое инвертированным индексом, который должен строится лишь один раз для всей коллекции безотносительно того сколько вертикалей будет добавлено в будущем. Добавление новой вертикали должно требовать только минимальных затрат по настройке ранжирующей (проверяющей выполнимость ограничений) функции, учитывающей специфику конкретной вертикали.

Во-вторых, так как мы рассматриваем поисковый сценарий и в то же

самое время осуществляем глубинный анализ страницы с целью определения того, что все ограничения на атрибуты объекта на странице выполнены, наше решение должно обладать разумным временем отклика и быть вычислительно эффективным. Традиционный подход к построению вертикальной поисковой системы на основе технологии извлечения информации не встречает таких требований, так как извлечение происходит оффлайн и пользовательские запросы не используются.

1.2 Обзор релевантной литературы

Основу данной работы составляют статьи и идеи из различных ветвей исследований, посвященных поисковым системам, такие как классический информационный поиск, вертикальный поиск, сущностный поиск. Также с точки зрения применяемых техник и методов работа связана с исследованиями по отбору и синтезу признаков на основе частых паттернов (поиск информативных закономерностей), извлечению информации, и машинному обучению ранжирующих функций.

1.2.1 Классический информационный поиск и Вероятностный Принцип Ранжирования (PRP)

Информационный поиск имеет уже достаточно долгую историю, однако, начиная с ранних работ Луна [24] об эффективности представления документов как векторов слов и выявлении статистическими методами их относительной информативности, далее в работах о вероятностном принципе и подходе к задаче ранжирования [26, 30], векторной модели ранжирования [31] и ранжировании на основе статистических языковых моделей [28] поисковый протокол не изменился. По-прежнему, на вход поисковой системе поступает поисковый запрос, а на выходе си-

стема возвращает ранжированный список страниц. При этом релевантность основана на простейшем текстовом соответствии. Например, в векторной модели ранжирования [31] это просто нормированное скалярное произведение (cosine similarity) векторов слов, соответствующих запросу и документу/странице. В методах на основе статистических языковых моделей [28] настраивается языковая модель для каждого документа, а далее они упорядочиваются согласно тому как вероятен запрос относительно соответствующей модели в вероятностном смысле (вероятность породить данный запрос, используя случайный процесс по выбору слов, задаваемый параметрами на основе данного документа).

В данной работе, также как и в информационном поиске, мы используем запрос (ключевые слова; значения, задающие ограничения) для того, чтобы направлять поисковый алгоритм к потенциально релевантным страницам и позициям конкретных значений на данных страницах. Это сделано для того чтобы повысить скорость отклика, так как при использовании инвертированного индекса мы значительно сужаем пространство поиска документов просто на основе бинарного вхождения слова в документ. Также данная идея, будучи примененной в контексте извлечения информации, позволяет повысить точность извлечения, так как данный процесс теперь происходит с шаблоном (конкретное значение из запроса), а не в открытом смысле на основе косвенных контекстных и структурных признаков.

В качестве руководства к поиску используется модифицированная версия инвертированного индекса [5, 9], который в свою очередь представляет собой хеш-отображение из множества слов в множество страниц, которые данное слово содержат. Однако, в отличие от информационного поиска в данной работе мы рассматриваем богатые структуриро-

ванные запросы и осуществляем глубинный анализ страницы (также как в алгоритмах, решеющих задачу извлечения информации) для того, чтобы предсказать является ли страинца релевантной, то есть является ли она страницей под некоторой объект и что все ограничения на атрибуты данного объекта выполнены.

Следует отметить, что текстовое соответствие, используемое в классическом информационном поиске, нацелено на поиск страниц тематически связанных с запросом, и следовательно, чем выше частота слова на странице, тем выше релевантность страницы. В то же самое время, в данной работе, в контексте поиска объектных страниц со структурированными ограничениями, наличие слова на странице, даже с большой частотой, не является достаточным. В дополнение мы также должны понять, что данное слово относится к описываемому объекту и удовлетворяет соответствующему ограничению, указанному в запросе с помощью оператора. Например, при построении вертикального поисковика профессоров наличие названия университета на странице может быть связано как с текущем местом работы «преподаетВ», так и с местом, где данный профессор обучался – «выпускник». Следовательно, в данной работе, также как и в сущностном поиске [9], мы производим поиск как на основе контента (непосредственное значение, используемое в запросе), так и на основе контекста, анализируя с какой целью данное слово/значение упоминается на странице. В то же время в классическом информационном поиске используется лишь контент – явно указанные ключевые слова из запроса.

Вероятностный Принцип Ранжирования (*Probability Ranking Principle, PRP*). При решении задачи информационного поиска возникает вопрос о том какой критерий использовать для того, чтобы ран-

жировать страницы. Ответ на этот вопрос дает вероятностный принцип ранжирования [29], согласно которому документы должны быть упорядочены по убыванию вероятности соответствия запросу на основе всей доступной исходной информации. Приведем обоснование данного принципа в рамках классического байесовского подхода.

В общем случае дано обучающее множество $D = \{(x_i, y_i)\}_{i=1}^{l}$ и требуется восстановить функцию $A_\theta : X \longrightarrow Y$ из некоторого семейства функций, параметризованное параметром θ. Зададим априорную вероятность на множестве параметров $P(\theta)$ и функцию правдоподобия $P(y|x, \theta)$. Далее выписывая правдоподобие данных согласно выбранной модели

$$P(D|\theta) = \prod_{i=1}^{l} P(y_i|x_i, \theta)$$

и применяя правило Байеса, найдем функцию распределения апостериорной вероятности параметров, выражаемую формулой

$$P(\theta|D) = P(D|\theta)P(\theta)/P(D)$$

и выпишем формулу для распределения финального предсказания согласно описанной модели

$$P(y|x, D) = \int_{\theta} P(y|x, \theta)P(\theta|D)d\theta.$$

Заметим, что данная формула описывает распределение на множестве значений, и для того, чтобы выбрать конкретное значение требуется определить функцию потерь $\mathcal{L}(y_{pred}, y)$ и выбрать соответствующий элемент пространства ответов, оптимизируя функцию байесовского риска. Согласно байесовскому выводу функция риска для алгоритма A относительно апостериорного распределения есть

$$\mathcal{R}(A) = \int_{y} \mathcal{L}(A(x), y)P(y|x, D)dy,$$

и оптимальное решение имеет вид

$$A^* = arg\min_a \mathcal{R}(a).$$

Если теперь в контексте задачи ранжирования принять, что $X = \mathfrak{F}(q,d)$ – пространство векторов объектов, определенных на парах запрос-документ, а $Y = \{0,1\}$ множество меток релевантности {нерелевантный, релевантный} соответственно, и определить $l = c_0$ при $Y = 0$ и $l = c_1$ при $Y = 1$, то, расписывая соответствующую функцию риска для оптимального решения, получим

$$R = c_0 P(y=0|q,d) + c_1 P(y=1|q,d) = (c_1 - c_0)P(y=1|q,d) + c_0.$$

1.2.2 Сущностный поиск

Данная работа идеологически следует тем же идеям, что изложены в исследованиях по сущностному поиску [9], который решает задачу поиска в Интернет понятий/сущностей (личность, организация, место), описываемых набором ключевых слов. Например, поддерживается поиск по запросам типа «*#телефон службы поддержки авиакомпании X*» или «*#профессор #интерес=(машинное обучение) в России*». Согласно данному подходу Веб рассматривается как хранилище понятий/сущностей, а не как хранилище страниц, и поиск должен быть построен вокруг них. Для этого авторами предложено расширить стандартный инвертированный индекс с помощью генеральных сущностей (личность, организация, место), которые предызвлекаются оффлайн с помощью генеральных скальперов (методов извлечения), таких как модели максимальной энтропии [11], скрытые марковские модели [37], либо условные случайные поля [42]. Структура индекса сохраняется для обычных слов, а постинг (список документов и позиций по термину) для сущностей со-

держит в дополнение вероятность, что извлеченное значение относится к тому или иному типу сущностей. Данная вероятность используется онлайн в процессе обработки запроса, чтобы найти наиболее правдоподобный ответ. В частности, сам процесс обработки запроса разбит логически на 3 этапа: (1) извлечение потенциально релевантных страниц на основе простейшего текстового соответствия с использованием расширенного сущностями инвертированного индекса (отметим, что сущности при таком подходе рассматриваются как обычные слова), (2) локальное распознавание релевантных подстрок на странице (при этом учитывается близость терминов и сущностей из запроса и вышеописанная вероятность), (3) аггрегация глобального ответа на основе локальных (при этом множество кандидатов-сущностей, найденных на отдельных страницах, проходит стадию дедупликации и финальный ответ находится на основе простейшего подсчета, учитывающего также вероятность быть релевантным для конкретного локального кандидата и репутацию[3] соответствующей страницы). Однако, структурированный поиск с ограничениями имеет несколько ключевх отличий.

Во-первых, сущностный поиск решает задачу поиска сущностей в Интернет и локальный анализ контекста вокруг позиции-кандидата является достаточным, чтобы найти ответ путем аггрегации (подсчета, дедупликации) подобных кандидатов. Однако, при структурированном поиске цель найти объектные страницы, которые задаются особой структурой, и поэтому помимо локального анализа контекста каждой потенциальной позиции, найденной при помощи инвертированного индекса, мы также производим анализ структуры страницы, чтобы понять является ли данная страница страницей объекта определенного типа.

