Обробка інформації в складних організаційних системах

УДК 004.032.26

Г.Г. Асеев

Харьковская государственная академия культуры, Харьков

МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ ПОТОКОВ ИНТЕРНЕТ-ДОКУМЕНТОВ

Рассмотрены особенности основных направлений методов web mining. Главными из них является классификация с обучением: деревья решений, нейронные сети и метод Naive Bayes. Приведены рекомендации к их использованию.

Ключевые слова: *гипертекстовые документы, методы web mining, классификация с обучением, деревья решений, нейронная сеть, байесовская классификация.*

Введение

В результате развития информационных технологий количество данных, накопленных человечеством в электронном виде, возрастает быстрыми темпами. Эти данные существует вокруг нас в различных видах: тексты, изображения, аудио, видео, гипертекстовые документы, реляционные базы данных и т.д. Огромное количество данных появилось в результате повсеместного использования сети Интернет, которая значительно облегчила доступ к информации из географически удаленных точек Земли.

В настоящее время Интернет стал местом, где сконцентрировано больше всего информации. Каждый день миллионы пользователей пользуются услугами, генерируют текстовый (блоги, социальные сети, форумы) и медиа-контент *(*youtube.com, flickr.com*)*, покупают товары, читают новости и т.д. С приходом эры Web 2.0 именно пользователи стали главным действующим лицом всемирной паутины, которое порождает и использует огромное количество данных. Наличие и использование такого большого объема данных порождает ряд проблем: поиск релевантной информации, получение новых знаний, изучение спроса посетителей. Здесь приходит на помощь такое направление как Web mining. Web mining – это набор методов data mining, позволяющих обрабатывать данные, находящиеся в web-среде.

Цель данной работы - рассмотрение и совершенствование основных направлений методов web mining. В данной статье ограничимся классификацией с обучением: деревья решений, нейронные сети и метод Naive Bayes.

Постановка задачи. Все задачи, решаемые методами обработки и анализа данных сети Интернет, можно разделить на следующие группы:

1) Обнаружение шаблонов поведения пользователей.

2) Поиск релевантной информации.

3) Извлечение информации (контента) из неструктурированных источников.

Каждое из трёх направлений в настоящее время активно развивается, появляются новые методы и алгоритмы решения таких задач, что позволяет использовать «интеллектуальные» приёмы анализа огромных массивов данных, возникающих в результате коллективных действий пользователей.

Основные этапы web mining

Рассмотрим основные этапы получения результата [1 - 3] в web mining с учетом специфики веб-среды:

1. Сбор данных (логи, специальная собираемая пользовательская статистика и пр.).

2. Предобработка данных. Поскольку собранные данные характеризуются значительной степенью зашумлённости, они нуждаются в очистке. Важнейшими критериями при выборе метода очистки являются устойчивость и эффективность метода.

3. Обработка и анализ данных. Применение конкретных методов (с учётом решаемой задачи) для получения практических результатов, анализ адекватности и применимости полученных результатов.

4. Интерпретация полученных результатов и их перенос на основу бизнес-задачи.

Весь вышеописанный цикл может происходить в течение одного сеанса работы пользователя с веб-приложением с «подстройкой» этого приложения под запросы пользователя, либо анализ может выполняться периодически, на основе действий большого количества пользователей (анализ эффективности различных блоков сайта).

Методы web mining

В сфере анализа веб-данных используется достаточно большое количество различных методов и алгоритмов обработки данных. Следует отметить, что такие методы должны обладать хорошей робастностью (устойчивостью), поскольку данные очень сильно зашумлены, могут измеряться в различных шкалах (дискретные, непрерывные, текстовые, даты и т.д.) и с достаточно большим разбросом значений.

Например, методы классификации используются для выделения групп пользователей сайта и предоставления каждой группе уникального контента, интересующего именно её. Различные методы поиска ассоциаций используются для выявления наиболее значимых паттернов в поведении пользователей с целью определения наиболее значимых страниц и ресурсов сайта, сокращения пути к ним и удаления ненужного «информационного мусора» со страниц. Методы анализа текстовых документов (text mining) применяются для получения сгруппированных документов с оглавлением и поиска одинаковых по смыслу страниц. Например, это используют роботы-сканеры новостей, настроенные на получение всех ссылок со всех новостных сайтов и т.д.

