IMPLEMENTASI JACCARD INDEX DAN N-GRAM PADA REKAYASA APLIKASI KOREKSI KATA BERBAHASA INDONESIA

Aida Indriani[1)], Muhammad[2)], Suprianto[3)], dan **Hadriansa[4)]**

[1,4]Teknik Informatika, STMIK PPKIA Tarakanita Rahmawati
[2]Sistem Informasi, STMIK PPKIA Tarakanita Rahmawati
[3]Magister Teknik Informatika, Universitas Ahmad Dahlan
[1,2,4]Jl. Yos Sudarso, Tarakan, 77113
[3]Jl. Dr. Soepomo Janturan, Yogyakarta, 55164
E-mail : aida@ppkia.ac.id[1)], muhammad@ppkia.ac.id [2)], rejectdiall@gmail.com [3)] , ansar@ppkia.ac.id [4)]

ABSTRAK

Banyaknya informasi diberbagai media, membuat pengguna harus jeli dalam mencari informasi yang benar. Informasi yang dikatakan benar bukan hanya dilihat dari sumber terpercaya, tetapi dalam penulisan tidak boleh terjadi kesalahan ejaan kata (*typo*) yang dapat mengakibatkan kesalahpahaman makna informasi yang dibaca. Untuk meminimalkan kesalahan ejaan kata dibutuhkan peran editor dengan melakukan koreksi kata secara satu per satu. Tujuan dari penelitian ini adalah untuk membuat aplikasi koreksi kata secara otomatis, dengan memanfaatkan teknik *text mining* yaitu *set based similarity measure.* Teknik yang digunakan yaitu *jaccard index* dan menggunakan bantuan fitur N-gram sebanyak 3 yaitu Bi-gram, Tri-gram dan Quad-gram. Selain itu, penelitian ini bertujuan untuk menentukan fitur N-gram yang tepat dalam melakukan koreksi kata. Dengan adanya aplikasi koreksi kata ini diharapkan dapat membantu tim editor dalam melakukan pengecekan kata sebelum dipubikasikan ke umum. Untuk analisa fitur N-gram yang tepat untuk melakukan koreksi kata adalah fitur Bi-gram.

Kata Kunci: Koreksi kata, *Text mining, Jaccard index, Fitur N-gram*

1. PENDAHULUAN

Setiap manusia pasti memerlukan informasi. Informasi dapat berupa berita, artikel, paper, jurnal, makalah dan lain-lain. Informasi dapat dikatakan baik apabila informasi yang disajikan bisa dapat dipertanggungjawabkan kebenarannya dan ditulis dalam kata yang baik dan benar. Sebelum dipublikasikan, informasi yang diperoleh mengalami proses *editing*. Proses *editing* dilakukan untuk mengecek kata demi kata pada sebuah informasi agar tidak terjadi kesalahan dalam ejaan kata yang dilakukan oleh penulis informasi.

Pada penelitian sebelumnya, metode *jaccard index* digunakan untuk melakukan pencarian artikel dengan cara melakukan perhitungan kemiripan kata kunci terhadap artikel yang ada. Himpunan *intersect* dan *union* diperoleh dari kata asli tanpa mengalami pemotongan karakter untuk setiap kata yang ada atau biasa disebut sebagai N-gram (Rinartha, 2017). Analisa koreksi kata juga dapat dilakukan dengan menggunakan metode levenshtein distance yaitu dengan cara menghitung jarak dengan menggunakan matriks 2 (dua) dimensi. Koreksi kata dilakukan dengan menggunakan fitur N-gram yaitu Bi-gram (Fahma, dkk, 2018). Ada beberapa algoritma yang bisa digunakan dalam proses pencocokan *string* antara lain, algoritma *Brute Force*, *Knuth-Morris-Pratt*, *Boyer-Moore*, *Karp-Rabin* dan *Shift O* (Yusnita dan Yunita, 2018).

Pada penelitian ini digunakan algoritma *set based similarity measure* yaitu teknik *jaccard index* dalam melakukan koreksi kata dengan mencari nilai *intersect* dan *union* antara dua kata yang dibandingkan. Berdasarkan penelitian sebelumnya yang melakukan koreksi kata tanpa fitur N-gram dan hanya menggunakan fitur Bi-gram, maka pada penelitian ini menggunakan 3 (tiga) fitur N-gram untuk melengkapi proses koreksi kata yaitu Bi-gram, Tri-gram dan Quad-gram. Penggunaan ketiga fitur N-gram pada penelitian ini bertujuan untuk mengetahui fitur N-gram mana yang tepat dalam melakukan koreksi kata.

