P. N. Bondarenko
Saint Petersburg Electrotechnical University «LETI»

STRUCTURE OF SETUP DEVICE WITH STATE ACTUATION IN TIME

Request for comment new method of designing calculation devices, permissive improve response and reliability computing at the cost of state actuation, both at level discrete elements and at level device and system different hierarchy level.

Calculation devices, information processes, state actuation, structure, time

УДК 004.89; 004.912

А. Н. Рукавицын
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Разработка модели классификации веб-страниц с использованием методов интеллектуального анализа данных

Описывается разработка модели классификации веб-страниц с использованием методов интеллектуального анализа данных. Модель позволяет совершать мягкую мультиклассовую классификацию веб-страниц. Для разработки модели использовалась комбинация существующих и разработанных методов. Эксперименты показали увеличение точности классификации.

Классификация веб-страниц, интеллектуальный анализ данных, машинное обучение, обработка текста

Классификация веб-страниц. В настоящее время Интернет занимает важную роль в жизни человека. Информационное пространство в сети насчитывает уже миллионы гигабайт данных разного рода и отличается высоким уровнем доступности для пользователей. Развитие мобильных средств доступа в Интернет и веб-технологий позволило увеличить популярность Интернета.

Легкость создания и редактирования контента в Интернете приводит к распространению нежелательной информации, в частности, запрещенного контента в соответствии с законом «Об информации, информационных технологиях и о защите информации» [1]. Подобный контент может содержать не только жестокие или националистические шутки, но также экстремистские материалы или призывы к насилию и свержению конституционного строя. В статье «304th Military Intelligence Battalion» [2], написанной организацией «Federation of American Scientists», описывается использование террористами социальных сетей в качестве метода коммуникации и планирования террористических актов.

Определение тематики контента веб-страниц является одной из важнейших задач многих интернет-компаний. При верной категоризации можно производить более точную выборку рекламных блоков пользователю, что позволит улучшить продажи как мест размещения рекламных баннеров, так и рекламируемого товара. Кроме того, защита детей от нежелательной информации также является одной из основных возможных сфер применения категоризации контента.

В соответствии с российским законодательством контроль и обеспечение выполнения закона «Об информации, информационных технологиях и о защите информации» осуществляет федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор). Для блокировки веб-

контента была разработана автоматизированная информационная система ведения и использования базы данных о сайтах, содержащих запрещенную к распространению в России информацию. Система представляет собой единый реестр доменных имен, указателей страниц сайтов в Интернете и сетевых адресов, позволяющих идентифицировать сайты в Интернете, содержащие информацию, распространение которой в Российской Федерации запрещено.

На данный момент работа подобных сервисов выполняется посредством приема сообщений от граждан, юридических лиц, индивидуальных предпринимателей, органов государственной власти, органов местного самоуправления о наличии на страницах сайтов в Интернете противоправной информации и только в этом случае рассматривает блокировку контента.

Для автоматизации проверки и классификации веб-контента, а также для вычисления нежелательных веб-страниц и веб-сайтов можно воспользоваться методами интеллектуального анализа данных (ИАД). Цель технологии ИАД – выявить структуры и найти закономерности в слабоструктурированных данных.

Информация в Интернете отличается высокой динамикой: создание нового контента, его изменение и удаление порой занимают несколько секунд. Учитывая количество пользователей, которые могут создавать нежелательный контент, представляется затруднительным использование традиционных методов обнаружения и категоризации подобной информации.

Цель описываемых исследований – повысить точность классификации веб-страниц посредством разработки программного прототипа на основе моделей, методов и алгоритмов ИАД. Программный прототип должен удовлетворять следующим требованиям: обученная модель должна относить веб-страницу к одной из 14 категорий с точностью предсказания более 70 %.

Релевантные работы. Сегодня существует множество работ в области классификации веб-страниц. Основным отличием классификации веб-страниц от обычного текста является гипертекст. В связи с этим классификацию можно разделить на классификацию по содержимому на целевой веб-странице или по содержимому соседних веб-страниц.

Атрибуты веб-страниц можно разделить на текстовые и визуальные [3]. Текстовая информация более удобна для использования в классификации. Для этого используются несколько вариантов выбора атрибутов, таких, как bag-of-words, TF-IDF и n-gram. Такие методы обычно применяются в text mining исследованиях. Веб-страница использует HTML-теги как контейнеры, в которых может находиться текст. Такие теги могут быть выбраны в качестве атрибутов [4]. Используя «весы» [5], [6] для каждого тега, можно повысить точность классификации. Веб-страницы можно представить иерархией визуальных элементов [7], таких, как навигация, контент и другие блоки. Не всегда внедрение такого метода позволяет увеличить точность. В результате можно объединить данные подходы [8] для повышения точности классификации. Улучшить производительность классификатора также возможно [3] уменьшив размерность данных.

