

# Pencarian Potongan Ayat Al-Qur'an dengan Perbedaan Bunyi pada Tanda Berhenti Berdasarkan Kemiripan Fonetis

Naufal Rasyad<sup>\*†</sup>, Moch. Arif Bijaksana<sup>‡§</sup>, Kemas Muslim Lhaksmana<sup>\*¶</sup>

<sup>\*</sup>Program Studi Sarjana Informatika, Fakultas Informatika, Universitas Telkom  
Jl. Telekomunikasi No. 01 Dayeuhkolot, Bandung, Indonesia

<sup>\*</sup>naufalrasyd@students.telkomuniversity.ac.id

<sup>¶</sup>kemasmuslim@telkomuniversity.ac.id

<sup>§</sup>arifbijaksana@telkomuniversity.ac.id

**Ringkasan**—Al-Quran merupakan kitab suci utama bagi umat Islam yang ditulis menggunakan bahasa Arab. Seiring dengan perkembangan teknologi, telah dikembangkan sistem pencarian ayat Al-Qur'an berdasarkan kemiripan fonetis salah satunya adalah Lafzi. Namun untuk menangani perbedaan bunyi pada tanda berhenti di pertengahan ayat, sistem Lafzi belum bisa menanganinya dengan baik. Maka dari itu, dibutuhkan sistem yang dapat membantu pengguna dalam melakukan pencarian ayat Al-Quran, terutama untuk perbedaan bunyi pada tanda berhenti sehingga pencarian bisa menemukan kata yang berbeda pengucapan pada tanda berhenti. Berdasarkan permasalahan tersebut, dari sistem Lafzi, dilakukan pengembangan supaya dapat melakukan pencarian yang bisa menangani perbedaan bunyi pada tanda berhenti. Digunakan pengindeksan trigram untuk memperkirakan kecocokan string antara kueri dengan transliterasi ayat Al-Qur'an serta dibuat aturan pada input dengan huruf akhir 'T' menjadi 'H'. Sistem yang sudah ada mendapatkan nilai *recall* sebesar 81% dan nilai *MAP* sebesar 65%. Sedangkan hasil dari penelitian ini diperoleh nilai *recall* sebesar 100% dan nilai *MAP* sebesar 84%.

**Kata Kunci**—Al-Qur'an, sistem pencarian, kemiripan fonetis, trigram

## I. PENDAHULUAN

Al-Qur'an merupakan kitab suci utama yang menjadi pedoman bagi seluruh umat Islam di seluruh dunia. Al-Qur'an diturunkan dalam Bahasa Arab dan ditulis dengan aksara Arab (*hijaiyyah*). Secara statistik, Al-Qur'an terdiri atas 114 surat, 6236 ayat, dan 77.845 kata [1]. Untuk jumlah yang cukup banyak tersebut, pencarian akan sulit dilakukan terutama untuk pencarian secara manual. Maka dapat digunakan alat bantu yaitu komputer untuk memudahkan melakukan pencarian potongan ayat dalam Al-Qur'an

Penelitian tentang pengembangan sebuah sistem yang dapat membantu memudahkan pencarian ayat Al-Qur'an sudah dilakukan sejak lama [2], salah satunya yaitu menggunakan metode pencarian secara sekuensial. Saat ini juga telah banyak dikembangkan aplikasi perangkat lunak untuk mempelajari Al-Qur'an. Bahkan telah dikembangkan

juga sistem pencarian potongan ayat berdasarkan kemiripan fonetis. Salah satu contoh yang berbasis *web* ialah Lafzi [3]

Pada aplikasi perangkat lunak Al-Qur'an yang telah disebutkan, dapat dilakukan pencarian ayat yang mengharuskan pengguna untuk memasukkan kata kunci pencarian berupa potongan ayat dalam aksara latin. Namun pada aplikasi tersebut belum bisa menangani pencarian untuk perubahan bunyi pada tanda berhenti secara maksimal terutama untuk kueri pendek. Beberapa kata yang diikuti waqaf atau kata pada akhir ayat biasanya berbeda pelafalannya terutama untuk kata yang huruf akhirnya adalah ta marbuttoh (ة) yaitu mengubah huruf *ta* menjadi *ha* seperti pada contoh lafadz الْمُضْعَفَةُ atau dalam tulisan latin yaitu al-mudhghota. Jika lafadz tersebut diwaqafkan, maka cara membacanya menjadi الْمُضْعَفَهَا atau dalam tulisan latin yaitu al-mudhghoh.