2. RUANG LINGKUP

Berdasarkan latar belakang yang telah dijelaskan, maka dapat dirumuskan permasalahan yang dilakukan dalam penelitian ini adalah bagaimana membangun sebuah aplikasi koreksi kata menggunakan algoritma *set based similarity measure* dengan teknik *jaccard index* dan dengan fitur N-gram.

Penelitian ini bertujuan untuk membantu tim editor dalam melakukan koreksi kata secara otomatis, agar informasi yang dipublikasikan terhindar dari kesalahan ejaan kata. Selain itu juga, penulis dapat menganalisa fitur N-gram mana yang tepat untuk digunakan dalam proses koreksi kata.

3. BAHAN DAN METODE

Untuk melakukan implementasi algoritma *jaccard index* dalam aplikasi koreksi kata, ada beberapa metode yang terkait dengan penelitian ini, yaitu:

3.1 *Text Mining*

Salah satu variasi dari data *mining* yaitu *text mining*. *Text mining* bekerja dengan cara menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar (Kurniawan, dkk, 2012). *Text mining* bertujuan untuk memperoleh informasi yang berguna dari sekumpulan dokumen yang diklasifikasikan secara otomatis. Permasalahan yang biasa ditangani oleh *text mining* selain klasifikasi yaitu *information extraction*, *clustering*, dan *information retrieval* (Prakasa, 2016). Selain itu juga *text mining* mempunyai tujuan yaitu untuk mencari kata dalam sekumpulan dokumen dan melakukan analisa keterhubungan kata dalam dokumen tersebut (Praseptian, 2014).

Pada penelitian ini tahapan *text mining* yang dilakukan hanya tahapan *pre-processing* yaitu tindakan *toLowerCase* dan *tokenizing*. *toLowerCase* yaitu mengubah semua karakter huruf menjadi huruf kecil, dan *Tokenizing* yaitu proses penguraian deskripsi yang semula berupa kalimat-kalimat menjadi kata-kata dan menghilangkan delimiter-delimiter seperti tanda titik (.), koma (,), spasi dan karakter angka yang ada pada kata tersebut (Indriani, 2014).

3.2 *Similarity Measure*

Similarity Measure adalah proses pengukuran kemiripan suatu objek terhadap objek acuan. Ada beberapa jenis *similarity measure* yang biasa digunakan antara lain *Distance-Based Similarity Measure*, *Probabilistic-Based Similarity Measure*, *Set-Based Similarity Measure*, *Feature-Based Similarity Measure* dan *Context-Based Similarity Measure* (Nugraheny, 2015).

Jaccard Index adalah indeks yang menunjukkan tingkat kesamaan antara suatu himpunan (set) data dengan himpunan (set) data yang lain. Jaccard Index dihitung menggunakan rumus (1) sebagai berikut:

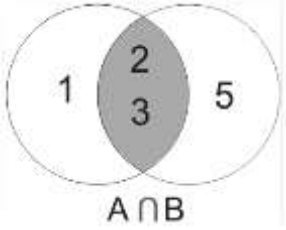
$$J(A,B) = (A \text{ INTERSECT } B) / (A \text{ UNION } B) \quad (1)$$

Intersect adalah operasi irisan dua himpunan A dan B (ditulis A ∩ B) adalah himpunan semua anggota A dan juga termasuk anggota B (Mumu dan Tanujaya, 2018). Berikut adalah contoh dari proses *intersect* dari himpunan A dan B.

A = {1, 2, 3}
B = {2, 3, 5}

A ∩ B = {2, 3}

Proses *intersect* himpunan A dan B dapat dilihat pada gambar 1.



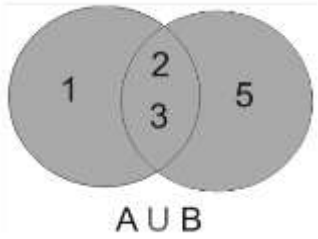
Gambar 1. Intersect himpunan A dan B

Union adalah dua himpunan A dan B (A U B) adalah himpunan dari semua anggota A atau B atau keduanya (Mumu dan Tanujaya, 2018). Berikut adalah contoh dari proses *union* dari himpunan A dan B.