Также следует отметить, что для решения многих задач используется целый набор методов и алгоритмов, которые в совокупности более эффективно позволяют решать бизнес-задачи.

Классификация с обучением

Пусть имеется набор объектов, каждый из которых принадлежит одному из m классов. В качестве примера можно привести клиентов банка, которые могут быть отнесены к классу добросовестных или недобросовестных заемщиков, или множество солдат на фотоснимке, которые можно разделить на «своих» и «чужих». Задачей классификации с обучением является составление правила, по которому для любого объекта можно с большой степенью достоверности определить класс, которому данный объект принадлежит.

Пусть $x_1,...,x_k$ – атрибуты объекта, m – количество классов. В результате классификации должна быть получена некоторая функция $f(x_1,...,x_k)$, значение которой принадлежит {1,..., m}, и задает номер (метку) класса, которому принадлежит объект с атрибутами $x_1,...,x_k$.

В распоряжении у исследователя обычно имеется некоторый набор объектов, у которых метка класса уже известна. Эти объекты могут быть использованы для обучения модели, то есть подбора параметров модели классификации, и для тестирования построенной модели классификации.

Классификация с обучением подразумевает следующие действия:

Подготовка данных. Имеющийся набор объектов с известными метками классов разбивается на 2 части: обучающую выборку и тестовую выборку. Желательно, чтобы это разбиение было произведено случайным образом. Чаще всего обучающая выборка имеет размер больше, чем тестовая.

Обучение модели. Параметры модели классификации подбираются на основе обучающей выборки таким образом, чтобы добиться наилучшего соответствия между предсказанными и фактическими метками классов.

3. *Тестирование модели*. Полученная в результате обучения модель проверяется на достоверность. Для этого вычисляется процент неверных результатов классификации объектов из тестовой выборки.

Классификация с обучением имеет множество приложений, например, в таких областях, как кредитование, медицинская диагностика, предсказание доходов, маркетинг и пр. Мы рассмотрим три метода классификации с обучением: деревья решений, нейронные сети и метод Naive Bayes для целей web mining.

Дерево решений

Дерево решений – это дерево, в котором каждой внутренней вершине поставлен в соответствие некоторый атрибут, каждая ветвь, выходящая из данной вершины, соответствует одному из возможных значений атрибута, а каждому листу дерева сопоставлен конкретный класс или набор вероятностей классов.

Для того чтобы классифицировать новый объект, необходимо двигаться по дереву сверху вниз, начиная с корня. При этом на каждом внутреннем узле дерева выбирается та ветвь, которая соответствует фактическому значению соответствующего атрибута. Добравшись до листа дерева, получаем тот класс, которому принадлежит объект согласно классифицирующему правилу.

Основная проблема состоит в том, чтобы построить достаточно хорошее дерево решений. Модифицированный алгоритм решения этой задачи, известный как алгоритм ID3 [4], заключается в следующем.

На шаге 1 данного алгоритма используется понятие информационного выигрыша атрибута. Пусть обучающая выборка S состоит из s объектов, m – это количество рассматриваемых классов, s_i – это число объектов из S, принадлежащих классу с номером i. Количество информации, необходимое для того, чтобы сообщить класс произвольного объекта, равно

$$I(s_1,...,s_m) = -\sum_{i=1}^{m} p_i \log_2(p_i),\ i = 1,$$

где p_i – это вероятность того, что произвольный объ-

ект принадлежит классу с номером i, оцениваемая величиной

$$p_i = \frac{s_i}{s_1 + \ldots + s_m}.$$

Пусть некоторый заданный атрибут A может иметь ν различных значений $\{a_1, a_2, \ldots, a_v\}$. Атрибут A может быть использован для разбиения множества S на ν подмножеств $\{S_1, \ldots, S_v\}$, где S_j содержит такие объекты из S, для которых атрибут A имеет значение a_j. Если на шаге 1 алгоритма выбрать атрибут A, то подмножества $S_1, \ldots, S_v$ соответствуют ветвям, идущим от вершины, содержащей множество S.