Использование соседних веб-страниц [9] позволяет значительно улучшить точность. Наиболее полезны для классификации веб-страницы, на которые ссылаются родительская веб-страница целевой, а также веб-страницы с теми же ссылками. При этом соседние веб-страницы также могут добавлять большое количество шума. При рассмотрении связей между веб-страницами можно построить граф и на основе этого получить вектор [10], используя методику как в TF-IDF.

Каждая веб-страница имеет свой уникальный адрес URL, по которому можно выполнить сравнительно быструю классификацию [11] без скачивания веб-страницы целиком. Хотя результат гораздо ниже, чем при обычной классификации текста. Использование n-gram при классификации по URL [12] веб-страницы может повысить эффективность.

Подготовка данных для обучения моделей классификации. Для достижения этой цели с помощью ИАД необходимо собрать и отфильтровать данные, обучить модель классификации с помощью метода машинного обучения (МО).

Одним из наиболее важных этапов при решении задач с помощью методов ИАД является сбор обучающей и тестовой выборки. Так как модель будет модифицироваться, необходимо сохранить данные, чтобы они были доступны для исследования влияния тех или иных параметров. В данном случае хранение в файле не столь удобно, так как необходимо будет использовать отношения, а в базах данных (БД) это реализовано очень просто. При сборе желательно хранить «сырые» данные, в рассматриваемом случае это исходный код веб-страниц. Первоначально было задумано собрать не только целевые веб-страницы, но и их

подстраницы, которые можно получить по ссылкам внутри веб-страницы. Оказалось, что это весьма трудоемкая задача по времени. Так, если одна страница скачивается в среднем секунду, то если взять 1000 веб-страниц с десятью подстраницами (ссылками), получим 10 000 страниц.

Перед тем как начать сбор данных, стоит получить доступ к ресурсу, откуда можно брать данные. Зачастую найти «нежелательный» или запрещенный контент можно только в том случае, если непосредственно заниматься подобной деятельностью. Такие категории, как терроризм или наркотики, довольно трудно найти с помощью стандартных средств поиска, таких, как поисковые веб-сайты. Поэтому были использованы данные с веб-сайтов, предоставляющие на бесплатной основе [13] ссылки на веб-страницы с категориями. В итоге были добавлены веб-страницы по 14 категориям.

Обучение моделей классификации на основе текстовых данных с веб-страниц. Существует ряд популярных инструментов для обработки полученных данных: RapidMiner, MATLab, Python, R, Theano, Weka. Возможности каждого примерно одинаковы, поэтому выбор можно считать субъективным. Было решено использовать Python и открытую библиотеку МО scikit-learn (sklearn).

При работе с текстом в библиотеке sklearn принято его оцифровать и представить в виде вектора (векторизовать), для чего есть 3 класса: HashingVectorizer, CountVectorizer, TfidfVectorizer. Основными параметрами при векторизации можно назвать количество атрибутов (n_features или max_features) и n_gram. Первоначально используем HashVectorizer с ограничением атрибутов на 20 000 и n_gram (1, 2). Для обучения воспользуемся алгоритмом Random Forest.

Для измерения точности классификации воспользуемся кроссвалидацией, которая уже реализована в sklearn. Для расчета используется матрица неточностей. Пусть tp – это true positive, tf – true false, тогда формула precision для бинарной классификации выглядит так:

$$P = \frac{\mathrm{tp}}{t\mathrm{p} + \mathrm{tf}}.$$

В случае мультиклассовой классификации используются [14] термины macro-averaging и micro-averaging:

$$P_{\mathrm{macro}} = \frac{\sum_{i=1}^{l} \frac{\mathrm{tp}_i}{\mathrm{tp}_i + \mathrm{fp}_i}}{l},$$

$$P_{\mathrm{micro}} = \frac{\sum_{i=1}^{l} \mathrm{tp}_i}{\sum_{i=1}^{l} \left(\mathrm{tp}_i + \mathrm{fp}_i\right)},$$

где l – классы.