[3]Может быть использован, например, алгоритм PageRank или HITS.

Во-вторых, сущностный поиск производит агрегацию предсказаний для каждого ответа-кандидата по всему Интернету, в то время как в подходе, предлагаемом в данной работе, в силу того, что осуществляется поиск страниц, агрегация осуществляется лишь в пределах страницы (отдельные локальные признаки, связанные с атрибутами и ограничениями на них, смешиваются алгоритмом машинного обучения в модель предсказания о выполнимости ограничений / финальную метку).

В-третьих, для того, чтобы построить расширенный инвертированный индекс, в сущностном поиске используются модули извлечения сущностей/понятий общего характера, такие как личность, организация, место. Но в тоже время из-за этого данный подход не может поддерживать поиск редких сущностей (проблема длинного хвоста). Развитие идей, предложенных в начальной статье о сущностном поиске, представлено в [40], где авторы разработали DoCQS систему и язык запросов, которые позволяют определять новые сущностные типы, например, университет на основе организации, или профессор на основе личности, в SQL-подобной нотации. Однако, данный подход также имеет свои недостатки. Так, например, новые сущностные типы и методы их извлечения определяются пользователем в момент формирования запроса, что согласно нашим экспериментам приводит к потере/пропуску множества релевантных ответов/сущностей/понятий. А следовательно, для повышения эффективности поиска необходимо решать проблему определения новых типов и методов их извлечения более системно. В данной работе мы предлагаем именно такой подход, при котором для каждого типа объектов (вертикали) мы обучаем отдельный алгоритм проверки выполнимости ограничений (включает в себя локальный алгоритм проверки позиций-кандидатов в ответы на основе контекстной и позициональной

информации). Более того, все алгоритмы обучаются статистическими методами, что позволяет покрыть значительную часть правильных ответов (позиций, описывающих соответствующие атрибуты объектов на страницах).

1.2.3 Извлечение информации

Задача извлечения информации из неструктурированного источника, например, веб-страницы, имеет долгую историю. Особенно активный этап развития данного направления исследований ассоциирован с разработкой методов извлечения информации с целью дальнейшего её использования для построения вертикальной поисковой системы или базы знаний. Можно выделить три основных подхода к построению методов извлечения информации: (a) на основе шаблонов, заданых оператором вручную, (b) основанные на методах машинного обучения по размеченным данным, (c) автоматические, основывающиеся на выявлении регулярностей в объекте из которого производится извлечение.

В частности, [21, 33] являются классическими примерами подхода на основе шаблонов. В группе работ [15, 22] описываются методы построения Веб-скальперов на основе размеченных данных. Так на обучающем множестве страниц указаны позиции всех атрибутов, которые необходимо извлечь, и алгортм учитывает такие признаки как разметка и структура страницы (DOM-дерево, взаимное расположение элементов), оформление отдельных элементов (цвет, размер шрифта), позиция элемента на страницы на основе словесного смещения или визуально при представлении страницы в браузере.

Ряд автоматических методов извлечения информации был предожен в последнее время. В [6] авторы предложили самообучаемый подход по

сегментированию страницы на основе визуальных признаков и дальнейшему извлечению информации из информативных сегментов. В [42] извлечение объектных записей и атрибутов отдельных объектов предложено решать в рамках одной задачи, что привело в свою очередь к повышению качества методов извлечения информации. Идея заключается в том, что алгоритм выявляет повторяющуюся структуру по DOM-дереву и с использованием визуальных признаков для того, чтобы найти объектные блоки/записи, и далее применяет методы нахождения соответствия деревьев на основе динамического программирования, что позволяет локализовать и извлечь отдельные атрибуты.

Некоторые работы подходят к задаче извлечения информации с использованием условных случайных полей [41, 36]. Так в [41] и [36] предложены 2D-CRF и Hierarchical-CRF модели соответственно, которые учитывают пространственные характеристики текста и веб-страницы в целом. В продолжение данной линии работ и с их использованием была разработана система объектного поиска [39]. В статье также как и в данной работе поддерживаются структурированные запросы, однако, предметом результатов является, также как и в сущностном поиске, отдельный уникальный объект, информация про который агрегируется и уточняется автоматически на основе алгоритмов разрешения сущностей и синтеза информации. Стоит отметить, что отличием сущностного поиска от данной работы является то, что в первом случае поддерживаются произвольные запросы, при условии что они содержат лишь слова и предындексированные сущности, и поиск ответа происходит онлайн в момент запроса с использованием инвертированного индекса, в то время как в работе по оъектному поиску на основе CRF строится централизованное хранилище и в момент запроса происходит лишь извлечение из

БД, что в свою очередь ограничивает тип объектов/сущностей, которые могут быть получены в качестве результата.

[32] решает проблему извлечения информации с помощью Марковских Логических Сетей, что позволяет учитывать предметные ограничения для повышения качества извлеченного результата. С целью сделать процесс извлечения информации легко расширяемым на новые вертикали и минимизировать размер обучаемого множества, задаваемого для страниц в выбранной вертикали, в [14] предложен метод, использующий слабые, но обощаемые признаки, и осуществляющий адаптацию обученного алгоритма извлечения с одного веб-сайта обучения на новые сайты в даной вертикали посредством unsupervised методов. В данной работе мы предлагаем подход, который также использует обобщаемые на всю вертикаль признаки и не требует обучающего множества больших размеров. Однако, в дополнение предлагаемый подход является расширяемым на новые вертикали с той точки зрения, что только модель извлечения должна быть создана, а коллекция остается без изменения. То есть в нашем подходе не требуется дополнительной обработки коллекции, так как мы смотрим на коллекцию через призму инвертированного индекса.

Однако, несмотря на значительную популярность подхода к построению вертикальной поисковой системы на основе методов извлечения информации, он обладает теми или иными недостатками, которые мы указали во введении.

1.2.4 Классификация на основе информативных закономерностей и частых паттернов

Частые паттерны (закономерности), представляющие собой, например, подмножества совместно покупаемых товаров, подпоследователь-

ности просмотров фильмов или слов есть в общем случае подструктуры, которые появляются в коллекции данных с частотой больше, чем заданный порог. Основная идея использования частых паттернов для решения задачи классификации заключается в том, что в множестве объектов, над которыми проводится классификация, каждая подструктура рассматривается как потенциальный признак для представления объекта, находятся частые подструктуры, и далее отбираются те из них, которые имеют сильную корреляцию с меткой класса. Также ранние алгоритмы, следующие данному подходу, вместо того, чтобы использовать частые паттерны как признаки, находят ассоциативные правила вида {комбинация подструктур -> метка класса} и устраивают различные методы принятия решений с использованием данных правил.

Первые алгоритмы, рассматривающие данный подход были независимо предложены в работах [23, 38, 25]. Один из самых простейших алгоритм CBA [25] находит частые паттерны и соответствующие ассоциативные правила, используя алгоритм Априори [3], и упорядочивает паттерны на основе надежности соответствующего правила. Надежностью правила называется вероятность появления постусловия по отношению к вероятности появляения пре- и пост-условий совместно. При классификации на основе подобных правил решение о присвоении метки класса принимается путем последовательного перебора правил в убывающем порядке надежности. Если не одно правило не сработало, присваивается метка класса с наибольшим числом объектов в обучающем множестве. Стоит отметить, что данный подход относится к множеству алгоритмов вида решающий список.

CMAR [23] развивает данную идею путем группирования правил в зависимости от того в какой класс происходит классификация и пред-

сказывает метку для нового объекта, используя взвешенное голосование. Весом группы выступует сумма надежностей всех правил за соответствующий класс. В алгоритме CPAR [38] происходит дополнительный отбор правил внутри группы. Также поиск правил осуществляется не только на основе частотного анализа паттернов (подструктур), но и с использованием различных критериев информативности (FOIL-gain, iGain) и эвристики последовательного покрытия обучающего множества, когда объекты, выделенные соответствующими правилами-закономерностями, убираются из рассмотрения.

Метод, в котором рассматривается использование частых паттернов непосредственно как признаков для алгоритма распознавания, представлен в работе [12]. Авторы провели анализ с целью понять полезность частых паттернов для задачи классификации и подтвердили разумное предположение, что если упорядочить все паттерны по возрастающему значению поддержки (число раз паттерн появляется в коллекции объектов обучения), то информативными являются паттерны с средними значениями поддержки. Данное наблюдение было также обосновано теоретически. Однако, подход предложенный в [12] как и все ранее предложенные алгоритмы, сталкивается с проблемой вычислительного характера - перебор конъюнкций является дорогой операцией. В [8] предлагается алгоритм DDPMine, решающий данную проблему комбинаторного взрыва, в котором поиск частых паттернов производится в рамках метода ветвей и границ. Отсекание бесперспективных ветвей осуществляется на основе iGain.

Данный алгоритм получил широкое развитие и был применен в контектсе поиска частых паттернов для классификации авторства текстов [18], фотографий [17], графов [19]. В заключение, было предложено теорети-

ческое обоснование, что при использовании алгоритма перебора частых паттернов в рамках метода ветвей и границ с метрикой CORK [34], алгоритм классификации является приближенно оптимальным.

