Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN: 2548-964X
Vol. 1, No. 6, Juni 2017, hlm. 498-506 http://j-ptiik.ub.ac.id

Deteksi Kesalahan Ejaan dan Penentuan Rekomendasi Koreksi Kata yang Tepat Pada Dokumen Jurnal JTIIK Menggunakan Dictionary Lookup dan Damerau-Levenshtein Distance

Tusty Nadia Maghfira[1], Imam Cholissodin[2], Agus Wahyu Widodo[3]

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: [1]tustynadia@gmail.com, [2]imamcs@ub.ac.id, [3]a_wahyu_w@ub.ac.id

Abstrak

Menulis merupakan sarana penyampaian dan bertukar informasi antar manusia. Kegiatan tersebut saat ini semakin mudah dilakukan dengan adanya perangkat canggih dan internet. Salah satu contohnya yaitu penulisan dan publikasi jurnal penelitian yang dibuat untuk mengembangkan ilmu pengetahuan. Publikasi jurnal penelitian umumnya ditampung oleh lembaga pendidikan baik nasional maupun internasional, salah satunya yaitu *website* JTIIK (Jurnal Teknologi Informasi & Ilmu Komputer) Fakultas Ilmu Komputer UB. Sebelum dipublikasi, jurnal yang masuk harus melalui tahap *editing* untuk mengoreksi apabila ada kesalahan dan kekurangan seperti adanya kesalahan ejaan kata. Namun, dalam kerjanya seorang *editor* juga dapat melakukan kesalahan yang berdampak dengan masih adanya kesalahan setelah *editing*. Adanya kesalahan ejaan kata dapat mengubah makna pengetahuan yang disampaikan penulis dan menimbulkan salah pemahaman informasi pada pembaca. Berdasarkan pemasalahan tersebut, peneliti bermaksud membantu kerja *editor* dalam analisis jurnal dengan mengusulkan sebuah sistem deteksi dan koreksi kesalahan ejaan kata menggunakan metode *Dictionary Lookup* dan *Damerau-Levenshtein Distance*. Metode *Dictionary Lookup* dinilai efektif dalam menentukan suatu ejaan kata bernilai benar atau salah berdasarkan *Lexical Resource*. Selain itu metode *Damerau-Levenshtein Distance* dapat mengoreksi kesalahan penulisan kata dengan tepat dan lebih unggul dibandingkan *Levenshtein Distance*. Hasil pengujian koreksi terbaik didapatkan pada skenario pengujian kedua yaitu presisi sebesar 0.78 dan *recall* sebesar 1.

Kata Kunci: *koreksi kata bahasa Indonesia, non-word error, kesalahan ejaan, deteksi kesalahan ejaan, koreksi kesalahan ejaan, dictionary lookup, damerau-levenshtein distance*

Abstract

Writing is a way to deliver and share information among people. It is now even easier to do because of the help of technology such as computers, smartphones and internet. For example, writing and publication of research journal is made to share and enhance knowledge. Generally, the publication of research journal is accommodated by educational institutions both national and international such as JTIIK (Jurnal Teknologi Informasi & Ilmu Komputer) Faculty of Computer Science UB. Before journal is published, a journal should pass editing process by editor to check if there is some mistake and deficiency such as spelling error. However, in their work, editor also accidentally making mistake that will lead to many error spelling that still exist even though editing process has been done. Some misspelled word can change the meaning of knowledge that the author want to deliver and cause misunderstanding of information among the readers. Based on these problems, researcher propose error spelling detection and correction system using Dictionary Lookup and Damerau-Levenshtein Distance. Dictionary Lookup method is considered effective in determining a word including validity or invalidity of the word based on availability or unavailability of the word in Lexical Resource. In addition, Damerau-Levenshtein Distance can provide better correction than Levenshtein Distance. The best precision and recall result for correction simultaneously are 0.78 and 1 from second test scenario.

