Deteksi Kemiripan Bagian-bagian Terjemah Al-Qur'an dengan Menggunakan Metode *Latent Semantic Analysis*

Ardhi Akmaludin Jadhira[1], Moch Arif Bijaksana [2], Bambang Ari Wahyudi [3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]ardhiakmaludinj@students.telkomuniversity.ac.id, [2]arifbijaksana@telkomuniversity.ac.id,
[3]bambangari@telkomuniversity.ac.id

Abstrak

Dalam kitab suci umat muslim, yaitu Al-Qur'an terdapat bagian-bagian terjemah yang memiliki kemiripan semantik antar halaman berbeda. Dalam memahami kemiripan semantik dan mengetahui keterkaitan bagian-bagian terjemah Al-Qur'an bukan sesuatu yang mudah dan cepat, kemiripan semantik dalam Al-Qur'an cukup sulit dimengerti karena maknanya yang sangat kompleks. Permasalahan yang akan diangkat dalam tugas akhir ini adalah bagaimana mengetahui nilai kemiripan semantik dari halaman terjemah Al-Qur'an dengan halaman-halaman yang lain. Dengan menerapkan metode latent semantic analysis yang dibantu dengan teknik singular value decomposition dan low rank approximation diharapkan dapat membantu dalam mencari pasangan-pasangan yang memiliki kemiripan semantik. Dalam mencari nilai kemiripan semantik latent semantic analysis menggunakan perhitungan cosine similarity. Dataset yang digunakan dalam penelitian ini adalah teks terjemah Al-Qur'an berbahasa Inggris, dengan keluaran sistem yaitu tingkat kemiripan dari dua buah atau lebih halaman yang dipasangkan. Dari hasil pengujian bahwa dengan menggunakan dimensi atau parameter Rank K yang maksimum didapatkan akurasi dan F-measure yaitu 100%. Jika semakin kecil dimensi atau parameter Rank K yang digunakan adalah minimum maka nilai kemiripan semantik akan semakin besar dan beragam serta semakin tidak relevan dengan dataset pasangan-pasangan halaman yang telah ditentukan.

Kata kunci : Terjemahan Al-Qur'an, Latent Semantic Analysis. Cosine Similarity

Abstract

In the Muslim holy book, Al-Qur'an contains translation parts that have semantic similarities between different pages. In understanding the semantic similarities and knowing the relevance of the parts of the translation of Al-Qur'an is not something that is easy and fast, the semantic similarity in Al-Qur'an is quite difficult to understand because of its very complex meaning. The problem that will be raised in this final project is how to find out the semantic similarity value of the translation pages of Al-Qur'an with other pages. By applying the latent semantic analysis method, which is assisted by singular value decomposition techniques and low rank approximation, it is expected to help in finding pairs that have semantic similarities. In looking for semantic similarity values, latent semantic analysis uses cosine similarity calculations. The dataset used in this research is the translation of Al-Qur'an in English, with the output of the system that is the level of similarity of two or more pages that are paired. From the results of testing that by using the maximum dimension or parameter of Rank K, accuracy and F-measure are 100%. If the smaller dimensions or the Rank K parameters used are the minimum, the semantic similarity value will be even greater and more diverse and increasingly irrelevant to dataset of predefined page pairs.

Keywords: Al-Qur'an Translation, Latent Semantic Analysis, Cosine Similarity

1. Pendahuluan

1.1 Latar Belakang

Al-Qur'an adalah kitab suci dan pedoman hidup manusia, Al-Qur'an terdiri dari 114 surah dan 6236 ayat. Pada kitab Al-Qur'an terdapat bagian-bagian terjemah yang memiliki kemiripan semantik atau makna tekstual yang serupa serta saling memiliki keterkaitan, seperti bagian-bagian terjemah yang diulang baik dalam surah yang sama ataupun surah yang berbeda dengan tingkat kemiripan semantik yang berbeda-beda. Dalam memahami kemiripan semantik dan mengetahui keterkaitan bagian-bagian terjemah Al-Qur'an bukan sesuatu yang mudah dan cepat, kemiripan semantik dalam Al-Qur'an cukup sulit dimengerti karena maknanya yang sangat kompleks. Permasalahan yang akan diangkat dalam tugas akhir ini adalah mengetahui nilai kemiripan semantik dari suatu halaman terjemah Al-Qur'an dengan halaman-halaman terjemah Al-Qur'an. Hakikatnya manusia dapat mengetahui dan menganalisa antara satu bagian terjemah dengan bagian terjemah yang lain apakah memiliki kemiripan semantik atau tidak, namun proses analisanya akan sulit dan lama apabila dilakukan secara manual untuk jumlah *dataset* yang cukup banyak. Manfaat dari deteksi kemiripan disini yaitu membantu bagi pihak yang membutuhkan dalam mencari adanya halaman, dimana dalam suatu halaman tersebut terdapat ayat yang diduga memiliki kemiripan serta keterkaitan, dimana ayat tersebut merupakan suatu tafsiran dari ayat yang lain dalam halaman terjemah Al-Qur'an.