Вариант macro-averaging дает каждому классу одинаковый вес в результирующей метрике, а micro-averaging – каждому документу. Если вес классов одинаков, с точки зрения стоимости ошибки имеет смысл использовать macro-averaging, иначе – micro-averaging и добавить больше документов этого класса в тестовую выборку. В данном случае классы по стоимости ошибки не равны, поэтому для оценки воспользуемся метрикой precision micro-averaging.

Очистим текст от всех символов, кроме букв, по шаблону: «[^a-zA-Z]», точность обученной модели – 0.660. Еще одним способом фильтрации является стемминг. Если используем стемминг из библиотеки NLTK, получим модель с точностью 0.665. Реализуем возможность фильтрации по стоп-словам – precision_micro 0.690. Используем все фильтры, а затем уберем окончания слов «ing|ly|ed|ious|ies|ive|es|s|ment». В таком случае получаем точность 0.697.

Попробуем заменить алгоритм классификации и протестируем модель. Рассмотрим алгоритмы SVM, Logistic Regression, kNN, Decision Tree. Результаты:

– kNN – 0.432;

– SVM – 0.725;

– Logistic Regression – 0.758;

– Decision Tree – 0.645.

Подбор параметров в целом можно назвать эмпирическим. Точность классификатора зависит как от векторизации, так и от алгоритма классификации. Попробуем улучшить модель добавив метаклассификаторы. Лучший результат показал алгоритм Bagging с Random Forest: 0.729.

Так как рассматривается задача мультиклассовой классификации, воспользуемся стратегией «one versus rest». Такая стратегия также реализована в библиотеке sklearn. В результате точность модели: 0.763.

На данном этапе установлено, что применение алгоритма Random Forest совместно с мета-алгоритмом Bagging и стратегией one versus rest позволяет получить наилучший результат. Точность полученной модели не достаточна для применения

на практике, поэтому попробуем улучшить ее, используя существующие методики классификации веб-страниц и предложив собственные.

В последующем все алгоритмы были применены повторно на каждом этапе. При этом высоким уровнем точности обладал алгоритм Logistic Regression, поэтому будут рассмотрены результаты с его применением.

Одним из важнейших этапов в подготовке данных на обучение является векторизация. Основными параметрами в этом случае выступают алгоритм получения атрибутов, количество слов (n-gram) и максимальное количество атрибутов. В библиотеке sklearn присутствует несколько реализаций векторизации данных: HashingVectorizer (HV) – преобразование входного текста в вектор со значениями количества вхождения слова в текст; TfidfVectorizer (TF) – преобразование в вектор со значениями отношения числа вхождений слова к общему количеству слов документа. В качестве дополнительного параметра в TF-IDF можно использовать сублинейную функцию (SB – sublinear), которая заменяет стандартный подсчет TF и позволяет убрать «обычные» слова из расчетов. Результаты указаны в таблице.

Число атрибутов	HV	TF	TF+SB
n-gram(1, 2)			
5000	747	789	790
20 000	767	778	781
max	767	772	778
n-gram (1, 3)			
5000	744	788	790
20 000	757	783	780
50 000	758	772	779
n-gram (2, 3)			
5000	529	687	699

Самую высокую точность дает использование векторизации с TF-IDF+sublinear с 5000 атрибутов и n-gram (1, 2).

При скачивании данных из Интернета проблема заключается в том, что данных достаточно много и определить, все ли веб-страницы были верно промаркированы, не кончилась ли аренда домена и т. д., достаточно сложно. Поэтому было решено провести фильтрацию веб-страниц по «правильно классифицируемым». Для этого векторизуем всю выборку, а затем обучаемся на полученном векторе и совершаем предсказание по нему же. При этом существует вероятность переобучения, число неверно предсказанных веб-страниц менее 5 %. Поэтому примем их за недоступные веб-сайты или категории, которые были изначально с ошибкой, и исключим их из выборки на обучение. Точность повысилась до 0.858.

В статье [3] утверждается возможность использования PCA (метод главных компонент). Это позволяет уменьшить анализируемое множество данных до размера, оптимального с точки зрения решаемой задачи. Этот метод используется для подготовки данных к классификации. Наилучший результат с разными алгоритмами и параметрами – 0.875. Результаты показывают повышение точности, но в дальнейшем метод не будет влиять на результаты исследования.

Использование специальных атрибутов веб-страницы для повышения точности классификации.

