В. Н. Тарасов, доктор технических наук, профессор, заведующий кафедрой «Программное обеспечение и управление в технических системах» ФГОБУ ВПО ПГУТИ
e-mail: vt@ist.psati.ru

Е. М. Мезенцева, аспирант, ассистент кафедры «Программное обеспечение и управление в технических системах» ФГОБУ ВПО ПГУТИ
e-mail: katya-mem@psuti.ru

МНОГОУРОВНЕВАЯ АРХИТЕКТУРА КЛИЕНТ-СЕРВЕРНОГО ПРИЛОЖЕНИЯ ФИЛЬТРАЦИИ СПАМА НА САЙТАХ

В работе описана архитектура клиент-серверного приложения, предназначенного для фильтрации спама. Алгоритм идентификации спама, основанный на применении классификации сообщения по методу Фишера. Тестирование модуля распределения хи-квадрат. Результаты тестирования обученного фильтра.

Ключевые слова: *спам, фильтр, метод Фишера, хи-квадрат распределение, клиент-сервер, сервер Apache, php, сервер MySQL, схема работы.*

Спам признан преступлением во многих странах, но, несмотря на это его объемы постоянно растут, технологии рассылки спама продолжают совершенствоваться, при этом в общем потоке возрастает доля ручного спама.

Поэтому разработка специализированных методов и алгоритмов, предназначенных для фильтрации спама в интерактивных разделах Интернета, актуальна и представляет научный и практический интерес в области защиты компьютерных сетей.

Для преодоления данной проблемы разработано множество фильтров спам сообщений. Но практически не существует универсальных программных комплексов, предназначенных для фильтрации спама на сайтах в комментариях, формах обратной связи и других интерактивных разделах. Основными недостатками существующих методов является высокая доля ложных срабатываний и ориентация «на запад». Нет решений, позволяющих производить фильтрацию на русскоязычных сайтах.

Поэтому целью данной статьи является разработка концепции построения и математической модели многомодульной, клиент – серверной системы фильтрации спама на сайтах.

Постановка задачи

Разрабатываемый анти-спам фильтр должен обеспечивать высокую скорость и стабильность работы, должны быть минимизированы ложные срабатывания, система фильтрации должна обеспечивать удобный интерфейс взаимодействия с любым интернет ресурсом.

Основными преимуществами системы должны стать гибкость изменения алгоритмов ее работы за счет многомодульности и простоты надстройки новых средств обработки нежелательных сообщений, для незамедлительного отражения новых видов спам атак сразу на всех подключенных интернет ресурсах, без каких-либо дополнительных действий со стороны администратора сайта.

Решение задачи. Трехуровневая архитектура клиент-серверного приложения, предназначенного для фильтрации спама

В результате анализа поставленной задачи получили спецификацию разрабатываемого фильтра, на её основе построим общую модель взаимодействия с пользователем и конкретизируем его основные функции.

Описанная в данной статье система реализована в виде технологии «клиент-сервер» – сетевой архитектуры, в которой задания и сетевая нагрузка распределены между поставщиками услуг (сервисов), называемыми серверами, и заказчиками услуг, называемыми клиентами.

Преимущества данной реализации в следующем:

- Делает возможным, распределение функций системы фильтрации спама между несколькими независимыми компьютерами в сети. Это позволяет упростить её обслуживание (замена, ремонт, модернизация или перемещение сервера не затрагивают клиентов).

- Все данные хранятся на сервере, который защищён гораздо лучше большинства клиентов. На сервере проще обеспечить контроль полномочий, чтобы разрешать доступ к данным только клиентам с соответствующими правами доступа.

- Низкие требования к производительности и техническим характеристикам клиентов (терминалов), как следствие, снижение их стоимости.

Терминалом может выступать не только компьютер, но и, например, мобильный телефон.

- Позволяет объединять различных клиентов, т.е. несколько клиентов используют ресурсы одного сервера фильтрации спама.

Ниже представлено описание взаимодействия компонентов трехуровневой архитектуры клиент-серверного приложения, предназначенного для фильтрации спама. Сервер БД представлен MySQL-сервером; сервер приложений следующими технологиями: интерпретатором PHP и web-сервером Apache. Роль клиента выполняет web-сервер клиента.