Oleh karena itu, untuk mengatasi masalah tersebut, dalam tugas akhir ini digunakan metode *Latent Semantic Analysis* (LSA). LSA adalah teknik dalam pemrosesan bahasa alami, khususnya dalam vektor semantik, LSA menganalisis hubungan antara sekumpulan dokumen dan istilah yang dikandungnya dengan menghasilkan serangkaian konsep yang terkait dengan dokumen [1]. Metode ini merupakan suatu model geometris yang merepresentasikan *term* sebagai konsep-konsep ke dalam ruang vektor. *Latent semantic analysis* menekankan pengamatan terhadap *term-term* yang menjadi acuan penilaian tanpa memperhatikan karakter linguistiknya dengan menggunakan proses-proses matematis yang menitikberatkan pada pengolahan matriks [2]. Dalam LSA terdapat teknik *Singular Value Decomposition* (SVD), SVD ini dapat menampakkan hubungan atau konsep yang mendasari *term* dan dokumen serta dapat membuang *noise* yang ada dalam ruang vektor [2]. Dalam mencari nilai kemiripan semantik *latent semantic analysis* dibantu dengan perhitungan *cosine similarity*. Berdasarkan permasalahan tersebut diperlukan suatu rancangan aplikasi. Diharapkan dengan menerapkan *latent semantic analysis* dapat mengetahui nilai kemiripan semantik dari bagian-bagian terjemah Al-Qur'an.

1.2 Topik dan Batasannya

Input yang diberikan kepada sistem adalah pasangan-pasangan halaman, dalam pasangan-pasangan tersebut terdapat halaman yang dicurigai mengandung kemiripan semantik yang akan dipasangkan dangan halaman-halaman lain sebagai korpusnya. Adapun *output* dari sistem adalah nilai kemiripan semantik dari masing-masing pasangan halaman serta presentase rasio yang mengandung kemiripan semantik yang tinggi. Contohnya adalah pasangan halaman dalam surah Ar-Rahman juz ke-27, *input* yang diberikan sistem adalah halaman 533 yang dipasangkan dengan halaman 532 dan 299. Keluaran dari sistem tersebut adalah nilai kemiripan pasangan 533 dengan 532, dan 533 dengan 299 serta rasio kemiripan semantik dari kedua pasangan tersebut.

Dalam menyelesaikan masalah pada tugas akhir ini, terdapat beberapa batasan masalah diantaranya:

1. *Dataset* yang digunakan adalah bagian-bagian terjemah Al-Qur'an berbahasa Inggris dengan jenis terjemah Saheeh International yang didapatkan dalam situs tanzil.net. Pengumpulan *dataset* didapatkan dari penelitian sebelumnya [3] dan ada yang dilakukan secara manual untuk menentukan kandidat halaman-halaman yang akan dipasangkan. Karakteristik kemiripan semantik dari suatu pasangan halaman dapat dilihat dengan adanya pasangan ayat yang mirip, data pasangan-pasangan tersebut dapat dilihat pada Lampiran 1.
2. Halaman-halaman yang akan dipasangkan telah ditentukan sebelumnya, adapun minimal jumlah korpus atau halaman yang dipasangkan adalah dua, dimana ada pasangan halaman yang memiliki kemiripan semantik dan ada pasangan halaman yang tidak memiliki kemiripan semantik. Pemilihan batasan halaman tersebut bertujuan untuk menguji jika pasangan-pasangan tersebut terjadi *missed detection* atau kesalahan dalam mendeteksi pasangan yang memiliki kemiripan semantik atau tidak. Adapun maksimal jumlah halaman dalam korpusnya adalah 10, jumlah tersebut dipilih karena untuk mempersempit ruang pengujian dalam menguji pengaruh jumlah dimensi dari parameter K yang digunakan terhadap perhitungan nilai *similarity*.
3. Kemiripan semantik ditandai dengan *rangking*, dimana pasangan yang memiliki kemiripan semantik yang tinggi menunjukkan nilai *similarity* mendekati angka satu, sedangkan pasangan yang tidak memiliki kemiripan semantik menujukkan nilai *similarity* mendekati angka nol.