Пусть s_{ij} – это количество объектов касса i в подмножестве S_j. Средняя информация, основанная на разбиении выборки с использованием атрибута A, равна

$$E(A) = \sum_{j=1}^{н} \frac{s_{1j} + \ldots + s_{mj}}{s} I\left(s_{1j} + \ldots + s_{mj}\right). \qquad (1)$$

Величина $\frac{s_{1j} + \ldots + s_{mj}}{s}$ выступает в качестве веса j-го подмножества и равна числу объектов в подмножестве S_j, деленное на общее число объектов из S. Чем меньше значение (1), тем более однородны (в среднем) множества S_j по классовой принадлежности. Заметим, что для заданного множества S_j:

$$I\left(s_{1j} + \ldots + s_{mj}\right) = -\sum_{i=1}^{m} p_{ij} \log_2 (p_{ij}),$$

где p_{ij} – вероятность того, что произвольный объект из S_j принадлежит классу i:

$$p_{ij} = s_{ij} / |S_i|.$$

Информационным выигрышем, соответствующим выбору атрибута A в качестве разбивающего множество S, назовем величину

$$Ent(A) = I\left(s_{1j} + \ldots + s_{mj}\right) - E(A).$$

Значение Ent(A) может рассматриваться, как среднее сокращение энтропии после того, как стало известным значение атрибута A.

Алгоритм ID3 на шаге 1 вычисляет информационный выигрыш каждого атрибута. Атрибут с наибольшим информационным выигрышем выбирается в качестве разбивающего атрибута для заданного множества S. Создается новая вершина, которая помечается этим атрибутом. Затем для каждого значения этого атрибута создаются ветви дерева, разбивающие S на соответствующие подмножества $S_1, \ldots, S_v$. Для каждой созданной ветви процедура повторяется вновь.

В результате работы данного алгоритма получается некоторое дерево решений, которое можно использовать для классификации. Однако часто полученное дерево бывает довольно громоздким, и его желательно упростить. Для этого требуется процедура упрощения дерева. Помимо того, что данная процедура позволяет получить более компактный и простой вид дерева решений, она часто позволяет значительно сократить время вычислений.

Можно предложить два основных подхода к проблеме упрощения дерева решений. Первый подход – упрощение дерева на этапе его создания. В рамках этого подхода на этапе рассмотрения какой-либо вершины может быть принято решение не создавать ветви, выходящие из нее, и не делить соответствующее множество объектов выборки. Это решение может быть принято, если информационный выигрыш от ветвления из данной вершины меньше установленного порога. В результате данная вершина становится листом, который может быть помечен меткой самого представительного класса в соответствующей выборке.

Второй подход предполагает удаление ветвей из уже «выращенного» дерева. Для каждой нелистовой вершины дерева рассматриваются два варианта: когда дерево остается неизменным, и когда из дерева удаляются все ветви, выходящие их данной вершины, а сама вершина становится листом и помечается меткой самого представительного класса для выборки, соответствующей вершине.

Нейронные сети

Задачи, которые могут решаться с помощью искусственных нейронных сетей, включают задачу классификации, кластерный анализ, аппроксимацию функций, задачу прогноза, оптимизации, поиска по содержимому и распознавания образов. Искусственные нейронные сети (ИНС) могут быть представлены, как взвешенные ориентированные графы, в которых вершины соответствуют нейронам, а ориентированные ребра с весами соответствуют связям между выходами нейронов и входами нейронов.

По структуре связей нейронные сети могут быть разделены на два класса:

1. *Сети прямого распространения*: соответствующий сети граф не имеет петель, то есть обратные связи невозможны. Примерами таких сетей являются однослойный перцептрон, многослойный перцептрон, сети Кохонена.

2. *Рекуррентные сети* (*сети обратного распространения*): возможны циклы, а значит обратные связи. Примером является сеть Хопфилда.