Web-сервер клиента (1)-> Сервер Apache (2) -> интерпретатор PHP (3) -> Провайдер данных MySQL (4) -> Сервер MySQL (5) -> Провайдер данных MySQL (6) -> интерпретатор PHP (7) -> Сервер Apache (8).

1 – web-сервер клиента отправляет HTTP-запрос;

2 – web-сервер Apache определяет тип запрашиваемого ресурса, и для случая запроса *.php (расширение файлов страниц PHP) загружает интерпритатор PHP. Сервер Apache также может осуществлять идентификацию и авторизацию пользователя, от которого поступил запрос. Далее, фабрика обработчиков создает объектную модель запрашиваемой страницы и обрабатывает действия пользователя.

3 – в ходе генерации ответа интерпретатору PHP может потребоваться обращение к БД, в этом случае используются библиотеки классов провайдера данных MySQL и выполняющая среда обращается к серверу БД;

4 – провайдер данных MySQL передает запрос на операцию с БД серверу MySQL;

5 – сервер MySQL осуществляет обработку запроса, выполняя соответствующие операции с БД;

6 – провайдер данных MySQL передает результаты запроса объекту страницы;

7 – объект страницы с учетом полученных данных осуществляет рендеринг графического интерфейса страницы и направляет результаты в выходной поток;

8 – Сервер Apache отправляет содержимое сгенерированной страницы web-серверу клиента.

Фильтрация спама на сайтах-участниках системы работает по следующей общей схеме, приведенной на рис. 1:

Пользователи, подключенные к сервису фильтрации спама, отправляют на сайт текстовые сообщения. Поступившие сообщения сохраняются на сайте, и отправляются запросы с текстом сообщений фильтру. Система фильтрации производит необходимые преобразования, вычисления и принимает решение, являются ли сообщения спамом. В зависимости от внутренней политики, программное обеспечение сайта либо скрывает сообщения пользователя, либо помечает их как спам, так же может приниматься решение об их удалении. Каждое новое сообщение сохраняется на сервере фильтра с присвоенным ему уникальным идентификатором.

В случае если произошло ложное срабатывание фильтра, администратор сайта корректирует результат его работы. Результат корректировки отправляется на сервер спам-фильтра. Сервис фильтрации спама производит обучение фильтра [1].

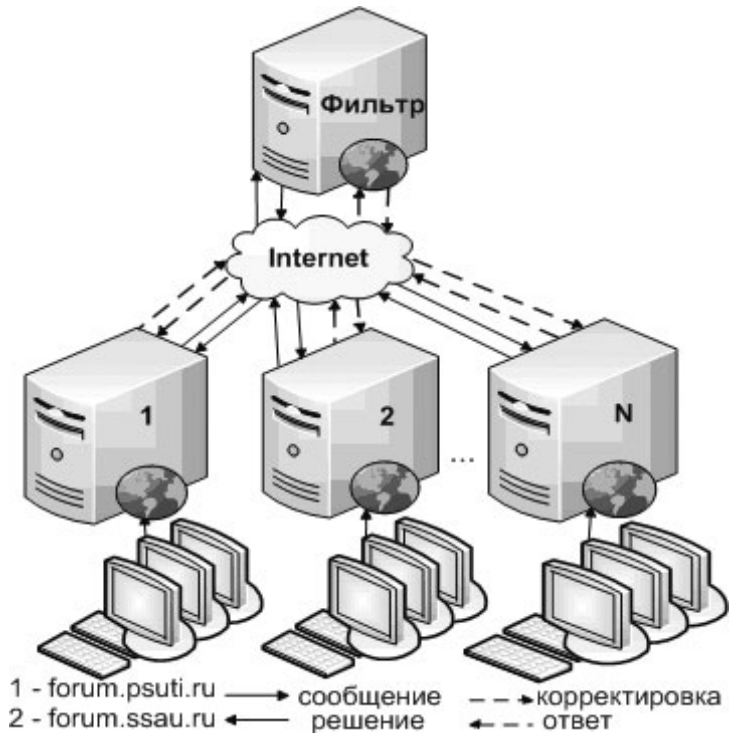


Рис. 1. Общая схема работы системы фильтрации спама на сайтах

Алгоритм идентификации спама, основанный на применении классификации сообщения по методу Фишера (R. A. Fisher)

Для того чтобы идентифицировать поступившее на сайт сообщение, будем исходить из словаря, полученного на этапе обучения фильтра. Для любого сообщения введем 2 гипотезы:

H_A – спам;
H_B – не спам.