1.3 Tujuan

Adapun tujuan dari tugas akhir ini adalah sebagai berikut:

1. Mengetahui nilai kemiripan semantik dengan menggunakan metode *latent semantic analysis* pada bagian-bagian terjemah Al-Qur'an.
2. Mengetahui pengaruh jumlah dimensi pada pencarian nilai kemiripan.
3. Mengetahui efektivitas *latent semantic analysis* dalam mendeteksi kemiripan semantik pada bagian-bagian terjemah Al-Qur'an.

2. Studi Terkait

2.1 Al-Qur'an

Al-Qur'an secara etimologis berasal dari kata kerja qaraa yang artinya bacaan atau sesuatu yang harus dibaca atau dipelajari [4]. Al-Qur'an adalah salah satu kitab suci yang terpelihara keasliannya, terdiri dari 6236 ayat, 114 surat dan 30 juz. Al-Qur'an merupakan sumber hukum utama dan sebagai pedoman hidup umat manusia. Dalam Al-Qur'an terdapat dua jenis surah yang dibedakan berdasarkan waktu dan tempat turunnya ayat yaitu Makkiyah dan Madaniyah.

2.2 Latent Semantic Analysis

Latent Semantic Analysis (LSA) adalah suatu teori atau metode statistik aljabar yang melakukan ekstraksi struktur semantik yang tersembunyi dari kalimat berupa himpunan *term* dari dokumen [2]. LSA menggunakan pendekatan matematis untuk mengetahui makna dari suatu teks dengan memanfaatkan fungsi statistik dari data berbasis korpus dengan pendekatan dimensi rendah (*low rank approximation*), konsep LSA mencari nilai kemiripan diantara dua buah segmen teks tanpa memperhatikan susunan kata. Metode ini mempunyai prinsip bahwa suatu konsep yang ada dalam teks bisa diketahui cukup dengan kemunculan katanya saja. Konteks kalimat bisa didapatkan dari diksi yang digunakan, karena tiap kata kunci memiliki makna yang memiliki kaitan dengan dokumen dan dianggap sudah cukup merepresentasikan ide [2]. Secara sederhananya dalam LSA teks direpresentasikan menjadi sebuah matriks dimana baris mewakili kata kunci atau *term* dan kolom mewakili dokumen, isi dalam matriks tersebut adalah frekuensi kemunculan kata dalam suatu dokumen [5]. Isi dalam matriks tersebut kemudian diberi bobot, yang kemudian bobot tersebut akan dilakukan proses dekomposisi matriks menggunakan *singular value decomposition* dan terakhir akan dilakukan perhitungan nilai kemiripan menggunakan *cosine similarity* [5].

Adapun langkah-langkah detil dari proses LSA adalah sebagai berikut:

1. *Text preprocessing*, tahap ini mempersiapkan teks menjadi data yang akan mengalami pengolahan lebih lanjut diantaranya, *case folding*, tokenisasi, *punctuation removal*, dan *stopwords removal* [6]. *Case folding* merupakan suatu proses menyamakan teks dalam dokumen, *case folding* digunakan untuk merubah huruf kapital menjadi huruf kecil. Tokenisasi melakukan pembentukkan suatu kalimat menjadi bagian atau unit terkecil, dalam tahap ini teks yang berupa kalimat dipotong menjadi per kata. *Punctuation removal* melakukan penghilangan digit dan simbol. *Stopwords removal* melakukan penghilangan terhadap kata-kata umum yang biasa sering diucapkan.
2. Mencari *Term Frequency* (TF) dan melakukan pembobotan *term*. Perhitungan bobot yang lazim digunakan adalah *Term Frequency* (TF)-*Inverse Document Frequency* (IDF). Adapun rumusnya [7] adalah sebagai berikut:

$$Wdt = TF\ x\ IDF \qquad (1)$$

$$Wdt = TF\ x \log \frac{n}{df} \qquad (2)$$

Keterangan:
Wdt : bobot *term* (kata) terhadap dokumen
TF : jumlah kemunculan *term* dalam suatu dokumen
n : jumlah semua dokumen yang ada dalam *dataset*
df : jumlah *term* yang terkandung dalam semua dokumen

3. *Latent semantic analysis* menggunakan teknik *Singular Value Decomposition* (SVD) dan *low rank approximation* dalam mendekomposisi dan mengurangi dimensi matriksnya. SVD akan menguraikan matriks Wdt menjadi tiga matriks yaitu matrik U, S, V**.** Adapun rumusnya [8] adalah sebagi berikut:

$$A_{td} \approx U_{tm} S_{mm} V_{md}^{T} \qquad (3)$$

Keterangan:
A: matriks asal
U: matriks eigenvector dari AA^T
S : matriks diagonal
V^T : transpose dari matriks V
t : jumlah baris matriks
d : jumlah kolom matriks
m: rank, dimana rank (< min(t,d))

Dekomposisi SVD memungkinkan dimensi matriks asal untuk dilakukan reduksi dimensi. Dengan proses reduksi dimensi terhadap matriks SVD, maka akan diperoleh penyederhanaan dari matriks asal dengan mengambil struktur penting antara kata kunci dengan kalimatnya [8].