1.2.5 Машинное обучение ранжированию

В отличие от классического информационного поиска, когда модели ранжирования основаны на различных наблюдениях о структуре языка [28] или мотивированы конкретными вероятностными предположениями [29], машинные методы обучения ранжированию следуют другому пути и решают непосредственно задачу предсказания релевантности в рамках дискриминантного подхода. Данный подход мотивирован тем, что с переходом от закрытых и чистых коллекций, используемых в классическом информационном поиске, к Веб-поиску качество классических моделей оказалось очень низким в силу различных явлений, таких как спам, и возникла потребность в введении новых признаков, интерпретация и количество которых уже не поддаются ручной настройке. Методы машинного обучения как раз способны решать задачу через настройку коэффициентов статистическими методами, не требующими моделирования явления детально.

Так в рамках факторного подхода к ранжированию задача формулируется как построение отображения из множества признаков, определенных на парах запрос-документ, в множество меток релевантности. Задачей обучения является настройка коэфициентов соответствующего семейства моделей ранжирования. В целом выделяется три основных подхода: точечные [10] методы, когда происходит редукция задачи ранжирования к задаче регрессии и векторы объектов не взаимодействуют; парные [4, 16], когда задача ранжирования сводится к задаче класси-

фикации правильного порядка на парах объектов; списочные [35], когда решается непосредственно задача ранжирования и алгоритм настраивается непосредственно на предсказание наиболее релевантного списка выдачи.

В контексте данной работы задача ранжирования решается при поиске на странице позиций, которые наиболее вероятно связаны с соответствующим атрибутом объекта. Применение подобной редукции основано на том, что правильные («настоящие» позиции атрибутов) имеют схожие статистические свойства (располагаются в определенном месте страницы, имеют регулярный контекст), и следовательно, при наличии обучающего множества данные общие свойства могут быть установлены. Выбор задачи ранжирования не случаен. Достаточно заметить, что для генерирования правильной метки о выполнимости ограничений нужно точно знать лишь одну «настоящую» позицию атрибута объекта, представленного на данной странице. А поскольку для новой страницы не известно какая позиция является «настоящей», то позиции ранжируются в убывающем порядке по вероятности быть «настоящими», и решение принимается с использованием Top-k позиций-кандидатов.

Также идеологически схожая стратегия используется в алгоритмах проверки выполнимости ограничений, когда на основе признаков, определенных по запросу, вертикали, и странице предсказывается финальная оценка о выполнимости на основе дискриминантной функции.

2 Запросо-ориентированный подход к поиску объектных страниц по структурным ограничениям на атрибуты объекта

В данной главе мы описываем предлагаемое решение проблемы поиска объектных страниц по структурным ограничениям на атрибуты объектов, которые они содержат. В частности, сначала, мы приводим описание общей модели, основываясь на вероятностном принципе ранжирования, и рассматриваем различные варианты факторизации условной вероятности релевантности документа запросу в контексте задачи. Далее, мы описываем ключевые алгоритмы, разработанные в рамках данной работы, для проверки ограничений на атрибуты объектов. Мы также кратко рассматриваем алгоритм распознавания объектных страниц.

2.1 Общая модель на основе вероятностного принципа ранжирования и факторизация задачи на подзадачи

Согласно вероятностному принципу ранжирования страницы должны быть упорядочены в убывающем порядке вероятности того, что данная страница удовлетворяет запросу, что может быть выражено следующей формулой – $P(rel = 1|q, d)$, где rel - является случайной величиной на множестве $\{0, 1\}$ моделирующей релевантность, q – запрос и d – документ/страница. В контексте задачи, рассматриваемой в данной работе, понятие релевантности вводится как свойство страницы быть объектной страницей и что все ограничения на объект, представленный на данной странице, выполнены. Также запрос q в нашем случае является

комплексным и содержит ограничение на вертикаль поиска v, ассоциированную с конкретным типом объектов, и ограничения на отдельные атрибуты объекта q_c. Следовательно, вероятностный принцип ранжирования, примененный к проблеме, рассматриваемой в данной работе имеет вид:

$$P(rel = 1|q, d) = P(rel = 1|v, q_c, d) = P(c_1 = 1, \ldots, c_k = 1, o = 1|v, q_c, d), \tag{1}$$

где $c_i = 1$ означает, что ограничение на i-ый атрибут выполнено, а $o = 1$, что страница является объектной страницей для объектов данной вертикали.

Далее данную формулу в зависимости от предпосылок и целей, которые хочется достичь, можно факторизовать различным образом.

1. Если в формуле 1 выделить множитель с вероятностью о выполнимости ограничений на атрибуты при условии, что страница является объектной страницей в данной вертикали, а также предположить независимость исходов о выполнимости ограничений, то получим

$$\Big(\prod_{i=1}^{k} P(c_i = 1|o = 1, v, q_c, d)\Big) P(o = 1|v, q_c, d). \tag{2}$$

   В этом случае, сначала, происходит предсказание о том, что страница является объектной с использованием поверхностных признаков, определенных на уровне страницы, и далее осуществляется проверка ограничений на атрибуты на основе глубинного анализа позиций-кандидатов, потенциально связанных с атрибутами объекта, на которые выражены ограничения в запросе. Преимуществом данного варианта является то, что, сначала, проверяется лишь одно ограничение – «объектность», и если удастся установить, что

страница не является объектной, то её обработку можно будет закончить, что в свою очередь может сэкономить время обработки запроса в целом.

2. Если же в формуле 1 выделить множитель с вероятностью того, что страница является объектной при условии, что ограничения выполнены, и также опять допустить независимость исходов о выполнимости атрибутных ограничений, то формула 1 примет вид

$$P(o = 1|c_1 = 1, \ldots, c_k = 1, v, q_c, d) \prod_{i=1}^{k} P(c_i = 1|v, q_c, d). \qquad (3)$$

В данном случае в отличие от предыдущего случая, сначала, происходит предсказание вероятностей о выполнимости ограничений на атрибуты, а далее данные предсказания используются при определении того является ли данная страница объектной. Преимуществом данного варианта является то, что при распознавании «объектности» страницы используется дополнительная информация, что может привести к большему качеству данного алгоритма распознавания объектности страницы и поиска в целом. Например, если на странице выполнены все ограничения, то это повышает вероятность того, что страница является объектной, так как на произвольной «необъектной» странице данное совпадение менее вероятно.

В вычислительном эксперименте мы сравниваем эти два случая с точки зрения качества оценки вышеописанных вероятностей. Вопрос эффективности каждого из подходов будет исследован в будущей работе при построении полноценной поисковой системы.

Заметим, что данную проблему можно решать и напрямую в рамках IBT (*inference-based training*) подхода, моделируя условную вероятность

страницы быть релевантной для заданой вертикали и запроса с ограничениями на атрибуты, то есть использовать непосредственно формулу $P(rel = 1|v, q_c, d)$. Однако, как было отмечено в [7] для сложных задач структурного обучения эффект от использования прямого протокола обучения[4] достигается только при наличии большого обучающего множества. В противном случае рекомендуется обучать отдельные классификаторы для подзадач индивидуально.

Так в контексте рассматриваемой проблемы, например, в случае 2, сначала, происходит предсказание вероятности о выполнимости ограничений на атрибуты и далее предсказанные значения используются как дополнительные признаки при определении объектности страницы. Данный подход типичен при решении сложных, многостадийных задач обучения по прецедентам, когда происходит разбиение задачи на подзадачи, и отдельный алгоритм разрабатывается для решения каждой конкретной подзадачи. Далее, при предсказании ответа для нового прецедента модели применяются последовательно (*pipelinening, cascading*) так, что предсказания ранних моделей используются как признаки при построении предсказаний поздними моделями. Например, подобная схема используется при решении машинными методами задачи поверхностного семантического парсинга (*Semantic Role Labelling, SRL*), в которой как подзадачи последовательно решаются задача синтаксического парсинга, которая в свою очередь основана на задаче о предсказании частей речи для слов предложения (*Part Of Speech tagging, POS*). В процессе семантического парсинга нового предложения модели, обученные на решение подзадач, генерируют значения признаков (деревья синтаксического раз-

[4]когда происходит оптимизация всего алгоритма, а не составных частей (локальных классификаторов) по отдельности.

бора предложения, части речи), которые используются в финальной модели для *SRL*.

Также прежде, чем мы обратимся к алгоритмам проверки выполнимости ограничений на атрибуты и объектности страницы, отдельно стоит описать в каком порядке происходит вычисление множителей-вероятностей в факторизованной формуле и как в первую очередь страница попадает в рассмотрение данного алгоритма при поиске.

2.1.1 Принцип извлечения страниц из инвертированного индекса

Заметим, что в формулах 2 и 3 множители, выражающие вероятности о выполнимости ограничений на атрибуты объекта, зависят от запроса с ограничениями на атрибуты, выбранной объектной вертикали, и самой страницы. При этом в отличие от классического поиска в данном случае возможны «пустые» запросы[5] или запросы с ограничениями только на численные атрибуты. Например, это может быть запрос типа «v =цифровая камера, $price < 300\$$». В этом случае по существу необходимо рассматривать все страницы и применять алгоритм проверки ограничений и объектности для каждой из них. Однако, данный процесс является очень вычислительно затратным и требуется более эффективное решение.

Аналогично подходу, используемому в классическом информационном поиске, мы предлагаем формировать множество страниц-кандидатов посредством использования инвертированного индекса и далее проверять ограничения только для страниц-кандидатов. Если запрос не со-

[5]когда задана только объектная вертикаль согласно постановке обязательная в каждом запросе, без ограничений на атрибуты объекта).