## II. STUDI TERKAIT

### A. Cara Mewaqafkan Bacaan

Ada beberapa aturan untuk membaca kalimat yang diwaqafkan [4] diantaranya

- 1) Jika terdapat huruf terakhir berharakat sukun atau mati, maka membacanya tidak ada perubahan sama sekali. Contoh lafadz ( أَعْمَالَهُمْ ) tetap dibaca "a'malahum"
- 2) Jika huruf terakhir berharakat fathah, kasrah, dan dhammah, maka huruf terakhir tersebut harus dibaca sukun atau mati. Contoh lafadz ( الْبَلَدِ ) ditulis "albalaadi", dan dibaca menjadi "albalaad"
- 3) Jika huruf terakhirnya ta marbuttoh (ة), baik letaknya di tengah ataupun di akhir kalimat, maka membacanya yaitu menggantikannya dengan huruf ha (ه) yang dibaca sukun.
- 4) Jika huruf terakhirnya berharakat atau hidup, tetapi sebelumnya didahului dengan huruf sukun atau mati, maka dua huruf tersebut dibaca mati dan huruf yang

terakhir dibaca dengan suara pelan. Contoh pada lafadz ( بِالْهَزْلِ ) ditulis "bilhazli", dibaca menjadi "bilhazl"

- 5) Jika di akhir kalimat didahului bacaan mad asli atau mad layyin (bacaan mad yang huruf sebelumnya berharakat fathah), maka cara membacanya yaitu dengan mematikan huruf yang terletak di akhir kalimat tersebut dengan dipanjangkan sedikit. Contoh pada lafadz ( الْحَكِيمُ ) ditulis "alhakiimu", dibaca menjadi "alhakiim"
- 6) Ketika berhenti di akhir kalimat, tetapi huruf akhirnya berharakat fathah tanwin, maka hanya harakat fathahnya saja yang dibaca sebanyak dua harakat. Contoh pada lafadz ( أَفْوَاجًا ) ditulis "afwajaan", dan dibaca menjadi "afwajaa"
- 7) Jika huruf terakhir bertasydid, maka membacanya yaitu mematikan tanpa menghilangkan fungsi tasydidnya. Contoh pada lafadz ( خَلَقَهُنَّ ) ditulis "kholaqohunna", dibaca menjadi "kholaqohunn"
- 8) Hamzah di akhir kata yang ditulis di atas waw ( ءُ ) dimatikan bila waqaf, dan dibaca pendek bila washal. Contoh pada lafadz ( يَتَفَيَّؤُا ) ditulis "yatafayyaw", dibaca menjadi "yatafayya"

**B. Pencocokan String Berdasarkan Kemiripan Fonetis**

Fonetik atau sistem bunyi bahasa merupakan suatu aspek bahasa yang harus dipelajari pertama kali ketika suatu bahasa diajarkan, karena kata maupun kalimat dari suatu bahasa tersusun dari bunyi-bunyi tersebut [5].

Pencocokan string berdasarkan kemiripan fonetis merupakan pencocokan string yang dilakukan berdasarkan kemiripan dari segi pengucapannya meskipun terdapat perbedaan penulisan pada string yang dibandingkan tersebut [6]. Contoh kata "eyes" dengan kata "ice" merupakan tulisan yang berbeda dengan makna yang berbeda, namun dari segi pengucapan, kedua string tersebut mirip, sehingga kedua string tersebut dianggap cocok.

