

Implementasi dan Analisis Konkordansi untuk Pencarian Kata dalam Terjemahan Al-Quran Berbahasa Inggris

Dwi Putri Lestari¹, Moch. Arif Bijaksana², Eko Darwiyanto³

¹²³ Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom, Bandung
Jalan Telekomunikasi 1, Dayeuh Kolot, Bandung 40257

¹ putridesrobb@student.telkomuniversity.ac.id, ²arifbijaksana@telkomuniversity.ac.id,
³ekodarwiyanto@telkomuniversity.ac.id

Abstrak

Al-Qur'an memiliki kedudukan yang penting serta memiliki cakupan pembahasan yang luas sebagai penjabar segala sesuatu dan petunjuk bagi seluruh umat Islam. Al-Qur'an diwahyukan kepada nabi Muhammad SAW dalam bahasa Arab, sehingga masih banyak orang yang kurang memahami makna keseluruhan dari Al-Qur'an tersebut. Berkembangnya zaman, banyak pakar agama di dunia membuat dokumen terjemahan Al-Qur'an ke dalam berbagai bahasa dan mempermudah seluruh umat Islam untuk memahami isi Al-Qur'an. Namun jika melakukan pencarian kata –atau suatu makna– dari dokumen terjemahan Al-Qur'an yang sudah ada secara konvensional ini akan sulit dan membutuhkan waktu yang lama. Salah satu cara untuk mengatasi permasalahan pencarian kata dalam terjemahan Al-Qur'an yaitu dengan menggunakan metode konkordansi. Konkordansi merupakan salah satu metode yang digunakan untuk menampilkan daftar kata-kata (*keyword*) yang diambil dari sejumlah teks atau dokumen. Pencarian dimulai dengan memasukkan *keyword* kemudian sistem konkordansi akan menampilkannya dalam bentuk *list* tanpa menghilangkan keseluruhan kalimat –ayat Al-Qur'an. Hasil pencarian menggunakan metode konkordansi akan menghasilkan lebih banyak kata dibandingkan dengan pencarian yang biasa. Hasil pengujian pencarian kata dengan sistem konkordansi memiliki rata-rata tingkat perolehan (*recall*) sebesar 0,96422 dan rata-rata tingkat ketepatan (*precision*) yang dihasilkan sebesar 0,9425 serta rata-rata akurasi yang dihasilkan sebesar 0,9993. Berdasarkan hasil pengujian yang dilakukan, maka dapat diambil kesimpulan bahwa tingkat efektifitas sistem pencarian kata konkordansi sudah bisa dikatakan efektif.

Kata Kunci: Konkordansi, Al-Quran, pencarian.

Abstract

The Quran has an important position and has a wide scope of discussion as an explanation of all things and guidance for the whole Muslims. The Quran was revealed to Prophet Muhammad SAW in Arabic, so there are still many people who lack understanding of meaning whole of the Quran. Rise of the times, there are religious experts in the world made the translation of the Quran into various languages and facilitated all Muslims to understand the contents of the Quran. However, if doing a word search –a meaning– from an existing Quranic translation document that already exists conventionally will be difficult and takes a long time. One way to solve the problem of word search in the translation of the Quran is by using the method of concordance. Concordance is one of the methods used to display a list of target words extracted from a given text, or set of texts. The search begins by entering the keyword then the concordance system will display it in a list form without omitting the entire sentence – verse of the Qur'an. Search results using the concordance method will generate more words than the usual searches. The result of the word search with the concordance system has the average of recall rate of 0.96422 and the average precision of 0.9425 and average accuracy of 0.9993. Based on the results of tests conducted, it can be concluded that the level effectiveness bag system search for the word concordance can already be said to be effective.

Keywords: Concordance, Quran, Searching.

1. Pendahuluan

Al-Qur'an merupakan sumber hukum kehidupan dan petunjuk bagi umat manusia. Untuk itu, khususnya umat Islam perlu memahami cakupan isi Al-Qur'an. Para ulama sudah banyak membuat dokumen/korpus terjemahan Al-Qur'an ke dalam berbagai bahasa. Namun, melakukan pencarian suatu kata dari terjemahan Al-Qur'an untuk keperluan tertentu jika mencarinya secara manual akan membutuhkan waktu yang sangat lama[1]. Seperti contoh kasus ada sebagian orang pernah mengetahui atau mengingat makna sebuah kata dalam suatu ayat tetapi tidak dapat mengenal bunyi ayat seluruhnya, atau mengetahui sebuah ayat sepenuhnya, tetapi tidak tahu itu merupakan ayat dari nomor berapa maupun dari nama surah apa disebabkan banyaknya surah serta ayat yang terkandung dalam Al-Qur'an. Maka dari itu, tidak mudah mencarinya dalam Al-Qur'an sehingga dibutuhkan cara untuk mengeksekusi langsung korpus terjemahan Al-Qur'an yang dapat melakukan pencarian dengan waktu yang jauh lebih singkat.