A = { 1, 2, 3}
B = {2, 3, 5}

A U B = {1, 2, 3, 5}

Proses *union* himpunan A dan B dapat dilihat pada gambar 2.



Gambar 2. Union himpunan A dan B

3.3 N-gram

N-gram adalah potongan n karakter dalam suatu *string* tertentu atau potongan n kata dalam suatu kalimat tertentu (Lisangan, 2015). N-gram dapat dibedakan berdasarkan berapa jumlah huruf yang dipergunakan dalam pemisahan huruf untuk setiap kata antara lain dengan nilai n = 1, 2, 3 atau 4. n = 1 biasanya disebut dengan Uni-gram, n = 2 disebut dengan Bi-gram, n = 3 disebut dengan Tri-gram dan n = 4 disebut Quad-gram.

Misalnya dalam kata "TYPO" akan didapatkan N-gram sebagai berikut:

Uni-gram : T, Y, P, O
Bi-gram : _T, TY, YP, PO, O_
Tri-gram : _TY, TYP, YPO, PO_, O__
Quad-gram : _TYP, TYPO, YPO, PO__, O___

3.4 Kamus Besar Bahasa Indonesia (KBBI)

Kamus merupakan sebuah karya yang mempunyai fungsi sebagai referensi. Kamus disusun secara alfabetis dan berupa senarai kata. Pada kamus juga terdapat informasi mengenai pelafalan, ejaan, makna kata, dan kelas kata. Dalam KBBI, kamus merupakan sumber rujukan untuk memahami makna kata suatu bahasa. Selain itu juga, kamus memuat perbendaharaan kata suatu bahasa dengan jumlah yang tidak terbatas (Setiawati, 2016). Kata yang terdapat pada KBBI

merupakan kata pembanding yang akan digunakan dalam proses koreksi kata dan akan digunakan sebagai rujukan kata yang benar. Beberapa contoh kata yang terdapat dalam kamus bahasa Indonesia dapat dilihat pada tabel 1.

Tabel 1. Kata dalam kamus bahasa Indonesia

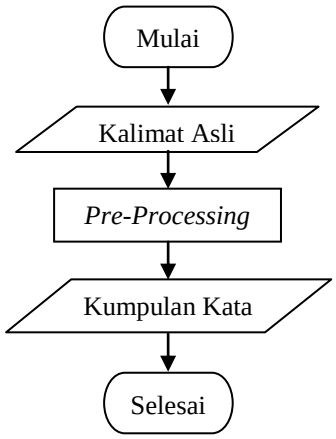
No.	Kata	No.	Kata
1	Ananda	11	Media
2	Ancam	12	Melia
3	Besar	13	Meningkat
4	Bagian	14	Pencari
5	Cantik	15	Penggunaan
6	Dewasa	16	Sebagai
7	Devisa	17	Sosial
8	Informasi	18	Semakin
9	Info	19	Semangat
10	ini	20	Tegar

4. PEMBAHASAN

Ada beberapa bahasan yang dituangkan dalam penelitian ini yaitu mengenai tahapan perancangan implementasi metode, database, dan hasil implementasi antar muka.

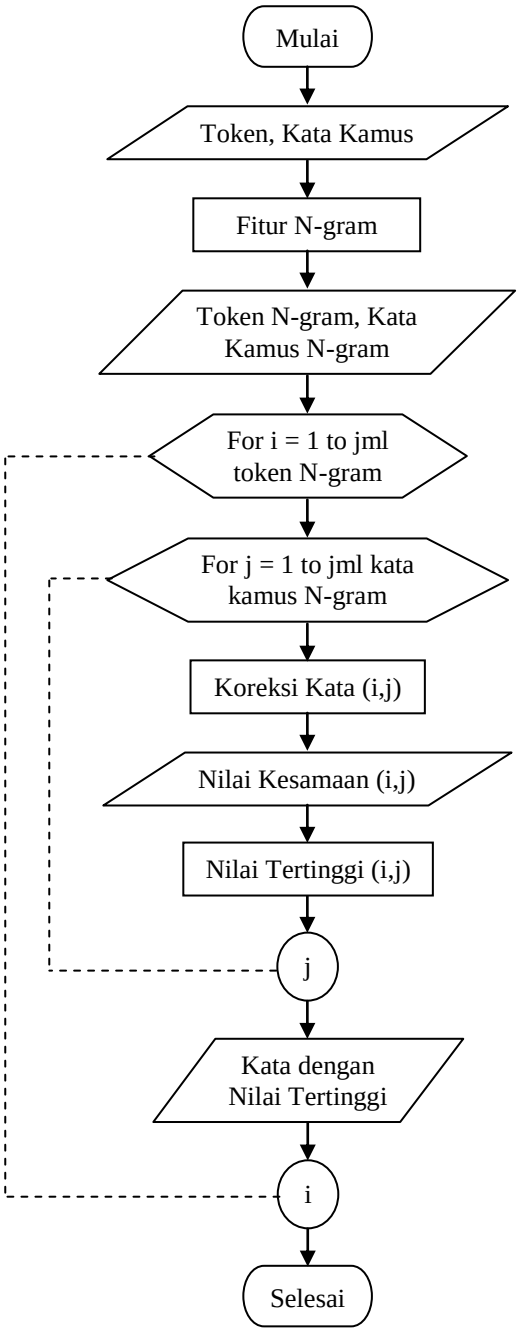
4.1 Tahapan Perancangan Implementasi Metode

