УДК 004.4'6 DOI: 10.31673/2412-4338-2018-0-2-40-45

Аронов А. О. *Державний університет телекомунікацій, Київ*

АЛГОРИТМ ВИЯВЛЕННЯ ЗАСТАРІЛОЇ ІНФОРМАЦІЇ НА ОСНОВІ АНАЛІЗУ ДАНИХ САЙТУ

Запропоновано алгоритм виявлення застарілої інформації сайту на основі еталонних шаблонів, які перетворюють текстові дані до єдиного уніфікованого представлення. Для виявлення застарілої інформації аналізуються не лише часові показники створення/оновлення сторінок сайту, а безпосередньо зміст текстової сторінки. Запропонований алгоритм призначений для використання системними адміністраторами сайту.

***Ключові слова**: супроводження сайту, застаріла інформація, інформаційно-аналітичний аналіз даних, адміністратор сайту, критерії пошуку застарілих даних, періодичність пошуку запиту.*

Aronov A. O. *State University of Telecommunications, Kyiv*

ALGORITHM OF THE DETECTION OF THE OUTDATED INFORMATION ON THE BASIS OF ANALYSIS OF DATA SITES

The paper proposes an algorithm for implementing the method of identifying outdated information on the basis of the analysis of text data of the pages of the site. The algorithm of the software application for the search of outdated information on the pages of the site, which describes its settings. The criteria for finding outdated information and the sequence of their checks are determined. It is foreseen to execute search queries in nested pages. The main criteria of relevance of the site information on different indicators are determined.

Describes the process of running search queries, which is governed by separate settings: date editing pages; start date of the search query; periodicity of the search query. After performing the preparatory steps to find out the outdated information in the page section, a search query is performed in the database to select the pages in which the texts will search for outdated information. As a result of the operation of the algorithm, templates are used that convert text data into a single unified representation. The scientific novelty of the results obtained is that an algorithm for the automatic detection of outdated information on the basis of information analytical analysis of the site's data, which differs from the existing ones, that the detection of outdated information is analyzed not only using time indices of the time of creation / updating of pages of the site, but directly the content of text page. The principle of the function in the software is described in detail, all regular expressions are described, which is used by the function to identify date markers in the text data of the analyzed pages. The proposed algorithm is intended for use by system administrators of the site.

***Keywords**: maintenance of the site, outdated information, information analytical analysis of data, site administrator, criteria for finding outdated information, periodicity of the search query.*

Аронов А. А. *Государственный университет телекоммуникаций, Киев*

АЛГОРИТМ ОБНАРУЖЕНИЯ УСТАРЕВШЕЙ ИНФОРМАЦИИ НА ОСНОВЕ АНАЛИЗА ДАННЫХ САЙТА

Предложен алгоритм выявления устаревшей информации сайта на основе эталонных шаблонов, которые превращают текстовые данные к единому унифицированному представлению. Для выявления устаревшей информации анализируются не только временные показатели создания / обновления страниц сайта, а непосредственно содержание текстовой страницы. Предложенный алгоритм предназначен для использования системными администраторами сайта.

***Ключевые слова**: сопровождение сайта, устаревшая информация, информационно-аналитический анализ данных, администратор сайта, критерии поиска устаревшей информации, периодичность поискового запроса.*

Вступ та постановка задачі дослідження

Разом із стрімким зростанням кількості сайтів збільшується й обсяг накопичуваної інформації на сайтах, що призводить до проблем ефективного функціонування сайту. До змісту інформації на сайті висуваються вимоги актуальності, цілісності, несуперечливості. Проте з моменту створення сайту на ньому зберігається все більше інформації, яка втрачає актуальність. Актуальність інформації на сайті – це відповідність тематики інформації потребам користувачів сайту. Таким чином, актуальність інформації є динамічною властивістю інформації, за якою потрібен постійний нагляд. За якість інформації на сайті відповідає системний адміністратор сайту, в обов'язки якого входить перевірка інформації на коректність та своєчасне оновлення а також видалення застарілої інформації. Враховуючи великі об'єми інформації дуже складно відстежити які з сторінок оновлені. Причина криється у редагуванні інформації: створення нових масивів інформації на сторінках сайту, виправлення допущених помилок, що робить неможливим визначення актуальної інформації за параметрами «дата останнього редагування». Крім, того складність полягає ще в тому, що фіксування дати безпосередньо у змісті сторінки, яка може визначати застарілість інформації, виражається природномовними засобами і не має уніфікованого подання, що ускладнює процес автоматизації виявлення застарілої інформації.