Классификация веб-страниц на основе тегов. Важнейшим фактором в ИАД является правильное использование атрибутов. На веб-страницах можно в качестве атрибутов выбрать теги [2]–[4], например «title», «a», «meta», и блоки «header», «content» и «footer», которые были получены селекторами по тегу, id и class. Так как на этот текст был акцент со стороны разработчиков веб-сайта, то, возможно, роль их гораздо выше. Использование отдельных классификаторов для каждого из атрибутов не дает повышения точности. Попробуем совместить их с существующим, тем самым увеличив количество вхождений ключевых слов. В результате опытным путем вычислено, что добавление тегов «title» и «meta» (description, keywords) дает точность 0.767.

Классификация веб-страниц на основе URL. Для повышения точности классификации попробуем воспользоваться другими атрибутами. В нескольких работах [9], [10] была указана возможность использования URL (Uniform Resource Locator) веб-страницы для ее классификации. URL является стандартизированным способом записи адреса ресурса в сети. В Интернете для веб-сайтов используется упрощенный формат записи URL: <схема>://<хост>:<порт>/<путь>?<параметры>#<якорь>. Каждая веб-страница использует свой уникальный URL (адрес), при этом появилось неформальное правило «хорошего тона» у разра-

ботчков делать человекопонятные адреса – адреса, содержащие читаемые слова, не аббревиатуры или непонятные идентификаторы.

В связи с этим использование URL является не только быстрым способом классификации. В адресе веб-страницы можно встретить основные ключевые слова. Воспользуемся такой возможностью и отфильтруем адрес от специальных символов и цифр, а также слов, которые присутствуют почти во всех адресах, таких, как расширения, домены первого уровня и протоколы. Обычно адрес состоит из нескольких слов без разделения, поэтому воспользуемся символьным n-gram. После выбора подходящих параметров была получена точность 0.631. Точность достаточно низкая, что ожидаемо, так как адреса могут состоять из аббревиатур, сокращений или выдуманных слов. Воспользуемся еще одним свойством веб-страниц – заголовком, в котором хранится в текстовой форме основная идея содержимого. Если произвести классификацию по заголовку, то точность достигает 0.669. Объединим атрибуты, тем самым добавив недостающие ключевые слова отдельных атрибутов. Получим точность 0.713.

Классификация веб-страниц на основе Word2vec. Одним из способов получения новых атрибутов является метод с использованием [15] word2vec. Данная библиотека представляет слова в виде числового вектора, где минимальное расстояние между векторами будет у наиболее схожих по смыслу слов. Для классификации текста можно использовать в качестве атрибута среднее арифметическое (average vector) или набор центроидов (bag of centroids). Результат классификации в обоих случаях соответственно 0.860 и 0.854. В данном случае точность сильно зависит от размера выборки обучения. Возможно, при условии использования более крупных баз данных результаты могли быть выше.

Классификация веб-страниц на основе двухуровневых ключевых слов. Еще одна сложность классификации заключается в определении принадлежности текста, который может относиться одновременно к множеству категорий, но при этом ни к одной из доступных, например «новости».

Сначала добавим категорию «новости». Выполним поиск слов с высоким весом используя в качестве ключевых слов news, finance, sport, political, politics, health, tech, technology, culture, art, weather, economy, business, lifestyle, world, national, travel, celebrity, movies, music, fashion.

Использование ключевых слов не ново и почти ничем не отличается от того, как определяются остальные категории. Поэтому введем критерий оценки и подсчета – производим вычисление при наличии хотя бы двух вхождений основного ключевого слова, в данном случае «news». Таким образом, имеется 2 уровня ключевых слов, где на первом уровне находится основное, а на втором – остальные слова, по которым выполняется проверка. Для увеличения процента соответствия берем многомерный вектор стоп-слов с использованием синонимов – это позволяет получить более корректный процент вхождения слов без подсчета слов с одинаковым смыслом.

Понижение уровня ошибки в случае неопределенности. При классификации очень часто используется матрица стоимости ошибки, которая позволяет оценить результаты классификации для каждой категории. При мультиклассовой классификации еще одной проблемой является выбор категории при отсутствии верной, поэтому обучение было выполнено с использованием присваивания «неизвестной».

Еще одним способом снижения процента ошибки является добавление в алгоритм предсказания проверки по объему. Если размер достаточно мал, то категория веб-страницы будет «неизвестной». Такими веб-страницами могут быть заглушки или веб-страницы, не содержащие текста.

Также для повышения точности добавим использование списка доменов с известными категориями.

