

## **Penerapan Metode *Focused Crawler* dan Algoritma *Porter Stemmer* Pada Pencarian Lirik Lagu**

Swono Sibagariang  
Program Studi Sistem Informasi, Universitas Sari Mutiara Indonesia  
bagariangswono@yahoo.co.id

### **Abstrak**

Metode *focused crawler* bertujuan untuk selektif mencari halaman yang relevan dengan satu set yang telah ditentukan topik dengan beberapa kata kunci. Pada metode ini pengguna cukup memasukkan kata kunci untuk menemukan kalimat atau lagu pada kidung jemaat. Pada penelitian ini membahas tentang metode pencarian *focused crawler* android pada aplikasi kidung jemaat. Metode *focused crawler* adalah metode yang menggunakan proses crawling. Dimana proses tersebut dilakukan berdasarkan skor tertinggi terlebih dahulu. Dengan mengimplementasikan algoritma *focused Crawler*, aplikasi akan mencari kata kunci pada masing-masing lirik, kemudian menyimpannya pada daftar kata kunci. Dari hasil penelitian terlihat bahwa dengan menggunakan handphone yang memiliki sistem operasi Android, pengguna yang ingin mengikuti kebaktian bisa lebih mudah dalam mencari "lirik" lagu pada aplikasi kidung jemaat yang sudah diinstal, selain itu pengguna juga bisa melakukan pencarian berdasarkan judul atau lirik.

Kata kunci : *Focused Crawler*, *Porter Stemmer*, Android

### **1. PENDAHULUAN**

#### **1.1 Latar Belakang**

Pada zaman globalisasi ini, teknologi berkembang dengan sangat pesat. Ini merupakan salah satu penyebab orang-orang lebih suka menggunakan alat komunikasi, telepon selular khususnya, untuk mendapatkan informasi dari pada harus membuka buku yang belum tentu mereka punya, ditambah lagi membawa buku dinilai lebih merepotkan dari pada membawa

telepon selular. Alkitab yang berbentuk aplikasi sekarang ini sudah dapat digunakan di telepon selular, agar lebih praktis dalam pemanfaatannya. Dari fakta tersebut penulis mencoba membuat aplikasi kidung jemaat. Dimana aplikasi ini berisikan lirik lagu-lagu rohani.

Dalam sebuah lagu terkadang tidak mengingat judul lagunya, tetapi hanya mengingat lirik lagu yang sering kita nyanyikan. Untuk melakukan pencarian

dengan manual, maka aplikasi akan melakukan pencarian pada seluruh lirik, sehingga membutuhkan waktu yang lebih banyak. Dengan menggunakan metode *focused crawler*, maka aplikasi akan menyimpan kata kunci pada masing-masing lagu, sehingga pada saat kita melakukan pencarian, maka aplikasi akan mengarahkan kepada kata kunci yang disimpan, kemudian hasilnya akan ditampilkan. Jika yang dicari tidak ditemukan, maka aplikasi akan mencari ke seluruh isi.

Pada penelitian ini akan diterapkan pada aplikasi kidung jemaat, karena penulis merasa sangat pentingnya mengetahui lagu rohani yang bisa dipakai untuk kemudahan dalam bernyanyi bagi umat kristen umumnya dan khususnya bagi seluruh pengguna telepon selular berbasis Android. Salah satu bentuk pemanfaatan teknologi untuk penyajian informasi ini adalah dengan telepon selular yang memiliki sistem operasi Android. Perangkat lunak untuk

mengembangkan aplikasi ini adalah Eclipse, dimana aplikasi ini diimplementasikan pada emulator Android.

## **1.2 Maksud Dan Tujuan**

Maksud dari tulisan ini adalah :  
Menerapkan Metode *Focused Crawler* dalam pencarian lirik pada aplikasi kidung jemaat.

Adapun tujuan perancangan dari Metode pencarian *Focused Crawler* Berbasis Android pada aplikasi kidung jemaat.

## **1.3 Rumusan Masalah**

Dengan menggunakan handphone yang memiliki sistem operasi Android, orang yang ingin mengikuti kebaktian sudah bisa melihat lirik lagu pada aplikasi lirik lagu kidung jemaat yang sudah diinstal, orang juga bisa melakukan pencarian berdasarkan judul atau lirik. Dengan mengimplementasikan algoritma *Focused Crawler* , aplikasi akan mencari kata kunci

pada masing-masing lirik, kemudian menyimpannya pada daftar kata kunci.

#### 1.4 Batasan Masalah

Agar pembahasan dalam Tugas Akhir ini tidak menyimpang dari apa yang telah dirumuskan maka diperlukan pembatasan masalah. Adapun batasan masalah pada Tugas Akhir ini, yaitu :

1. Membangun aplikasi kidung jemaat Berbasis Android
2. Mengimplementasikan algortima focused crawler untuk pencarian berdasarkan judul dan isi pada aplikasi
3. Pencarian hanya berfokus pada teks yang ada pada kidung jemaat.