Keywords: *Indonesian spell checker, non-word error, spelling error, spelling error detection, spelling error correction, dictionary lookup, damerau-levenshtein distance*

1. PENDAHULUAN

Kegiatan tulis menulis sebagai media komunikasi dan informasi dalam peradaban manusia telah dikenal lama yaitu sejak masa sebelum Masehi. Pada masa tersebut, manusia berkomunikasi hanya dengan membuat tulisan gambar (hieroglif) dan simbol-simbol pada berbagai objek di alam seperti pada batu, kayu, daun, dan dinding-dinding gua (Muradmaulana, 2014). Namun, seiring waktu berjalan tradisi tulis menulis pun juga ikut mengalami perkembangan. Di awal Masehi, manusia mulai menemukan kertas serta mengenal bermacam bahasa dan huruf berdasarkan daerah mereka tinggal. Tidak berhenti sampai disitu, hingga sampai saat ini di mana perkembangan teknologi semakin canggih di mana masyarakat mulai banyak menggunakan komputer, *smartphone* dan internet, budaya tulis menulis juga terus berkembang. Hanya saja penggunaan kertas mulai berkurang dan beralih pada tulis menulis menggunakan media-media canggih seperti komputer dan *smartphone*. Contohnya banyak berbagai diktat kuliah, jurnal, dan ilmu pengetahuan yang disimpan dalam *electronic book* (*E-book*), berita dan surat kabar yang tersaji dalam *website-website*, serta surat menyurat dengan menggunakan *electronic mail* (*email*).

Dengan penggunaan komputer, *smartphone* dan internet yang saat ini semakin luas tersebut, memberikan lebih banyak kemudahan bagi masyarakat dalam kegiatannya sehari-hari termasuk kegiatan yang melibatkan tulis menulis. Hal tersebut mendorong meningkatnya jumlah masyarakat yang mampu menulis dengan baik sehingga permintaan dokumen tertulis pun juga ikut meningkat pada berbagai aspek (N et al. 2011). Salah satunya yaitu pada bidang pendidikan. Banyak peneliti di kalangan mahasiswa, ilmuan dan dosen yang berlomba-lomba dalam menyumbangkan idenya untuk peningkatan pendidikan dan teknologi melalui penyusunan jurnal penelitian. Jurnal penelitian umumnya berisi usulan solusi yang baru atau perbaikan dari solusi yang sudah ada terhadap suatu masalah. Banyak lembaga baik nasional maupun internasional yang menyediakan wadah bagi peneliti untuk mengakses dan mengusulkan jurnal, salah satunya yaitu website JTIIK (Jurnal Teknologi Informasi & Ilmu Komputer) FILKOM UB. JTIIK memuat naskah hasil-hasil penelitian di bidang Teknologi Informasi dan Ilmu Komputer yang belum pernah diterbitkan di media manapun. Setiap jurnal yang masuk, diwajibkan untuk melalui tahap *editing* untuk mengecek apabila ada kekurangan dan kesalahan pada jurnal seperti adanya kesalahan ejaan kata atau informasi yang belum lengkap. Namun seperti manusia pada umumnya, seorang *edit*or dalam tugasnya juga tidak luput dari kesalahan yang tidak disengaja. Kesalahan ejaan kata dapat saja ditemukan lagi setelah tahap *editing* dikarenakan kealpaan *edit*or atau pengetahuan yang kurang terhadap penulisan ejaan kata yang paling *update* dan sesuai dengan KBBI. Selain itu seorang *edit*or lebih berfokus pada kesalahan besar terkait isi jurnal sehingga proses deteksi kesalahan ejaan kata dapat luput dari pengecekan. Padahal hal kecil seperti kesalahan ejaan kata dapat berpengaruh terhadap informasi dan pengetahuan yang disajikan dalam jurnal. Kemudian waktu pengecekan dokumen yang terbatas dan banyaknya naskah jurnal yang masuk dengan jumlah halaman yang berbeda-beda memungkinkan *edit*or kurang teliti dan cenderung tergesa-gesa dalam menganalisis dokumen. Kesalahan penulisan ejaan kata yang fatal dapat mengubah makna pengetahuan yang ingin disampaikan penulis dalam jurnalnya dan menimbulkan penyerapan informasi yang salah pada pembaca. Oleh karena itu untuk membantu kerja tim *edit*or dalam melakukan *editing* dibutuhkan suatu sistem otomatis yang dapat mendeteksi apabila ada kesalahan ejaan kata pada jurnal serta memberikan rekomendasi kandidat kata yang benar untuk kata yang salah ejaan tersebut.