2.3 Cosine Similarity

Pada tugas akhir ini kemiripan antar pasangan halaman menggunakan *cosine similarity*. *Cosine similarity* merupakan proses perhitungan kemiripan antar dua vektor yang umumnya digunakan dalam pencarian kemiripan antar dokumen [9]. Dengan teks yang direpresentasikan ke dalam matriks yang berisi nilai-nilai vektor, perbedaan sudut kosinus dapat diketahui dari dua buah vektor, besar sudut tersebut yang mengindentifikasikan bersarnya perbedaan makna dalam teks. Rumus perhitungannya [9] adalah sebagai berikut:

$$\cos\theta = \frac{A\,B}{|A|\,|B|} = \frac{\sum_{i=1}^{n} Ai\ x\ Bi}{\sqrt{\sum_{i=1}^{n}(Ai)^2}\ x\ \sqrt{\sum_{i=1}^{n}(Bi)^2}} \qquad (4)$$

Keterangan:
A = vektor A
B = vektor B
|A| = panjang vektor A
|B| = panjang vektor V

2.4 Evaluation Measure

Evaluasi bertujuan untuk menilai performansi yang dapat dicapai oleh sistem. Evaluasi dalam tugas akhir ini digunakan untuk mengetahui apakah suatu sistem telah optimal dalam mendeteksi halaman yang terindikasi memiliki kemiripan semantik terhadap halaman yang lain. Evaluasi yang digunakan adalah *precision*, *recall*, dan *F-Measure*. *Precision* mengidentifikasikan kualitas dari klasifikasi sistem, sedangkan *recall* mengidentifikasikan kuantitas dari sistem, dan *F-Measure* merupakan pengukuran kualitas dari akurasi sebuah klasifikasi biner [10]. Rumus perhitungan evaluasinya [10] adalah sebagai berikut:

$$precision = \frac{TP}{TP+FP} \qquad (5)$$

$$recall = \frac{TP}{TP+FN} \qquad (6)$$

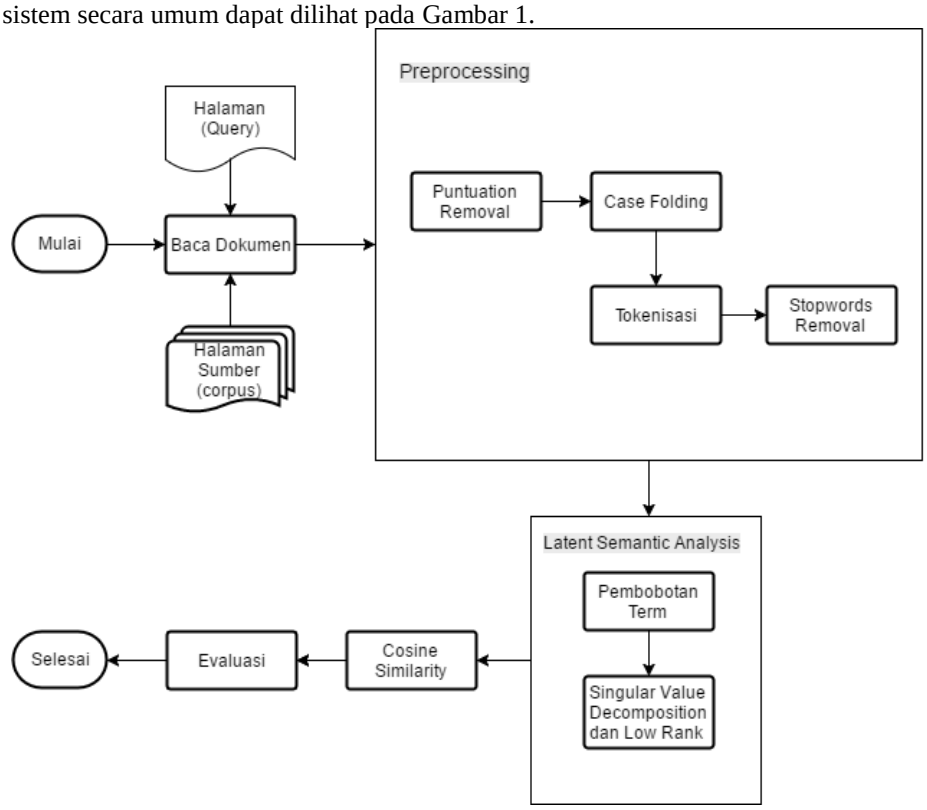
$$F_1 = 2\ x\ \frac{precision\ x\ recall}{precision+recall} \qquad (7)$$

