ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И
ПРОГРАММИРОВАНИЕ

УДК 681.324

А. В. Аксенов – магистрант кафедры вычислительных систем и сетей
А. Ю. Козловский – научный руководитель

ПРИНЦИПЫ ПОСТРОЕНИЯ САМООРГАНИЗУЮЩЕГОСЯ ВЕБ-СООБЩЕСТВА

В настоящее время весь Интернет, и блогосфера в частности, страдает от неконтролируемого потока некачественной информации, что снижает ее ценность как коммуникативного, образовательного и делового ресурса. Так, в Интернете процветает спам, бескультурие, конфликты. Существующие подходы (модерация, автоматический анализ текстов, ранжирование пользовательским сообществом) либо малоэффективны, либо чересчур неудобны, либо однобоки и не учитывают всех аспектов задачи.

Проблема не нова, поэтому для ее решения уже выдвинут ряд способов, отличающихся большей или меньшей эффективностью.

Можно отбирать контент, анализируя его содержание или характеристики отправителя. Такой принцип лежит в основе работы различных спам-фильтров, встроенных в клиенты электронной почты, а также используемых на почтовых серверах. Они эффективны далеко не всегда. Автоматическая классификация текстов лежит на стыке двух областей: информационного поиска и машинного обучения. Существующие концепции классификаторов (регрессионный классификатор, ДНФ-классификатор, классификатор на нейронных сетях), в особенности на начальном периоде обучения, подвержены ошибкам, которые сильно снижают ценность ресурса.

Иной способ предполагает использование модераторов – виртуально (или реально) трудоустроенных пользователей, в чьи обязанности входит контроль контента на предмет соответствия его некоторым оговоренным для конкретного сайта правилам, общепринятым нормам морали и этики, а также, что является минусом данного подхода, личным взглядам самого модератора. Модерация делится на два вида: постмодерация, когда модераторы оценивают уже опубликованный контент и при необходимости удаляют его, и премодерация, когда сообщения публикуются только после подтверждения возможности этого модератором.

Оба подхода являются плохо масштабируемыми и зачастую (при больших объемах информации) нецелесообразными, когда модераторы просто не в состоянии справиться с лавинообразно возрастающим потоком информации [2].

Это заставляет разработчиков искать новые пути решения поставленной проблемы. Рассмотрим наиболее оригинальное из них. В интернете существует множество так называемых «управляемых пользовательским сообществом» («user-driven») веб-сайтов [3]. Их основной принцип заключается в том, что контент отбирается самими пользователями, а не специально выделенными модераторами. Наиболее известными являются digg.com, slashdot.org, reddit.com. К этому разряду можно также отнести wikipedia.org. Российский эквивалент – news2.ru. Там воплощен принцип «коллективного определения ценности информации» путем ведения рейтингов пользователей и их сообщений. Таким образом можно достичь нескольких целей: вычислять спам-ботов, «троллей» [3] и других нежелательных блоггеров, а также ранжировать информацию по степени заинтересованности в ней сообщества. Последнее открывает перед сайтами, применяющими такой подход, широкие возможности. Самая ценная информация демонстрируется чаще, а не вызвавшая интереса и одобрения отходит на второй план. Это повышает ценность ресурса в глазах посетителей.

Но такой подход не лишен серьезных проблем. Большинство подобных сайтов страдают от того, что происходит выделение «элиты», т. е. пользователей с чрезвычайно высоким рейтингом (как правило, их число составляет менее 1 % общего количества пользователей), которые влияют на жизнь сайта, при этом сводя результаты деятельности большинства к нулю [4, 5].

Эта проблема обусловлена тем, что каждый пользователь может напрямую влиять на рейтинг другого прямым голосованием, что позволяет пользователям объединяться в группы для достижения своих целей, как правило отрицательных для сайта и других пользователей.

Для решения этой проблемы в данной работе предпринимается попытка определения таких принципов организации веб-сообщества, которые выполняли бы функцию повышения «общего уровня качества» информации, т.е. своего рода упорядочивания ее по степени ценности, и отбрасывания информации, не удовлетворяющей определенным показателям критерия качества. При этом система должна выполнять большинство функций (отбор «хорошей» информации, препятствование распространению «плохой») автоматически, не требуя услуг модераторов, чье вмешательство в коммуникационный процесс не всегда возможно, а если возможно, то не всеми приветствуется. Кроме того, алгоритм должен быть достаточно устойчивым, чтобы исключить успешность попыток воздействия на него со стороны недоброжелателей. Он должен включать в себя средства преодоления тех трудностей, с которыми столкнулись первопроходцы идеи ранжирования блоговой информации (сайт digg.com и другие). В частности, роковой ошибкой для них стало введение возможности пользователям напрямую влиять на рейтинги других пользователей.