Таким чином, актуальною є задача надання системному адміністратору сайту інструменту, який автоматично дозволяв би виявляти застарілу інформацію на сайті та приймати рішення щодо її подальшої долі (видаляти, архівувати, переміщувати у спеціальні рубрики тощо).

Метою роботи є розробка алгоритму автоматичного виявлення застарілої інформації на основі аналізу даних сайту, який би надав системному адміністратору можливість усувати неактуальну інформацію зі сторінок сайту.

Основна частина

Відповідно до поставленої в роботі мети запропоновано алгоритм для створення програмного забезпечення (далі Додаток) для пошуку застарілої інформації на сторінках сайту, в якому передбачено такі налаштування:

1) Розділ пошуку:
- розділ «Новини»;
- розділ «Сторінки».

2) Ідентифікатор окремої сторінки на випадок, якщо пошуковий запит стосується певної новини чи сторінки на сайті.

3) Критерії пошуку:
- виконання пошуку лише у назві сторінки обраного розділу;
- виконання пошуку лише у тексті сторінки обраного розділу;
- виконання пошуку, використовуючи дати створення та редагування сторінки обраного розділу;
- виконання пошуку у вкладених сторінках;

4) Дія, яку має виконати система зі знайденими сторінками, які відповідають критеріям пошуку, а саме:
- створити сповіщення для системного адміністратора про необхідність оновлення сторінки;
- видалити сторінку з сайту;
- скопіювати зміст сторінки до архіву;
- перемістити зміст сторінки до архіву.

5) Ідентифікатор сторінки-архіву, у випадку, коли при створенні пошукового запиту адміністратор обирає дії «Скопіювати до архіву» чи «Перемістити до архіву».

Слід зазначити, що виконання пошуку у вкладених сторінках можливе для розділів сайту, у яких передбачено створення ієрархічного зв'язку між сторінками [1].

Алгоритм виявлення застарілої інформації реалізовано на основі методу автоматизації виявлення застарілої інформації [2].

Процес запуску роботи пошукових запитів регулюється окремими налаштуваннями, серед яких передбачено:

- дату редагування сторінок, починаючи з якої відбираються матеріали для аналізу;
- дату запуску пошукового запиту;
- періодичність запуску пошукового запиту, а саме:
  - одноразовий запуск;
  - один раз на день;
  - один раз на тиждень;
  - один раз на місяць;
  - один раз на рік.

Системна частина пошуку застарілої інформації для розділу «Сторінки» виконує підготовку до запиту до бази даних, а саме:

– у випадку, коли пошук інформації має відбуватись лише на одній сторінці, до запиту додається параметр з ідентифікатором необхідної для дослідження сторінки;

– у випадку, коли в налаштуваннях пошукового запиту використаний параметр «Пошук у вкладених сторінках», Додаток повинен знайти всі ідентифікатори сторінок, які необхідно включити в пошук.

Після виконання підготовчих дій для пошуку застарілої інформації у розділі «Сторінки» виконується пошуковий запит до бази даних для вибору тих сторінок, у текстах яких буде виконано пошук застарілої інформації.

В залежності від обраних критеріїв пошуку для результуючих сторінок застосовується функція «check_outdate()», результатом роботи якої можливі три варіанти:

- «outdated» – знайдена застаріла інформація;
- «actual» – інформація актуальна;
- «nodate» – не знайдені часові відмітки у вхідному тексті.