держит ограничений на текстовые атрибуты, то используются высокоинформативные характеристические слова, специально отобранные для данной вертикали алгоритмами отбора признаков. Например, для вертикали объектов «цифровые камеры» ключевыми словами могут быть *камера, зум, мегапикселей*. Если же запрос содержит ограничения на текстовые атрибуты, то мы дополнительно используем соответствующие ключевые слова/значения. Постинги для всех слов пересекаются и полученное множество страниц-кандидатов передается в алгоритм проверки ограничений на атрибуты и распознавания объектности страницы.

Далее мы рассмотрим как предсказать множители-вероятности в случае 2^6. Отметим, что при этом, сначала, происходит проверка ограничений на атрибуты, и далее определяется является ли страница объектной. Полученные вероятности перемножаются и поисковая выдача формируется из страниц-кандидатов, упорядоченных в убывающем порядке вероятности выполнения всех условий, сформулированных в запросе.

Мы также предлагаем отфильтровывать страницы, для которых хотя бы один из множителей-вероятностей в формуле 3 меньше, чем пороговая величина, определенная согласно процессу кросс валидации по обучающему множеству. Это позволяет достигнуть двух дополнительных преимуществ. Во-первых, в этом случае мы повышаем точность алгоритма поиска, так как в ответ попадают только страницы-кандидаты, для которых все ограничения выполнены с большой вероятностью. Во-вторых, это позволяет проводить «раннее» отсечение нерелевантных страниц и снизить время отклика поисковой системы на запрос в целом, так

[6]Случай 1 абсолютно аналогичен, за исключением того, что при предсказании объектности страницы не используются признаки, связанные с выполнимостью ограничений на атрибуты объекта, и порядок применения алгоритмов обратный, то есть, сначала, проверяется объектность страницы, а потом ограничения на атрибуты.

как, например, при последовательной проверке ограничений на данной странице малая вероятность в выполнимости текущего ограничения делает страницу уже нерелевантной.

2.2 Алгоритмы проверки ограничений на атрибуты объектов

В данной секции мы опишем как предсказать вероятность о том, что конкретное ограничение на атрибут из запроса выполнено.

Согласно классическому подходу к построению алгоритмов распознавания, мы рассматриваем данную задачу как задачу обучения отображения. Так как рассматриваемая вероятность имеет вид $P(c_i = 1|v, q_c, d)$, то соответствующий алгоритм есть отображение вида A_c: $\vec{x} \longrightarrow y$, где $\vec{x} = \mathfrak{F}(v, q_c, d)$ является вектором признаков, построенных для данного запроса и страницы- кандидата, а y есть случайная величина из множества $[0, 1]$.

Ниже мы приводим ключевые наблюдения и замечания, лежащие в основе предлагаемых признаков, а далее мы перейдем непосредственно к алгоритмам проверки выполнимости ограничений.

2.2.1 Наблюдения и Замечания

Предлагаемый алгоритм проверки атрибутных ограничений основан на трех ключевых наблюдениях.

1. Локальные признаки. Как мы заметили во введении для проверки структурированных ограничений недостаточно стандартного для информационного поиска подхода, при котором для определения тематической релевантности страницы используются лишь глобальные признаки, определенные на уровне страницы на основе ключевых слов из

запроса. Причиной этого является то, что помимо значения (ключевое слово, число) структурированное ограничение содержит также оператор, и следовательно, необходим более детальный анализ страницы (позиций, вхождений) для снятия неоднозначности об употреблении конкретного значения.

Так, например, при классическом подходе на основе $TFIDF$ не возможно дифференцировать означает ли слово «*МФТИ*» на странице место работы или вуз окончания. Ситуация еще сложнее в случае числовых атрибутов, так как при этом не известно даже явное значение – ограничения в общем случае задают полуинтервалы и потенциально полезным является любое число со страницы.

По результатам проделанных экспериментов было также выявлено, что для предсказания выполнимости ограничений на числовые атрибуты полезными являются только признаки, основанные на числах. Корреляция и взаимная информация метки о выполнимости ограничения с любым из чисто словарных признаков ниже 0.08.

2. Вертикальная регулярность. При создании шаблонов веб-страниц[7] в одной вертикали веб-програмисты и дизайнеры следуют некоторым общим принципам. Мы заметили, что контент представлен на множестве различных веб-сайтов в одинаковой форме и значения атрибутов на объектных страницах употребляются в одинаковом контексте. Например, разрешение камеры цифрового фотоаппарата указывается как «*#number Мпикс.*» в заголовке веб-страницы, или «*с разрешением #number мегапикселей*» в теле страницы. Более того, информативными

[7]В последнее время практикуется разделение логики обработки и представления информации. Поэтому веб-страницы строятся на основе шаблонов, содержимое которых наполняется на основе серверных скриптов.

также являются позиционные признаки, такие как (1) смещение[8], например, если значение находится среди первых 200 слов веб-страницы, и (2) нахождение слова внутри некоторого HTML-тега, например, в *<title>* или *<h1>*.

Следовательно, «настоящие» позиции (значения, выражающие атрибуты) могут быть отделены от остальных значений на основе контекста и позиционной информации, а все многообразие естественного языка и верстки сводится к нескольким типам фраз и стилям разметки, что может быть выучено алгоритмом. Более того, данная регулярность обосновывает идею о том, что алгоритм проверки ограничений, обученный на конечном множестве прецедентов (веб-сайтов в одной вертикали) будет применим на всей вертикали.

3. Избыточность атрибутной информации на странице.

На большинстве страниц информация об атрибутах повторяется. Так, например, разрешение цифровой камеры может быть описано в заголовке страницы, в секции о технических спецификациях, в отзывах, и др. Следовательно, если использовать предсказания о выполнимости ограничения, сгенерированные всеми локальными алгоритмами распознавания, то можно предсказать правильную метку с большей точностью. Даже если локальный алгоритм сделал ошибку и определил «ненастоящую» позицию-значение, как «настоящую», то при условии, что данный алгоритм в целом имеет высокую точность, подобных предсказаний будет мало, и в результате, предсказания всех локальных алгоритмов, будучи агрегированы, позволят предсказать правильную метку о выполнимости для всей страницы.

Такая же идея лежит в основе всех алгоритмов построения алгорит-

[8] индекс слова в документе, считая от начала.

мических композиций. Более того, согласно схожему принципу построены признаки в классическом информационном поиске, где большая частота слова на странице (избыточность) сигнализирует о более высокой тематической релевантности страницы запросу.

Вывод: На основе описанных наблюдений мы предлагаем двухуровневый подход к процессу генерации признаков. При этом на локальном уровне решается множество задач распознавания о том, что позиции слов из запроса, указанные инвертированным индексом, являются «настоящими», то есть что они описывают значение соответствующего атрибута объекта, представленного на данной странице. Данная стадия необходима для снятия неоднозначности употребления конкретных слов и значений и восстановления их реальной семантики[9]. Далее, на глобальном уровне, предсказания локальных алгоритмов агрегируются в глобальные признаки, и наряду с другими глобальными признаками, определенными по запросу и странице в целом, они используются отображением A_c для предсказания вероятности того, что ограничение на выбранный атрибут выполнено.

Стоит отметить, что принципиально мы выделяем 2 типа атрибутов и соответственно ограничений – текстовые и числовые. Данное разделение вызвано тем, что в случае текстовых атрибутов мы можем использовать слова из запроса непосредственно для нахождения на странице позиций, которые нужно рассмотреть на предмет быть «настоящими». В то же самое время в случае числовых атрибутов подобный способ нахождения позиций не возможен, так как ограничения на числовые атрибуты в общем случае задают полупространства типа $(<, 300)$, то есть нет яв-

[9]Задачи построения локальных классификаторов для каждого атрибута рассматриваются отдельно.

ного значения для поиска позиций-кандидатов. Поэтому мы используем поиск позиций-кандидатов по типу, используя в расширенном инвертированном индексе постинг для сущностного класса *#number*.

2.2.2 Разметка данных

Для построения алгоритма проверки выполнимости ограничений на атрибуты объекта по запросу, странице, и для выбранной объектной вертикали мы рассматриваем конечное множество объектных страниц из данной вертикали, загруженных с конечного множества веб-сайтов. На каждой странице мы указываем все позиции для каждого из атрибутов объекта для того, чтобы учесть избыточность атрибутной информации, а также иметь больше «настоящих» (положительных) прецедентов для обучения локальных алгоритмов распознавания[10]. Пример размеченной объектной страницы приведен на рис. 2.

Заметим, что несмотря на то, что мы решаем проблему поиска, обучающее множестово запросо-независимо, поскольку по сути мы решаем задачу извлечения информации. Однако, на основе данного запросо-независимого размеченного множества мы можем искусственно сгенерировать запросо-зависимое обучающее множество посредством семплирования запросов (ограничений) из множества с фиксированной структурой (пары оператор/значение) и сравнения известного значения атрибута на странице из обучающего множества с порогом из ограничения для порождения меток о выполнимости ограничений. Именно, это множество используется для обучения алгоритмов проверки ограничений, так как

[10]При классическом способе разметки, используемом в работах по извлечению информации, указывается только одна позиция для каждого из атрибутов объекта интереса на странице, либо для каждой страницы создается XML-файл или запись базы данных, в которой указываеются соответствующие значения всех атрибутов.