**C. N-Gram**

N-gram merupakan potongan karakter sebanyak N yang diambil dari suatu string yang lebih panjang [7]. Untuk membentuk suatu N-gram yang utuh biasanya ditambahkan penanda pada awal dan akhir string. Salah satu penanda untuk awal dan akhir kata bisa menggunakan karakter *underscore* ("\_"). Dengan demikian, suatu string "WORD" akan terdiri dari N-gram berikut :

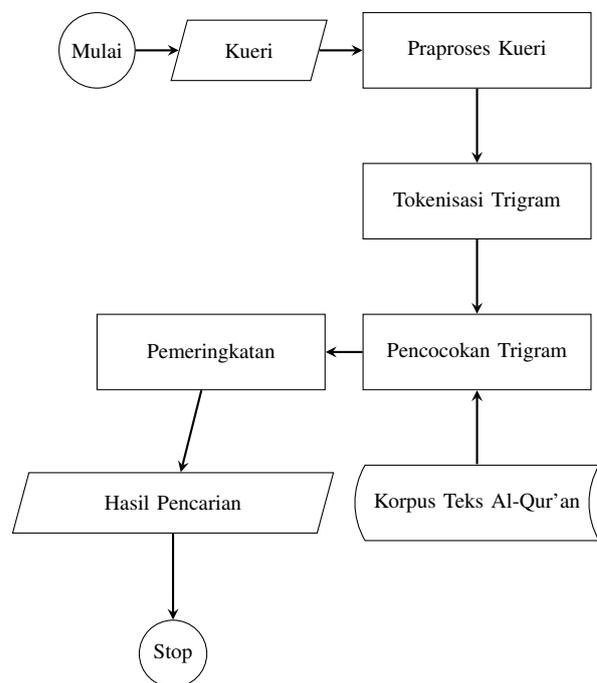
- bigram : \_W, WO, OR, RD, D\_
- trigram : \_WO, WOR, ORD, RD\_, D\_\_
- quadgram : \_WOR, WORD, ORD\_, RD\_\_, D\_\_\_

Secara umum, string dengan panjang k yang memiliki penanda, akan memiliki panjang k+1 bigram, k+1 trigram, k+1 quadgram, dan seterusnya. Manfaat utama yang bisa didapat dengan menggunakan N-gram yaitu berdasarkan

karakteristik N-gram itu sendiri sebagai bagian dari string. Karena setiap string [8] terurai menjadi bagian-bagian kecil, setiap kesalahan cenderung hanya mempengaruhi beberapa bagian saja.

**III. SISTEM YANG DIBANGUN**

Alur sistem secara keseluruhan digambarkan pada gambar 1. Pertama, dilakukan proses tokenisasi per kata terhadap teks kueri, untuk memudahkan dalam pengkodean fonetis. Hasil pemisahan perkata kemudian akan diubah menjadi kode fonetis. Kode fonetis yang ada dalam korpus dan kueri masukan kemudian diukur nilai kesamaannya. Jika kedua kode fonetis memiliki nilai kesamaan yang tinggi, maka antara korpus dan kueri tersebut dinyatakan cocok dan kemudian dikeluarkan sebagai hasil.



Gambar 1. alur sistem pencarian ayat Al-Qur'an berdasarkan kemiripan fonetis

**A. Pengkodean Fonetis Kueri**

Setiap kueri yang dimasukan merupakan teks latin yang harus dilakukan praproses salah satunya yaitu diubah kedalam kode fonetis supaya dapat dicocokkan dengan hasil pengkodean fonetis pada teks Al-Qur'an. Teks kueri terlebih dahulu diubah kedalam huruf kapital, kemudian spasi ganda dihapus, dan karakter selain huruf serta beberapa tanda baca (tanda kutip tunggal, apostrof, dan tanda hubung) dihapus. Beberapa prosedur praproses untuk teks latin dijelaskan pada langkah-langkah berikut [3]:

1) Substitusi Huruf Vokal

Pada aksara Arab, hanya terdapat tiga jenis huruf vokal yaitu A, I dan U saja [9]. Sedangkan pada aksara latin terdapat vokal lainnya yaitu E dan O. Maka

dari itu, vokal harus disubstitusi. Huruf vokal O, misalnya pada kata "ARROHMAN" diganti menjadi A sehingga kata tersebut menjadi "ARRAHMAN". Untuk huruf vokal E, misalnya pada kata "MUTTAQIEN" diganti menjadi I sehingga kata tersebut menjadi "MUTTAQIIN".