Функція «check_outdate()» потребує два параметри для своєї роботи:

- «text» – вхідні текстові дані для аналізу
- «date_add» – дата створення чи редагування сторінки, яка аналізується

З вхідних текстових даних видаляються всі елементи мови розмітки (наприклад HTML) а також стилі та порожні рядки. Після цього текстові дані перетворюються у нижній регістр для зменшення кількості порівнянь у випадках, коли часові маркери вказані у тексті різними регістрами.

Функція «check_outdate()» здатна сприймати часові показники природомовних видів:

- G1: = <число> + <місяць> + <рік> (наприклад: «2 червня 2018»);
- G2: = <число1-число2> + <місяць> + <рік> (наприклад: «22-23 травня 2018»);
- G3: = <число1-число2> + <місяць> (наприклад: «2-7 лютого відбувся»);
- G4: = <місяць> + <рік> (наприклад: «у вересні 2016»);
- G5: = <місяць> (наприклад: «у січні відбувся»).

Для забезпечення виявлення часових показників використовуються регулярні вирази [3].

Наступний регулярний вираз виявляє перші три з наведених часових показників:

$reg_expr = "$((([0-9]{1,2})(\.|\/|-|))((([0-9]{1,2})(\.|\/|-|)((".mb_strtolower(JANUARY, "utf-8")."|".mb_strtolower(FEBRUARY, "utf-8")."|".mb_strtolower(MARCH, "utf-8")."|".mb_strtolower(APRIL, "utf-8")."|".mb_strtolower(MAY, "utf-8")."|".mb_strtolower(JUNE, "utf-8")."|".mb_strtolower(JULY, "utf-8")."|".mb_strtolower(AUGUST, "utf-8")."|".mb_strtolower(SEPTEMBER, "utf-8")."|".mb_strtolower(OCTOBER, "utf-8")."|".mb_strtolower(NOVEMBER, "utf-8")."|".mb_strtolower(DECEMBER, "utf-8")."))|(([0-9]{1,2})(\.|\/|-|))|((".mb_strtolower(JANUARY, "utf-8")."|".mb_strtolower(FEBRUARY, "utf-

```
8")."|".mb_strtolower(MARCH, "utf-8")."|".mb_strtolower(APRIL, "utf-
8")."|".mb_strtolower(MAY, "utf-8")."|".mb_strtolower(JUNE, "utf-
8")."|".mb_strtolower(JULY, "utf-8")."|".mb_strtolower(AUGUST, "utf-
8")."|".mb_strtolower(SEPTEMBER, "utf-8")."|".mb_strtolower(OCTOBER, "utf-
8")."|".mb_strtolower(NOVEMBER, "utf-8")."|".mb_strtolower(DECEMBER, "utf-8").")
)))(20([0-9]{1,2}))?)$";
```

В тексті регулярного виразу використовуються змінні, що зберігають назви місяців року, які перетворюються у нижній регістр. Це дозволяє без зміни коду Додатку використовувати дану функцію на всіх мовах, що підтримуються сайтом. Результати пошуку у текстових даних записуються до змінної «matches_date».

```
if(preg_match_all($reg_expr, $text, $matches_date)) {...}
```

Після того, як в текстових даних, переданих функції, знайдено часові показники типів G1, G2, G3 необхідно визначити який саме вид часових часового показнику знайдено. Для визначення типів G2 та G3 використовується наступний регулярний вираз:

```
$reg_expr_few_days = "$([0-9]{1,2})(\.|\/|-| )((([0-9]{1,2})(\.|\/|-|
))(".mb_strtolower(JANUARY, "utf-8")."|".mb_strtolower(FEBRUARY, "utf-
8")."|".mb_strtolower(MARCH, "utf-8")."|".mb_strtolower(APRIL, "utf-
8")."|".mb_strtolower(MAY, "utf-8")."|".mb_strtolower(JUNE, "utf-
8")."|".mb_strtolower(JULY, "utf-8")."|".mb_strtolower(AUGUST, "utf-
8")."|".mb_strtolower(SEPTEMBER, "utf-8")."|".mb_strtolower(OCTOBER, "utf-
8")."|".mb_strtolower(NOVEMBER, "utf-8")."|".mb_strtolower(DECEMBER, "utf-8").")(
20([0-9]{1,2}))?)$";
if(preg_match_all($reg_expr_few_days, $text, $matches_few_dates)) {...}
```