Ketengan:
True Positive (TP) = suatu kondisi dimana sistem mendekteksi kelas positif dan faktanya pun positif
True Negative (TN) = suatu kondisi dimana sistem mendekteksi kelas negatif dan faktanya pun negatif
False Positive (FP) = suatu kondisi dimana sistem mendekteksi kelas positif namun faktanya negatif
False Negative (FN) = suatu kondisi dimana sistem mendekteksi kelas negatif namun faktanya positif

3. Sistem yang Dibangun

3.1 Gambaran Umum Sistem

Gambaran sistem secara umum dapat dilihat pada Gambar 1.



Gambar 1 Diagram alir

Berikut penjelasan dari gambaran umum sistem:

1. Sistem membaca *dataset* yang berupa pasangan-pasangan halaman yang berisi terjemah Al-Qur'an dalam bahasa Inggris. *Input* dipasangkan dari sebuah halaman sebagai kuerinya dengan kedua atau lebih halaman sumber sebagai korpusnya. *Dataset* tersedia dalam format .txt.
2. Sistem melakukan *preprocessing* terhadap data, yaitu dengan melakukan penghilangan terhadap tanda baca (*punctuation*), *case folding,* tokenisasi, *stopwords removal*.
3. Setelah *preprocessing* didapatkan data bersih yang kemudian akan diterapkan metode *latent semantic analysis.*
4. Setelah menerapkan metode, sistem melakukan perhitungan nilai kemiripan semantik dari pasangan-pasangan halaman menggunakan *cosine similarity.*
5. Sistem melalukan evaluasi dengan membaca nilai kemiripan dari hasil *cosine similarity* dan membaca nilai kemiripan antar halaman dari *gold standard*. Evaluasi yang digunakan adalah *precision, recall,* dan F-Measure.

3.2 Rancangan Sistem

3.2.1 Preprocessing

Dataset pada setiap halaman yang diambil dari situs tanzil.net belum siap digunakan, oleh karena itu dibutuhkan *preprocessing* data terlebih dahulu. Berikut ini merupakan penjelasan dari Gambar 1:

a. *Puntuation Removal,* Pada tahap ini *puntuation removal* melakukan penghilangan terhadap digit dan simbol. Tahap ini diperlukan sebagai proses awal supaya didaptkan teks yang murni berisi kata-kata agar pemrosesan selanjutnya menjadi lebih mudah, contohnya dapat dilihat pada Tabel 1.

Tabel 1 Contoh Punctuation Removal

Sebelum	Sesudah
And they say, "When is this promise, if you should be truthful?" (38).	And they say When is this promise if you should be truthful

b. *Case Folding*, Pada tahap ini teks dalam dokumen dilakukan pengelompokkan kata menjadi semua huruf kecil, contohnya dapat dilihat pada Tabel 2.

Tabel 2 Contoh Case Folding

Sebelum	Sesudah
And they say When is this promise if you should be truthful	and they say when is this promise if you should be truthful

c. Tokenisasi, Pada tahap ini teks dalam dokumen dipecah menjadi per kata ditandai dengan spasi dan tanda " |", contohnya dapat dilihat pada Tabel 3.

Tabel 3 Contoh Tokenisasi

Sebelum	Sesudah
and they say when is this promise if you should be truthful	| and | they | say | when | is | this | promise | if | you | should | be | truthful |

d. *Stopwords Removal*, Pada tahap ini teks yang mengadung kata *stopword* atau kata umum dihilangkan atau tidak diikutkan pada proses. Dari sekumpulan kata yang didapatkan dari hasil tokenisasi tidak semuanya bekontribusi untuk makna ayat, oleh karena itu perlu dilakukan *stopwords removal*. contohnya dapat dilihat pada Tabel 4.