Мы рассмотрим применение многослойного перцептрона к задаче классификации. Многослойный перцептрон состоит из нескольких слоев нейронов: входного слоя, выходного слоя и нескольких скрытых слоев.

Нейронная сеть может рассматриваться, как вычислительная система, которой на вход подается вектор ввода, а результатом вычислений является

вектор вывода. При этом каждая компонента вектора ввода подается через соответствующий нейрон входного слоя, а вектор вывода соответствует нейронам выходного слоя.

Все слои нейронной сети пронумерованы последовательно от 0 до m, где номер 0 соответствует входному слою, а номер m – выходному. Обозначим n_k – количество нейронов в слое k.

Нейроны каждого слоя соединены со всеми нейронами смежных слоев. Для каждой пары связанных нейронов определен вес этой связи $w_{ij}^{(k)}$, где i – номер нейрона слоя $k-1$, j – номер нейрона слоя k.

Выходом каждого нейрона является величина $x_i^{(k)}$, где i – номер нейрона слоя k. Она рассчитывается на основе входов нейрона и связей этого нейрона с нейронами предыдущих слоев:

$$x_i^{(k)} = f\left(S_j^{(k)}\right),$$

где

$$S_j^{(k)} = \sum_{i=1}^{n_{k-1}} x_i^{(k-1)} w_{ij}^{(k)}, \quad f(x) = \frac{1}{1+e^{-\alpha x}}.$$

Функция f(x) называется логистической функцией, ее применение гарантирует, что величина $x_i^{(k)}$ принадлежит отрезку [0, 1]. Параметр α выбирается пользователем.

Компоненту вектора ввода с номером i обозначим $x_i^{(0)}$. Считаем, что входные данные преобразованы таким образом, что $x_i^{(0)} \in [0, 1]$ для всех i. Выходом нейронной сети в соответствии с используемыми обозначениями является вектор, i-я компонента которого равна $x_i^{(m)}$.

Процесс обучения нейронной сети состоит в том, чтобы подобрать ее веса $w_{ij}^{(k)}$ таким образом, чтобы для обучающей выборки результаты на выходе нейронной сети как можно меньше отличались от требуемых результатов. Мерой ошибки является величина

$$E = \frac{1}{2}\sum_{p=1}^{n_m}\left(x_p^{(m)} - d_p\right)^2, \qquad (2)$$

где d_i – требуемые результаты на выходе. Например, для задачи классификации $d_i = 1$, если рассматриваемый элемент обучающей выборки с атрибутами $x_i^{(0)}$ принадлежит классу i, и $d_i = 0$ в обратном случае.

Алгоритм обучения нейронной сети, который называется алгоритмом обратного распространения ошибки, основан на методе градиентного спуска. Это означает, что величины $w_{ij}^{(k)}$ на каждом шаге «немного» сдвигаются в сторону антиградиента функции ошибок E:

$$w_{ij}^{(k)} := w_{ij}^{(k)} + \Delta w_{ij}^{(k)}, \quad \Delta w_{ij}^{(k)} = -\varepsilon\frac{\partial E}{\partial w_{ij}^{(k)}},$$

где ε – некоторое небольшое положительное число, называемое скоростью обучения, обычно лежащее в интервале от 0 до 1. Если ε слишком мало, то процесс обучения занимает слишком много времени, если ε слишком велико, то процесс может быстро «свалиться» к некоторому неадекватному локальному минимуму, или осциллировать между такими локальными минимумами. Часто в качестве ε выбирается величина 1/t, где t – номер итерации алгоритма.