Banyak peneliti yang terdorong untuk melakukan penelitian guna menentukan solusi permasalahan dengan menyusun aplikasi *automatic spell checker*. Salah satu penelitian tentang deteksi dan koreksi adalah deteksi dan koreksi *non-word error* kata Bahasa Indonesia dilakukan oleh Soleh & Purwarianti pada tahun 2011 dengan menggunakan metode *Dictionary Lookup* untuk deteksi dan untuk proses koreksi menggunakan metode *Probability of Similarity* dan *Forward Reversed Dictionary* sebagai pembanding. Hasil menunjukkan bahwa metode *Probability of Similarity* lebih unggul dengan nilai akurasi sebesar 98.55%. Berdasarkan latar belakang masalah yang ada dan analisis penulis terhadap penelitian sebelumnya, penulis mengusulkan penelitian untuk deteksi dan penentuan rekomendasi kandidat kata yang

benar pada dokumen jurnal yang diusulkan pada JTIIK menggunakan teknik *Dictionary Lookup* untuk deteksi kesalahan kata dan koreksi kesalahan menggunakan salah satu metode dari *Minimum Edit Distance* yaitu *Damerau-Levenshtein Distance*. Penulis berharap sistem ini dapat membantu editor jurnal dalam mendeteksi dan memberikan rekomendasi kandidat koreksi kesalahan kata yang tepat pada kesalahan penulisan ejaan kata yang tanpa sengaja sering dilakukan oleh manusia dan dapat menambah wawasan pengetahuan terutama di bidang *Text Mining.*

2. DASAR TEORI

2.1. Text Mining

Text Mining merupakan proses pencarian pola atau penggalian informasi dari data teks untuk menghasilkan informasi baru. Tujuan dari text mining adalah menemukan informasi yang penting dari teks dengan mengubah teks menjadi data yang dapat digunakan untuk analisis yang lebih lanjut (Expertsystem, 2016). Berbeda dengan *Data Mining*, *Text Mining* mengolah data berupa teks yang tidak terstruktur, tidak memiliki bentuk, dan sulit ditangani secara algoritme (Witten 2002). Oleh karena itu pada proses *Text Mining* memerlukan beberapa tahap awal (*preprocessing*) yang bertujuan untuk mempersiapkan agar teks dapat diubah menjadi bentuk yang lebih terstruktur. Beberapa tahapan dari *preprocessing* yaitu:

- Tokenisasi
- *Filtering*
- *Stemming*

Proses tokenisasi merupakan proses pemisahan setiap kata dari dokumen. Pada proses ini juga dilakukan penghilangan angka dan simbol pada dokumen yang dinilai tidak berpengaruh terhadap pengambilan informasi dari dokumen. Selain itu juga dilakukan pengubahan setiap huruf menjadi huruf kecil.

Selanjutnya hasil proses tokenisasi diseleksi lagi dengan menghilangkan kata-kata tidak penting (*stop word*). Proses ini disebut sebagai proses *filtering.*

Hasil dari proses *filtering* diproses lagi pada tahap selanjutnya yaitu proses *stemming* (mengubah setiap kata menjadi kata dasar).

Dari beberapa tahapan *preprocessing* yang sebelumnya dijelaskan, tidak semua tahapan harus dilaksanakan. Tahapan *preprocessing* yang dilakukan bergantung pada kebutuhan analisis teks yang ingin dilakukan. Setelah *preprocessing* selesai, data teks dapat dilanjutkan pada proses analisis untuk menentukan informasi penting pada teks. Pada *Text Mining* terdapat beberapa kategori analisis teks yang umum digunakan meliputi: pengorganisasian dan *clustering* dokumen, klasifikasi dokumen, *Information Extraction* (IE), *Web Mining*, *Named Entity Recognition* (NER), *Natural Language Processing* (NLP), *Information Retrieval* (IR). Analisis teks yang berperan penting dalam proses deteksi dan koreksi kesalahan penulisan kata yaitu *Natural Language Processing* (NLP) dan *Information Retrieval* (IR).