Отнесение сообщения, пришедшего на сайт, к спаму/не спаму будет осуществляться по приведенному ниже алгоритму.

На первом этапе нам необходимо вычислить вероятность того, что отдельное слово сообщения принадлежит некоторой категории. Для этого можно разделить найденное количество сообщений с признаком i в данной категории, на общее число сообщений в той же категории, но мы воспользуемся другим методом, преимущества которого определились на этапе эксперимента, описанном ниже.

Пусть:

F_{ai} – количество сообщений с признаком i в группе спама;

F_{bi} – количество сообщений с признаком i в группе не спама.

Тогда статистическая вероятность появления признака i в спам сообщении

$$p_{ai} = \frac{F_{ai}}{F_{ai} + F_{bi}}, \quad (1)$$

а вероятность появления признака i в не спам сообщении

$$p_{bi} = \frac{F_{bi}}{F_{ai} + F_{bi}}. \quad (2)$$

Таким образом, количество сообщений с признаком i в одной из категорий делим на общее количество сообщений с данным признаком i.

При применении формул (1) и (2) учитывается тот факт, что со временем может накопиться одинаковое количество сообщений в обеих категориях, т.е. эти формулы не зависят от того, сколько сообщений накоплено в той или иной категории.

Заметим, что приведенные выше формулы дают точный результат только для тех признаков, которые фильтр уже встречал в обеих категориях. Это делает спам-фильтр слишком чувствительным на ранних этапах обучения в отношении к редко встречающимся словам. Чтобы справиться с данной проблемой, вычислим новую вероятность, начиная с предполагаемой априорной вероятности – ($P_{пр}$) и веса – (w), приданного этой вероятности, а затем добавим рассчитанные по формулам (1) и (2) вероятности.

Если вероятность $P_{пр}=0,5$ и $w = 1$ – вес предполагаемой вероятности равен одному слову, то определяем средневзвешенные вероятности, используя выражения (1), (2):

$$\overline{p_{ai}} = \frac{(w * P_{пр}) + p_{ai} * (F_{ai} + F_{bi})}{(w + F_{ai} + F_{bi})},$$

$$\overline{p_{bi}} = \frac{(w * P_{пр}) + p_{bi} * (F_{ai} + F_{bi})}{(w + F_{ai} + F_{bi})}.$$

Такой подход позволяет избегать деления на ноль в формуле Байеса, а также учитывать редко встречающиеся слова.

Для получения объединенных вероятностей всего документа (сообщения), будем исходить из словаря, полученного на этапе обучения спам фильтра. Введем следующие события: A – сообщение относится к спаму; B – не спам-сообщение. Мы предположили, что вероятности независимы, поэтому возможно их перемножение:

$$P(A) = \overline{p_{a1}} \times \overline{p_{a2}} \times \ldots \times \overline{p_{an}}, \quad (3)$$

для вероятности совместного появления слов в спаме;

$$P(B) = \overline{p_{b1}} \times \overline{p_{b2}} \times \ldots \times \overline{p_{bn}} \quad (4)$$

для вероятности совместного появления слов в не спаме, где n – количество слов в документе [2].

По методу Фишера, от произведения вероятностей берется натуральный логарифм, и результат умножается на – 2. Для этого введем переменную *hisqv*, которая будет определена выражениями, приведенными ниже:

$hisqv = -2 * \ln(P(A))$ или $hisqv = -2 * \ln(P(B))$,

где $P(A)$ и $P(B)$ вычисляются по формулам (3) и (4).