Рассмотрим вопрос вычисления величины $\frac{\partial E}{\partial w_{ij}^{(k)}}$. Обозначим

$$z_j^{(m)} = f'\left(S_j^{(m)}\right)\left(x_j^{(m)} - d_j\right), \qquad (3)$$

и определим последовательно для $k = m-1, m-2$ ..., 1 величины $z_j^{(k)}$ по формуле

$$z_j^{(k)} = f'\left(S_j^{(k)}\right)\sum_{p=1}^{n_{k+1}} z_p^{(k+1)} w_{jp}^{(k+1)}. \qquad (4)$$

Теорема. Для всех слоев нейронной сети k от 1 до m, всех нейронов i слоя $k-1$, всех нейронов j слоя k выполняется равенство

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = z_j^{(k)} x_i^{(k-1)}. \qquad (5)$$

Доказательство: Докажем сначала, что для любого r такого, что $k \le r \le m$, выполняется

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_r} \frac{z_p^{(r)}}{f'\left(S_p^{(r)}\right)} \frac{\partial x_p^{(r)}}{\partial w_{ij}^{(k)}}. \qquad (6)$$

Будем доказывать данное утверждение по индукции. Пусть сначала r = m. Из (2), (3) и (6) следует, что

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \frac{\partial \frac{1}{2}\sum_{p=1}^{n_m}\left(x_p^{(m)} - d_p\right)^2}{\partial w_{ij}^{(k)}} = \qquad (7)$$

$$= \sum_{p=1}^{n_m}\left(x_p^{(m)} - d_p\right)\frac{\partial x_p^{(r)}}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_{mr}} \frac{z_p^{(m)}}{f'\left(S_p^{(m)}\right)} \frac{\partial x_p^{(m)}}{\partial w_{ij}^{(k)}},$$

то есть при r = m соотношение (6) выполнено. Покажем теперь, что если оно выполнено при некотором r, что оно выполняется и для $r' = r-1$, если $r - 1 \ge k$. Действительно, пусть

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_r} \frac{z_p^{(r)}}{f'\left(S_p^{(r)}\right)} \frac{\partial x_p^{(r)}}{\partial w_{ij}^{(k)}}.$$

Тогда

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_r} \frac{z_p^{(r)}}{f'\left(S_p^{(r)}\right)} f'\left(S_p^{(r)}\right) \frac{\partial S_p^{(r)}}{\partial w_{ij}^{(k)}} =$$

$$= \sum_{p=1}^{n_r} z_p^{(r)} \frac{\partial \sum_{q=1}^{n_{r-1}+1} x_q^{(r-1)} w_{qp}^{(r)}}{\partial w_{ij}^{(k)}} = \sum_{q=1}^{n_{r-1}+1} \left(\sum_{p=1}^{n_r} z_p^{(r)} w_{qp}^{(r)} \right) \frac{\partial x_q^{(r-1)}}{\partial w_{ij}^{(k)}}.$$

Отсюда из (4) и (7), получим

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \sum_{q=1}^{n_{r-1}} \frac{z_q^{(r-1)}}{f'\left(S_q^{(r-1)}\right)} \frac{\partial x_q^{(r-1)}}{\partial w_{ij}^{(k)}} = \sum_{q=1}^{n_r} \frac{z_q^{(r)}}{f'\left(S_q^{(r)}\right)} \frac{\partial x_q^{(r)}}{\partial w_{ij}^{(k)}}.$$

Таким образом, соотношение (6) доказано для всех $r \geq k$. Следовательно, оно выполняется и для $r = k$. Значит,

$$\frac{\partial E}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_k} \frac{z_p^{(k)}}{f'\left(S_p^{(k)}\right)} \frac{\partial x_p^{(k)}}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_k} \frac{z_p^{(k)}}{f'\left(S_p^{(k)}\right)} f'\left(S_p^{(k)}\right) \frac{\partial S_p^{(k)}}{\partial w_{ij}^{(k)}} =$$

$$= \sum_{p=1}^{n_k} z_p^{(k)} \frac{\partial \sum_{q=1}^{n_{k-1}+1} x_q^{(k-1)} w_{qp}^{(k)}}{\partial w_{ij}^{(k)}} = \sum_{p=1}^{n_k} z_p^{(k)} \sum_{q=1}^{n_{k-1}+1} x_q^{(k-1)} \frac{\partial w_{qp}^{(k)}}{\partial w_{ij}^{(k)}} =$$

$$= \sum_{p=1}^{n_k} z_p^{(k)} \sum_{q=1}^{n_{k-1}+1} x_q^{(k-1)} \delta_{iq} \delta_{jp} = z_j^{(k)} x_i^{(k-1)},$$