Подобные решения сложно проверить на существующей выборке из-за возможного переобучения – список доменов берется из обучающей выборки, а также проверка по объему возможна лишь при наличии страниц в выборке в качестве отдельной категории.

Метод иерархической классификации. Применение стратегии «one versus rest» позволило улучшить качество модели. Поэтому используем схожий метод, отличающийся тем, что можно использовать модели с отдельно подобранными параметрами и алгоритмом. Для этого возьмем одиночные бинарные классификаторы, обученные строго под свои категории. Для получения итоговой категории применим метод голосования. Добиться повышения точности удалось за счет округления выходных данных классификаторов. При классификации без обучения по «правильно предсказанным» точность составила 0.879. При классификации с обучением по «правильно предсказанным» – 0.928.

Метод голосования является не самым лучшим, поэтому попробуем его улучшить. В качестве замены воспользуемся еще одним классификатором, который будет обучен по результатам предыдущих. В таком случае необходимо разделить обучение на несколько уровней. Для начала данные делят на первый (L_1) и второй (L_2) уровни. Выборка делится при полном смешивании порядка (позволяет более корректно определять точность), причем на каждом уровне присутствует равное соотношение данных из каждой категории. Первому уровню устанавливаются категории в бинарном виде (0, 1) в соответствии с принадлежностью. При обучении L_1 (атомарных классификаторов) обучаются по каждой категории из L_1. При обучении L_2 (рефери) используются результаты атомарных классификаторов (процент совпадения с категорией). В итоге на первом уровне получаем предсказание по каждой категории, а затем на втором уровне рефери выдает окончательное решение по ним.

Данная методика позволяет связать несколько классификаторов, обученных на разных атрибутах, с разными алгоритмами. Используя ранее полученные атрибуты можно построить модель для классификации. В качестве алгоритма подведения итога высокую точность показал SVM. При классификации без обучения по «правильно предсказанным» точность 0.887. При классификации с обучением по «правильно предсказанным» точность 0.961. Таким образом, удалось выбрать наиболее подходящие алгоритмы и параметры для каждой категории и для рефери, тем самым повысив точность классификации до 0.961.

Метод классификации с помощью «соседних» веб-страниц. Гипертекстовые особенности, такие, как ссылки, также можно использовать [9] в качестве атрибутов. Простейший способ – использование набора подстраниц в качестве атрибутов для обучения, поэтому сначала рассмотрим его. Можно попробовать классифицировать целевую веб-страницу по ее подстраницам, затем методом голосования по полученному списку результатов классификации получить результат. Процесс несложный, но трудоемкий, появляется необходимость скачивать все подстраницы целевой веб-страницы, поэтому была использована лишь часть выборки. Для того чтобы существенно сократить временные затраты, можно попробовать ограничить количество скачиваемых подстраниц используя бинарную выборку – собрать 6 случайных ссылок, две из которых находятся в начале страницы, следующие две – в середине, а остальные – в конце. Для начала проведем классификацию по новой выборке с использованием только целевой веб-страницы (точность 0.847). Выполним классификацию по подстраницам (точность 0.690). Результат оказался гораздо ниже. Можно попробовать объединить текст всех подстраниц (точность 0.771). Данный подход не позволил достичь желаемого результата из-за ограниченности подстраниц, не все подстраницы могли быть доступны или относились к контенту целевой. Поэтому была использована та же часть выборки с бинарной категорией, но уже со всеми подстраницами. При классификации по подстраницам результат оказался выше: 0.850.

Проведя классификацию только по контенту целевой веб-страницы, среди результатов предсказаний учитывалась уверенность в правильности. Это позволило разделить веб-страницы по результатам на группы с высокой и низкой уверенностью в верности. Разделив выборку на эти группы, рассмотрим взаимосвязь с классификацией по подстраницам. Для веб-страниц с высокой уверенностью подстраницы совпадали с целевой в 90 % случаев как для верно, так и неверно классифицированных. При классификации веб-страниц с низкой уверенностью подстраницы совпали в 44 % случаев. Проведем классификацию по подстраницам только для веб-страниц с низкой уверенностью, а по целевым – с высокой (точность 0.852). Данный метод позволяет повысить точность и использовать новый вид атрибутов, но точность и скорость получения результата зависят от количества подстраниц. Веб-страницы с более сложным контентом для классификации, например «news», не дают хорошего результата из-за содержания смешанного набора категорий. Также целевые веб-страницы, содержащие изображения, не позволяют получить подстраницы, поэтому рассмотрим возможность использования соседних веб-страниц. В статье [9] авторы рассматривают схему (рис. 1) того, как можно построить отношения между веб-страницами используя ссылки. На схеме расположены веб-страницы со связями на двух уровнях (radius 1-2) в зависимости от их дистанции относительно целевой веб-страницы (Target Page). На первом уровне целевая веб-страница может иметь родительскую (Parent) и дочерние (Child) веб-страницы. На втором уровне Parent веб-страницы могут иметь родительские (Grandparent) и дочерние (Sibling). Аналогично, Child веб-страницы имеют дочерние (Grandchild) и родительские (Spouse). Также авторы утверждают, что при классификации важны именно