Dengan perkembangan teknologi informasi yang sangat pesat saat ini dan solusi dari permasalahan di atas adalah dengan membuat sistem pencarian berbasis konkordansi. Konkordansi adalah daftar target kata yang diambil dari teks yang diberikan dalam penelitian ini adalah korpus Al-Qur'an terjemahan bahasa Inggris. Sistem konkordansi juga memiliki beberapa proses pengolahan yang bisa mengekstraksi dan menampilkan informasi lainnya seperti menghitung jumlah frekuensi kata, mengidentifikasi anggota kata lainnya dari target kata, membuat kombinasi kelompok kata (*word clusters*), serta menemukan sinonim dari target kata, sehingga memberikan proses pencarian yang lebih luas karena tidak hanya terbatas pada kata yang dicarinya saja.

Pencarian dilakukan dengan memasukkan term yang ingin dicari, kemudian hasil pencarian sistem konkordansi akan menampilkan dalam bentuk *list* dari term tersebut. Term yang dimasukkan berupa kata.

2. Dasar Teori dan Perancangan Sistem

2.1. Konkordansi

Menurut Kamus Besar Bahasa Indonesia (KBBI), konkordansi adalah daftar alfabetis kata pokok dari sebuah buku atau karya seorang penulis dengan konteks terdekat. Konkordansi juga diartikan sebagai sebuah daftar kata-kata (*keyword*) yang diambil dari sekumpulan teks/korpus, disajikan untuk menunjukkan konteks dimana kata tersebut digunakan[2]. Kombinasi antara korpus dan konkordansi memberikan masa depan yang menjanjikan di bidang pengajaran bahasa dan dengan mempelajarinya akan memungkinkan peserta didik untuk menemukan pola-pola dan menyelesaikan permasalahan salah konsep dengan cara mengamati secara luas contoh yang terjadi dalam teks-teks yang nyata.