- 2) Penghilangan Karakter Ganda  
Huruf-huruf konsonan atau vokal sama yang berdampingan dijadikan satu huruf saja. Tahap ini setara dengan penghilangan tasydid untuk huruf konsonan dan penghilangan bacaan panjang untuk huruf vokal.
- 3) Substitusi Diftong  
Huruf vokal yang membentuk diftong disesuaikan dengan pengkodean teks Al-Qur'an yang sudah dibuat. Diftong AI diubah menjadi AY, sedangkan AU diubah menjadi AW.
- 4) Substitusi Bacaan Idgham  
Aturan pada tajwid idgham yaitu pada saat huruf nun mati ( ن ) bertemu dengan huruf-huruf idgham, diantaranya ya, nun, mim, wau, lam, dan ra ( ي , ن , و , ل , dan ر ). Pada kasus demikian, huruf N dihilangkan apabila bertemu dengan huruf idgham.
- 5) Substitusi Bacaan Iqlab  
Aturan pada tajwid iqlab yaitu pada saat huruf nun mati ( ن ) bertemu dengan huruf ba ( ب ) sehingga huruf NB akan berubah bunyi menjadi MB.
- 6) Substitusi Bacaan Ikhfa  
Aturan pada tajwid idgham yaitu pada saat huruf nun mati ( ن ) bertemu dengan huruf-huruf ikhfa. Bacaan ikhfa yaitu bunyi huruf N yang dibaca samar-samar bila diikuti huruf konsonan tertentu, dan terkadang dituliskan dengan bunyi NG. Misalnya pada kata "ANGFUSINA". Karakter G dihilangkan supaya setara dengan pengkodean teks Al-Qur'an.
- 7) Pemadanan Huruf  
Pemadanan huruf dari teks latin perlu mempertimbangkan huruf arab yang direpresentasikan kedalam lebih dari satu huruf latin konsonan. Misalkan huruf tsa dituliskan TS, dan huruf dza dituliskan DZ [10]. Oleh karena itu, dibuat aturan pemadanan huruf supaya bisa disesuaikan dengan pengkodean teks Al-Qur'an. Pemadanan huruf dituliskan dalam tabel I.
- 8) Penghilangan Spasi  
Seluruh spasi harus dihilangkan supaya setara dengan hasil pada prosedur pengkodean teks Al-Qur'an.
- 9) Penambahan aturan untuk tanda berhenti Korpus teks Al-Qur'an memiliki beberapa ciri salah satu diantaranya adalah tidak ada tanda berhenti di tengah ayat. Maka dari itu, dibuat sebuah aturan supaya bisa menemukan kata yang diikuti oleh tanda berhenti terutama untuk kata dengan akhiran ta marbutthoh ( ة ) yang dilafalkan menjadi ha.

Tabel I  
ATURAN PEMADANAN AKSARA LATIN

Aksara Latin	Padanan
SH, TS, SY	S
KH, CH	H
ZH, DZ	Z
DH	D
TH	T
GH	G
NG ('ain)	X
F, V, P	F
Q, K	K
J, Z	Z
' , ' (apostrof)	X

Sebagai contoh, kata "ghisyawatun" ( غِشْوَةٌ ) jika berada di akhir ayat atau tanda berhenti maka akan menjadi "ghisyawah". Supaya sistem dapat menampilkan hasil untuk kueri "ghisyawah", maka dibuat aturan atau rule sebagai berikut :

- a) Setiap kueri yang berakhiran huruf 'h' akan diubah menjadi huruf 't'. Kata "ghisyawah" akan diproses sebagai "ghisyawat"
- b) Setiap kueri yang berakhiran huruf 'h' meskipun tidak terletak pada tanda berhenti akan tetap ditampilkan sebagai output dengan akhiran huruf 'h'. Sebagai contoh, untuk kueri "alhamdulillah" akan tetap menampilkan hasil sesuai dengan kueri tersebut.

Prosedur pengkodean fonetis pada kueri dilakukan secara berurutan. Contoh penerapan prosedur pengkodean fonetis ini tercantum pada tabel II untuk kueri "maliki yaumiddiin" setelah dirubah kedalam huruf kapital. Pengkodean fonetis kueri ini dilakukan supaya menghasilkan string yang setara dengan pengkodean pada teks Al-Qur'an atau korpus.