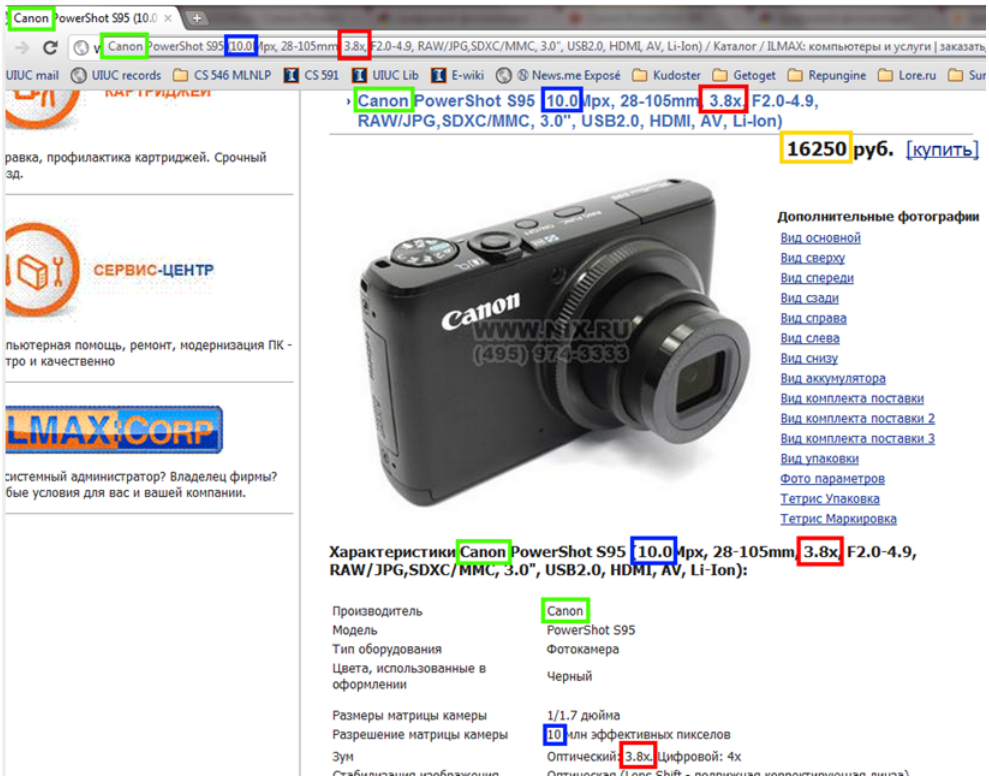

Рис. 2: Пример размеченной страницы.

мы используем запросо-зависимые признаки.

2.2.3 Проверка ограничений на текстовые атрибуты

Согласно вышеописанному общему подходу алгоритм проверки каждого ограничения состоит из 2 стадий. Мы, сначала, опишем ключевую локальную стадию, а затем перейдем к финальному, глобальному алгоритму. Стоит отметить, что для каждого атрибута объекта данной вертикали мы обучаем отдельный алгоритм, то есть результатом стадии обучения является множество алгоритмов мощностью равной числу атрибутов объекта.

Так как мы предлагаем запросо-ориентированный подход к проверке

ограничений, то мы используем значения из запроса для поиска позиций-кандидатов. Например, при вертикальном поиске профессоров запрос («*преподаетВ*», «*МФТИ*») с ограничением на вуз, в котором профессор преподает на данный момент, будут рассмотрены все позиции слова «*МФТИ*». Далее для каждой из данных позиций будет проверено является ли данная позиция «настоящей», то есть выражающей, действительно, место трудоустройства профессора. Для этого необходимо построить локальный алгоритм распознавания, который формально решает задачу A_{loc}: $\mathfrak{F}(pos) \longrightarrow y$, где $\mathfrak{F}(pos) \in \mathbb{R}^n$ есть признаковое описание данной позиции-кандидата, а $y \in \{0, 1\}$ метка класса, выражающая, что данная позиция есть «настоящая» позиция. Следовательно, требуется ответить на 2 вопроса:

1. Какие признаки использовать для представления позиций-кандидатов в виде векторов, учитывая необходимость в индексируемости данных признаков для минимизации времени отклика конечной поисковой системы?

2. Какую модель обучения выбрать и как ее обучить?

1. Представление позиций. На основе наблюдений и замечаний, описаных выше, мы предлагем использовать контекстные и позициональные признаки для представления каждой отдельной позиции. Данные признаки могут быть заданы вручную, просто посредством анализа страниц из обучающего множества и отбора фраз, которые могут сигнализировать о том, что позиция «настоящая». Однако, поскольку одной из целей данной работы является разработка алгоритмов легко расширяемых на новые объектные вертикали и поскольку при ручном подходе сложно учесть все возможные случаи и зависимости, мы предлагаем

синтезировать признаки автоматически. При этом при добавлении новой вертикали не потребуется дополнительной деятельности за исключением разметки обучающего множества страниц.

Следуя методологии синтеза информативных закономерностей для задач классификации в данной работе мы представляем каждую позицию как транзакцию кодов и проводим синтез информативных закономерностей (паттернов). Коды присваиваются каждому отдельному признаку. Мы рассматриваем следующие типы признаков[11]:

- *Присутствие слова в определенной зоне документа. Например, в заголовке, метаданных, теле страницы.*
- *Присутствие определенного слова в окрестности рассматриваемой позиции-кандидата. Например, «цена находится в окне шириной 3».*
- *Присутствие слова в определенном кластере смещения позиции. Например, позиция находится среди первых 200 слов страницы.*

Метка класса для каждой транзакции известна, так как на размеченном множестве страниц мы указали все «настоящие» позиции, и следовательно, неотмеченные позиции являются отрицательными. Формально вход алгоритма синтеза признаков имеет следующий вид.

Пусть дано конечное множество кодов признаков $I = \{i_1, i_2, \ldots, i_m\}$ и пара меток классов $C = \{0, 1\}$, означающих «ненастоящие» и «настоящие» позиции соответственно. Множество позиций формирует базу данных транзакций D_{db} для отбора признаков для локального распознавания так, что $D_{db} = \{\mathbf{n}_i, y_i\}_{i=1}^{N}$, где $\mathbf{n}_i \in 2^I$ множество кодов и $y_i \in C$.

[11] Отметим, что все признаки могут быть эффективно вычислены только с использованием инвертированного индекса.

В этой форме алгоритм синтеза дискриминантных признаков DDPMine [8] может быть успешно применен. В результате мы получим конечное множество признаков, которые и будут использоваться для представления позиций-кандидатов.

2. Обучение локального алгоритма распознавания «настоящих» позиций. Мы моделируем данную проблему как задачу ранжирования, при которой запросы и документы в классической постановке соответственно заменены на страницы и позиции-кандидаты на этих страницах. Данная редукция основана на том, что «настоящие» позиции, имеющие положительные метки и характеризуемые схожим признаковым описанием по всей объектной вертикали, будут ранжированы выше, чем «ненастоящие» отрицательные позиции. Например, заголовок часто содержит число, описывающее значение атрибута разрешение цифровой камеры, и поэтому активация данного признака в признаковом описании позиции-кандидата может быть сильным основанием рассматривать данную позицию «настоящей» и ранжировать ее выше, чем остальные позиции-кандидаты. Далее на основе позиций из *mona* можно принять решение о выполнимости ограничения непосредственно, либо использовать в качестве признака для глобального алгоритма предсказания выполнимости соответствующего огрпничения. Формально задача ранжирования в контексте данной работы имеет следующий вид.

Дано обучающее множество страниц $D_{dlp} = \{s_i\}_{i=1}^{n}$. На каждой странице размечены все положительные позиции расматриваемого атрибута и определенное множество отрицательных позиций, в целом составляющие множество позиций, ассоциированных с данной страницей $P_{s_i} = \{\vec{p}_{s_i,1}, \ldots, \vec{p}_{s_i,n(s_i)}\}$. Каждая позиция есть вектор $\vec{p}_{s_i,j}$ в пространстве признаков синтезированных на предыдущей стадии алгоритмом синтеза дис-

криминантных закономерностей. Для каждой пары страница-позиция указана метка $y_{s_i,j}$, обозначающая является ли данная позиция «настоящей» или «ненастоящей».

Цель восстановить функцию ранжирования позиций, оптимизирующую функционал вида

$$Q(A_{pr}) = \frac{1}{|D_{dlp}|} \sum_{s_i \in D_{dlp}} \mathcal{L}\big(\vec{A}_{pr}(s_i), \vec{y}_{s_i}\big) \longrightarrow \min, \tag{4}$$

где $\mathcal{L}$ специфичная для задачи ранжирования функция потерь, а $\vec{A}_{pr}(s_i)$ и $\vec{y}_{s_i}$ предсказанные и реальные векторы позиций на странице s_i, соответственно.

Для решения данной задачи мы используем двуклассовый SVM с вероятностным выходом. Стоит также отметить, что, используя данную модель алгоритмов, мы следуем точечному подходу к задаче ранжирования и сводим задачу ранжирования к задаче вероятностной двуклассовой классификации. Выбор данного подхода основан на том, что в случае когда информация об атрибуте отсутствует на странице совсем, согласно нашей договоренности при постановке задачи, страницу необходимо считать нерелевантной, и вероятностный выход позволяет ввести порог для осуществления данной операции. Парный и списочный подходы к задаче ранжирования не предоставляют такой возможности, так как они оптимизируют функционалы качества либо на парах, либо на списках, а не оценивают вероятность позиции быть релевантной непосредственно.