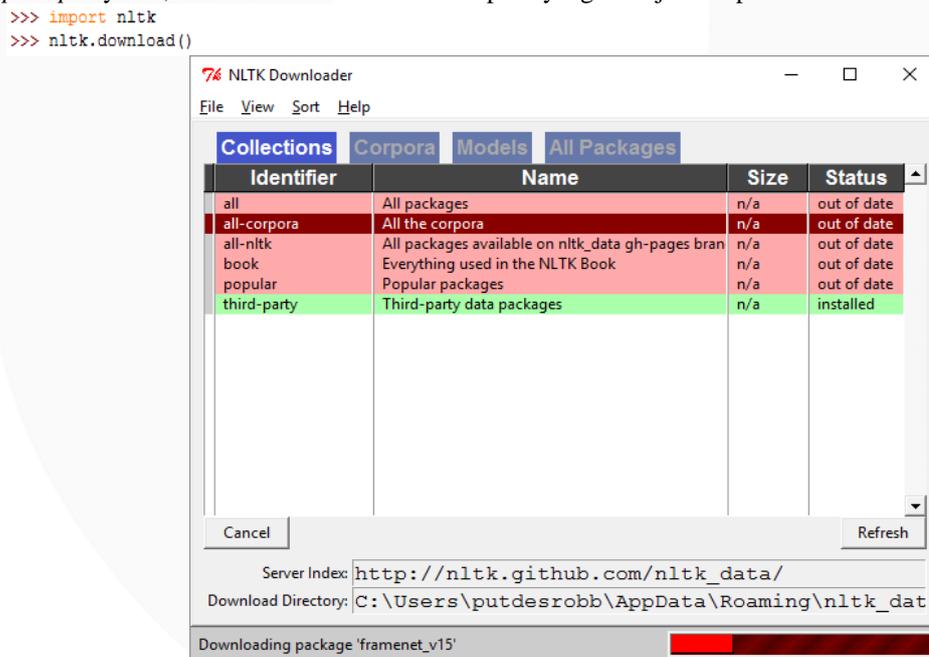
2.2. Pengenalan Python

Python merupakan bahasa pemrograman tingkat tinggi (*high level language*) yang dikembangkan oleh Guido van Rossum pada tahun 1989 dan diperkenalkan untuk pertama kalinya pada tahun 1991[3]. Python lahir atas dasar keinginan untuk mempermudah seorang programmer dalam menyelesaikan tugas-tugasnya dengan cepat. Python dirancang untuk memberikan kemudahan yang sangat luar biasa kepada *programmer* baik dari segi efisiensi waktu, maupun kemudahan dalam pengembangan program dan dalam hal kompatibilitas dengan sistem. Python bisa digunakan untuk program *standalone* dan pemrograman skrip.

2.3. NLTK

Meski Python memiliki kemampuan untuk melakukan tugas-tugas NLP dasar, namun tidak cukup powerful untuk melakukan tugas-tugas standar NLP, maka dari itu muncullah modul *Natural Language Toolkit* (NLTK). Modul ini menyediakan berbagai fungsi dan *wrapper*, serta corpora standar baik itu mentah atau pun *pre-processed* yang digunakan dalam materi pengajaran NLP. *Natural Language Toolkit* (NLTK) adalah sebuah *platform* yang digunakan untuk membangun program analisis teks[4]. *Platform* ini awalnya dirilis oleh Steven Bird dan Edward Loper dalam kaitannya dengan mata kuliah komputasi linguistik di Universitas Pennsylvania pada tahun 2001. NLTK menyediakan antarmuka yang mudah digunakan untuk lebih dari 50 sumber korpora dan leksikal seperti WordNet, bersama dengan seperangkat pustaka (*library*) pemrosesan teks untuk klasifikasi (*classification*), tokenisasi (*tokenization*), penumpukan (*stemming*), penandaan (*tagging*), penguraian (*parsing*), dan penalaran semantik (*semantic reasoning*), pembungkus (*wrapper*) pustaka NLP, dan forum diskusi aktif[5].

Modul NLTK bisa di-download di <http://www.nltk.org>. Setelah itu bisa langsung di-*install* dan mulailah dengan Python *interpreter* lalu *install* data yang diperlukan dari buku dengan mengetikkan dua perintah berikut pada *prompt* Python, lalu memilih koleksi buku seperti yang ditunjukkan pada Gambar 1.



Gambar 1: Downloading the NLTK Book Collection

Pada Gambar 1 tab *Collections* di *downloader* menunjukkan bagaimana paket dikelompokkan menjadi beberapa set, dan kita bisa memilih baris berlabel *book* untuk mendapatkan semua data yang diperlukan untuk contoh dan latihan. Setelah data diunduh, kita dapat memuat sebagian data menggunakan Python *interpreter*. Langkah pertama adalah mengetikkan perintah khusus pada *prompt* Python, yang memberitahu *interpreter* untuk memuat beberapa teks yang dapat kita jelajahi: `from nltk.book import *`. Maksudnya adalah "dari modul buku NLTK, muat semua item."

```

>>> from nltk.book import *
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
>>> |

```

Kapan pun kita ingin mencari tahu tentang teks-teks ini, kita hanya perlu memasukkan nama mereka pada *prompt* Python:

```

>>> text1
<Text: Moby Dick by Herman Melville 1851>
>>> text7
<Text: Wall Street Journal>
>>> text9
<Text: The Man Who Was Thursday by G . K . Chesterton 1908>
>>> |