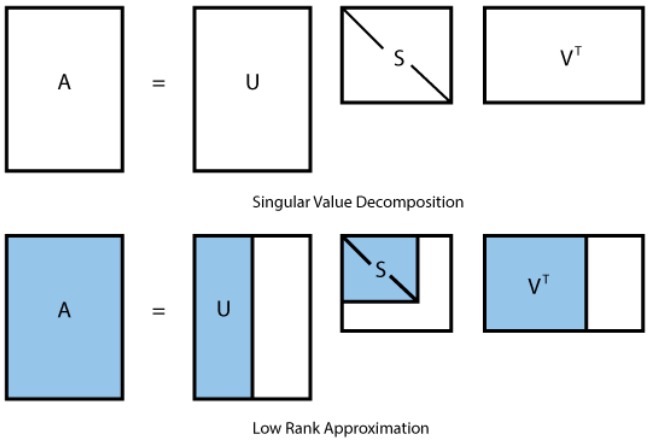
Tabel 4 Contoh Stopwords Removal

Sebelum	Sesudah
| and | they | say | when | is | this | promise | if | you | should | be | truthful |	| promise | truthful |

3.2.2 Latent Semantic Analysis

Berikut merupakan penjelasan *latent semantic analysis* pada Gambar 1:

1. Pembobotan *Term*
   Pada tahap ini teks dalam dokumen yang telah dilakukan *preprocessing* akan dihitung kemunculan suatu kata pada setiap dokumen, lalu akan dilakukan perhitungan bobot atribut atau *term* yang terdapat dalam dokumen, gabungan perhitungan bobot yang lazim digunakan adalah *Term Frequency* (TF)-*Inverse Document Frequency* (IDF). Pembobotan ini penting dilakukan untuk mengatasi masalah dalam perhitungan *similarity*, karena jika setiap kata tidak diberikan bobot maka seluruh kata tersebut akan dianggap penting. Pembobotan *term* direpresentasikan ke dalam matriks, dalam sistem pembobotan ini sangat penting karena nilai dalam matriksnya akan dipakai dalam perhitungan dekomposisi matriks dengan menggunakan *singular value decomposition*.
2. *Singular Value Decomposition* (SVD)
   Latent semantic analysis menggunakan SVD dan *low rank* dalam mendekomposisi dan mengurangi dimensi matriks hasil pembobotan. Nilai matrik U, S, dan V^T serta perkalian ketiga matriksnya didapatkan dengan bantuan referensi Bluebit.Matrix. Setelah matriks terdekomposisi, matriks A akan dikurangi dimensinya menjadi K atau dalam rumus m dimensi. Ini merupakan salah satu keunggulan dari metode LSA, dimana menjadi lebih cepat jika terjadi pengurangan dimensi. Adapun untuk illustrasi SVD dan *low rank* dapat dilihat pada Gambar 2.



Gambar 2 Illustrasi SVD dan Low Rank

Berdasarkan batasan masalah yang telah dijelaskan diatas, jika dalam korpus terdapat dua halaman maka nilai Rank K maksimal adalah satu, sedangkan jika dalam korpus terdapat sepuluh halaman maka nilai Rank K maksimal adalah sembilan. Maksimal nilai Rank K yang digunakan tidak bisa sama atau melebihi jumlah kolom, dimana kolom merupakan dokumen atau halaman. Adapun untuk mendapatkan hasil kualitas sistem yang baik tidak ada ketentuan atau cara yang pasti dalam pemilihan parameter Rank K sebagai dimensi yang digunakan, melainkan yang perlu dilakukan adalah dengan *trial and error*.

3.2.3 Cosine Similarity

Hasil dari dekomposisi pada *singular value decomposition* dan pengurangan jumlah dimensi pada *low rank approximation* akan direpresentasikan dalam bentuk matriks. Kemudian sistem melakukan perhitungan nilai kemiripan kosinus (*cosine similarity*) antara pasangan-pasangan halaman. *Cosine similarity* digunakan karena pada dasarnya matriks merupakan vektor, dimana suatu vektor dengan vektor lainnya akan membentuk suatu sudut. Dari hasil perhitungan tersebut, nilai kemiripan pasangan-pasangan halaman berkisar antara nol hingga satu, dimana semakin mendekati nol maka pasangan halaman tersebut mempunyai kemiripan rendah dan dapat diidentifikasi tidak mirip, sedangkan semakin mendekati satu maka pasangan halaman tersebut mempunyai kemiripan tinggi dan dapat diidentifikasi mirip.

3.2.4 Evaluasi

Nilai kemiripan pasangan-pasangan halaman dari hasil *cosine similarity* akan dibandingkan dengan nilai kemiripan antar halaman yang dilakukan secara semi manual yang mengacu terhadap data *gold standard* pasangan-pasangan halaman yang dapat dilihat pada Lampiran 2. Hasil nilai tersebut dari setiap pengujian yang dilakukan akan dievaluasi dengan menggunakan *precision, recall,* dan *F-Measure* untuk mengetahui kualitas dari sistem yang telah dibuat.