Pada tahapan perancangan implementasi metode akan dijelaskan langkah-langkah penggunaan metode *jaccard index* dan fitur N-gram dalam melakukan koreksi kata. Tahapan awal yaitu melakukan *pre-processing* yang digambarkan dengan *flowchart* seperti pada gambar 3.



Gambar 3. Tahapan *Pre-Processing*

Pada gambar 3, dijelaskan tahapan *pre-processing* yang dilakukan pada kalimat asli yang dimulai dari *toLowerCase* dan dilanjutkan dengan proses *tokenizing* sehingga mnghasilkan sebuah kumpulan kata (*token*) yang nantinya akan digunakan pada tahapan selanjutnya yaitu koreksi kata. Tahapan koreksi kata ditunjukkan pada gambar 4.



Gambar 4. Tahapan Koreksi Kata

Pada gambar 4, dijelaskan tahapan koreksi kata yaitu dimulai dari mengubah token dan kata dalam kamus Bahasa Indonesia menjadi kata N-gram. Setiap token dihitung tingkat kesamaan dengan seluruh kata yang ada dalam kamus Bahasa Indonesia dengan menggunakan jaccard index. Kata yang mempunyai nilai kesamaan tertinggi merupakan hasil dari koreksi kata.

4.2 Desain *Database*

Untuk membangun aplikasi koreksi kata, penulis menggunakan 1 (satu) database dengan nama DBKoreksi dan 3 (tiga) tabel, antara lain:

4.2.1 Tabel Kamus Bahasa Indonesia

Tabel kamus Bahasa Indonesia digunakan untuk menampung kata-kata yang terdapat pada kamus Bahasa Indonesia yang nantinya akan digunakna sebagai kata pembanding dari setiap kata yang dikoreksi. Struktur tabel kamus Bahasa Indonesia dapat dilihat pada tabel 2.

Tabel 2. Struktur Tabel Kamus

Nama_Field	Tipe_Data	Panjang	Keterangan
kata	*Varchar*	20	Kata yang terdapat pada kamus Bahasa Indonesia

4.2.2 Tabel Kalimat Asli

Tabel kalimat asli digunakan untuk menampung kalimat yang akan dikoreksi, untuk kemudian dilakukan proses *pre-processing*. Struktur tabel kalimat asli dapat dilihat pada tabel 3.

Tabel 3. Struktur Tabel Kal_Asli

Nama_Field	Tipe_Data	Panjang	Keterangan
kalasli	*Varchar*	200	Kalimat-kalimat yang terdapat pada informasi

4.2.3 Tabel *Pre-Processing*

Tabel *pre-processing* digunakan untuk menyimpan kumpulan kata yang terdapat pada kaliman asli, untuk kemudian secara satu per satu kata dibandingkan dengan kata kamus Bahasa Indonesia untuk menghitung tingkat kemiripan kata dengan menggunakan teknik *jaccard index* dan fitur N-gram. Struktur tabel *pre-processing* dapat dilihat pada tabel 4.

Tabel 4. Struktur Tabel *Preproc*

Nama_Field	Tipe_Data	Panjang	Keterangan
katapre	*Varchar*	20	Kata Pre-processing

4.2.4 Tabel Kalimat Koreksi

Tabel kalimat koreksi digunakan untuk menampung kalimat yang telah melalui proses koreksi kata. Struktur tabel kalimat koreksi dapat dilihat pada tabel 5.