В процессе предсказания для всех позиций-кандидатов, указанных инвертированным индексом, на новой странице мы применяем вышеописанный алгоритм и упорядочиваем позиции согласно вероятности быть «настоящими», то есть выражающими истинное значение атрибута объекта, представленного на данной странице. Упорядоченность позиций

используется при генерации некоторых глобальных признаков.

Перейдем к описанию глобальных признаков. Так как конечной целью является не извлечение конкретного значения атрибута, а предсказание вероятности о выполнимости ограничения на конкретный атрибут объекта, представленного на странице, аналогично подходу, используемому в Веб-поиске, мы определяем множество признаков по запросу, вертикали, и странице и строим отображение из вектора глобальных признаков в финальное множество оценок о выполнимости этого ограничения. При этом данный подход базируется на идее, что существует определенная скрытая зависимость между выполнимостью ограничения и значениями признаков. Мы определяем следующие глобальные признаки:

- TF_{body} – частота слова-значения из запроса в теле страницы;
- TF_{title} – частота слова-значения из запроса в заголовке страницы;
- $isTop1InTitleReal$ – является ли первая предсказанная позиция-кандидат в заголовке страницы «настоящей»;
- $isTop1InBodyReal$ – является ли первая предсказанная позиция-кандидат в теле страницы «настоящей»;
- $TF_{context.body}$ – число позиций в теле страницы, для которых вероятность быть «настоящими» больше, чем некоторый порог. Порог настраивается с помощью кросс валидации по обучающему множеству. Можно проынтерпретировать данный признак, как дисконтированную частоту слова, учитывающую контекстую и позициональную информацию, то есть после снятия семантической неоднозначности.

В заключение опишем процедуру построения обучающего множества для обучения глобального алгоритма предсказания выполнимости ограничений. Для этого искусственным образом порождается множество запросов путем семплирования операторов и значений атрибута с размеченного множества страниц[12]. Далее для сгенерированного множества запросов и размеченных страниц вычисляются локальные, а затем глобальные признаки, и записывается матрица объекты-признаки в классическорй постановке. Далее по этой матрице может быть обучен любой алгоритм классификации, в частности в данной работе используется алгоритм SVM.

2.2.4 Проверка ограничений на числовые атрибуты

Как мы описали выше общий подход к проверке числовых и текстовых ограничений один и тот же. Однако, в случае числовых ограничений существует определенная особенность, связанная с тем, что ограничения, представляющие собой пары (*оператор, значение*), задают полуинтервалы, и поэтому не возможно осуществлять отбор позиций-кандидатов быть «настоящими» с помощью значения из запроса и инвертированного индекса.

Вместо этого мы предлагаем использовать постинг для сущностного типа *#number*, то есть дополнительный постинг в расширенном инвертированном индексе, введенный в работе о сущностном поиске [9]. Однако, стоит отметить, что при таком отборе мы вынуждены рассматривать большое количество позиций-кандидатов в силу того, что на страницах (особенно в вертикали онлайн шоппинга) присутствует очень много чи-

[12]Заметим, что для каждой размеченной страницы мы можем извлечь реальные значения всех атрибутов, так как разметкой указаны все «настоящие» позиции.

сел. Поэтому, чтобы сократить пространство поиска возможных «настоящих» позиций, мы предлагаем проводить предфильтрацию позиций на основе высокочастотных слов-паттернов. Данная идея основана на наблюдении о регулярности контекста, в частности, что «настоящие» позиции числовых атрибутов рядом содержат определенные слова почти всегда. Например, в случае вертикального поиска цифровых камер для атрибута «цена» такими словами-паттернами являются *цена, $, РУБ.*

Для того чтобы найти такие паттерны, мы решаем задачу поиска частых закономерностей. Рассматривается *положительное* множество транзакций (те же, что используются для синтеза признаков для локального алгоритма распознования «настоящих» позиций). Каждая транзакция есть конечное множество кодов из $I = \{i_1, i_2, \ldots, i_m\}$, и задача состоит в поиске частых паттернов (наборов, закономерностей) в данной базе данных. Мы используем высокоэффективный алгоритм FP-Growth [13] и задаем очень высокий порог надежности паттернов. Порог выбирается из тех соображений, чтобы с помощью найденных паттернов можно было отфильтровать заведомо «ненастоящие» позиции, но в то же время гарантировать, что с помощью такой высокоэффективной, но, возможно, неаккуратной операции не произойдет потери «настоящих» позиций. Более качественное распознавание происходит локальным алгоритмом, так как он уже обучен по большому множеству и использует богатое признаковое представление позиций (здесь транзакций).

Теперь также как и в случае текстовых атрибутов определим семейство глобальных признаков, которые используются финальным алгоритмом для оценки вероятности о выполнимости ограничений на числовые атрибуты. Заметим, что в отличие от локальных позиций-кандидатов в случае текстовых атрибутов, когда «настоящая» позиция может толь-

ко увеличить вероятность в выполнимости ограничения, для числовых атрибутов «настоящая» позиция может также способствовать отрицательному результату, так как ограничения на числовые атрибуты требуют логического сравнения. Например, можно установить «настоящую» позицию цены на странице, и, сравнив ее с пороговым значением из соответствующего ограничения, принять решение, что ограничение не выполнено. Поэтому различные позиции-кандидаты могут локально предсказывать метки о выполнимости разных знаков.

- $isTop1TitleSat$ – означающий, что позиция-кандидат, ранжированная первой в заголовке страницы, предсказывает, что ограничение выполнено;
- $isTop1BodySat$ – означающий, что позиция-кандидат, ранжированная первой в теле страницы, предсказывает, что ограничение выполнено;
- $isTopKVoteSat$ – означающий, что на основе взвешенного голосования *Top-k* позиций-кандидатов в теле страницы ограничение выполнено;
- $isTopKRatioSat$ – равный отношению суммы вероятностей положительных позиций-кандидатов из *Top-k* на сумму вероятностей всех позиций-кандидатов из *Top-k* в теле страницы.

В качестве глобального алгоритма предсказания выполнимости ограничений мы мспользуем SVM.

2.3 Распознавание объектных страниц

В данной секции мы опишем алгоритмы проверки объектности страниц. Предварительно отметим, что в рамках данной работы мы рассматриваем частный случай общей Веб-коллекции, при котором страницы загружены с множества различных интернет-магазинов, содержащих только ограниченное множество типов страниц – каталог товаров, объектная страница, страница рекламы и др., а не общую Веб-коллекцию. Следовательно, данное допущение делает задачу распознавания объектности страницы более легкой, и мы планируем рассмотреть методы решения общей задачи в будущей работе.

Формально задача распознавания объектности страницы есть задача построения отображения $A_{obj}\colon \mathfrak{F}(c_1, \ldots, c_k, v, q_c, d) \longrightarrow y$, где $y \in \{0, 1\}$. Согласно нашим экспериментам при рассматриваемой в данной работе постановке достаточно использовать чисто словарные признаки, извлеченные со страниц обучающего множества, и не обращаться к структурным (число фотографий, число одинаковых HTML-поддеревьев, и др.).

В качестве обучающего множества мы рассматриваем множество размеченных объектных страниц и любые другие страницы, загруженные с сайтов интернет-магазинов. Для того, чтобы учесть требования к времени отклика конечной поисковой системы, мы отбираем признаки согласно различным критериям информативности. В итоге, согласно нашим экспериментам требуется порядка 10-20 слов-признаков для решения задачи распознавания объектности в данной постановке.

3 Модифицированная архитектура инвертированного индекса для задачи поиска со структурированными ограничениями

В данной секции мы описываем, как мы расширяем существующий инвертированный индекс для решения задачи структурированного Вебпоиска с логическими и числовыми ограничениями. Предварительно мы кратко опишем структуру традиционного индекса и далее представим предлагаемую надстройку, а также подтвердим ее использование в главе посвященной вычислительному эксперименту.

3.1 Традиционный инвертированный индекс

Традиционный инвертированный индекс используется для ускорения процесса поиска и представляет собой отображение { термин/ключ -> [список документов, в которых данный термин встречается] }. В инвертированном индексе для сущностного поиска в дополнение к словам в индексе служат базовые сущности, такие как личность, место, дата, число, телефон, и др. Индекс строится по коллекции документов в режиме оффлайн. При исполнении поискового запроса, рассматриваются страницы, которые содержат слова из запроса, и далее происходит пересечение постингов данных слов и формирование конечного множества страниц-кандидатов. Затем к отобранным с помощью инвертированного индекса страницам применяется ранжирующая функция и документы упорядочиваются согласно предсказанной оценке релевантности. Структура традиционного индекса мотивирована тем, что присутствие слова на странице принимается как свидетельство о том, что страница являет-

ся тематически релевантной и поэтому необходимо уметь быстро получать документы, содержащие ключевое слово. Позиционный инвертированный индекс содержит помимо кодов страниц, также и позиции слова в документе, что позволяет исполнять запросы с заданием точной последовательности указанных слов в запросе. Примером, позиционального запроса может быть название города «New York».