```

2.3.1. Pencarian Teks

Ada banyak cara untuk memeriksa konteks teks selain hanya membacanya. Tampilan konkordansi menunjukkan kepada kita setiap kemunculan kata yang diberikan, bersama dengan beberapa konteks. Contohnya kita bisa melihat kata "affection" dari teks *Sense and Sensibility* dengan memasukkan teks2 diikuti oleh periode, kemudian istilah *concordance*, dan kemudian menempatkan "affection" dalam tanda kurung:

```

>>> text2.concordance("affection")
Displaying 25 of 79 matches:
, however , and , as a mark of his affection for the three girls , he left them
t . It was very well known that no affection was ever supposed to exist between
deration of politeness or maternal affection on the side of the former , the tw
d the suspicion -- the hope of his affection for me may warrant , without impru
high forbade the indulgence of his affection . She knew that his mother neither
rd she gave one with still greater affection . Though her late conversation wit
can never hope to feel or inspire affection again , and if her home be uncomfo
m of the sense , elegance , mutual affection , and domestic comfort of the fami
, and which recommended him to her affection beyond every thing else . His soci
ween the parties might forward the affection of Mr . Willoughby , an equally st
the most pointed assurance of her affection . Elinor could not be surprised at
he natural consequence of a strong affection in a young and ardent mind . This
opinion . But by an appeal to her affection for her mother , by representing t
every alteration of a place which affection had established as perfect with hi
e will always have one claim of my affection , which no other can possibly shar
f the evening declared at once his affection and happiness . " Shall we see you
ause he took leave of us with less affection than his usual behaviour has shewn
ness ." " I want no proof of their affection ," said Elinor ; " but of their en
onths , without telling her of his affection ;-- that they should part without
ould be the natural result of your affection for her . She used to be all unres
distinguished Elinor by no mark of affection . Marianne saw and listened with i
th no inclination for expense , no affection for strangers , no profession , an
till distinguished her by the same affection which once she had felt no doubt o
al of her confidence in Edward ' s affection , to the remembrance of every mark
was made ? Had he never owned his affection to yourself ?" " Oh , no ; but if
>>>

```

Pencarian kata "affection" dalam teks *Sense and Sensibility* ditemukan sebanyak 79 buah dan dimunculkan sebanyak 25 buah secara tampilan konkordansi. Konkordansi memungkinkan kita melihat kata-kata dalam konteks. Sebagai contoh, kita melihat bahwa "affection" terjadi dalam konteks seperti *his ___for*.

2.4. Pembobotan Kata

Pembobotan kata terdapat beberapa jenis, yaitu pembobotan lokal, pembobotan global, dan normalisasi. Pembobotan lokal diberikan pada suatu kata berdasarkan jumlah kemunculannya pada satu dokumen. Pembobotan global diberikan pada suatu kata atas kemunculannya di semua dokumen, sedangkan normalisasi adalah sebuah cara untuk menormalkan panjang vektor dokumen sehingga vektor tersebut independen terhadap panjangnya.

2.4.1. Term Frequency (TF)

Term Frequency merupakan metode untuk menghitung bobot tiap term dalam teks. Dalam metode ini, tiap term diasumsikan memiliki nilai penting yang sebanding dengan jumlah kemunculan term tersebut pada teks [6]. Adapun rumus untuk pembobotan term adalah sebagai berikut.

$$W(d, t) = TF(d, t)$$

Dimana $TF(d, t)$ adalah *term frequency* dari term t dalam teks d .

Tabel 1: Contoh *Term Frequency* pada Dokumen

Term	Dok1	Dok2
Nama	34	12
Sepeda	3	1

Belajar	7	6
---------	---	---

2.5. Korpus

Korpus (*corpus*) adalah bagian dari pustaka teks elektronik, tetapi korpus dibangun dengan kriteria untuk tujuan khusus. McEnery dan Wilson[7] menjelaskan korpus merupakan kumpulan dari beberapa teks sebagai sumber penelitian bahasa dan sastra. Kilgarriff dan Grefenstette[8] mendeskripsikan bahwa kumpulan teks disebut korpus dengan syarat kumpulan teks tersebut digunakan sebagai objek dari penelitian bahasa dan sastra. Korpus dibangun memerlukan energi yang cukup besar. Korpus memiliki anggota dokumen teks dalam jumlah yang cukup banyak, bisa sampai jutaan dokumen. Pengumpulan jutaan dokumen memerlukan waktu yang lama. Teks-teks dalam korpus disusun dengan sistematis untuk memudahkan pengelolaan[9]. Beberapa hal yang perlu diperhatikan dalam membangun korpus adalah sebagai berikut : a) perencanaan dan rancangan korpus; b) pemilihan sumber data; c) ijin dari pemilik data; d) pengumpulan data dan pengkodeannya; dan e) penanganan korpus.

2.6. Regular Expression

Regular Expression (ekspresi regular, regexp, regex, RE) adalah sebuah bahasa mini untuk mendiskripsikan string atau teks[10]. Regex dapat dipakai untuk mencocokkan sebuah string dengan sebuah pola. *Regular expression* (regex) memungkinkan kita melakukan pencarian, substitusi, atau pemisahan string dalam kasus yang kompleks. Contoh misalkan mencari teks berawalan karakter tertentu, memiliki jumlah perulangan dari suatu teks, dan lain sebagainya. Selain itu, regex juga sangat manjur (*powerful*) terutama untuk proses penguraian kata (*text parsing*).