Tabel 5. Struktur Tabel Kal_Koreksi

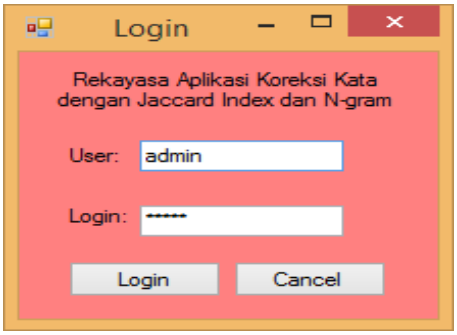
Nama_Field	Tipe_Data	Panjang	Keterangan
kalkoreksi	*Varchar*	200	Kalimat yang telah dikoreksi

4.3 Hasil Impelemntasi Antar Muka

Aplikasi koreksi kata yang dibangun dengan menggunakan Microsoft Visual Studio 2012. Terdapat 3 (tiga) *form* pada aplikasi yang dibangun yaitu *form login*, *form pre-processing* dan *form* koreksi kata. Secara detail akan dijelaskan ketiga form yang dibuat, sebagai berikut:

4.3.1 *Form Login*

Form login merupakan tampilan awal pada aplikasi koreksi kata. Pada *form login* terdapat 2 (dua) masukan yaitu *user* dan *login*. Pada saat login berhasil, maka berpindah ke *form pre-processing* yaitu yang merupakan tahapan awal dari aplikasi koreksi kata. Desain *form login* ditunjukkan pada gambar 3.

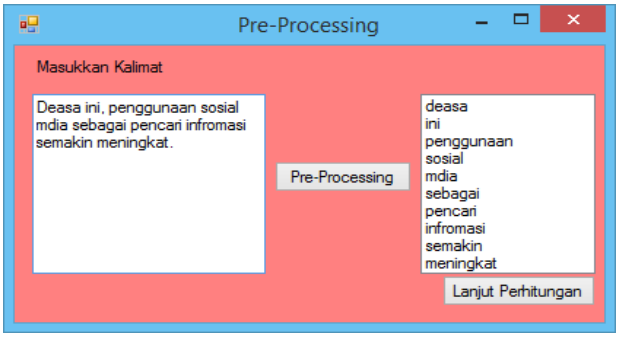


Gambar 3. *Form Login*

4.3.2 *Form Pre-Processing*

Form pre-processing merupakan tahapan awal dari proses koreksi kata, yaitu dengan membagi kalimat menjadi potongan kata-kata (*token*) dengan cara menghilangkan tanda baca dan spasi. Selain itu juga dalam tahapan *pre-processing* kata diubah menjadi huruf kecil.

Pada *form pre-processing* terdapat 1 (satu) masukan yang berguna untuk memasukkan kalimat asli yang akan dilakukan koreksi kata. Setelah semua kalimat dimasukkan, langkah selanjutnya yaitu memilih tombol "pre-processing", yang nantinya akan menghasilkan potongan kata yang terdapat pada komponen ListBox. Tombol "lanjut perhitungan" digunakan untuk menuju *form* koreksi kata. Desain *form pre-processing* ditunjukkan pada gambar 4.



Gambar 4. *Form Pre-Processing*

4.3.3 *Form* Koreksi Kata

Form koreksi kata digunakan untuk melakukan pencocokan kata *pre-processing* dengan kata pada kamus besar Bahasa Indonesia. Hasil akhirnya yaitu menampilkan kata dengan memiliki tingkat kesamaan tertinggi dengan menggunakan 3 (tiga) fitur N-gram yaitu Bi-gram, Tri-gram dan Quad-gram.

Kata *pre-processing* dan kata dalam kamus tampil secara otomatis pada *ListBox* dengan mengambil data yang tersimpan pada tabel preproc dan tabel kamus. Desain form koreksi kata ditunjukkan pada gambar 5.

Gambar 5. Perhitungan Koreksi Kata

5. UJI COBA DAN HASIL ANALISA

Pada uji coba akan dijelaskan bagaimana penerapan terhadap data uji sesuai dengan tahapan implementasi *jaccard index* dengan beberapa fitur N-gram yang telah dijelaskan dan yang telah diterapkan pada aplikasi yang dibuat dalam hal koreksi kata. Hasil analisa melakukan analisa terhadap hasil uji coba dan pemilihan fitur N-gram yang tepat dalam hal koreksi kata dari 3 (tiga) N-gram yang digunakan.