2.2. Natural Language Processing (NLP)

Natural Language Processing (NLP) merupakan salah satu komponen dari *Text Mining* yang berfokus pada hubungan antara komputer dan bahasa alami manusia. Tujuan dari NLP adalah untuk mendesain dan membangun perangkat lunak yang dapat menganalisis, memahami, dan menghasilkan bahasa manusia yang dapat digunakan secara alami, sehingga saat berkomunikasi dengan komputer seakan-akan berkomunikasi dengan manusia lain (Mishra & Kaur 2013). Dalam mencapai tujuan tersebut NLP menggunakan beberapa metode yaitu *Automatic Summarization*, *Part-of-Speech Tagging*, *Disambiguation, Machine Translation*, *Parsing*, *Optical Recognition*, dan *Question Answering*. Kerja dari NLP membutuhkan basis pengetahuan yang isinya konsisten dan benar seperti *thesaurus*, *lexicon of words*, dan berbagai macam data set lainnya yang berisi aturan linguistik dan gramatikal yang valid dan terbaru (Expertsystem, 2016). Aplikasi Text Mining tidak dapat terpisah dengan konsep NLP karena secara tidak langsung NLP bekerja pada background aplikasi untuk membantu sistem dapat membaca teks. Suatu aplikasi text mining yang menggunakan konsep NLP, tidak hanya berupa aplikasi pencarian yang sederhana memberikan daftar hasil pencarian yang sesuai namun juga memberikan informasi detil tentang teks tersebut dan mengungkap pola dari dokumen pada *data set*.

2.3. Information Retrieval (IR)

Information Retrieval (IR) adalah aktivitas untuk memperoleh sumber informasi (biasanya dokumen) dari sebuah data tidak terstruktur (biasanya teks) yang relevan atau memenuhi

kebutuhan informasi dari sebuah data berukuran besar (biasanya tersimpan pada komputer) (Manning et al. 2009). IR didesain untuk memfasilitasi pengguna dalam mendapatkan kembali informasi yang behubungan dengan apa yang dibutuhkan pengguna secara efektif dan efisien (Liyana et al. 2010). Di dalam sistem IR menggabungkan metode penyimpanan, indeksasi, *filtering*, pengorganisasian, pencarian dan menampilkan informasi. Berikut ini adalah fungsi utama sistem IR:

- Untuk mengidentifikasi sumber informasi yang relevan dengan target pengguna
- Untuk menganalisis isi dari sumber dokumen
- Untuk merepresentasikan isi dari sumber yang dianalisis sehingga dapat sesuai dengan query dari pengguna
- Untuk menganalisis *query* pengguna dan untuk merepresentasikannya ke dalam bentuk yang sesuai dengan database
- Untuk mencocokkan statemen pencarian dengan yang tersimpan dalam *database*
- Untuk mendapatkan kembali informasi yang relevan
- Untuk membuat pengaturan yang dibutuhkan dalam sistem berdasarkan feedback dari pengguna.

Pada IR sebuah *query* tidak hanya mengidentifikasi satu objek unik dalam kumpulan data, melainkan satu *query* dapat sesuai dengan beberapa objek namun dengan derajat kesesuaian yang berbeda. Sebuah objek merupakan satu entitas yang merepresentasikan informasi dari kumpulan data atau *database*. Berbeda dengan hasil yang disajikan pada sistem klasik *query* SQL *database*, hasil pada IR disajikan dalam bentuk peringkat. Sistem IR pada umumnya melakukan komputasi untuk menghitung seberapa besar nilai kecocokan setiap objek yang ada di dalam *database* dengan *query*, dan membuat sistem peringkat untuk hasil tersebut berdasarkan hasil perhitungan. Hasil dengan peringkat tertinggi ditampilkan pada pengguna dan dianggap sebagai hasil yang paling relevan.