2.7. WordNet

WordNet adalah sistem referensi leksikal *online* yang desainnya terinspirasi oleh teori psikolinguistik terkini tentang memori leksikal manusia[11]. Kata benda, kata kerja, dan kata sifat dalam bahasa Inggris disusun dalam bentuk sinonim, masing-masing mewakili satu konsep leksikal yang mendasarinya. Perbedaan antara WordNet dengan kamus bahasa pada umumnya adalah kamus bahasa memfokuskan pada kata sedangkan WordNet memfokuskan diri kepada makna kata. Satu makna dalam WordNet dapat dinyatakan dengan *synset* (*synonym set*), yaitu kumpulan kata yang merepresentasikan suatu makna. Selain dari representasi makna, di dalam WordNet juga terdapat relasi/hubungan antar makna seperti hipernim, hiponim, holonim, meronim, dll.

2.8. Precision dan Recall

Menurut Kurniawan (2010) *Precision* adalah perbandingan jumlah dokumen yang relevan terhadap *query* dengan jumlah dokumen yang terambil dari hasil pencarian. *Recall* adalah perbandingan jumlah dokumen relevan yang terambil sesuai dengan *query* yang diberikan dengan total kumpulan dokumen yang relevan dengan *query*.

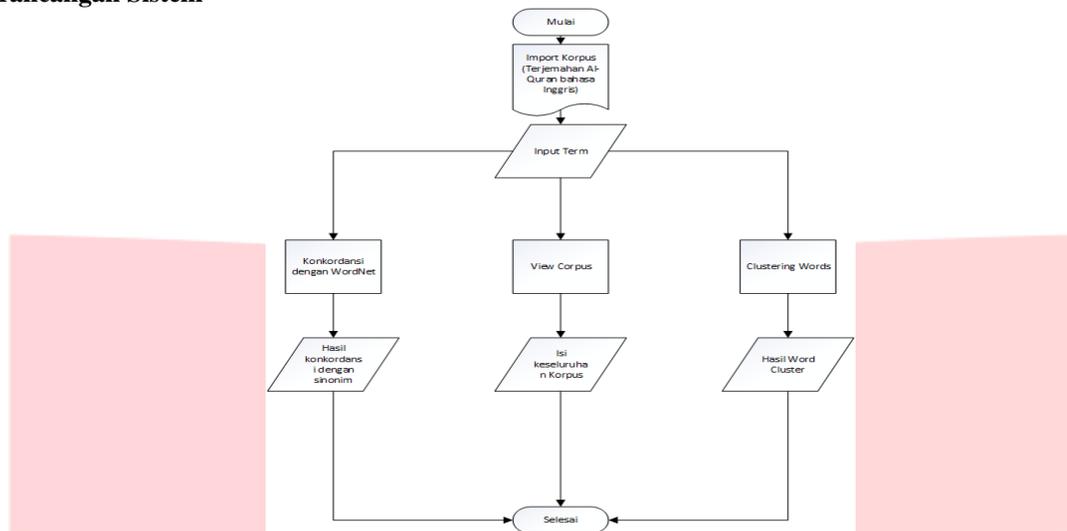
Pada penelitian pencarian kata berbasis konkordansi ini bagaimana menilai keefektifan hasil penelusuran menggunakan *precision-recall* bisa didefinisikan bahwa *precision* adalah perbandingan jumlah ayat Al-Qur'an yang relevan terhadap kata (*keyword*) dengan jumlah ayat Al-Qur'an yang terambil dari hasil pencarian. *Recall* adalah perbandingan jumlah ayat Al-Qur'an relevan yang terambil sesuai dengan *keyword* yang diberikan dengan total kumpulan ayat Al-Qur'an yang relevan dengan *keyword*.

Perhitungan performansi sistem ini akan membandingkan hasil penelusuran yang ditemukan oleh sistem dengan total keseluruhan yang sebenarnya yang terdapat pada Al-Qur'an. Berikut merupakan pengklasifikasin perhitungan untuk performansi sistem pada Tabel 2.