5.1.1 *Pre-processing*

Contoh kalimat yang dimasukkan sesuai dengan gambar 3 adalah "Deasa ini, penggunaan sosial mdia sebagai pencari infromasi semakin meningkat". Dari contoh kalimat dilakukan proses *pre-processing*, hasil proses *pre-processing* dapat dilihat pada gambar 3 yaitu menjadi potongan kata-kata "deasa, ini, penggunaan, sosial, mdia, sebagai, pencari, infromasi, semakin dan meingkat".

5.1.2 Bi-gram

Kata yang telah dilakukan proses *pre-processing*, dilakukan pemotongan karakter dengan jumlah 2 (dua) karakter. Hasil pemotongan karakter dengan menggunakan fitur Bi-gram dapat dilihat pada tabel 6.

Tabel 6. Potongan Karakter Bi-gram

No.	Kata	Bi-gram
1	deasa	_d, de, ea, as, sa, a_
2	ini	_i, in, ni, i_
3	penggunaan	_p, pe, en, ng, gg, gu, un, na, aa, an, n_
4	sosial	_s, so, os, si, ia, al, l_
5	mdia	_m, md, di, ia, a_
6	sebagai	_s, se, eb, ba, ag, ga, ai, i_
7	pencari	_p, pe, en, nc, ca, ar, ri, i_
8	infromasi	_i, in, nf, fr, ro, om, ma, as, si, i_
9	semakin	_s, se, em, ma, ak, ki, in, n_
10	meningkat	_m, me, en, ni, in, ng, gk, ka, at, t_

Kata yang terdapat pada kamus besar Bahasa Indonesia juga mengalami proses pemotongan karakter dengan Bi-gram.

5.1.3 Tri-gram

Kata yang telah dilakukan proses *pre-processing*, dilakukan pemotongan karakter dengan jumlah 3 (tiga) karakter. Hasil pemotongan karakter dengan menggunakan fitur Tri-gram dapat dilihat pada tabel 7.

Tabel 7. Potongan Karakter Tri-gram

No.	Kata	Bi-gram
1	deasa	_de, dea, eas, asa, sa_, a__
2	ini	_in, ini, ni_, i__
3	penggunaan	_pe, pen, eng, ngg, ggu, gun, una, naa, aan, an_, n__
4	sosial	_so, sos, osi, sia, ial, al_, l__
5	mdia	_md, mdi, dia, ia_, a__
6	sebagai	_se, seb, eba, bag, aga, gai, ai_, i__
7	pencari	_pe, pen, enc, nca, car, ari, ri_, i__
8	infromasi	_in, inf, nfr, fro, rom, oma, mas, asi, si_, i__
9	semakin	_se, sem, ema, mak, aki, kin, in_, n__
10	meningkat	_me, men, eni, nin, ing, ngk, gka, kat, at_, t__

Kata yang terdapat pada kamus besar Bahasa Indonesia juga mengalami proses pemotongan karakter dengan Tri-gram.

5.1.4 Quad-gram

Kata yang telah dilakukan proses *pre-processing*, dilakukan pemotongan karakter dengan jumlah 4 (empat) karakter. Hasil pemotongan karakter dengan menggunakan fitur Quad-gram dapat dilihat pada tabel 8.

Tabel 8. Potongan Karakter Quad-gram

No.	Kata	Bi-gram
1	deasa	_dea, deas, easa, asa_, sa__, a___
2	ini	_ini, ini_, ni__, i___
3	penggunaan	_pen, peng, engg, nggu, ggun, guna, unaa, naan, aan_, an__, n___
4	sosial	_sos, sosi, osia, sial, ial_, al__, l___
5	mdia	_mdi, mdia, dia_, ia__, a___
6	sebagai	_seb, seba, ebag, baga, agai, gai_, ai__, i___
7	pencari	_pen, penc, enca, ncar, cari, ari_, ri__, i___
8	infromasi	_inf, infr, nfro, from, roma, omas, masi, asi_, si__, i___
9	semakin	_sem, sema, emak, maki, akin, kin_, in__, n___
10	meningkat	_men, meni, enin, ning, ingk, ngka, gkat, kat_, at__, t___

Kata yang terdapat pada kamus besar Bahasa Indonesia juga mengalami proses pemotongan karakter dengan Quad-gram.

5.1.5 Jaccard Index

Berikut adalah proses perhitungan jarak menggunakan metode *jaccard index* antara kata dalam naskah berita terhadap kata yang terdapat pada kamus berbahasa Indonesia, dapat dilihat pada Tabel 9.