4. Evaluasi

4.1 Hasil Pengujian

Dari 40 pasangan halaman yang telah ditentukan, skenario pengujian pertama dilakukan dengan mempasangkan satu halaman sebagai *query* ke dua halaman sebagai *corpus*, dimana dalam *corpus* tersebut terdapat salah satu halaman yang mengandung kemiripan semantik yang tinggi, secara lengkap skenario pengujian pertama dapat dilihat pada Lampiran 2. Skenario pengujian kedua, ketiga, dan keempat dilakukan terhadap *dataset* yang sama dengan mempasangkan satu ke sepuluh halaman sebagai *corpus*, dimana hanya terdapat satu halaman dalam *corpus* yang mengandung kemiripan semantik, yang menjadi pembeda dalam pengujian tersebut adalah parameter Rank K yang digunakan sebagai jumlah dimensinya**,** adapun pengujian lengkapnya dapat dilihat pada Lampiran 3**.** Rekapitulasi hasil pengujian secara keseluruhan dapat dilihat pada Tabel 5. Pada tabel 6 merupakan perbandingan nilai kemiripan dari kelima data uji dengan menggunakan metode dan cara yang berbeda tetapi masih dalam lingkungan *dataset* yang sama. Warna abu-abu pada tabel tersebut menunjukkan data pasangan halaman yang memiliki kemiripan semantik yang tinggi sedangkan warna *orange* menunjukkan data pasangan halaman yang sama sekali tidak memiliki kemiripan semantik. Perbandingan nilai similarity antara LSA dengan *rolling hash* & n-gram secara lengkap dapat dilihat pada Lampiran 4.

Tabel 5 Rekapitulasi Hasil Pengujian.

No	Skenario Pengujian	Rank K	Precision	Recall	Accuracy	F1-Score
1	40 kali pengujian (1 ke 2 halaman)	1	1	1	100%	100%
2	1 kali pengujian (1 ke 10 halaman)	9	1	1	100%	100%
3	1 kali pengujian (1 ke 10 halaman)	7	0.25	1	70%	40%
4	1 kali pengujian (1 ke 10 halaman)	5	0.17	1	50%	29%

Tabel 6 Perbandingan Similarity

No	Halaman (Query)	Halaman Sumber (Corpus)	Similarity LSA	Similarity Rolling Hash & N-Gram	Similarity Manual	Similarity Gold Standard
1	Halaman 383	Halaman 214	0.95	0.52	0.94	0.79
		Halaman 551	0.31	0.10	0.00	0.00
2	Halaman 545	Halaman 551	1.00	0.55	1.00	0.83
		Halaman 564	0.04	0.09	0.00	0.00
3	Halaman 588	Halaman 564	0.96	0.56	0.93	0.78
		Halaman 265	0.28	0.08	0.00	0.00
4	Halaman 522	Halaman 265	0.97	0.61	0.92	0.76
		Halaman 537	0.24	0.11	0.00	0.00
5	Halaman 568	Halaman 537	0.96	0.57	0.95	0.72
		Halaman 367	0.28	0.11	0.00	0.00

4.2 Analisis Hasil Pengujian

Hasil skenario pengujian pertama pada Tabel 5 menunjukkan bahwa dari 40 pasangan halaman yang diujikan dengan menggunakan parameter Rank K=1 tidak terjadi kesalahan dalam pengidentifikasian kemiripan semantik, pengujian tersebut mendapatkan nilai akurasi sistem dan *F-Measure* sangat baik yaitu 100%. Sedangkan pada skenario pengujian kedua, ketiga, dan keempat terhadap *dataset* yang sama menunjukkan hasil terbaik jatuh pada parameter Rank K=9 dengan nilai akurasi dan *F-Measure* yaitu 100%. Pada skenario pengujian pertama dan kedua hasil yang didapatkannya optimal karena parameter Rank K yang digunakan sebagai jumlah dimensinya maksimum yaitu total jumlah halaman dalam korpus dikurangi satu, sehingga matriks dari hasil *low rank approximation* sesuai dengan matriks dekomposisi SVD dan matriks asal yaitu matriks hasil pembobotan. Dalam skenario pengujian ketiga dan keempat mendapatkan hasil yang tidak optimal dengan nilai *F-Measure* 40% dan 29%, hal ini terjadi akibat kecenderungan nilai *similarity* yang besar dengan menggunakan parameter Rank K yang minimum, yang mengakibatkan terjadinya *missed detection* atau kesalahan dalam pengidentifikasian nilai *similarity*. Inilah yang menyebabkan akurasinya kedua pengujian tersebut cukup tinggi tetapi tidak bernilai karena tidak bisa mendeteksi sesuatu yang kita cari dan inginkan yaitu pasangan halaman yang memiliki kemiripan semantik yang tinggi.