Information Retrieval secara cepat mendominasi akses informasi, menggantikan metode pencarian tradisional *database*. Banyak pemanfaatan IR yang dapat dijumpai dalam kehidupan sehari-sehari. Salah satu institusi yang mengadopsi sistem IR adalah perpustakaan. Penggunaan IR pada perpustakaan terletak pada sistem pencarian buku, jurnal atau majalah pada komputer yang memungkinkan pencarian berdasarkan judul dan nama pengarang. Hasil dari pencarian akan menampilkan sejumlah daftar buku, jurnal atau majalah yang terdaftar pada sistem perpustakaan Selain itu penerapan lain dari IR yang umum digunakan yaitu pada web search engine.

2.4. Spelling Error

Spelling error merupakan keadaan di mana terjadi kesalahan penulisan susunan kata. Berdasarkan sejarahnya, awalnya keadaan ini berhubungan dengan kesalahan penulisan kata secara manual, namun saat ini hal tersebut juga dapat terjadi pada proses pengetikan yang dilakukan dengan bantuan mesin ketik dan komputer. Hal tersebut dapat terjadi dikarenakan kesalahan mekanik atau keluputan tangan atau jari saat mengetik, selain itu terkadang juga disebabkan oleh ketidaktahuan seseorang tentang bagaimana pengejaan tulisan yang benar.

Berdasarkan jenis katanya *spelling error* dapat dibedakan menjadi 2 tipe yaitu *non-word spelling error* dan *real-word spelling error*. *Non-word spelling error* merupakan kesalahan penulisan kata di mana kata tersebut tidak dapat ditemukan dalam kamus (tidak memiliki makna). Sedangkan *real-word spelling error* merupakan kesalahan penulisan kata di mana kata tersebut dapat ditemukan dalam kamus (memiliki makna) namun bukan kata yang dimaksud dalam dokumen (N et al. 2011). Dalam menangani *non-word spelling error*, dibutuhkan kandidat kata *real-word* yang mirip dengan kata yang salah dengan ketentuan memiliki nilai *edit distance* terpendek. Sedangkan untuk mengatasi *real-word spelling error* dibutuhkan kandidat kata dengan pengucapan pengejaan yang mirip.

2.5. Error Spelling Detection

Error Spelling Detection merupakan proses pengecekan validitas suatu kata dalam bahasa tertentu, suatu kata disebut valid jika kata tersebut dapat ditemukan dalam *lexical resource* (N et al. 2011). *Lexical resource* merupakan *database* di mana data di dalamnya dapat berupa *corpus*, *lexicon*, *wordlist* atau bentuk lain. Proses

utama dari *error detection* adalah membandingkan kata dalam teks dengan kata yang terdapat pada *lexical resource*. Banyak metode yang dapat digunakan untuk proses deteksi kesalahan penulisan kata, namun yang umumnya digunakan untuk deteksi *non-word error* adalah deteksi menggunakan *Dictionary Lookup* dan *N-Gram Analysis*.

Metode *dictionary lookup* merupakan metode yang sering digunakan dalam menentukan *non-word error*. Proses yang dilakukan pada metode ini yaitu melakukan pengecekan apakah kata yang dimaksud terdaftar dalam kamus atau tidak, jika tidak ada maka kata ini dianggap sebagai *non-word*. Cara ini termasuk cara yang efektif untuk menentukan kata termasuk salah penulisannya atau tidak, namun jumlah kata dalam kamus yang banyak dapat berakibat pada proses pengecekan menjadi lama, oleh karena itu dibutuhkan teknik optimasi pada teknik pencarian kata. Teknik optimasi dapat dilakukan dengan penggunaan *binary search* dan *hash* seperti pada penelitian (Soleh & Purwarianti 2011) dan (N et al. 2011). Permasalahan lain dari penggunaan kamus yaitu adanya kata yang sudah diberi imbuhan pada kamus, serta adanya kata asing, turunan kata dan kata baru yang tidak dapat diprediksi kemunculannya pada dokumen, selain itu pada beberapa bidang seperti kesehatan, ekonomi, biologi memiliki istilah khusus yang terkadang tidak ada dalam kamus umum. Menanggapi masalah tersebut, untuk mendapatkan kata yang relevan sesuai kebutuhan pengguna dibutuhkan *domain specificity* yaitu menentukan *domain* atau tema dokumen secara spesifik.