что и требовалось доказать. Нетрудно показать, что $f'(x) = \alpha f(x)(1 - f(x))$, поэтому

$$f'\left(S_j^{(k)}\right) = \alpha f\left(S_j^{(k)}\right)\left(1 - f\left(S_j^{(k)}\right)\right) = \alpha x_j^k \left(1 - x_j^k\right).$$

Следовательно (3), (4) могут быть записаны в более удобном для вычислений виде:

$$z_j^m = \text{б} x_j^m \left(1 - x_j^m\right)\left(x_j^m - d_j\right), \qquad (8)$$

$$z_j^k = \text{б} x_j^k \left(1 - x_j^k\right) \sum_{p=1}^{n_{k+1}} z_p^{(k+1)} w_{jp}^{(k+1)}. \qquad (9)$$

Условием окончания обучения может быть, например, истечение времени, отведенного на обучение, или то, что процент неверно классифицированных объектов обучающей выборки не превысил заданной величины. Топология нейронной сети (количество слоев, количество нейронов в каждом слое) обычно выбирается эмпирически, и строгих указаний для такого выбора не имеется.

Обучение нейронной сети занимает обычно продолжительное время, поэтому она может применяться только в тех областях, где это приемлемо. Другим существенным недостатком нейронных сетей является то, что результаты обучения плохо интерпретируемы, так как для человека трудно интерпретировать символическое значение весов.

Байесовская классификация

Метод Байесовской классификации является статистическим методом. Он позволяют предсказать вероятность принадлежности объекта к заданному классу. Метод Байесовской классификации основан на теореме Байеса, приведенной ниже. Достоинствами метода являются как точность, так и скорость при работе с большими массивами данных.

Пусть X – некоторый объект, класс которого неизвестен. Пусть H – гипотеза, заключающаяся в том, что X принадлежит к классу C. Для проблемы классификации мы хотим определить P (H | X), вероятность выполнения гипотезы H при наблюдаемых данных X.

На языке теории вероятностей P (H|X) – это вероятность a posteriori наступления H при условии X. Например, рассмотрим в качестве множества объектов футбольные мячи и бильярдные шары, описываемых в базе данных цветом и формой. Предположим, что X – коричневого цвета и круглой формы, а H – гипотеза, что X – это футбольный мяч. Тогда P (H|X) – степень достоверности того, что X – это футбольный мяч при том, что мы видим, что X – коричневого цвета и круглое.

В то же время P (H) – это вероятность a priori наступления H. Для нашего примера P (H) – это вероятность, что произвольно взятый объект из нашей базы данных будет являться футбольным мячом. Вероятность a posteriori P (H | X) базируется на большем количестве информации, чем вероятность a priori P (H), которая не зависит от X.

Аналогично, P (X | H) – это вероятность a posteriori наступления X при условии H. То есть это вероятность, что X – круглой формы и коричневого цвета притом, что мы знаем, что X – футбольный мяч.

Теорема Байеса гласит, что

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}. \qquad (10)$$

Рассмотрим так называемый наивный метод Байесовской классификации, как очень простой и эффективный при больших размерах базы данных. В нем предполагается, что все атрибуты независимы друг от друга.

Пусть любой объект задан с помощью n атрибутов, то есть объект X может быть представлен в виде вектора $X = (x_1, ..., x_n)$. Предполагаем для простоты, что все атрибуты категориальные, то есть могут принимать лишь конечное число значений. Пусть m – это количество классов.

Мы должны для произвольного заданного объекта X с неизвестной меткой класса, определить

вероятности его вхождения в классы 1,..., m. Класс, которому соответствует наибольшая вероятность, и будет оценкой по методу Байесовской классификации.