Hasil pengujian pada Tabel 6 di atas menunjukkan bahwa metode *latent semantic analysis* cukup baik dalam mendeteksi kemiripan pada pasangan halaman yang mengandung kesamaan semantik yang tinggi, hal dibuktikan dengan nilai setiap pasangan pada warna abu-abu tersebut diatas nilai kemiripan dengan metode *rolling hash* & n-gram dan hampir mendekati nilai kemiripan yang dilakukan secara manual. Nilai kemiripan dengan menggunakan metode *rolling hash* & N-Gram diambil dari hasil pengujian penelitian Winda Eka Samodra. Sedangkan nilai kemiripan dengan menggunakan perhitungan yang dilakukan secara manual terhadap *dataset* pasangan ayat yang telah ditentukan dicari dan didapat dengan menggunakan *dot product* dimana perhitungan tersebut membutuhkan waktu yang sangat lama karena dilakukan secara manual, untuk pasangan yang tidak mirip pada warna *orange* dengan nilai kemiripan nol dari perhitungan manual dan gold standard, hal ini terjadi karena mengacu kepada pasangan ayat yang tidak sama sekali ada mirip maka perhitungan untuk nilai kemiripan pasangan ayat tersebut

adalah total ayat yang mirip dibagi dengan total ayat dalam kedua halaman, maka nilainya adalah nol. Untuk perhitungan semi manual antar halaman dapat dilihat pada Lampiran 2.

5. Kesimpulan dan Saran

Dari pengujian-pengujian yang dilakukan, secara kesuluruhan penerapan metode *latent semantic* analysis sudah efektif dibandingkan dengan metode *rolling hash* dan n-gram dalam mendeteksi pasangan-pasangan halaman yang memiliki nilai kemiripan yang tinggi, terlihat dari nilai kemiripan semantik yang hampir mendekati angka satu. Hal ini sudah baik, karena LSA merupakan *unsupervised learning* yang tidak menggunakan pengetahuan tambahan seperti *wordnet* dan *thesaurus.* Untuk pemilihan parameter Rank K yang digunakan sebagai penentuan jumlah dimensi dalam mengukur nilai kemiripan sangat berpengaruh terhadap kualitas sistem, semakin kecil dimensi yang digunakan maka nilai *similarity* akan semakin besar dan beragam serta semakin tidak relevan dengan *dataset* pasangan-pasangan halaman yang telah ditentukan sebelumnya.

Untuk tugas akhir atau penelitian yang lebih baik dari sistem ini kedepannya diharapkan terdapat beberapa pengembangan, seperti memperluas *dataset* tugas akhir ini, membuat kamus thesaurus yang bersini hiponim, hipernim, dan sinonim dari terjemah Al-Qur'an berbahasa Inggris dimana kamus tersebut digunakan sebagai *knowledge* tambahan untuk LSA, mencoba dengan menggunakan metode lain yang dapat mengembalikan hasil berupa kalimat bukan lagi angka dari bagian-bagian atau ayat terjemah Al-Qur'an mana yang memiliki kemiripan semantik.

Daftar Pustaka

[1] K. a. o. Shams, *Plagiarism detection using semantic analysis*, Dhaka, Bangladesh: BRAC University, 2010.

[2] P. Nakov, "Latent semantic analysis of textual data," 2000.

[3] M. Aisyah, *Sistem Pendeteksi Plagiarisme Menggunakan Pendekatan Text Alignment Menggunakan Sentence Similarity*, Open LIbrary Telkom University, 2017.

[4] A. Wahab Khallaf, *Ilmi Ushul Al-Figh*, 1956.

[5] T. K. a. F. P. W. a. L. D. Landauer, "An introduction to latent semantic analysis," *Discourse processes*, Vols. 2-3, pp. 259-284, 1998.

[6] A. Rusydiana, "TEXT MINING CENTER," 01 September 2016. [Online]. Available: http://textmining-center.blogspot.com/2016/09/4-tahap-proses-text-mining.html. [Accessed 14 Juni 2018].

[7] G. a. B. C. Salton, "Term-weighting approaches in automatic text retrieval," *Information processing \& management*, vol. 24, pp. 513-523, 1988.

[8] S. a. D. S. T. a. F. G. W. a. L. T. K. a. H. R. Deerwester, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, pp. 391-407, 1990.

[9] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, 2008, pp. 49-56.

[10] D. L. a. D. D. Olson, Advanced data mining techniques, Springer Science \& Business Media, 2008.

[11] J. Tawisa, *Sistem Pendeteksi Plagiarisme Pada Dokumen Teks Bahasa Indonesia Dengan Menggunakan Metode Latent Semantic Analysis*, open library telkom university, 2012.