Вкратце о принципе организации блога. Автор блога публикует на своей странице сообщения («посты»), посвященные какой-либо проблеме, конкретизирует проблему в заголовке и метках («тэгах»). Метки служат для тематической группировки сообщений, позволяющей найти все сообщения, посвященные той или иной теме. Сообщения практически не имеют ограничений по объему и могут состоять из одной строки, ссылки или нескольких ссылок на другие веб-страницы или же иметь форму развернутой статьи.

Важную часть любого блога составляют дискуссии, которые поднимает та или иная запись блога. Другие пользователи системы, ведущие свои блоги, могут беспрепятственно просматривать сообщения и оставлять на них комментарии. Комментарии также имеют вид сообщений, но, как правило, менее развернуты и носят характер согласия или несогласия с автором сообщения, иногда развивая тему, затронутую в нем. Обилие их говорит об интересности сообщения, вызвавшего такую бурную дискуссию.

На комментарии также можно оставлять комментарии, и так далее. Вложенность комментариев не ограничена. Таким образом, мы можем представить себе дискуссию в виде не линейной, что практикуется на многих форумах, а древовидной структуры. В чем ее преимущества? Первое, и самое очевидное, – структурированность беседы. Обсуждения в блогосфере, в противовес линейным дискуссиям, имеют четкую структуру, что подразумевает наличие у каждого сообщения (кроме корневых сообщений) «родителя», т.е. сообщения, ответом на которое оно является, логически продолжая дискуссию, а также набора «потомков», т.е. сообщений, являющихся ответами на данное сообщение, иначе говоря, комментариями к нему. Таким образом, можно легко найти ту ветку дискуссии, которая представляет наибольший интерес, и оставить сообщение именно в ней, избегая «каши», характерной для форумов, применяющих линейные темы («топики») и систему цитат.

Вторым преимуществом является легкость нахождения таких «узловых» сообщений, из которых «произрастает» наибольшее дерево беседы, иными словами – сообщений, вызвавших наибольшую дискуссию. Можно сделать предположение, что среди всех узлов дерева беседы именно такие несут наибольшую ценность. На этой идее основывается самоорганизуемость предлагаемой системы.

В основе данной работы лежит отказ от прямой манипуляции рейтингом в пользу расчета рангов пользователей и информации самим сайтом на основе анализа их авторской деятельности, активности в обсуждениях и ссылок на сообщения других авторов.

Для ранжирования пользователей и их сообщений выбран аналог алгоритма PageRank, разработанного основателями компании Google Л. Пейджем и С. Брином. Его суть состоит в анализе направленного графа, вершинами которого являются веб-страницы, а ребрами – гиперссылки между ними с

получением в качестве выходных данных весов страниц, влияющих на порядок вывода результатов запроса к поисковой системе компании.

Выбор PageRank в качестве прототипа алгоритма вычисления рангов объясняется похожестью моделей совокупности веб-страниц, соединенных гиперссылками и сообщества пользователей и их сообщений, соединенных связями «автором – его сообщение», «сообщением - ответ на сообщение», «удаленное сообщения - пользователь, его удаливший».

В более строгом математическом представлении модель путешествия пользователя представляет собой направленный граф, вершинами которого являются страницы, а ребрами – гиперссылки, ведущие с одной страницы на другую. Будем моделировать передвижение пользователя по сети следующим образом: пользователь стартует в случайной вершине. С вероятностью *d*=0,85 пользователь переходит по одному из случайных исходящих ребер, а с вероятностью (1-*d*)=0,15 он переходит в случайную вершину графа.

Представим себе, что этот пользователь бродит так бесконечно долго. Для каждого *k* можно определить PageRank $PR_k(p_i)$ как вероятность оказаться в вершине p_i через *k* шагов. Для каждой вершины есть вероятность того, что пользователь окажется в ней через бесконечно большое число шагов. Тогда предельная вероятность оказаться в вершине p_i и есть PageRank:

$$PR(p_i) = \lim_{k\to\infty} PR_k(p_i)$$

Итак, будем рассматривать PageRank как вероятность попадания пользователя в вершину в произвольный момент времени.

PageRank $PR(p_i)$ вершины p_i выражается как

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)} \quad (1)$$

где *d* – «демпинг-фактор», параметр затухания, который принимается равным 0.85 и выражает вероятность того, что пользователь, придя в вершину, совершит переход по одному из исходящих ребер; $PR(p_i)$ – PageRank вершины p_i; $M(p_i)$ – совокупность вершин, от которых ведут ребра, входящие в вершину p_i; $PR(p_j)$ – PageRank вершины p_j, соединенной исходящим ребром с p_i; $L(p_j)$ – число исходящих ребер в вершине p_j; $\frac{1}{L(p_j)}$ – вероятность того, что пользователь, находящийся в вершине p_j, из $L(p_j)$ доступных ему ребер совершит переход именно по ребру, ведущему к вершине p_i; $d\frac{PR(p_j)}{L(p_j)}$ - поток «теоретической посещаемости», который дойдет к вершине p_i от вершины p_j (суммирование идет по всем вершинам, соединенным исходящими ребрами с p_j); $\frac{1-d}{N}$ – минимальный PageRank вершины (не равен нулю за счет того, что пользователь регулярно совершает перескоки в случайную вершину графа).