## 2. DASAR TEORI

### 2.1 *Focused Crawler*

Dengan menggunakan metode ini, tidak bisa dipastikan halaman mana yang terlebih dahulu dilakukan proses crawling, karena proses *crawling* dilakukan berdasarkan skor tertinggi terlebih dahulu.

Untuk jumlah halaman tidak bisa dipastikan berapa banyak yang akan menjalani proses *crawling*, karena halaman pada level selanjutnya ketergantungan terhadap level sebelumnya.

Berikut ini proses yang dilakukan Web *crawler* pada saat berkerja:

- a. Mengunduh halaman web.
- b. Memparsing halamana yang didownload dan mengambil semua link
- c. Untuk setiap link yang diambil, ulangi proses.

Dalam langkah pertama, sebuah web *crawler* mengambil URL dan mengunduh halaman dari Internet berdasarkan URL yang diberikan. Seringkali halaman yang diunduh disimpan ke sebuah file atau ditempatkan di basisdata. Dengan menyimpan halaman web, maka *crawler* atau program yang lain dapat memanipulasi halaman itu untuk diindeks (dalam kasus mesin pencari) atau untuk pengarsipan untuk

digunakan oleh pengarsip otomatis. Tahap kedua, Web *crawler* memparsing keseluruhan halaman yang diunduh dan mengambil link-link ke halaman lain.

Tiap link dalam halaman didefinisikan dengan sebuah penanda HTML yang serupa dengan yang ditunjukkan disini :

```
<A  
HREF="http://www.host.com/directory/file.  
html  
>Link</A>
```

Setelah *crawler* mengambil link dari halaman, tiap link ditambahkan ke sebuah daftar untuk dicrawl. Langkah ketiga dari Web *crawling* adalah mengulangi proses.

Web *crawler* mengambil URL dan mengunduh halaman web a yang telah disimpan. Halaman web a mempunyai 2 halaman web, yaitu halaman web b dan web c, halaman web b dan c akan dibandingkan mana yang memiliki nilai relevansi tertinggi. Pada gambar ditemukan nilai tertinggi adalah halaman web b, maka halaman web c akan dimasukkan kedalam antrian. Proses dilanjutkan web *crawler* memparsing

halaman yang di unduh dan mengambil link-link ke halaman lain, dan ditemukan 3 halaman web yang lain yaitu halaman web d,e dan f. Maka. Kemudian halaman web d,e dan f akan dimasukkan kedalam antrian. Antrian sekarang akan berisi halaman web c,d,e dan f. Setelah halaman web b selesai, maka akan dicari skor tertinggi dari nilai c,d,e dan f. pada gambar 3.11 ditemukan skor tertinggi adalah halaman web e, sehingga proses akan dilakukan pada halaman web e. prose dilanjutkan ditemukan halaman web g,h dan i, kemudian dimasukkan kedalam antrian, sehingga pada antrian akan berisi halaman web c,d,f,g,h dan i. Setelah halaman web e selesai, maka akan dicari skor tertinggi dari halaman c,d,f,g,h dan i untuk dilakukan proses *crawling* selanjutnya. Proses ini akan terus berlangsung, sehingga halaman yang memiliki nilai skor tertinggi akan menjadi prioritas utama dalam proses *crawling*.

## 2.2 Android

### 2.2.1 Pengertian Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, middleware dan aplikasi (Ivan Siregar, 2010) Android adalah sistem operasi untuk *mobile device* yang awalnya dikembangkan oleh Android Inc. Perusahaan ini kemudian dibeli oleh Google pada tahun 2005. Android dibuat berdasarkan kernel Linux yang dimodifikasi. Aplikasi Android ditulis dengan bahasa Java, menggunakan Java Core Libraries. Aplikasi Android dijalankan di atas VM bernama Dalvik Virtual Machine.

Untuk mengembangkan Android, dibentuk OHA (*Open Handset Alliance*), konsorium dari 34 perusahaan peranti keras (Hardware), peranti lunak (Software), dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia (Ivan Siregar, 2010).

### 2.2.2 Arsitektur Android

Pada Gambar II.2 berikut akan digambarkan lapisan system perangkat lunak Android.



Gambar II.7 Lapisan Perangkat Lunak Android

Berikut akan dijabarkan masing-masing lapisan pada sistem operasi Android mulai dari yang paling dekat dengan perangkat keras.