Tabel 2: Klasifikasi Performansi Sistem Konkordansi

Sistem	Gold Standard	
	Mengandung <i>Keyword</i>	Tidak Mengandung <i>Keyword</i>
Mengandung <i>Keyword</i>	<i>True Positive</i>	<i>False Positive</i>
Tidak Mengandung <i>Keyword</i>	<i>False Negative</i>	<i>True Negative</i>

2.9. Perancangan Sistem



Gambar 2 Gambaran umum sistem

Berikut penjelasan tahapan dan proses utama sistem pada Gambar 2:

1. Sistem membuka (*Open*) korpus yang berupa file .txt yaitu terjemahan Al-Qur'an dalam bahasa Inggris.
2. Sebelum melakukan semua proses pengolahan pengguna memasukkan term yang ingin dicari berdasarkan keperluan.
3. Pencarian term menggunakan konkordansi akan dikombinasikan dengan WordNet maka akan dihasilkan pencarian secara konkordansi dengan sinonim dari term.
4. Pencarian term juga bisa dilakukan seperti pencarian biasa.
5. Pencarian term dengan n-gram sebelumnya pengguna akan mengatur seberapa banyak jumlah *clustering* ke kanan yang dibutuhkan untuk term yang dicari dan juga bisa mengatur minimum frekuensi dari kata yang dicarinya tersebut.

3. Pembahasan

Sistem dibangun sesuai perancangan sistem. Sistem yang mengimplementasikan konkordansi dalam pencarian kata dalam terjemahan Al-Quran dan beberapa pemrosesan tambahan lainnya seperti halnya sinonim kata dengan WordNet dan pengelompokkan kata dengan n-gram.

3.1. Skenario Pengujian

1. *Blackbox testing*
Pengujian dilakukan untuk membandingkan keluaran yang diharapkan dengan hasil keluaran riil. Pada blackbox testing, pengujian lebih berfokus pada percobaan berbagai kombinasi kemungkinan data masukkan yang mungkin diterima sistem.
2. Pengujian performansi
Pengujian dilakukan untuk mengetahui performansi dari sistem yang telah dibangun. Pada tugas akhir ini digunakan perhitungan menggunakan akurasi-precision-recall - untuk menentukan performansi setiap fitur yang ada di sistem yang dibangun.
3. Pengujian kecepatan
Pengujian dilakukan untuk mengetahui kecepatan proses eksekusi sistem yang telah dibangun dengan data yang digunakan.

3.2. Analisis Hasil Pengujian

Berikut merupakan hasil pengujian yang telah dilakukan.

3.2.1. Analisis hasil *blackbox testing*

Tabel 3 : Hasil Analisis *Blackbox Testing*

Pengujian		Kondisi
Skenario	Hasil	
Impor korpus terjemahan Al-Quran bahasa Inggris	Tampilan nama File Korpus	Sesuai
Masukkan Kata	Tampil seluruh list kata beserta surah, ayat, jumlah frekuensi, nama file ditemukannya kata, repon waktu yang dibutuhkan	Sesuai
Memilih Sinonim	Tampilan sinonim kata	Sesuai

Memilih File View	Tampil keseluruhan isi korpus terjemahan Al-Quran bahasa Inggris	Sesuai
Memilih Word Clusters	Tampil tabel yang berisi rincian informasi kata-kata setelah kata input dengan panjang sesuai yang telah ditentukan oleh user sebelumnya.	Sesuai

3.2.2. Analisis performansi

Pada pengujian ini akan membandingkan hasil keluaran sistem untuk pencarian kata berdasarkan konkordansi dengan data seluruh kata yang terdapat di dalam korpus terjemahan Al-Quran bahasa Inggris dari Yusuf Ali yang sebenarnya yang sudah di hitung secara manual. Cara pengujiannya adalah pengguna cukup memasukan kata kunci kedalam kotak pencarian yang telah tersedia. Kata kunci tersebut akan dicocokkan dengan korpus oleh sistem. Jika ada yang cocok dengan kata kunci yang dimasukan, maka akan dimunculkan ayat Al-Quran yang relevan oleh sistem konkordansi. Berikut merupakan beberapa contoh penelusuran kata pada Tabel 4.