Результати пошуку записуються у змінну «matches_few_dates». Оскільки у тексті може зустрічатись декілька різних дат, необхідно визначити найбільшу.

```
$max_date = 0;
 if($max_date < strtotime($matches_few_dates[3][0][0]) )
 {$max_date = strtotime($matches_few_dates[3][0][0]); }
```

На основі визначеної найбільшої дати функція повертає результати роботи. Якщо поточна дата більша за максимальну знайдену у тексті – дані вважаються актуальними. В іншому випадку відбувається перевірка на наявність дати редагування чи створення сторінки. Якщо така дата відсутня – дані вважаються застарілими. Якщо така дата присутня і менша за максимальну, знайдену у тексті – дані вважаються застарілими.

Якщо за результатами

```
if(time() > $max_date) {
if($date_add != 0) {
        if($max_date > strtotime($date_add))
                return "outdated"; }
        else    return "outdated";}
return "actual";
```

виконання пошуку типів G2 та G3 дати не знайдені – це означає, що у змінній «matches_date» знаходиться дата типу G1, для визначення актуальності якої проводяться аналогічні дії.

```
$max_date = 0;
for($i = 0; $i < count($matches_date[0]); $i++) {
if(     $max_date < strtotime($matches_date[0][$i][0]) ) {
                $max_date = strtotime($matches_date[0][$i][0]); }}
        if(time() > $max_date) {
        if($date_add != 0) {
        if($max_date > strtotime($date_add))
                        return "outdated";}
```

```
        else     return "outdated"; }
    return "actual".
```

Для визначення присутності часових даних типу G4 та G5 використовується регулярний вираз:

```
$reg_expr_only_month = "$((".mb_strtolower(MONTH_IN_JANUARY, "utf-
8")."|".mb_strtolower(MONTH_IN_FEBRUARY, "utf-
8")."|".mb_strtolower(MONTH_IN_MARCH, "utf-
8")."|".mb_strtolower(MONTH_IN_APRIL, "utf-8")."|".mb_strtolower(MONTH_IN_MAY,
"utf-8")."|".mb_strtolower(MONTH_IN_JUNE, "utf-
8")."|".mb_strtolower(MONTH_IN_JULY, "utf-
8")."|".mb_strtolower(MONTH_IN_AUGUST, "utf-
8")."|".mb_strtolower(MONTH_IN_SEPTEMBER, "utf-
8")."|".mb_strtolower(MONTH_IN_OCTOBER, "utf-
8")."|".mb_strtolower(MONTH_IN_NOVEMBER, "utf-
8")."|".mb_strtolower(MONTH_IN_DECEMBER, "utf-8")." ?) ?(20([0-9]{1,2}))?)$";
if(preg_match_all($reg_expr_only_month, $text, $matches))
{...}
```

Оскільки пошук проводиться за місяцем та роком при порівнянні поточної дати зі знайденою у тексті Додаток встановлює 1 число наступного місяця для знайденої дати. Модифікована знайдена дата порівнюється з поточною і у випадку якщо поточна дата менша за модифіковану – інформація актуальна. В іншому випадку – інформація застаріла.

```
$max_date = 0;
for($i = 0; $i < count($matches[0]); $i++)
if(     $max_date < strtotime( date("Y-m-01 00:00:00", strtotime($matches[0][$i][0])) ))
        $max_date = $matches[0][$i][0];
if(time() > strtotime(date("Y-m-01 00:00:00", strtotime($max_date) + 60*60*24*31)))  {
        if($date_add != 0){
                if(strtotime(date("Y-m-01 00:00:00", strtotime($max_date) + 60*60*24*31))
>                       strtotime($date_add))
                        return "outdated"; }
        else     return "outdated"; }
return "actual"
```

Якщо використання всіх регулярних виразів не принесло результатів – функція повертає повідомлення про те, що у тексті дати не знайдені.