Фишером доказано, что если есть набор независимых и случайных вероятностей (3) или (4), то величина $-2 * \ln(P(A))$ подчинялась бы распределению x^2 с $2n$ степенями свободы (n-количество слов в сообщении):

$$F(x) = \int_0^x \frac{t^{n-1} e^{-t/2}}{2^n \Gamma(n)} dt. \quad (5)$$

где $\Gamma(n)$ – гамма-функция.

С учетом сказанного выше и представления гамма-функции от четного аргумента, перепишем интеграл (5) в виде (6):

$$F(hisqv) = \frac{1}{2^n (n-1)!} \int_0^{hisqv} x^{n-1} e^{-x/2} dx. \quad (6)$$

В программе вычисление вероятности производится по выражению (6), т.е. обратной функции распределения хи-квадрат, выполняется с помощью квадратурной формулы Гаусса с 15 узлами:

$$\int_a^b f(t)dt \approx \frac{b-a}{2}\sum_{i=1}^{n} A_i f(t_i),$$

где $t_i = (b+a)/2+(b-a)x_i/2$, а x_i – узлы квадратурной формулы Гаусса;

A_i – гауссовы коэффициенты, (i = 1,2,…,15) [3]. Для нашего случая a = 0, b = *hisqv*.

Число, возвращаемое функцией F(*hisqv*), будет малым в случае, если в тексте много признаков спама. Для верной классификации сообщения, нам необходим обратный результат. Для этого от единицы вычтем значение F(*hisqv*). Соответственно, вычитая из 1 значение функции F(*hisqv*) для большого количества не спам признаков, и получим вероятность того, что сообщение является не спамом.

Однако метод Фишера не является симметричным. Значит, нам необходимо скомбинировать вероятности спама и не спама, путем объединения вероятностей в одно число, которое даст нам значение спам/не спам от 0 до 1. Для этого воспользуемся т.н. индикатором Фишера:

$$I = \frac{1+P(H'_A)-P(H'_B)}{2},$$

где: $P(H'_A) = 1 - F(-2\ln(P(A))$ – вероятность принадлежности документа к спаму;

$P(H'_B) = 1 - F(-2\ln(P(B))$ – вероятность принадлежности документа не к спаму.

Для окончательного принятия решения, в методе классификации Фишера необходимо задать начальные значения нижнего и верхнего порогов.

Пусть T и L величины, измеряющиеся в долях и определяющие соответственно верхний и нижний пороги принятия решений.

Тогда:

- будем считать, что документ принадлежит группе спама, если $I \geq T$;
- документ принадлежит группе не спама, если $I \leq L$;
- если же выполняется $T \geq I \leq L$, то нельзя принять никакого решения.

Тестирование модуля распределения хи-квадрат

При тестировании работы функции хи-квадрат на вход подаются значения x^2 и число степеней свободы n, на выходе получаем вероятности. Сравнение выходных данных с табличными значениями вероятностей, приведенными в таблице 1, подтверждает их полную идентичность. При этом табличные значения вероятностей в свою очередь рассчитаны с помощью функции chi2inv пакета MATLAB.

Экспериментальное исследование степени корректности фильтрации сообщений методом Фишера

Исследование эффективности использования выбранного алгоритма фильтрации проводилось на сообщениях форума Медиацентра Самарского Государственного Аэрокосмического Университета имени академика С. П. Королева (http://forum.ssau.ru/). Для этого к фильтру был подключен дамп базы данных, который является файлом с её содержимым, позволяющим воссоздать базу данных «с нуля». Дамп состоит из 58659 сообщений, пригодных для анализа.

Рассмотренный в данной статье метод фильтрации сообщений способен обучаться. Для этого классификатору необходимо предоставить при-

Таблица 1

Тестирование вычисления распределения хи-квадрат

Данные на входе функции распределения хи-квадрат hiSguare(x^2, n)	Выходные данные	Табличные значения
(3,9403, 10)	0,05	0,05
(9,3418, 10)	0,50	0,50
(18,3070, 10)	0,95	0,95
(10,8508, 20)	0,05	0,05
(19,3374, 20)	0,50	0,50
(31,4104, 20)	0,95	0,95
(18,4927, 30)	0,05	0,05
(29,3360, 30)	0,50	0,50
(43,7730, 30)	0,95	0,95
(26,5093, 40)	0,05	0,05
(39,3353, 40)	0,50	0,50
(55,7585, 40)	0,95	0,95
(34,7643,50)	0,05	0,05
(49,3349, 50)	0,50	0,50
(67,5048, 50)	0,95	0,95

меры правильных ответов. Чем больше сообщений и верных способов классификации будет предоставлено приведенным алгоритмам, тем точнее получится итоговая классификация сообщений. Обучим фильтр на выборке, состоящей из 300 спам и 600 не спам сообщений форума.