Tabel 4 : Hasil penelusuran

NO	Kata yang dicari	Data Al-Qur'an	Relevan	Tidak Relevan	Ditemukan (Sistem)
1	praises	45	44	0	44
2	islam	39	31	0	31
3	persons	25	25	0	25
4	creatures	23	20	0	20
5	prophets	23	23	0	23
6	peoples	22	22	0	22
7	person	20	20	27	47
8	worshipped	16	16	0	16
9	muhammad	4	4	0	4
10	paradise	4	4	0	4

Untuk mengetahui bagaimana *recall* dan *precision* pada sistem konkordansi untuk terjemahan Al-Quran maka penulis mengolah dan menganalisis data di atas dengan mengacu pada rumus yang sudah penulis paparkan sebelumnya.

Tabel 5 : Analisis hasil penelusuran

No.	TP	TN	FP	FN	Precision	Recall	Akurasi
1	44	6192	0	1	1	0,977777778	0,999839667
2	31	6205	0	8	1	0,794871795	0,99871877
3	25	6211	0	0	1	1	1
4	20	6216	0	3	1	0,869565217	0,999519154
5	23	6213	0	0	1	1	1
6	22	6214	0	0	1	1	1
7	20	6216	27	0	0,425531915	1	0,995688967
8	16	6220	0	0	1	1	1
9	4	6232	0	0	1	1	1
10	4	6232	0	0	1	1	1
Rata-rata					0,942553191	0,964221479	0,999376656

Dari Tabel 5 dapat dilihat bahwa rata-rata nilai *precision* dari data tersebut di atas adalah sebesar 0,9425 dan nilai *recall* -nya adalah sebesar 0,96422 dari skala 0-1, sehingga dapat diketahui bahwa nilai *precision* lebih rendah dari pada nilai *recall* berdasarkan kata kunci yang digunakan oleh pengguna. Walaupun nilai *precision* lebih rendah dari pada nilai *recall*, tingkat keefektifan dari sistem konkordansi sudah dikatakan efektif. Keefektifan suatu sistem temu kembali informasi dinilai berdasarkan teori yang dicetuskan oleh Lancaster (1991) dalam Pendit (2008) yaitu relevan dan tidak relevan, jadi efektifitas temu kembali informasi dibedakan menjadi efektif jika nilai di atas 50% dan tidak efektif jika nilai dibawah nilai 50%. Kemudian Kondisi ideal dari keefektifan suatu sistem temu

kembali informasi adalah apabila rasio *recall* dan *precision* sama besarnya (1:1) (Lee Pao, 1989). Selain itu, suatu sistem temu kembali dinyatakan efektif apabila hasil penelusuran mampu menunjukkan ketepatan (*precision*) yang tinggi sekalipun perolehannya (*recall*) rendah (Rowley, 1992).

Dari sejumlah kata kunci yang dicari, kata kunci yang nilai *recall* dan *precision* yang tinggi terdapat pada kata kunci "persons", "prophets", "peoples", "worshipped", "muhammad", dan "paradise" yang nilai *recall* dan *precision* adalah 1. Hal tersebut dikarenakan jumlah yang relevan lebih besar atau seimbang jika dibandingkan dengan yang tidak relevan. Hal ini membuktikan bahwa nilai *recall* dan *precision* tinggi karena sistem dapat mengendalikan pencarian yang dilakukan dengan menggunakan kata kunci. Sedangkan yang mempunyai nilai *precision* paling rendah adalah kata kunci "person", yang nilai *recall* 1 dan *precision* 0,4255 hal tersebut dikarenakan jumlah yang tidak relevan lebih besar atau tidak seimbang dari jumlah yang relevan. Nilai *recall* dan *precision* tidak seimbang disebabkan oleh faktor-faktor, diantaranya: tidak sesuai pencocokan kata kunci yang dicari dengan kata yang ada di korpus, kesalahan pengembang dalam membangun sistem (*human errors*), dan masih banyak faktor-faktor lainnya.

3.2.3. Analisis kecepatan

Pengujian dilakukan untuk mengetahui kecepatan proses eksekusi sistem saat melakukan pencarian. Berikut hasil waktu respon sistem saat melakukan pencarian pada Tabel 6.