Визначення застарілої інформації у відібраних сторінках виконується шляхом виклику функції «check_outdate» для кожного тексту окремо, який необхідно дослідити. У випадку пошуку по даті порівнюються поточна дата та найбільша серед дати створення і дати редагування сторінки.

```
$result_title = "-"; $result_text = "-"; $result_date = "-";
if($report["search_title"]){$result_title = check_outdate($new["title"], $new["date"]);}
if($report["search_text"]){ $result_text = check_outdate($new["text"], $new["date"]);}
if($report["search_date"]){if($new["date"] < $report["report_date"])     $result_date =
"outdated"; }
if(($result_title == "outdated") || ($result_text == "outdated") || ($result_date ==
"outdated"))
{...}
```

На основі вказаних перевірок, якщо хоча б один з критеріїв пошуку повідомив про наявність застарілої інформації – Додаток переходить до виконання автоматичних дій зі знайденими сторінками. Сповіщення про необхідність оновлення заноситься до таблиці звітів роботи Додатку, де записується: ідентифікатор пошукового запиту, дата перевірки, розділ

сайту та ідентифікатор сторінки розділу, автоматична дія. Оскільки всі записи у таблиці звітів переглядає адміністратор під час кожного відвідування візуального інтерфейсу додаткових дій для сповіщення виконувати не має необхідності.

Висновки

В статті запропоновано алгоритм реалізації методу автоматичного виявлення застарілої інформації на основі інформаційно-аналітичного аналізу даних сайту, який відрізняється тим, що для виявлення застарілої інформації аналізуються не лише часові показники часу створення/оновлення сайту, а безпосередньо зміст текстової сторінки. Для аналізу текстової інформації побудовані шаблони на основі регулярних виразів, які дозволяють автоматизувати процес виявлення застарілої інформації та оновлення сторінок сайту.

Список використаної літератури

1. Аронов А. О. Розробка моделі структурно-логічного представлення даних сайту вищого навчального закладу на основі ієрархічного класифікатора / А. О. Аронов // Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. – 2018. – Вип. №59. – С. 70-75.

2. Аронов А. О. Метод автоматизації виявлення застарілої інформації на основі інформаційно-аналітичного аналізу даних сайту / А. О. Аронов, В. В. Вишнівський, І. В. Замаруєва // Сучасні інформаційні системи. – 2018. – №1. – С. 28-31.

3. Гойвертс Ян. Регулярные выражения. Сборник рецептов / Ян Гойвертс, Стивен Левитан. – Символ-Плюс, 2010. – 608 с.

References (MLA)

1. Aronov A. O. "Development of the Model of Structural-Logical Representation of the Data of the Site of the Higher Educational Institution on the Basis of the Hierarchical Classifier." *Collection of scientific works of the Military Institute of Taras Shevchenko Kyiv National University* 59 (2018): 70-75. Print.

2. Aronov A. O., Vyshnivskyi V. V., and Zamaruieva I. V. "The Method of the Detection of Outdated Information on the Basis of Information Analytical Analysis of the Data of the Site." *Suchasni Informatsiyni Systemy* 1 (2018): 28-31. Print.

3. Gojverst Yan, and Levitan Stiven. *Regular Expressions. A Collection of Recipes*. Symvol-Pius, 2010. Print.

Автор статті

Аронов Андрій Олексійович – аспірант, Державний університет телекомунікацій, Київ. Тел.: +380 (91) 900 08 00. E-mail: info.dut.edu.ua@gmail.com.

Author of the article

Aronov Andrii Oleksiiovych – post graduate student, State University of Telecommunications, Kyiv. Tel: +380 (91) 900 08 00. E-mail: info.dut.edu.ua@gmail.com.

Дата надходження в редакцію: 19.02.2018 р.

Рецензент: доктор технічних наук, професор В. В. Вишнівський *Державний університет телекомунікацій, Київ*