Ясно, что искомая вероятность вхождения X в класс с номером i равна P (H_i | X), где H_i – это гипотеза, что объект X относится к классу i. По теореме Байеса (10)

$$P(H_i|X) = \frac{P(X|H_i)P(H_i)}{P(X)}.$$

Вычисление P (X | H_i) в общем случае – очень сложная задача. Но если считать, что все атрибуты независимы, то данная задача упрощается, так как в этом случае

$$P(X) = \prod_{k=1}^{n} P(x_k),$$

$$P(X|H_i) = \prod_{k=1}^{n} P(x_k|H_i),$$

где P (x_k) – вероятность a priori того, что значение атрибута с номером k равно x_k,а P (x_k | H_i) – вероятность a posteriori того, что для объекта, принадлежащего классу i, значение атрибута с номером k равно x_k.

Величины P (x_k), P (x_k | H_i) могут быть вычислены на основе обучающей выборки следующим образом:

$$P(x_k|H_i) = \frac{s_{ik}(x_k)}{s_i}, \quad P(x_k) = \frac{\sum_{i=1}^{m} s_{ik}(x_k)}{\sum_{i=1}^{m} s_i},$$

где $s_{ik}(x_k)$ – количество записей в обучающей выборке, принадлежащих классу i, таких, что значение атрибута с номером k равно x_k; s_i – количество всех записей, принадлежащих классу i.

Выводы и заключение

В статье предложены два варианта модификация работы алгоритма ID3 [4] с деревьями решений: упрощение дерева на этапе его создания и удаление ветвей из уже «выращенного» дерева.

В обоих вариантах рассчитывается средний процент ошибок классификации и выбирается вариант с наименьшим процентом ошибок.

К преимуществам использования нейронных сетей относится то, что они универсальны для разных видов данных, и дают хорошие результаты даже при наличии «зашумленности» в выборке. Данные факторы говорят в пользу использования нейронных сетей в задачах классификации.

Теоретически, метод Байесовской классификации имеет минимальную степень ошибок по сравнению с другими классификаторами. Однако на практике это не всегда верно, так как условие независимости атрибутов – слишком сильное условие. Кроме того, часто необходимых статистических данных не хватает для выполнения классификации. Тем не менее, различные эмпирические исследования и сравнения данного метода с деревьями решений и с нейронными сетями показывают, что в ряде областей метод Байесовской классификации вполне применим.

Список литературы

1. Асеев Г.Г. Методы интеллектуальной предобработки данных в электронных хранилищах / Г.Г. Асеев // Радіоелектроніка. Інформатика. Управління. – 2010. - №2(23). – С. 106-111.

2. Асеев Г.Г. Методы интеллектуального анализа данных в электронных хранилищах: генетические алгоритмы / Г.Г. Асеев // Радіоелектроніка. Інформатика. Управління. – 2011. - №2(25). – С. 82-86.

3. Асеев Г.Г. Нейросетевой анализ непрерывных потоков данных из электронных хранилищ / Г.Г. Асеев // Системи обробки інформації. – Х.: ХУ ПС, 2013. – Вип. 4(111). – С. 52-56.

4. Степанов Р.Г. Технология Data Mining: Интеллектуальный Анализ Данных / Р.Г.Степанов. – Казань: КГУ, 2008. – 64 с.

Поступила в редколлегию 25.08.2014

Рецензент: д-р техн. наук, проф. И.В. Шостак, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

МЕТОДИ ІНТЕЛЕКТУАЛЬНОЇ ОБРОБКИ ПОТОКІВ ІНТЕРНЕТ- ДОКУМЕНТІВ

Г.Г. Асєєв

Розглянуто особливості основних напрямів методів web mining. Головними з них є класифікація з навчанням: дерева рішень, нейронна мережі й метод Naive Bayes. Наведено рекомендації до їхнього використання.

Ключові слова: *гіпертекстові документи, методи web mining, класифікація з навчанням, дерева рішень, нейронна мережа, байесовська класифікація.*

METHODS FOR INTELLIGENT PROCESSING THREADS INTERNET DOCUMENTS

G.G. Aseyev

The features of the basic methods of web mining. The main of them is the classification learning: decision trees, neural network and method Naive Bayes. Provides recommendations for their use.

Keywords: *hypertext documents, methods, web mining, classification learning, Dere vias solutions, neural network, Bayesian classification.*