соседние Sibling и Spouse (рис. 1) веб-страницы. На практике получить выборку по подобной схеме можно лишь с использованием поискового робота для сбора огромного числа веб-сайтов разных тематик, что требует значительных временных затрат. Такой подход практически не применим, так как информация (а точнее, веб-страницы) появляется и изменяется ежесекундно. Практическое применение данного метода остается под вопросом.

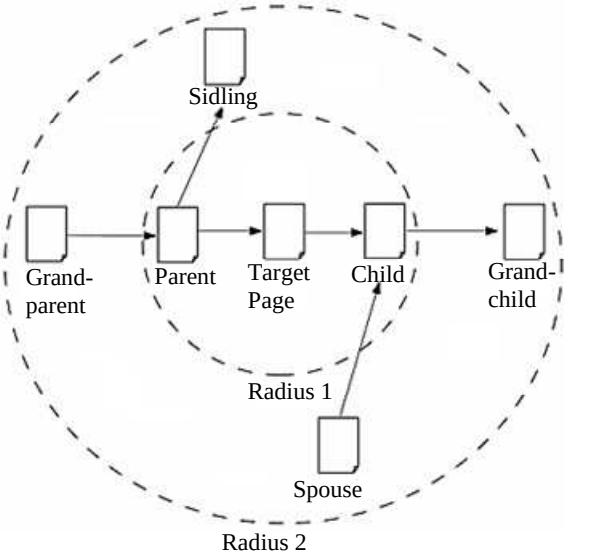

Рис. 1

Для получения родительских страниц воспользуемся доступными «backlink» сервисами. В результате точность классификации: 0.795.

При рассмотрении методов скачивания возникла идея подсчета ссылок. Списки известных доменов веб-сайтов с категориями можно использовать в качестве атрибутов. Таким образом получаем вектор, в котором атрибуты – это колонки с количеством ссылок на категорию. Данный метод не доказал свою состоятельность. Причиной тому послужило малое содержание ссылок на внешние источники или их полное отсутствие на целевых веб-страницах.

Обсуждение. Таким образом, добавление адреса или заголовка в виде дополнительных атрибутов не позволило увеличить точность модели. Возможно, для быстрой классификации этот метод удобен, но точность настолько низка, что это не позволяет использовать его в данном случае.

Использование word2vec не внесло значительных изменений в результаты. Возможно, если выборка данных на обучение была бы гораздо больше, то точность могла бы увеличиться, благодаря наличию большего количества важных слов в центроиде.

Описанные методы, связанные со ссылками, не позволяют получить ожидаемое улучшение точности. Это связанно с правильным отбором ссылок. Важно исследовать, какие ссылки наиболее полезны. Кроме того, использование родительских веб-страниц возможно лишь в виде исследовательской работы, когда в идеальных условиях можно собрать за долгий период необходимые данные. В случае практического применения этот метод сложно повторить из-за отсутствия механизма быстрого получения родительских веб-страниц. Также продолжительность классификации при использовании соседних веб-страниц занимает значительное время из-за необ-