Tabel II  
CONTOH PENGKODEAN FONETIS KUERI

Langkah	Hasil	Deskripsi
1	MALIKI YAUMIDDIIN	tidak ada substitusi vokal E dan O
2	MALIKI YAUMIDIN	karakter ganda dihilangkan
3	MALIKI YAWMIDIN	substitusi diftong AU dengan AW
4	MALIKI YAWMIDIN	tidak ada substitusi bacaan idgham
5	MALIKI YAWMIDIN	tidak ada substitusi bacaan iqlab
6	MALIKI YAWMIDIN	tidak ada substitusi bacaan ikhfa
7	MALIKI YAWMIDIN	tidak ada pemadanan huruf
8	MALIKIYAWMIDIN	spasi dihilangkan

### B. Tokenisasi Trigram

Setiap string yang dihasilkan dari hasil kode fonetis Al-Qur'an maupun kueri, dilakukan tokenisasi untuk mengambil trigram. Pembentukan trigram tidak membutuhkan

penanda awal atau penanda akhir string karena kueri pasti merupakan bagian atau substring dari teks Al-Qur'an [3]. Untuk hasil tokenisasi dari contoh pada tabel II, trigram yang didapatkan yaitu {MAL, ALI, LIK, IKI, KIY, IYA, YAW, AWM, WMI, MID, IDI, DIN}, sesuai dengan string "maliki yaumiddiin" yang telah diubah kedalam kode fonetis.

C. Pencocokan Trigram

Trigram yang dihasilkan dari kueri kemudian dibandingkan dengan trigram yang terdapat pada indeks. Trigram yang ada pada indeks tersebut telah tersedia pada sistem Lafzi. Trigram dari dokumen dan trigram dari kueri dibandingkan kemudian jumlah trigram yang sama dihitung [3]. Perhitungan dilakukan dengan memanfaatkan informasi yang tersimpan dalam indeks. Trigram yang memiliki jumlah sama paling banyak kemudian dikalikan dengan nilai ambang batas dimana hasilnya digunakan sebagai minimal skor. Kecocokan trigram yang memiliki jumlah lebih dari sama dengan minimal skor akan dimasukkan sebagai kandidat output.

D. Pemeringkatan Dokumen

Untuk pemeringkatan dokumen, dibuat mekanisme pemberian skor sedemikian sehingga dokumen yang posisi kemunculannya terurut dan rapat memiliki skor yang lebih tinggi dibandingkan dengan trigram yang posisi kemunculannya teracak dan terpisah-pisah. Untuk mencari kandidat hasil, digunakan *longest increasing subsequence* (LIS) [11] yang diterapkan pada posisi kemunculan trigram untuk memberi skor keterurutan. LIS dari sebuah sekuens S adalah subsekuens monoton naik dari S dengan panjang maksimum [12]. Index yang memiliki keterurutan paling panjang, maka akan dijadikan kandidat hasil. Misalnya terdapat suatu baris ayat dengan index seperti berikut 4[11, 12, 13, 14, 15]. Index tersebut memiliki skor LIS dengan panjang 5. Kemudian dihitung nilai kerapatannya yaitu dari rata-rata dari invers dari selisih antar elemen yang berdampingan. Nilai kerapatan *c* dari suatu LIS sepanjang *n* dengan elemen {*t*<sub>1</sub>, *t*<sub>2</sub>, ..., *t*<sub>*n*</sub>} [3] dirumuskan pada persamaan 1.

$$c = \frac{1}{n-1} \sum_{i=1}^{n-1} \frac{1}{(t_{i+1}) - t_i} \tag{1}$$

Dari contoh diatas, didapatkan nilai *n* adalah 5, dan nilai *t* adalah elemen dari index tersebut. Hasil dari rumus 1 diatas kemudian dikalikan dengan jumlah trigram yang sesuai antara trigram kueri dengan trigram teks Al-Qur'an untuk mendapatkan skor keterurutan. Skor dengan nilai paling tinggi akan diletakkan paling atas pada output.

IV. EVALUASI

Evaluasi dilakukan untuk mendapatkan hasil akurasi sistem berdasarkan nilai *recall* dan nilai *Mean Average Precision* (*MAP*).