3.2 Предлагаемая модификация

В данной работе мы также используем признаки, основанные на близости слов, и поэтому разумно предположить, что позиционный индекс может быть использован также и для целей данной работы. В этом случае при решении локальной задачи распознавания «настоящих» позиций, для каждого поступающего запроса мы должны инициализировать постинги и соответствующие указатели на слова, которые используются в качестве признаков. Однако, порядок размерности признакового описания для локального алгоритма распознавания «настоящих» позиций равен тысячам. Следовательно, данное решение не представляется возможным с вычислительной точки зрения.

Мы провели эксперимент для того, чтобы проверить можно ли понизить размерность вектора признаков и получить то же качество при распознавании «настоящих» позиций. Однако, при понижении размерности пространства признаков до уровня, способного быть обработанным с помощью позиционального инвертированного индекса с использованием zigzag-join операции, качество локального алгоритма распознавания значительно падает. В результате, мы заключили, что данный вариант не является подходящим.

Для преодаления данной сложности мы предлагаем модифицировать

позициональный инвертированный индекс посредством введения ***контекстного блока***. Так для каждой позиции слова на странице мы также индексируем контекст в окне заданной ширины. Например, если на странице присутствует текст «*..today our price is 299.99$ with free shipping and no taxes..*» и задано окно ширины 3, то в контекстном блоке для значения *299.99* будет следующая информация: левыйКонтекст = «*our price is*», правыйКонтекст = «*with free shipping*». Для удобства мы также приводим схему модифицированного индекса на рис. 3.

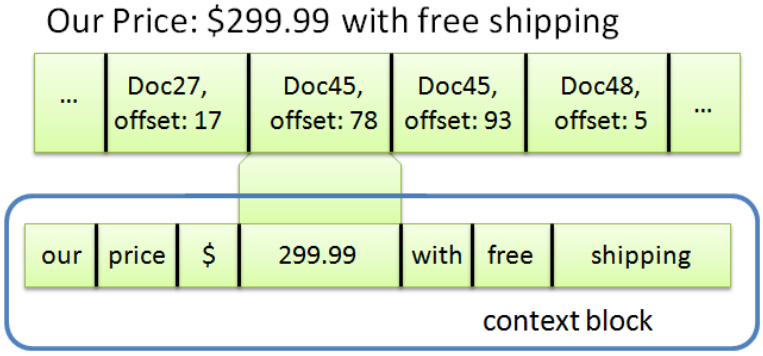

Рис. 3: Предлагаемая структура модифицированного индекса.

Отдельно стоит отметить, что так как мы индексируем для каждой позиции контекст размер инвертированного индекса вырастает, причем данный рост может быть значительным. Поэтому, чтобы снизить данный эффект и найти правильный баланс между качеством распознавания и размером индекса, размер окна стоит определять на основе вычислительного эксперимента.

4 Вычислительный эксперимент

В данной секции мы описываем проделанные эксперименты. В частности, мы анализируем зависимости качества алгоритмов проверки ограничений от размера окна; размера обучающего множества и числа сайтов в одной вертикали, использованных для обучения. Мы также приводим статистику по качеству распознавания «настоящих» позиций локальным алгоритмом с использованием полного набора признаков и после отбора (понижения размерности). В заключение, мы также оцениваем эффективность операции предфильтрации в случае числовых атрибутов. Во всех случаях приводится величина точности при уровне полноты 100%.

4.1 Описание данных, используемых в эксперименте

Для целей эксперимента по предсказанию выполнимости ограничений была сформированна новая коллекция данных. Коллекция представляет собой множество Веб-страниц, загруженных алгоритмом краулинга с множества Веб-сайтов интернет-магазинов. В данной работе мы рассматриваем применение методов и алгоритмов, описанных выше, на примере объектной вертикали цифровые фотоаппараты. Использовалось 10 различных интернет-магазинов: adorama, amazon, newegg, bestbuy, buy, compusa, ebay, pcnation, sears, walmart. В общем коллекция содержит 235 веб-страниц. В среднем на каждый интернет-магазин приходится равное количество страниц (порядка 25).

Для рассматриваемого типа объектов (объектной вертикали) мы определили 4 ключевых атрибута – цена, разрешение, зум, бренд. Каждая из 235 страниц прошла стадию разметки, так что мы указали все «настоящие» позиции для всех 4 ключевых атрибутов.

4.2 О выполнимости атрибутных ограничений

4.2.1 Зависимость качества предсказаний от размера размеченного множества страниц

В данном эксперименте анализируется *сколько* размеченных страниц необходимо алгоритму распознавания для того, чтобы достичь определенного уровня качества предсказаний о выполнимости ограничений. Мы последовательно увеличиваем размер обучающего множества страниц и проводим замеры качества предсказаний о выполнимости ограничений. Графическое представление результатов эксперимента приведено на рис. 4. А также в сводной таблице, которая следует ниже.

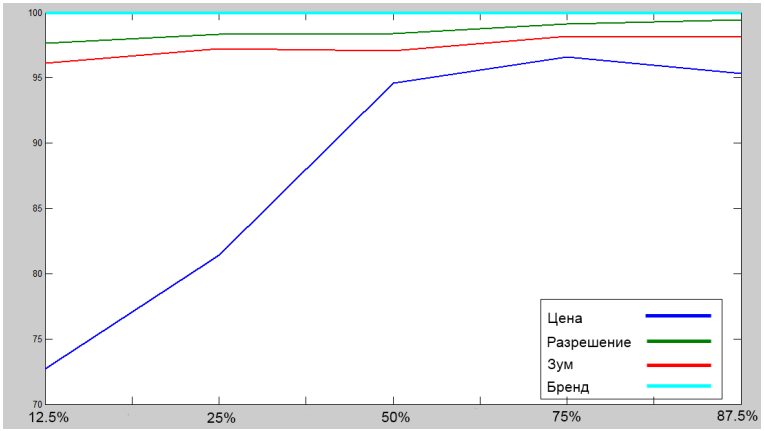

Рис. 4: Зависимость качества алгоритма предсказания выполнимости ограничений от размера обучающего множества.

Атрибут	12.5%	25%	**50%**	**75%**	**87.5**
Цена	72.72	81.41	94.57	96.61	95.33
Разрешение	97.63	98.34	98.41	99.15	99.44
Зум	96.13	97.22	97.07	98.17	98.11
Бренд	100.0	100.0	100.0	100.0	100.0

Также в таблице ниже мы приводим качество локального алгоритма распознавания «настоящих» позиций для рассматриваемых числовых атрибутов.

Атрибут	12.5%	25%	50%	75%	87.5
Цена	97.36	97.78	98.91	98.89	98.90
Разрешение	93.31	94.68	95.17	97.28	97.18
Зум	87.16	90.05	90.64	93.12	93.59

Согласно данному эксперименту разрешение и зум являются легкими атрибутами и не требуют большого обучающего множества. В то же время качество предсказаний для атрибута цена начинает быть достойным при более серьезном размере обучающего множества. Также по графику можно заметить, что насыщение наступает в районе 50% обучающего множества, что соответствует 120 страницам.

Данное наблюдение согласуется с выдвинутой гипотезой в связи с тем, что на страницах значение цены упоминается лишь несколько раз, тогда как разрешение и зум описываются в многих секциях страницы.

4.2.2 Зависимость качества распознавания от размера окна

Размер окна имеет непосредственное влияние на размер индекса, так как окно определяет размер контекстного блока. Также размер окна, который в свою очередь неявно связан с размерностью пространства признаков для локального алгоритма распознования «настоящих» позиций, определяет и качество распознавания этого алгоритма. В данном эксперименте мы анализируем именно влияние размера окна на *качество предсказаний* о выполнимости ограничений и оставляем анализ объема индекса на рассмотрение в будущей работе[13].

[13]Можно грубо оценить, что индекс растет в число раз равное размеру окна.

Атрибут	Окно +/−3	Окно +/−4
Цена	0.8837	**0.9457**
Разрешение	0.9756	**0.9841**
Зум	0.9536	**0.9707**

Как мы видим увеличение размера окна приводит к повышению качества распознавания. Однако, слишком большой размер окна не представляется возможны в силу роста инвертированного индекса из-за контекстного блока. Поэтому мы выбираем размер окна 4, как балансирующий качество алгоритма и затраты на хранение контекстных блоков (неявно связанных с признаками).

4.2.3 Зависимость качества распознавания от размерности признакового описания

Данный эксперимент имеет своей целью *проверить возможность использования стандартного позиционального индекса* для исполнения алгоритмов, предлагаемых в данной работе. В частности, сравниваются алгоритмы с полным и редуцированным пространством признаков (которое может быть исполнено инвертированным индексом) и замеряется качество предсказаний о выполнимости ограничений.

Атрибут	Пониженная размерность (*top*-35 признаков)	Окно +/−3
Цена	76.03	**88.37**
Разрешение	95.45	**97.56**
Зум	85.35	**95.36**

По результатам данного эксперимента мы заключили, что использование позиционального инвертированного индекса в данной ситуации не

возможно, так как понижение размерности приводит к значительному понижению качества алгоритма предсказания выполнимости ограничений. Для разрешения данной проблемы мы предложили модифицированную структуру инвертированного индекса, которая позволяет исполнять алгоритмы с полным признаковым описанием.