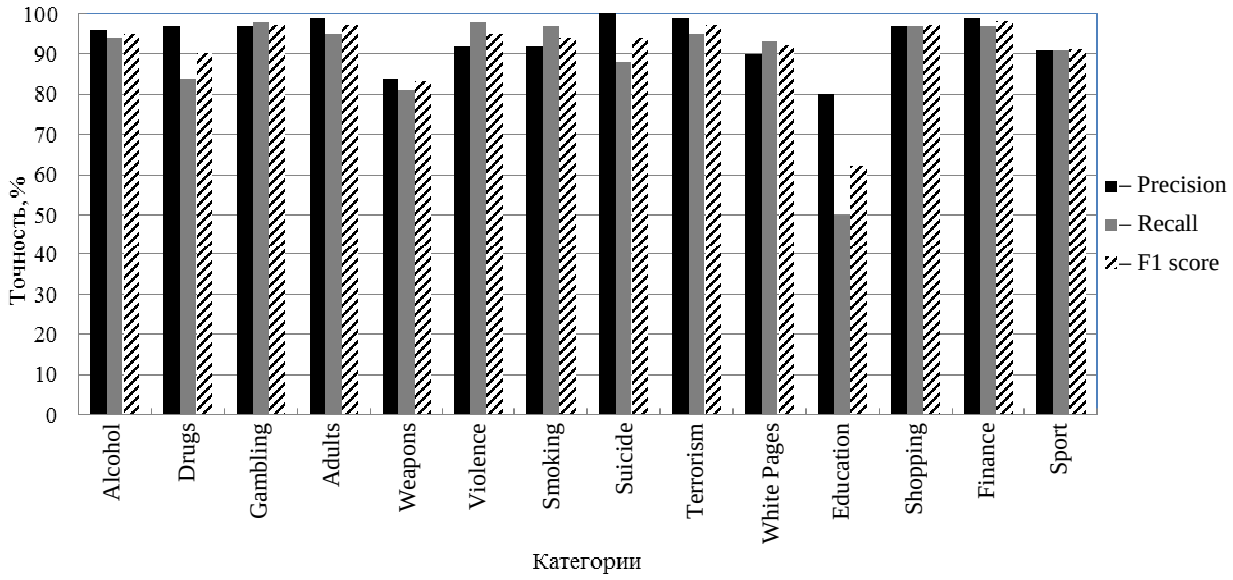

Рис. 2

ходимости скачивания их. Подстраницы, напротив, могут быть получены простым способом, но не дают высокой точности, а лишь позволяют решить вопрос выбора категории для страниц с низким уровнем уверенности.

Как выяснилось, одним из важнейших факторов является использование качественной выборки, которая не будет содержать пустых или неверно промаркированных данных. Это было показано после фильтрации выборки при обучении по правильно предсказанным. Классификацию по нескольким категориям можно улучшить, используя атомарные классификаторы, а также рефери для расстановки весов. График метрик представлен на рис. 2.

В статье было рассмотрено большинство известных методов классификации веб-страниц, а также предложены собственные. Также предложена модель для классификации веб-страниц с возможностью определения одной из 14 категорий и точностью precision micro-averaging – 96 %. Точность полученной модели не является идеальной. Основную сложность при классификации веб-страниц представляют те, которые не содержат текста. Обычно человек оценивает содержимое визуально. Возможно, добавление атрибутов такого типа поможет улучшить качество классификации, поэтому одним из возможных направлений в исследовании классификации может быть использование новых видов атрибутов, таких, как изображения.

Публикация выполнена в рамках государственной работы «Организация проведения научных исследований» базовой части государственного задания Минобрнауки России, а также проектной части государственного задания Минобрнауки России (ЗАДАНИЕ № 2.136.2014/К) и поддержана грантом РФФИ №16-07-00625.

СПИСОК ЛИТЕРАТУРЫ

1. Об информации, информационных технологиях и о защите информации: федер. закон от 27.07.2006 № 149-ФЗ (ред. от 13.07.2015) (с изм. и доп., вступ. в силу с 10.01.2016) // Рос. газ. 2006. 29 июля.

2. Sample Overview: alQaida-Like mobile discussions & potential creative uses // Site of Federation Of American Scientists. URL: http://fas.org/irp/eprint/mobile.pdf. (Дата обращения 15.11.2015).

3. Qi X., Davison B. D. Web page classification: Features and algorithms // J. ACM Computing Surveys. 2009. Vol. 41, iss. 2, № 12.

4. A fuzzy system for the web page representation / A. Ribeiro, V. Fresno, C. M. Garcia-Alegre, D. Guinea // Intelligent exploration of the Web. Physica-Verlag HD. 2003. P. 19–37.

5. Kwon O. W., Lee J. H. Web page classification based on k-nearest neighbor approach // IRAL '00 Proc. of the fifth Intern. workshop on Information retrieval with Asian languages, Hong Kong, China, 2000. P. 9–15.

6. Kwon O. W., Lee J. H. Text categorization based on k-nearest neighbor approach for Web site classification // Inform. Process. Manage. 2003. Vol. 39, № 1. P. 25–44.