Tabel 6 :

Kata yang dicari	Sistem Penelusuran	Respon Time (S)
<i>earth</i>	409	1,8
<i>sin</i>	335	2,1
<i>heaven</i>	239	1,4
<i>world</i>	104	1,2
<i>angel</i>	99	0,9352
<i>judge</i>	52	0,5510
<i>river</i>	49	0,5480
<i>sky</i>	47	0,9398
<i>praises</i>	44	0,7306
<i>islam</i>	31	0,4879
<i>persons</i>	25	0,3773
<i>creatures</i>	20	0,3678
<i>prophets</i>	23	0,4506
<i>peoples</i>	22	0,4074
<i>worshipped</i>	16	0,3177
<i>muhammad</i>	4	0,1291
<i>devil</i>	2	0,0862

Dari Tabel 4.4 diatas dapat disimpulkan bahwa melakukan pencarian kata dengan sistem konkordansi tidak memakan banyak waktu melainkan hanya membutuhkan selang waktu yang cepat dengan hitungan detik (s).

4. Kesimpulan dan Saran

4.1. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan dalam penelitian ini dapat disimpulkan:

1. Berdasarkan hasil pengujian dapat disimpulkan bahwa melakukan pencarian kata dengan konkordansi tidak membutuhkan waktu yang lama jika dibandingkan dengan pencarian secara konvensional. Tampilan yang dihasilkan sistem konkordansi juga terlihat lebih terstruktur dan rapi.
2. Sistem konkordansi dikombinasikan dengan proses pengolahan lainnya seperti WordNet dan *Word Cluster* akan menyediakan informasi yang lebih dari sekedar pencarian kata saja seperti informasi sinonim kata serta pengelompokan kata.
3. Berdasarkan hasil pengujian yang dilakukan, maka dapat diambil kesimpulan bahwa tingkat efektifitas sistem pencarian kata konkordansi sudah bisa dikatakan efektif. Hal ini terlihat dari rata-rata tingkat perolehan (*recall*) sebesar 0,96422 dan rata-rata tingkat ketepatan (*precision*) yang dihasilkan sebesar 0,9425 serta rata-rata akurasi yang dihasilkan sebesar 0,9993.

4.2. Saran

Saran penulis yang diperlukan untuk pengembangan Tugas Akhir ini adalah:

1. Sistem pencarian konkordansi dapat melakukan pencarian tidak hanya untuk mengeksekusi korpus terjemahan melainkan dengan keseluruhan Al-Qur'an agar informasi yang disajikan lebih lengkap lagi.
2. Sistem konkordansi juga dapat dikembangkan untuk menganalisis sejumlah korpus lainnya.

3. Penambahan fitur tambahannya lainnya seperti kolokasi, konkordansi plot, dll.

5. Daftar Pustaka

[1]	F. Tataran, "Aplikasi panduan kata dalam mencari ayat al-quran juz 30 berbasis mobile," pp. 1–2, 2010.
[2]	Y.-C. Sun and L.-Y. Wang, "Concordancers in the efl classroom: Cognitive approaches and collocation difficulty," <i>Computer Assisted Language Learning</i> , vol. 16, no. 1, pp. 83–94, 2003.
[3]	Sianipar and H. Wadi, <i>Pemrograman Python (Teori dan Implementasi)</i> . Penerbit INFORMATIKA, 2015.
[4]	S. Bird, E. Klein, and E. Loper, <i>Natural language processing with Python: analyzing text with the natural language toolkit.</i> "O'Reilly Media, Inc.", 2009.
[5]	S. Bird, "Nltk: the natural language toolkit," in <i>Proceedings of the CO- LING/ACL on Interactive presentation sessions</i> , pp. 69–72, Association for Computational Linguistics, 2006.
[6]	M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper.," in <i>FLAIRS conference</i> , vol. 1999, pp. 235–239, 1999.
[7]	T. McEnery and A. Wilson, <i>Corpus linguistics: An introduction</i> . Edinburgh University Press, 2001.
[8]	A. Kilgariff and G. Grefenstette, "introduction to the special issue on the web as corpus," <i>Computational Linguistics</i> , vol. 29, no. 3, pp. 333–348, 2003.
[9]	N. Nesselhauf, <i>Collocations in a learner corpus</i> , vol. 14. John Benjamins Publishing, 2005.
[10]	F. Samopa, D. H. Murti, and O. Oktanio, "Sistem query pada dokumen xml dengan menggunakan bahasa sql," <i>JUTI: Jurnal Ilmiah Teknologi Informasi</i> , vol. 4, no. 2, pp. 133–140, 2005.
[11]	G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to wordnet: An on-line lexical database," <i>International journal of lexicography</i> , vol. 3, no. 4, pp. 235–244, 1990.