Tabel 9. Beberapa Contoh Kata untuk Perhitungan Jaccard Index

No.	Kata pre-processing (A)	Kata dalam Kamus Bahasa Indonesia (B)	
1	deasa	1	dewasa
		2	devisa
		3	delusi
2	mdia	4	melia
		5	media
		6	medan
3	infromasi	7	informasi
		8	informan
		9	info

Perhitungan *jaccard index* dengan fitur Bi-gram
Diketahui himpunan:
A1 = { _d, de, ea, as, sa, a_ }
B1 = { _d, de, ew, wa, as, sa, a_ }

Intersect himpunan A1 dan B1 (A1 ∩ B1) adalah sebagai berikut:

(A1 ∩ B1) = { _d, de, as, sa, a_ } = 5

Union himpunan A1 dan B1 (A1 U B1) adalah sebagai berikut:

(A1 U B1) = { _d, de, ea, as, sa, a_, ew, wa} = 8
J(A1,B1) = 5/8 = 0.625

Perhitungan *jaccard index* dengan fitur Tri-gram
Diketahui himpunan:
A1 = { _de, dea, eas, asa, sa_, a__ }
B1 = { _de, dew, ewa, was, asa, sa_, a__}

Intersect himpunan A1 dan B1 (A1 ∩ B1) adalah sebagai berikut:

(A1 ∩ B1) = { _de, asa, sa_, a__} = 4

Union himpunan A1 dan B1 (A1 U B1) adalah sebagai berikut:

(A1 U B1) = { _de, dea, eas, asa, sa_, a__, dew, ewa, was} = 9

J(A1,B1) = 4/9 = 0.444

Perhitungan *jaccard index* dengan fitur Quad-gram
Diketahui himpunan:
A1 = { _dea, deas, easa, asa_, sa__, a___ }
B1 = { _dew, dewa, ewas, wasa, asa_, sa__, a___ }

Intersect himpunan A1 dan B1 (A1 ∩ B1) adalah sebagai berikut:

(A1 ∩ B1) = { asa_, sa__, a___ } = 3

Union himpunan A1 dan B1 (A1 U B1) adalah sebagai berikut:

(A1 U B1) = { _dea, deas, easa, asa_, sa__, a___, _dew, dewa, ewas, wasa} = 10

J(A1,B1) = 3/10 = 0.3

Hasil keseluruhan dari perhitungan jarak antara himpunan A dan B dapat dilihat pada tabel 10.

Tabel 10. Nilai Kesamaan Kata

No.	Kata dalam Naskah Berita (A)	Kata dalam Kamus Bahasa Indonesia (B)		Hasil Perhitungan N-gram		
				Bi	Tri	Quad
1	deasa	1	dewasa	0.625	0.444	0.3
		2	devisa	0.444	0.3	0.182
		3	delusi	0.182	0.083	0
2	mdia	4	melia	0.375	0.222	0.222
		5	media	0.571	0.375	0.375
		6	medan	0.1	0	0
3	infromasi	7	informasi	0.538	0.429	0.333
		8	informan	0.267	0.118	0.056
		9	info	0.250	0.154	0.071

Dari hasil perhitungan pada tabel 10 dapat dijelaskan hasil uji coba metode *jaccard index* dengan beberapa fitur N-gram sebagai berikut:

1. Untuk fitur Bi-gram, kata "deasa" terhadap kata "dewasa" memiliki tingkat kesamaan sebesar 0.625, terhadap kata "devisa" memiliki tingkat kesamaan sebesar 0.444 dan 0.182 terhadap kata "delusi". Nilai tertinggi sebesar 0.625 untuk kata "dewasa". Untuk fitur Tri-gram, nilai tertinggi sebesar 0.444 yaitu kata "dewasa". Untuk fitur Quad-gram, nilai tertinggi sebesar 0.3 juga untuk kata "dewasa".
2. Dari ketiga fitur untuk kata "dewasa" yaitu 0.625 untuk Bi-gram, 0.444 untuk Tri-gram dan 0.3 untuk Quad-gram. Nilai yang tertinggi adalah 0.625 dengan menggunakan fitur Bi-gram. Untuk kata "mdia" menghasilkan kata koreksi "media" yang memiliki nilai tertinggi 0.571 dengan menggunakan fitur Bi-gram. Sedangkan kata "infromasi" menghasilkan kata koreksi "informasi" yang memiliki nilai tertinggi 0.538 dengan menggunakan fitur Bi-gram.