7. Visual Adjacency Multigraphs – a Novel Approach for a Web Page Classification / M. Kovacevic, M. Diligenti, M. Gori, V. Milutinovic // Proc. of SAWM 2004 workshop, ECML 2004, New York, NY USA, 2004.

8. Web-page classification through summarization / D. Shen, Z. Chen, Q. Yang, H. J. Zeng, B. Zhang, Y. Lu, W. Y. Ma // Proc. of the 27th annual Intern. ACM SIGIR conf. on Research and development in information retrieval, UK, Sheffield, 2004. P. 242–249.

9. Qi X., Davison B. D. Web page classification: Features and algorithms // Computing Surveys. 2009. Vol. 41, № 2. P. 12.

10. Belmouhcine A., Benkhalifa M. Implicit Links based Web Page Representation for Web Page Classification // Proc. of the 5th Intern. conf. on Web Intelligence, Mining and Semantics. Larnaca, Cyprus, 2015. P. 12.

11. Kan M. Y., Thi H. O. N. Fast webpage classification using URL features // Proc. of the 14th ACM Intern. conf. on Information and knowledge management. Bremen, Germany, 2005. P. 325–326.

12. Abdallah T. A., Iglesia B. URL-based web page classification-a new method for URL-based web page classification using n-gram language models // SCITEPRESS Digital Library-KDIR 2014-Intern. conf. on Knowledge Discovery and Information Retrieval. Rome, Italy, 2014. P. 14.

13. Черный список веб-адресов // Официальный сайт проекта SquidGuard фильтрации и редиректа запросов. URL: http://www.squidguard.org/blacklists.html.

14. Sokolova M., Lapalme G. A systematic analysis of performance measures for classification tasks // Information Processing & Management. 2009. Vol. 45, № 4. P. 427–437.

15. Learning word vectors for sentiment analysis / A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, C. Potts // Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011. Vol. 1. P. 142–150.

A. N. Rukavitsyn
Saint Petersburg Electrotechnical University «LETI»

THE DEVELOPMENT OF WEB PAGE CLASSIFICATION MODEL BASED ON DATA MINING TECHNIQUES

Describes the development of web page classification model using data mining techniques. The model allows to make multi-label soft classification of the web pages. For developing of this classification model we used the combination of developing methods with existing methods. The experiments show increasing of classification precision with describing of metrics.

Web page classification, data mining, machine learning, text processing

УДК 681.322

В. В. Цехановский, В. Д. Чертовской
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Программная структура модели процесса оптимального планирования в многоуровневой системе

Рассмотрена программная реализация математической модели функционирования производства. Модель охватывает бизнес-процесс «Производство» с помощью однородного метода, описывающего процессы планирования и управления средствами динамического линейного программирования. Сформулированы требования к составляющим структуры, по которым выбрана система программных средств.

Математическое описание, однородный метод, процесс планирования, требования к структуре, система программных средств

С появлением фактически нового класса адаптивных систем управления производством с качественно изменяющейся целью в процедуре функционирования было сформировано их математическое системное описание на основе однородного метода [1]–[3]. Под производством понимается совокупность подсистем технико-экономического планирования и оперативного управления основным производством – при подсистемном представлении [1] или бизнес-процесс «Производство» – при процедурном представлении [2], [3].

Ввиду сложности предложенных авторами математических моделей и большого объема данных важное значение приобретает разработка и реализация компьютерных моделей, которые позволяют проводить исследование динамических свойств и проектирование многоуровневых автоматизированных адаптивных систем.

Постановка задачи. В настоящее время имеющиеся модели используют [4], как правило, одноуровневые варианты с решением задач «прямого счета». В то же время перспективным с позиций повышения конкурентоспособности предприятий являются многоуровневые адаптивные системы, использующие оптимизационные алгоритмы. Такой класс систем и рассматривается в данной статье.

Сам процесс проектирования процедур функционирования и адаптации может быть представлен схемой переходов

$$M(0a) \rightarrow M(0) \rightarrow M(1a) \rightarrow M(1) \rightarrow M(2) \rightarrow M(3).$$

Дело в том, что простота описания в многоуровневых системах противоречит необходимости учета ее важнейших координат и связей. Следствием являются различные уровни математической абстракции описания системы, определяемые целью системы и не зависящие от уровня иерархии управления: теоретико-множественный M(0a), M(0); абстрактно-алгебраический M(1a); топологический M(1); структурный M(2); параметрический M(3).