2.6. Error Spelling Correction

Error spelling correction merupakan proses yang dilakukan setelah proses *error spelling detection* selesai. Setelah menentukan bahwa suatu kata memiliki ejaan penulisan yang salah, pada proses ini dilakukan pencarian kata sebagai kandidat untuk mengoreksi kata yang salah ejaan tersebut. Dalam mengatasi *real-word error* dibutuhkan pengetahuan tambahan meliputi susunan kalimat yang benar dan sumber lain untuk mengekstraksi konteks kalimat (N et al., 2011). Sedangkan pada *non-word error* tidak membutuhkan pengetahuan dalam konteks kebahasaan untuk mencari kandidat koreksi kata karena dalam menentukan benar atau salahnya ejaan kata dapat dilihat dari ada atau tidaknya kata tersebut dalam *lexical resource*. Banyak metode yang dapat digunakan untuk koreksi *non-word error* antara lain *rule based methods, similarity key techniques,* dan *Minimum Edit Distance* yang terdiri dari metode *Levenshtein, Hamming, Damerau-Levenshtein,* dan *Longest Common Subsequence* (LCS). Pada penelitian ini metode koreksi yang digunakan adalah *Damerau-Levenshtein*.

2.6.1 Damerau-Levenshtein Distance

Algoritme *Damerau-Levenshtein Distance* merupakan algoritme pengembangan dari algoritme *Levenshtein Distance*. *Damerau-Levenshtein Distance* menentukan jumlah minimum operasi yang dibutuhkan untuk mengubah satu string menjadi string lain, di mana operasi yang digunakan sama dengan *Levenshtein Distance* yaitu *insertion*, *deletion*, *substitution* namun dengan penambahan operasi *transposition* diantara dua karakter (Damerau dalam Jupin, Shi, & Obradovic, 2013). Damerau tidak hanya membedakan 4 operasi *edit* tersebut, namun juga menyatakan bahwa operasi pada algoritme yang dikembangkan dapat sesuai dengan sekitar 80% dari semua kesalahan penulisan manusia. Setiap kesalahan berupa hilangnya karakter huruf, kelebihan karakter huruf, atau kesalahan urutan huruf dari dua karakter huruf yang berbeda (contoh: seharusnya tertulis “ka”, diketik dengan “ak”) dianggap sebagai 1 kesalahan sedangkan pada *Levenshtein* dianggap sebagai 2 kesalahan (Thang & Huy 2010).

Berikut ini merupakan perhitungan algoritme *Damerau-Levenshtein Distance* (Setiadi 2013).

$$D[0,0]=0 \quad (1)$$

$$D[i,0]=i \text{ for } i \in 0...m \quad (2)$$

$$D[0,j]=j \text{ for } j \in 0...n \quad (3)$$

$$\text{if } s[i]==t[j] \text{ then cost } =0 \text{ else cost}=1 \quad (4)$$

$$D[i,j]=\min \begin{cases} D[i-1][j]+1 \\ D[i][j-1]+1 \\ D[i-1][j-1]+cost \end{cases} \quad (5)$$

$$\text{if } (i>1 \text{ and } j>1 \text{ and } s[i]==t[j-1] \text{ and } s[j-1] ==t[i] \text{ then} \quad (6)$$

$$D[i,j]=\min \begin{cases} D[i][j] \\ D[i-2][j-2] \end{cases} \quad (7)$$

2.7. Penghitungan Kinerja Koreksi Kesalahan Penulisan Ejaan Kata

Perhitungan kinerja sistem koreksi kata dilakukan dengan menggunakan metode presisi dan *recall*. Presisi merupakan jumlah dokumen hasil *retrieve* sistem yang relevan, sedangkan *recall* merupakan jumlah dokumen relevan yang dihasilkan dari proses *retrieve* sistem (Manning et al., 2009). Perhitungan presisi dan *recall* secara berturut-turut ditunjukkan pada Persamaan 8 dan Persamaan 9.

$$\text{Presisi} = \frac{TP(Relevant\ Retrieved}{TP+FP(Retrieved)} \quad (8)$$

$$\text{Recall} = \frac{TP(Relevant\ Retrieved}{TP+FN(Relevant)} \quad (9)$$

Persamaan presisi dan recall pada Persamaan 8 dan Persamaan 9 dapat disusun dengan mengacu pada tabel kontingensi yang ditunjukkan pada Tabel 1.