## 3. PEMBAHASAN

### 3.1 Analisis sistem

#### 3.1.1 Metode *focused crawler*

Metode *focused crawler* adalah metode yang menggunakan proses crawling, dimana proses tersebut dilakukan berdasarkan skor tertinggi terlebih dahulu. Contoh kasus:

**Haleluya! Pujilah Allah yang Agung,  
Maha Esa! Dalam Kristus kita kenal  
Allah yang hidup, Bapa kekal!**

Proses metode *focused crawlernya* adalah sebagai berikut:

1. Menghapus special karakter

Pada tahap ini dilakukan penghapusan tanda baca seperti , . : ; ‘ “ ?, pada kalimat yang akan di *crawling*, yang akan dijadikan kata kunci. Sehingga kalimat tersebut tidak memiliki special karakter.

**Haleluya Pujilah Allah yang Agung Maha Esa Dalam Kristus kita kenal Allah yang hidup Bapa kekal**

2. Pisah kalimat berdasarkan spasi

Pada tahap ini kalimat yang telah di hapus special karakternya akan dipisahkan berdasarkan spasinya.

**Array [0] = Haleluya  
Array [1] = Pujilah  
Array [2] = Allah  
Array [3] = yang  
Array [4] = Agung  
Array [5] = Maha  
Array [6] = Esa  
Array [7] = Dalam  
Array [8] = Kristus**

**Array [9] = kita  
Array [10] = kenal  
Array [11] = Allah  
Array [12] = yang  
Array [13] = hidup  
Array [14] = Bapa  
Array [15] = Kekal**

3. *Stemming* dengan *porter stemmer*

Proses menghilangkan kalimat yang memiliki awalan, akhiran dan imbuhan dengan menggunakan algoritma *porter stemmer*.

**Array [0] = Haleluya  
Array [1] = puji  
Array [2] = Allah  
Array [3] = yang  
Array [4] = Agung  
Array [5] = Maha  
Array [6] = Esa  
Array [7] = Dalam  
Array [8] = Kristus  
Array [9] = kita  
Array [10] = kenal  
Array [11] = Allah  
Array [12] = yang  
Array [13] = hidup  
Array [14] = Bapa  
Array [15] = Kekal**

4. Menghapus kata yang kurang dari 3 karakter

Pada tahap ini dilakukan proses penghapusan pada setiap kata yang kurang dari 3 karakter.

Array [0] = Haleluya  
 Array [1] = puji  
 Array [2] = Allah  
 Array [3] = yang  
 Array [4] = Agung  
 Array [5] = Maha  
 Array [6] = Esa  
 Array [7] = Kristus  
 Array [8] = kita  
 Array [9] = kenal  
 Array [10] = Allah  
 Array [11] = yang  
 Array [12] = hidup  
 Array [13] = Bapa  
 Array [14] = Kekal

5. Penghapusan kata yang tidak

memiliki arti(*stop word*)

Kata kunci yang mempunyai lebih dari 3 karakter akan diproses kembali, dengan menghapus kata kunci yang tidak memiliki arti.

Array [0] = Haleluya  
 Array [1] = puji  
 Array [2] = Allah  
 Array [3] = Agung  
 Array [4] = Maha  
 Array [5] = Esa  
 Array [6] = Kristus  
 Array [7] = kenal  
 Array [8] = Allah  
 Array [9] = hidup  
 Array [10] = Bapa  
 Array [11] = Kekal

6. Menghitung skor kata kunci

Haleluya = 1  
 Puji = 1  
 Allah = 2  
 Agung = 1

Maha = 1  
 Esa = 1  
 Kristus = 1  
 Kenal = 1  
 Hidup = 1  
 Bapa = 1  
 Kekal = 1

7. Mengambil 5 kata kunci yang memiliki skor tertinggi

Allah = 2  
 Haleluya = 1  
 Puji = 1  
 Agung = 1  
 Maha = 1

8. Total skor

Tahap ini tahap penyelesaian.

Pada tahap ini dilakukan penghitungan skor dari 5 kata kunci yang tertinggi.

Skor = 2+1+1+1+1=6

**Algoritma Dasar *Focused Crawler***

**Input:** *Seed* = {*u*<sub>1</sub>, *u*<sub>2</sub>, ..., *u*<sub>*n*</sub>} daftar URL awal  
*URL\_Pool*  $\leftarrow$  *Seed*  
*Visited*  $\leftarrow$   $\emptyset$ , URL yang telah di kunjungi  
**while** *URL\_Pool*  $\neq$   $\emptyset$   
*u*  $\leftarrow$  **Select** (*URL\_Pool*, Kriteria Pemilihan)  
*p*  $\leftarrow$  **Download** (*u*)  
*Visited*  $\leftarrow$  *Visited*  $\cup$  *u*  
*out\_link*  $\leftarrow$  **Extract\_Outgoing\_Link** (*p*)  
**for each** *q*  $\in$  *out\_link*  
**if** (*q*  $\notin$  *Visited*) and (*q*  $\in$  *URL\_Pool*)  
*URL\_Pool*  $\leftarrow$  *URL\_Pool*  $\cup$  *q*  
**end if**  
**end for**

end while

## 4. IMPLEMENTASI

### 4.1 Hasil Program

Tampilan yang dihasilkan dari program dalam proyek tugas akhir ini merupakan hasil setelah aplikasi diinstal pada telepon selular.