*Recall* merupakan jumlah dokumen relevan yang diambil dibagi dengan sejumlah dokumen yang relevan dalam korpus [13]. Skor dari *recall* menentukan tingkat keberhasilan sistem dalam melakukan pencarian potongan ayat. Skor maksimum untuk *recall* adalah 1 dan skor minimumnya adalah 0. Jika nilai *recall* sistem adalah 1 artinya sistem berhasil melakukan pencarian sesuai dengan dokumen relevan dalam korpus. Dokumen relevan pada korpus disebut sebagai standar emas. Untuk menghitung skor *recall*, digunakan rumus 2.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Dimana TP adalah jumlah output yang sesuai dengan standar emas, sedangkan FP adalah jumlah standar emas yang tidak dikeluarkan oleh sistem.

*Mean Average Precision* atau *MAP* merupakan skor yang didapatkan dengan menghitung rata-rata AP atau *average precision*. AP dihitung untuk setiap dokumen relevan yang diambil [13]. AP juga memperhitungkan posisi dokumen yang relevan dalam daftar hasil. Contoh penghitungan AP tertera pada tabel III

Tabel III  
TABEL CONTOH UNTUK MENGHITUNG AP

Output ke-	Benar/Salah	Presisi	Keterangan
1	Benar	1/1	Presisi pada urutan 1
2	Salah	-	Tidak presisi
3	Salah	-	Tidak presisi
4	Benar	2/4	Presisi pada urutan 4
5	Benar	3/5	Presisi pada urutan 5

Dari tabel contoh diatas, maka didapatkan skor AP yaitu  $\frac{\frac{1}{2} + \frac{2}{4} + \frac{3}{5}}{3} = 0.53$

A. Hasil Pengujian

Tabel IV menunjukkan hasil pengujian pada penelitian ini. Pengujian juga dibandingkan dengan penelitian sebelumnya yaitu aplikasi Lafzi dengan menggunakan test set yang sama. Test set dijelaskan pada analisis hasil pengujian. Tabel V menampilkan nilai *recall* dan nilai *MAP* sebagai percobaan untuk mendapatkan nilai ambang batas paling optimal mulai dari 0.7 sampai 1.0 menggunakan kumpulan data yang sama.

Tabel IV  
TABEL HASIL PENGUJIAN MENGGUNAKAN EVALUASI *recall* DAN *MAP* DENGAN NILAI AMBANG BATAS 0.7

Sistem	Dengan Tanda Berhenti (skenario pertama)		Bacaan Aslinya (skenario kedua)	
	Recall	MAP	Recall	MAP
Lafzi+ (this work)	100%	84%	85%	58%
Lafzi [3]	81%	65%	85%	58%

Tabel V  
TABEL HASIL PERCOBAAN UNTUK MENCARI NILAI AMBANG BATAS YANG OPTIMAL

Nilai Ambang Batas	Recall	MAP
1.0	98%	84%
0.9	98%	84%
0.8	98%	84%
0.7	100%	84%

B. Analisis Hasil Pengujian

Pada skenario pertama, pengujian dilakukan dengan menggunakan 50 kueri yang merupakan kata pendek dan kalimat yang diikuti dengan tanda berhenti setelahnya. Hasil pengujian pada skenario pertama didapatkan nilai *recall* sebesar 100% dan *MAP* sebesar 84%. Lebih tinggi dari sistem Lafzi yang hanya mendapatkan mendapatkan nilai *recall* sebesar 81% dan *MAP* sebesar 65% seperti yang tertera pada tabel IV. Hal ini dikarenakan sistem lafzi belum bisa menangani pencarian pada tanda berhenti secara maksimal terutama perubahan bunyi pada *ta marbuttoh* (ة) yang diikuti tanda berhenti. Contoh kueri untuk skenario pertama dapat dilihat pada tabel VI.