Tabel 1. Tabel Kontingensi

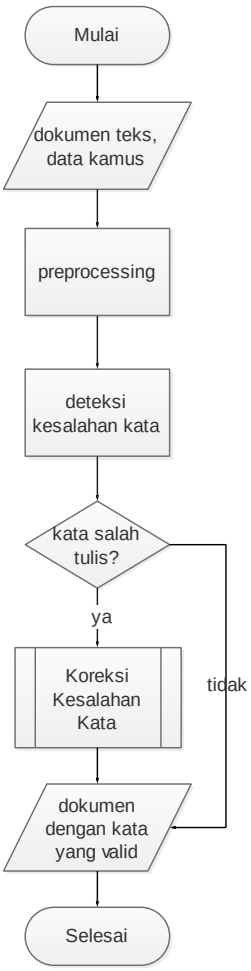
		Actual	
		Relevant	Nonrevelant
Prediction	Retrieved	True Positive (TP)	False Positive (FP)
	Not retrieved	False Negative (FN)	True Negative (TN)

Keterangan:
TP: hasil prediksi positif dan nilai sebenarnya juga bernilai positif (*true positive)*
TN: hasil prediksi negatif dan nilai sebenarnya juga bernilai negatif (*true negative)*
FN: hasil prediksi negatif sedangkan nilai sebenarnya positif *(false negative)*
FP: hasil prediksi positif sedangkan nilai sebenarnya negatif (*false positive)*

Nilai presisi terhadap nilai *recall* berbanding terbalik. Ketika nilai presisi mencapai nilai optimal maka nilai *recall* rendah, hal tersebut berlaku sebaliknya. Kondisi presisi yang tinggi berarti bahwa setiap hasil yang didapatkan dari hasil pencarian merupakan hasil yang relevan (namun tidak dapat dipastikan apakah semua dokumen relevan didapatkan pada hasil pencarian). Sedangkan kondisi *recall* yang tinggi berarti bahwa semua dokumen yang relevan didapatkan pada hasil pencarian (namun tidak dapat dipastikan berapa banyak dokumen yang tidak relevan yang juga dihasilkan pada proses pencarian).

3. PERANCANGAN DAN IMPLEMENTASI

Alur proses deteksi dan penentuan kandidat koreksi kesalahan ejaan kata menggambarkan langkah-langkah koreksi kesalahan penulisan kata yang diawali dengan proses deteksi kesalahan sampai dengan mendapatkan keluaran berupa kandidat koreksi kesalahan ejaan kata yang tepat. Dari sejumlah kandidat kata yang dihitung nilai jaraknya terhadap kata yang salah, sistem mengambil kata dengan nilai *edit distance* terkecil untuk dijadikan sebagai keluaran dari sistem. Gambar 1 merupakan alur proses deteksi dan koreksi kesalahan penulisan kata.



Gambar 1. Alur Proses Deteksi dan Koreksi Kesalahan Penulisan Kata Menggunakan *Dictionary Lookup* dan *Damerau-Levenshtein Distance*

Berdasarkan Gambar 1, proses deteksi dan koreksi kesalahan penulisan kata diawali dengan memberi masukan pada sistem yaitu berupa data dokumen jurnal yang ingin dideteksi kesalahannya dan data kamus Bahasa Indonesia sebagai pedoman ejaan kata yang tepat. Pada penelitian ini kamus yang digunakan didapatkan dari penelitian Sholeh & Purwarianti (2011). Setelah itu, proses dilanjutkan dengan *preprocessing*. Pertama dilakukan proses tokenisasi yaitu memisahkan setiap kata dari dokumen, menghilangkan angka dan simbol serta mengubah setiap huruf menjadi huruf kecil. Selanjutnya dilakukan *filtering* untuk menyeleksi kata-kata berbahasa Inggris. Hasil dari proses ini digunakan sebagai masukan pada proses deteksi kesalahan kata. Untuk menentukan suatu kata salah atau tidak yaitu dengan melakukan pencarian kata yang dimasukkan pada kamus. Jika kata ada dalam kamus maka kata tersebut termasuk kata yang valid atau benar. Sedangkan jika tidak ada dalam kamus maka kata tersebut dianggap sistem sebagai kata yang salah. Dari proses deteksi kesalahan kata maka akan didapatkan daftar kata yang salah. Setiap kata yang salah menjadi data masukan untuk proses koreksi dengan mencari kata pada kamus yang memiliki jarak *edit* minimum terhadap kata yang salah. Keluaran dari sistem adalah kandidat kata dengan jarak *edit* terkecil.