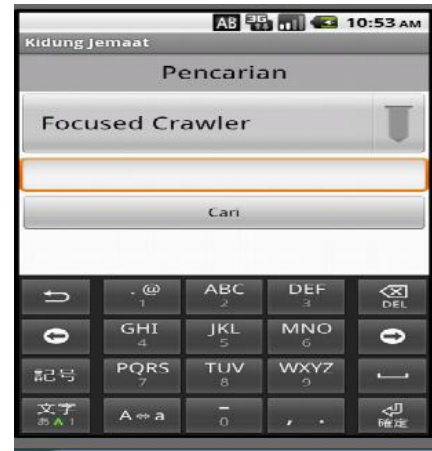
#### 4.1.1 Menu Utama

Gambar IV.1 berikut adalah tampilan awal metode pencarian *focused crawler* berbasis android pada aplikasi kidung jemaat.



Gambar IV 1 Tampilan Menu Utama

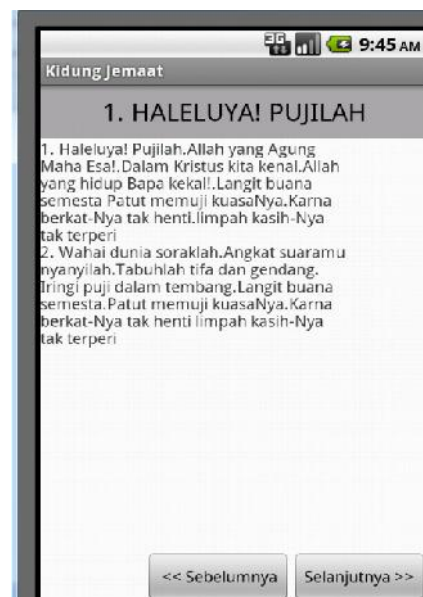
#### 4.1.2 Tampilan Pencarian berdasarkan metode *Focused Crawler*



Gambar IV.7 Tampilan pencarian berdasarkan metode *focused crawler*

Pada tampilan ini merupakan hasil pencarian menurut metode *focused crawler*. Dipencarian inipencarian berdasarkan kata kunci(*key word*).

#### 4.1.3 Tampilan Lirik lagu



Gambar IV.3 Tampilan Menu Lirik Lagu



## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Setelah dilakukan analisis, perancangan, implementasi serta uji coba Metode Pencarian *Focused Crawler* berbasis android pada aplikasi kidung jemaat ditarik kesimpulan sebagai berikut :

1. Dapat memudahkan pengguna dalam mencari halaman yang relevan.
2. Dalam melakukan pengembangan aplikasi kidung jemaat, *focused crawler* ternyata sangat efektif untuk digunakan karena dapat memudahkan pengguna dalam mencari judul-judul lagu kidung jemaat.

### 5.2 Saran

Penulis menyadari bahwa masih terdapat kekurangan dalam aplikasi ini, untuk itu melalui bagian ini disarankan pengembangan dapat digunakan algoritma dan metode yang lain.

### DAFTAR PUSTAKA

- [1] [www.computerknowledgeforyou.com](http://www.computerknowledgeforyou.com)
- [2] [http://elinux.org/Android\\_Architecture](http://elinux.org/Android_Architecture)
- [3] <http://vincommunity.wordpress.com/2009>

/04/18/perancangan-database/

- [4] <http://www.scribd.com/doc/25322890/Materi-UML-Dan-Use-Case>.<http://referensi.dosen.narotama.ac.id/files/2011/12/Pengantar-Membuat-Aplikasi-Android.pdf>.

- [5] [https://www.cs.umd.edu/~sayyadi/files/papers/4-A\\_Method\\_for\\_Focused\\_Crawling\\_Using\\_Combination\\_of\\_Link\\_Structure\\_and\\_Content\\_Similarity.pdf](https://www.cs.umd.edu/~sayyadi/files/papers/4-A_Method_for_Focused_Crawling_Using_Combination_of_Link_Structure_and_Content_Similarity.pdf)

- [6] <http://androidteam.googlecode.com/files/Anatomy-Physiology-of-an-Android.pdf>

- [7] <https://www.yumpu.com/en/document/view/22587922/focused-crawling-for-downloading-learning-objects-a-an->