Тестирование работы фильтра при различных порогах принятия решения показало, что самыми оптимальными являются:

- верхняя граница $T = 0,95$;
- нижняя граница $L = 0,4$.

Мы установили жесткие рамки по спаму и обычные для не спама. Это было сделано для того, чтобы как можно меньше нормальных сообщений проходило в спам, т.е. во избежание ложных срабатываний.

После определения порогов принятия решения, переходим к тестированию работы выбранных методов фильтрации спама. Для этого программно создадим 3 файла, сюда произведем запись проанализированных сообщений форума, по категориям – спам, не определено, не спам. Ниже приведен формат представления сообщений в файлах:

- 0.1 / 0.9;
- http://forum.ssau.ru/index.php?s=&showtopic=2596&view=findpost&p=65475;
- «Собрание по АРАБИКЕ для всех участников 31 МАЯ 19:00 СГАУ 205-16».

Здесь:

- 0.1 / 0.9 – значение вероятности (спам / нужные);
- http://forum.ssau.ru/… – адрес сообщения на форуме СГАУ;
- «Собрание по АРАБИКЕ …» – текст сообщения на форуме.

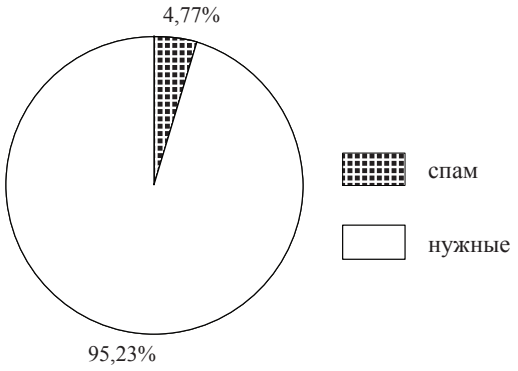


Рис. 2. Процентное соотношение количества сообщений спама и нужных сообщений от общего количества сообщений

Результат анализа входящего потока 58659 сообщений выглядит следующим образом:

- Спам – 1304 сообщения;
- Не спам – 40982 сообщения;
- Не определено – 16373 сообщения.

Процент ложных срабатываний (ошибочная классификация нужных сообщений как спам) и доля пропуска спама – являются основными критериями при оценке работы алгоритмов фильтрации спама. Полностью избавиться от них невозможно, но необходимо максимально уменьшить их число. Этого мы и добились, применяя алгоритм фильтрации спама по методу Фишера. Как видно из таблицы 1, ложные срабатывания составили 138 и пропуск спама 225 сообщений из 58659 сообщений.

После наглядного анализа результатов на этапе фильтрации входящего потока сообщений, от общего количества всех сообщений, можно сделать выводы, что система фильтрации получила на выходе следующие результаты.

Для сообщений спама:

- не смогла распознать 50,3% сообщений;
- количество пропущенных сообщений спама оказалось равным 8%.

Для нужных сообщений:

- не смогла распознать 26,79%;
- Количество ложных срабатываний оказалось равным 0,25%

Обобщенный наглядный анализ от общего потока сообщений выглядит следующим образом:

- Количество найденных нужных сообщений составило 69,48%;
- Не смогла распознать 21,51% нужных сообщений;
- Количество ложных срабатываний составило 0,2%;