Однако на PageRank наложено ограничение:

$$\sum_{i=1}^{N} PR(p_i) = 1$$

где *N* – общее количество вершин в графе.

Из этого напрямую следует, что средний PageRank равен $\frac{1}{N}$.

В формуле (1), если рассматривать ее как способ определения вероятности нахождения пользователя в вершине, первое слагаемое характеризует вероятность того, что пользователь непосредственно перед этим совершил прыжок в случайную вершину графа. Вероятность этого прыжка равна, как было указано выше, (1-*d*), тогда как вероятность случайного попадания в конкретную вершину (все вершины при прыжке равноправны) составляет $\frac{1}{N}$. Произведение этих вероятностей дает первое сла-

гаемое. По аналогии, второе слагаемое – произведение вероятности перехода по случайному исходящему ребру графа d на сумму вероятностей перехода по входящему ребру из одной из ссылающихся вершин [6].

Решая систему уравнений, можно найти PageRank всех вершин графа. Расчет можно вести разными методами.

Наиболее часто используется итерационный метод расчета PageRank. Он состоит в численном решении системы уравнений.

1. Выбираем структуру графа, расстановку ребер, систему уравнений.
2. Задаемся начальными значениями PageRank для каждой вершины. Они могут быть любыми.
3. Рассчитываем новый набор значений PageRank по следующему уравнению, полученному из (1):

$$PR_k(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR_{k-1}(p_j)}{L(p_j)}$$

4. Находим сумму PageRank по всему набору вершин, и делим PageRank каждой вершины на полученную величину. В результате средний PageRank становится равным $\frac{1}{N}$.

5. Если набор значений PageRank изменился по сравнению с исходным набором шага 3 на величину, большую заранее заданного критерия, возвращаемся к шагу 3. Если нет, то расчет заканчиваем.

Для теоретического обоснования рассмотрен матричный метод расчета PageRank. Для его использования составляется матрица всех ссылок $S = \{s_{ij}\}$, причем если нет ребра из p_i в p_j, то $s_{ij} = \frac{1-d}{N}$, а если ребро есть, то $s_{ij} = \frac{1-d}{N} + d\left(\frac{1}{L(p_i)}\right)$. Также задаются некоторые начальные значения элементам вектора рангов:

$$PR_0 = \begin{pmatrix} PR_0(p_1) \\ PR_0(p_2) \\ \dots \\ PR_0(p_N) \end{pmatrix}.$$

Далее итерационно вычисляются следующие значения вектора рангов по формуле

$$PR_{k+1} = S \cdot PR_k,$$

причем при $k \to \infty$ формула принимает вид

$$PR = S \cdot PR,$$

что соответствует утверждению, что предельный вектор рангов является собственным вектором матрицы всех ссылок, которому соответствует собственное значение, равное 1. Существование такого собственного значения доказывается теоремой Перрона-Фробениуса.

Нетрудно показать, что матрица всех ссылок в том случае, если в графе нет вершин без исходящих ребер, является положительной стохастической, а у таких матриц, согласно теореме Перрона-Фробениуса, существует вещественное собственное значение $\lambda = 1$, а все остальные собственные значения удовлетворяют неравенству $|\lambda| < 1$. Также доказывается существование собственного вектора ν, который можно нормировать таким образом, что все его элементы будут лежать в диапазоне [0, 1] и сумма их будет равна 1. Этот вектор и соответствует предельному вектору рангов, а также является равновесным распределением для цепи Маркова, задаваемой соответствующей матрицей переходов, т.е. существует матрица ранга 1

$$S_\infty = \lim_{k \to \infty} S^k,$$

столбцы которой представляют собой вектор ν. Таким образом, это можно считать третьим способом вычисления вектора рангов для матриц небольших порядков.

Библиографический список

1. Википедия – Блогосфера (http://ru.wikipedia.org/wiki/Блогосфера)
2. Энциклопедия сайтостроения - Борьба со спамом на сайтах (http://site.nic.ru/content/view/212/193/)

3. Trolls (http://www.paulgraham.com/trolls.html)
4. Digg the rigged? A closer look at Digg's democratic model (http://jesusphreak.infogami.com/blog/is_digg_rigged)
5. Digg, Wikipedia, and the myth of Web 2.0 democracy. - By Chris Wilson - Slate Magazine (http://www.slate.com/id/2184487/)
6. Ian Rogers – Google Page Rank – Whitepaper (http://www.ianrogers.net/google-page-rank/)