Tabel VI  
CONTOH KUERI UNTUK SKENARIO PERTAMA

No	Teks Arab	Teks Latin	Kemunculan pada Al-Qur'an (surah:ayat)
1	غَشْوَةٌ	gisyawah	2:7
2	خَلِيفَةٌ	kholifah	2:30
3	قَشْوَةٌ	qoswah	2:74
4	وَحْجَارَةٌ	wal hijaroh	2:24, 66:6
5	مُطَهَّرَةٌ	mutohharoh	2:25, 4:57, 80:14, 98:2

Dengan diterapkannya aturan untuk menangani pencarian kata dan kalimat yang diikuti tanda berhenti, sistem ini dapat menghasilkan nilai *recall* dan nilai *MAP* yang lebih baik daripada Lafzi aslinya. Sebagian besar kueri yang mengandung *ta marbuttoh* (ة) yang diikuti tanda berhenti pada pertengahan ayat tidak ditemukan pada sistem Lafzi. Kata kunci "ghisyawah" (غَشْوَةٌ) seharusnya ditemukan pada surah Al-Baqoroh ayat 7. Perbedaan antara trigram yang dihasilkan kueri dengan trigram pada teks Al-Qur'an menyebabkan kueri tersebut tidak ditemukan pada sistem Lafzi. Setelah diterapkan aturan untuk tanda berhenti pada kueri, ada kenaikan terhadap nilai *recall* yaitu sebesar 19% dan nilai *MAP* sebesar 19%.

Terdapat beberapa hasil pencarian yang kurang relevan dengan kueri. Hal ini disebabkan karena skor pencocokan antara trigram kueri dengan trigram teks Al-Qur'an berada diatas nilai ambang batas. Dalam penelitian ini, nilai ambang batas yang ditentukan yaitu sebesar 0.7, karena nilai tersebut dianggap nilai paling optimal dengan menghasilkan nilai *recall* sempurna sebesar 100% sesuai dengan hasil percobaan pada tabel V.

Perbedaan nilai ambang batas antara 0.7 dan 0.8 yang menyebabkan perubahan nilai *recall* berada pada kueri "khofiyah". Untuk kueri tersebut terdapat hasil pencarian yang menghasilkan skor kerapatan yang berada pada rentang 0.7 dan 0.8 sehingga menyebabkan hasil tidak ditemukan pada nilai ambang batas yang lebih besar dari 0.8.

Pada skenario kedua, sistem diuji dengan menggunakan bacaan aslinya. Bacaan aslinya yaitu dimana tulisan arab tersebut dibaca sesuai dengan aslinya. Misalkan ( غَشْوَةٌ ) tetap dibaca "ghisyawatun". Kasus ini diuji sebagai perbandingan apakah dengan menerapkan aturan tanda berhenti dapat mengubah hasil pencarian yang seharusnya. Pengujian ini dilakukan menggunakan 100 kueri dimana 50 diantaranya adalah bacaan asli dari kueri yang diuji sebelumnya, dan 50 lainnya adalah kueri yang ada pada dokumentasi lafzi [3]. Kueri tambahan yang diperoleh dari dokumentasi Lafzi mengandung beberapa variasi penulisan seperti untuk kueri "asshobiriin", "ash shoobiriin", "ashshabiriina", "ash-shaabiriin". Contoh kueri untuk skenario kedua dapat dilihat pada tabel VII.

Tabel VII  
CONTOH KUERI UNTUK SKENARIO KEDUA

No	Teks Arab	Teks Latin	Kemunculan pada Al-Qur'an (surah:ayat)
1	غَشْوَةٌ	gisyawatun	2:7
2	خَلِيفَةٌ	kholifatan	2:30, 38:26
3	وَحْجَارَةٌ	wal hijarotu	2:24, 66:6
3	الصَّبْرِينَ	asshobiriin	21:85, 47:31, 16:126, 3:17, 2:153, 3:142, 22:35, 8:46, 2:155, 3:146, 37:102, 8:66, 33:35, 2:177, 2:249
5	يَسْتَطِيعُونَ	yastathi'un	26:211, 36:75, 7:192, 36:50, 68:42, 25:9, 17:48, 21:40, 18:101, 7:197, 21:43, 4:98, 16:73, 11:20, 2:273

Hasil nilai *recall* dan nilai *MAP* yang diperoleh masing-masing mendapatkan nilai yang sama yaitu 85% dan 58%. Hasil tersebut menunjukkan bahwa dengan ditamharkannya rule untuk tanda berhenti, tidak mempengaruhi pencarian untuk kueri seharusnya.