4.3 Об эффективности высокочастотных слов в процессе предфильтрации для сужения множества позиций-кандидатов для числовых атрибутов

С целью уменьшения времени отклика конечной поисковой системы мы используем операцию предфильтрации для сужения с помощью очень вычислительно дешевых методов числа позиций-кандидатов. Данная идея актуальна только в случае числовых атрибутов, поскольку изначально позиции находятся с помощью #number постинга, который является очень генеральным ключом и приводит к очень большому множеству позиций-кандидатов.

Ниже мы приводим коэффициенты редукции пространства поиска для каждого из рассматриваемых числовых атрибутов.

Атрибут	До фильтрации	После фильтрации	Отношение
Цена	30243	5104	5.9254
Разрешение	18079	1502	12.0366
Зум	18079	1325	13.6445

Следовательно, предложенная эвристика сокращает вычислительные затраты на анализ локальных позиций в диапазоне от 5 до 14 раз.

4.4 О качестве распознавания объектных страниц

Для эксперимента по классификации объектных и необъектных страниц (среди множества страниц интернет-магазинов) было сформировано дополнительное обучающее множество, которое состоит из всех размеченных страниц и случайно отобранных страниц, загруженных краулером. Напомним, что в рамках данной работы решается частный случай общей задачи обнаружения объектности страницы, когда рассматриваются только страницы, загруженные с интернет-магазинов и используются чисто словесные признаки. По результатам численного эксперимента алгоритм достигает 92.56% значения по метрике accuracy[14].

[14]Число правильных классификаций деленое на общее число классификаций.

Заключение

В настоящей работе рассматривается новая задача поиска веб-страниц с структурированными ограничениями на атрибуты объектов, а также предлагается соответствующее решение. Решение впервые объединяет идеи, используемые в информационном поиске и извлечении информации, в рамках одного подхода, что позволяет решать сложную задачу с превосходным качеством.

В работе алгоритмизирована схема добавления новой объектной вертикали поиска и детально описаны все шаги, которые необходимо преодолеть для того, чтобы использовать предложенное решение на практике (построение полноценной поисковой системы). А именно, предложены алгоритмы проверки выполнимости ограничений на текстовые и числовые атрибуты, разработаны алгоритмы распознавания «настоящих» позиций для снятия семантической неоднозначности употреблений терминов на веб-странице, описан алгоритм распознавания «объектности» страниц, а также предложена модифицированная архитектура инвертированного индекса. В рамках данной работы также был разработан плагин-расширение к Google Chrome для разметки данных, что позволило значительно ускорить процесс создания обучающих множеств для алгоритмов. Стоит отметить, что в рамках данной работы тестовые и обучающие коллекции были созданы и размечены автором собственноручно, в отличие от традиционного случая, когда используются уже существующие коллекции.

Проведен вычислительный эксперимент, подтверждающий гипотезы, наблюдения и замечания, заложенные в основу алгоритмов. Проанализированы закономерности о поведении качества алгоритма предсказаний выполнимости ограничений от различных параметров. Из числен-

ных экспериментов установлено, что предложенный подход показывает превосходные результаты и является более чем конкурентным по сравнению альтернативными подходами, так как он не имеет присущих им недостатков.

В ходе работы над данной задачей было также выявлено дальнейшее направление исследований и начаты работы по построению полноценной поисковой системы.

Список литературы

[1] G. Agarwal, G. Kabra, and K. C.-C. Chang. Towards rich query interpretation: walking back and forth for mining query templates. In *Proceedings of the 19th international conference on World Wide Web*, WWW'10, Raleigh, North Carolina, 2010.

[2] E. Agichtein, S. Cucerzan, and E. Brill. Analysis of factoid questions for effective relation extraction. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR'05, Salvador, Brazil, 2005.

[3] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB'94, 1994.

[4] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, ICML'05, 2005.

[5] M. J. Cafarella and O. Etzioni. A search engine for natural language applications. In *Proceedings of the 14th international conference on World Wide Web*, WWW'05, Chiba, Japan, 2005.

[6] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma. Block-based web search. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR'04, 2004.

[7] M.-W. Chang, L. Ratinov, N. Rizzolo, and D. Roth. Learning and inference with constraints. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 3*, AAAI'08, 2008.

[8] H. Cheng, X. Yan, J. Han, and P. S. Yu. Direct discriminative pattern mining for effective classification. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, 2008.

[9] T. Cheng, X. Yan, and K. C.-C. Chang. Entityrank: searching entities directly and holistically. In *Proceedings of the 33rd international conference on Very Large Data Bases*, VLDB'07, Vienna, Austria, 2007.

[10] W. S. Cooper, F. C. Gey, and D. P. Dabney. Probabilistic retrieval based on staged logistic regression. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR'92, 1992.

[11] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld. Open information extraction from the web. *Commun. ACM*, Vol. 51, December 2008.

[12] J. H. H. Cheng, X. Yan and C. Hsu. Discriminative frequent pattern analysis for effective classification. In *Proceedings of ICDE*, ICDE'07, 2007.

[13] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, SIGMOD'00, 2000.

[14] Q. Hao, R. Cai, Y. Pang, and L. Zhang. From one tree to a forest: a unified solution for structured web data extraction. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR'11, 2011.

[15] N. Ireson, F. Ciravegna, M. E. Califf, D. Freitag, N. Kushmerick, and A. Lavelli. Evaluating machine learning for information extraction. In *Proceedings of the 22nd international conference on Machine learning*, ICML'05, Bonn, Germany, 2005.

[16] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD'02, 2002.

[17] S. Kim, X. Jin, and J. Han. DisIClass: discriminative frequent pattern-based image classification. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, MDMKDD'10, Washington, D.C., 2010.

[18] S. Kim, H. Kim, T. Weninger, J. Han, and H. D. Kim. Authorship classification: a discriminative syntactic tree mining approach. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, SIGIR'11, Beijing, China, 2011.

[19] X. Kong and P. S. Yu. Semi-supervised feature selection for graph classification. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD'10, Washington, DC, USA, 2010.

[20] R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. Zhu. Using structured queries for keyword information retrieval. 2007.

[21] N. Kushmerick. *Wrapper induction for information extraction.* PhD thesis, 1997.

[22] K. Lerman, S. N. Minton, and C. A. Knoblock. Wrapper maintenance: a machine learning approach. *J. Artif. Int. Res.*, Vol. 18, February 2003.

[23] W. Li, J. Han, and J. Pei. CMAR accurate and efficient classification based on multiple class-association rules. In *Proceedings of the 2001 IEEE International Conference on Data Mining*, ICDM'01, 2001.

[24] H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM J. Res. Dev.*, 1(4), Oct. 1957.

[25] B. L. W. H. Y. Ma. Integrating classification and association rule mining. In *Proceedings of the AAAI'98*, 1998.

[26] M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing and information retrieval. *J. ACM*, Vol. 7, July 1960.

[27] M. Paşca. Organizing and searching the world wide web of facts – step two: harnessing the wisdom of the crowds. In *Proceedings of the 16th international conference on World Wide Web*, WWW'07, Banff, Alberta, 2007.

[28] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR'98, 1998.

[29] S. E. ROBERTSON. A probability ranking principle in information retrieval, 1977.

[30] J. Rocchio. Relevance feedback in information retrieval. 1971.

[31] G. Salton and M. E. Lesk. Computer evaluation of indexing and text processing. *J. ACM*, Vol. 15, January 1968.

[32] S. Satpal, S. Bhadra, S. Sellamanickam, R. Rastogi, and P. Sen. Web information extraction using markov logic networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD'11, San Diego, California, 2011.

[33] S. Soderland. Learning to Extract Text-based Information from the World Wide Web. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, KDD'97, 1997.

[34] M. Thoma, H. Cheng, A. Gretton, J. Han, H.-P. Kriegel, A. J. Smola, L. Song, P. S. Yu, X. Yan, and K. M. Borgwardt. Near-optimal supervised feature selection among frequent subgraphs. In *Proceedings of Proceedings of SIAM International Conference on Data Mining*, SDM'09, 2009.

[35] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, ICML'08, 2008.

[36] X. Xin, J. Li, J. Tang, and Q. Luo. Academic conference homepage understanding using constrained hierarchical conditional random fields. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM'08, 2008.

[37] A. Yates, M. Cafarella, M. Banko, O. Etzioni, M. Broadhead, and S. Soderland. Textrunner: open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, NAACL'07, Rochester, New York, 2007.

[38] X. Yin and J. Han. Cpar: Classification based on predictive association rules. In *Proceedings of SIAM International Conference on Data Mining*, SDM'03, 2003.

[39] J. Zheng and Z. Nie. Language models for web object retrieval. In *Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing*, WiCOM'09, 2009.

[40] M. Zhou, T. Cheng, and K. C.-C. Chang. Docqs: a prototype system for supporting data-oriented content query. In *Proceedings of the 2010 international conference on Management of data*, SIGMOD'10, Indianapolis, Indiana, 2010.

[41] J. Zhu, Z. Nie, J.-R. Wen, B. Zhang, and W.-Y. Ma. 2d conditional random fields for web information extraction. In *Proceedings of the 22nd international conference on Machine learning*, ICML'05, 2005.

[42] J. Zhu, Z. Nie, J.-R. Wen, B. Zhang, and W.-Y. Ma. Simultaneous record detection and attribute labeling in web data extraction. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, KDD'06, Philadelphia, PA, 2006.

[43] Y. I. Zhuravlev. On algebraic approach to solution of recognition and classification problems. 1978.