6. KESIMPULAN

Setelah melakukan uji coba dan hasil analisa terhadap implementasi *jaccard index* dan fitur N-gram pada aplikasi koreksi kata, maka dapat disimpulkan bahwa aplikasi yang dibuat dapat membantu tim editor dalam melakukan koreksi kata. Selain itu juga, diperoleh nilai kesamaan kata tertinggi untuk penggunaan ketiga fitur N-gram pada penelitian ini dengan menggunakan beberapa data uji coba. Dapat disimpulkan bahwa fitur Bi-gram yang lebih tepat dalam melakukan koreksi kata

dikarenakan memiliki nilai tertinggi dibanding dengan fitur Tri-gram dan Quad-gram.

7. SARAN

Aplikasi koreksi kata dapat dikembangkan dengan menggunakan aplikasi berbasis web atau android. Selain itu, penggunaan metode lain seperti *cosine similarity* juga dapat digunakan dalam melakukan pencocokan kata. Aplikasi juga dapat dikembangkan dengan melakukan koreksi kata dengan mengolah teks berbahasa inggris.

8. DAFTAR PUSTAKA

Fahma, A.I., Cholissodin, M., & Perdana, R.S., 2018, Identifikasi Kesalahan Penulisan Kata (*Typographical Error*) pada Dokumen Berbahasa Indonesia menggunakan Metode N-gram dan Levensthein Distance. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 2 No. 1, e-ISSN: 2548-964X, 53-62.

Indriani, Aida, 2014, Maximum Marginel Relevance untuk Peringkasan Teks Otomatis Sinopsis Buku Berbahasa Indonesia. Seminar Nasional Teknologi Informasi dan Multimedia, ISSN: 2302-3805, 28-34, Yogyakarta.

Kurniawan, B., Effendi, S., & Sitompul, O.S., 2012, Klasifikasi Konten Berita dengan Metode Text Mining. Jurnal Dunia Teknologi Informasi, Vol. 1 No. 1, 14-19.

Lisangan, E.A., 2015, Implementasi n-Gram Technique dalam Deteksi Plagiarisme pada Tugas Mahasiswa. Jurnal Tematika, Vol. 1 No. 2, ISSN: 2303-387824-30.

Mumu, J., & Tanujaya, B., 2018, Desain Pembelajaran Materi Operasi pada Himpunan menggunakan Permainan "Lemon Nipis". Journal of Honai Math, Vol. 1 No. 1, p-ISSN: 2615-2185 e-ISSN: 2615-2193, 14-23.

Nugraheny, D., 2015, Metode Nilai Jarak guna Kesamaan atau Kemiripan Ciri suatu Citra (kasus deteksi awan cumulonimbus menggunakan principal component analysis). Jurnal Angkasa, Vol. VII,21-30.

Prakasa, S.A., 2016, Text Mining. Sekolah Tinggi Teknologi Telematika Telkom Purwokerto.

Praseptian, M.D., & Indriani, A., 2014, Implementasi Text Mining dalam Klasifikasi Buku dengan Metode Naïve Bayes Classifier Studi Kasus pada Perpustakaan STMIK PPKIA Tarakanita Rahmawati. Seminar Nasional Inovasi dan Tren, 243-247.

Pusat Bahasa Departemen Pendidikan Nasional, 2008, Kamus Bahasa Indonesia. ISBN: 978-979-689-779-1.

Rinartha, K., 2017, Simple Query Suggestion untuk Pencarian Artikel menggunakan Jaccard Similarity. Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi, Vol. 3 No. 1, e-ISSN: 2502-8995 p-ISSN: 2460-8181, 30-34.

Setiawati, S., 2016, Penggunaan Kamus Besar Bahasa Indonesia (KBBI) dalam Pembelajaran Kosakata Baku dan Tidak Baku pada Siswa Kelas IV SD. Jurnal Gramatika, Vol. 2 No.1, ISSN: 2442-8485 e-ISSN: 2460-6319, 44-48.

Yusnita, A., & Yunita, 2018, Penelusuran Katalog Perpustakaan pada SMA IT Yabis Bontang dengan Algoritma Boyer-Moore. Sebatik STMIK WICIDA, ISSN: 1410-3737 e-ISSN: 2621-069X, 15-21, Samarinda.