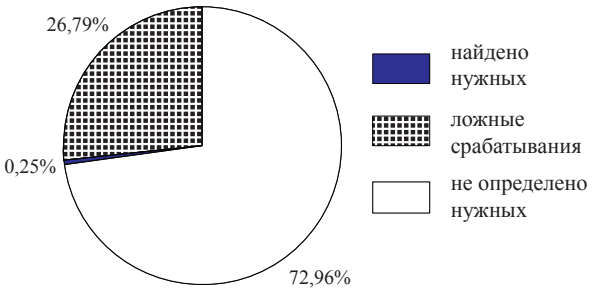


Рис. 3. Процентное соотношение количества нужных сообщений от общего количества нужных сообщений

Таблица 2

Результаты фильтрации входящего потока сообщений

Общее количество сообщений		Найденное кол-во писем спама	Найденное кол-во писем не спама	Не определено		Ложные срабатывания	Пропуск спама
Спам	Не спам			спам	Не спам		
2799	55860	1166	40757	1408	14965	138	225

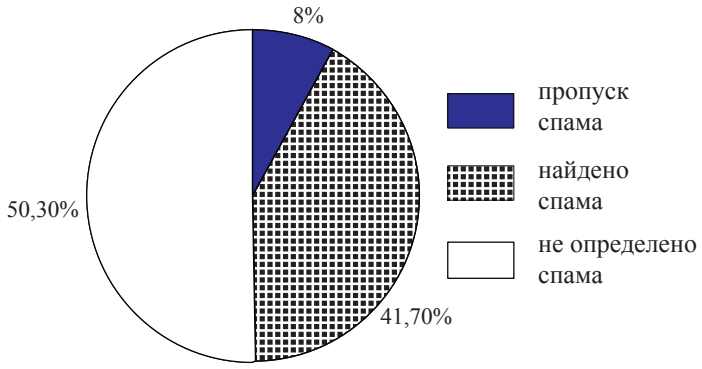


Рис. 4. Процентное соотношение количества сообщений спама от общего количества сообщений спама

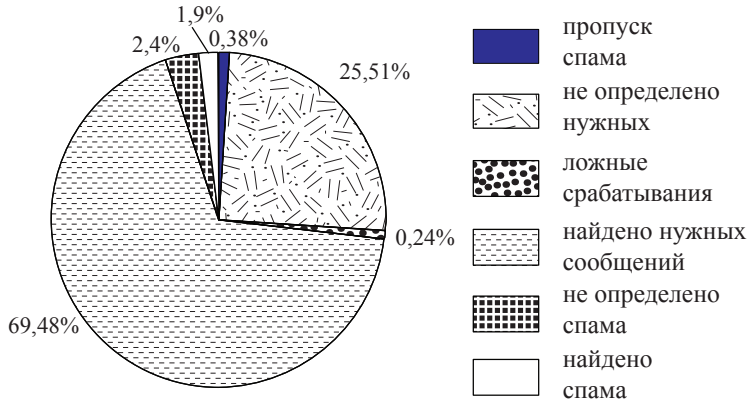


Рис. 5. Процентное соотношение количества писем спама и нужных сообщений от общего количества сообщений

- Определила 1,99% сообщений спама;
- Количество пропущенных сообщений спама равно 0,38%;
- Не смогла распознать 2,4% сообщений спама.

Заключение

Даже при обучении фильтра на небольшой выборке метод Фишера сводит к минимуму возникновение ложных срабатываний и пропуск спама. Проведенное тестирование показывает, что в ходе эксперимента были получены результаты, подтверждающие целесообразность использования автоматизированной системы фильтрации спама на сайтах. В дальнейшем необходимо обучить систему фильтрации на ложных срабатываниях, пропуске спама и на сообщениях, отнесенных к неизвестным, провести внедрение многомодульного спам-фильтра на форумы СГАУ, ПГУТИ.

Литература

1. Мезенцева, Е. М. Разработка специализированной модели, методов и алгоритмов, предназначенных для фильтрации спама / Е. М. Мезенцева // Интеллект. Инновации. Инвестиции. – 2011. – Специальный выпуск по материалам международного форума «Россия как трансформирующееся общество: экономика, культура, управление». – С. 73–77.

2. Seibel, P. Practical Common Lisp / P. Seibel. – New York : Apress, 2005. – 528 с.

3. Никольский, С. М. Квадратурные формулы / С. М. Никольский. – М. : Наука, 1974. – 224 с.