Nilai *MAP* sebesar 58% dikarenakan hasil dari kueri tambahan tersebut banyak yang tidak terurut sesuai dengan standar emas yang telah ditetapkan. Misalnya untuk pencarian dengan standar emas [2:173, 2:182, 2:192, 2:199, 2:226, 3:89, 5:3, 5:39, 8:69, 9:5, 9:99, 9:102, 16:115, 24:5, 24:62, 49:14, 58:12, 60:12, 64:14, 73:20], namun sistem mengeluarkan hasil [60:12, 24:62, 2:199, 8:69, 73:20, 9:102, 2:192, 9:5, 2:173, 2:182, 3:89, 5:39, 24:5, 2:226, 5:3, 9:99, 16:115, 49:14, 58:12, 64:14, 16:18, 5:34, 24:33]. Meskipun seluruh hasil yang ditemukan sesuai dengan standar emas dan nilai *recall* nya adalah 1, tapi nilai *Average Precision* nya menjadi 0 karena keterurutannya tidak ada yang sesuai dengan standar emas. Pada pemeringkatan dokumen, hasil dengan skor keterurutan paling tinggi akan

dikeluarkan paling atas.

#### V. KESIMPULAN

Berdasarkan analisis hasil pengujian, maka dapat ditarik kesimpulan bahwa sistem Lafzi yang telah dikembangkan rata-rata dapat meningkatkan nilai *recall* dan nilai *MAP* untuk pencarian potongan ayat yang diikuti tanda berhenti. Untuk menangani pencarian potongan ayat yang diikuti tanda berhenti, Sistem yang dikembangkan mendapatkan nilai *recall* dan nilai *MAP* masing-masing sebesar 100% dan 84%. Hasil ini lebih baik daripada sistem Lafzi yang hanya mendapatkan nilai *recall* dan nilai *MAP* masing-masing sebesar 81% dan 65%.

#### PUSTAKA

- [1] B. Hammo, A. Sleit, and M. El-Haj, "Effectiveness of query expansion in searching the holy quran," 2007.
- [2] Z. Yahya, M. T. Abdullah, A. Azman, and R. A. Kadir, "Query translation using concepts similarity based on quran ontology for cross-language information retrieval," *Journal of Computer Science*, vol. 9, no. 7, p. 889, 2013.
- [3] M. A. Istiadi, "Sistem Pencarian Ayat Al-Quran Berbasis Kemiripan Fonetis," Master's thesis, Final Project, IPB, Bogor, 2012.
- [4] Nadiazhr, "13 macam tanda waqaf yang wajib kamu ketahui," accessed: 2019-06-22. [Online]. Available: <https://tabbayun.com/tanda-waqaf/>
- [5] A. Lundeto, "Analisis metode pengajaran fonetik dan morfologi bahasa arab," *Jurnal Ilmiah Iqra'*, vol. 3, no. 1, 2018.
- [6] M. Syaroni and R. Munir, "Pencocokan string berdasarkan kemiripan ucapan (phonetic string matching) dalam bahasa inggris," in *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, 2005.
- [7] W. B. Cavnar, J. M. Trenkle *et al.*, "N-gram-based text categorization," in *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, vol. 161175, 1994.
- [8] B. Zhao and E. Zhang, "Modifying a tokenizer based on pseudo data for natural language processing," May 24 2016, uS Patent 9,348,809.
- [9] S. Hadi, S. C. Soeratno, M. Ramlan, and I. D. P. Wijana, *Perubahan Fonologis Kata-kata Serapan dari Bahasa Arab dalam Bahasa Indonesia*. Gadjah Mada University, 2003.
- [10] A. M. Ismail and M. U. Nawawi, "Pedoman ilmu tajwid," *Surabaya: Karya Abditama*, 1995.
- [11] D. Romik, *The surprising mathematics of longest increasing subsequences*. Cambridge University Press, 2015, vol. 4.
- [12] D. Liben-Nowell, E. Vee, and A. Zhu, "Finding longest increasing and common subsequences in streaming data," *Journal of Combinatorial Optimization*, vol. 11, no. 2, pp. 155–175, 2006.
- [13] D. Kelly *et al.*, "Methods for evaluating interactive information retrieval systems with users," *Foundations and Trends in Information Retrieval*, vol. 3, no. 1–2, pp. 1–224, 2009.