Perhitungan nilai jarak *edit Damerau-Levenshtein Distance* dapat diilustrasikan pada matriks seperti pada Tabel 2 dengan kata yang salah adalah "HUSUS" dan kata target "KURSUS". Nilai jarak *edit* dihitung pada setiap pertemuan baris dan kolom dimulai dari posisi indeks baris dan kolom pertama hingga baris dan kolom terakhir sebagai hasil akhir nilai jarak *edit*. Berikut ini merupakan contoh perhitungan nilai jarak *edit* pada posisi baris dan kolom terakhir.

Tabel 2. Contoh Manualisasi Perhitungan Nilai Jarak Edit Damerau-Levenshtein Distance

s/t		H	U	S	U	S
	0	1	2	3	4	5
K	1	1	2	3	4	5
U	2	2	1	2	3	4
R	3	3	2	2	3	4
S	4	4	3	2	3	3
U	5	5	4	3	2	3
S	6	6	5	4	3	2

$$s[5] = t[6] -> cost = 0$$

$$D[6,5] = min\begin{cases} D[6-1][5]+1 \\ D[6][5-1]+1 \\ D[6-1][5-1]+cost \end{cases}$$

$$\mathrm{D}[6,5] = \min\begin{cases} \mathrm{D}[6-1][5]+1 \\ \mathrm{D}[6][5-1]+1 \\ \mathrm{D}[6-1][5-1]+\mathrm{cost} \end{cases}$$

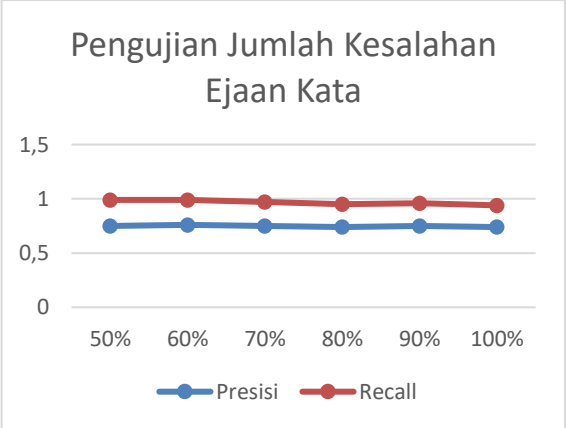
$$\mathrm{D}[6,5] = \min\begin{cases} 4 \\ 4 = 2 \\ 2 \end{cases}$$

4. PENGUJIAN DAN ANALISIS

Pada penelitian ini dilakukan pengujian terhadap hasil koreksi kesalahan dengan dua skenario pengujian yaitu pengujian untuk mengetahui pengaruh jumlah kesalahan ejaan kata dan jumlah kata terhadap nilai presisi dan *recall*.

4.1. Pengujian Jumlah Kesalahan Ejaan Kata

Pengujian jumlah kesalahan ejaan kata digunakan untuk mengetahui pengaruh jumlah kesalahan ejaan kata terhadap nilai presisi dan *recall* proses koreksi sistem. Pada pengujian ini data diambil dari 5 jurnal berbeda yang bersumber dari JTIIK dengan mengecualikan persamaan dan daftar pustaka jurnal. Pengujian dilakukan dengan membuat variasi jumlah kesalahan kata yang diujikan yaitu 50%, 60%, 70%, 80%, 90%, dan 100% dari keseluruhan kesalahan kata pada setiap dokumen. Pada Gambar 2 dipaparkan grafik hasil pengujian jumlah kesalahan ejaan kata.



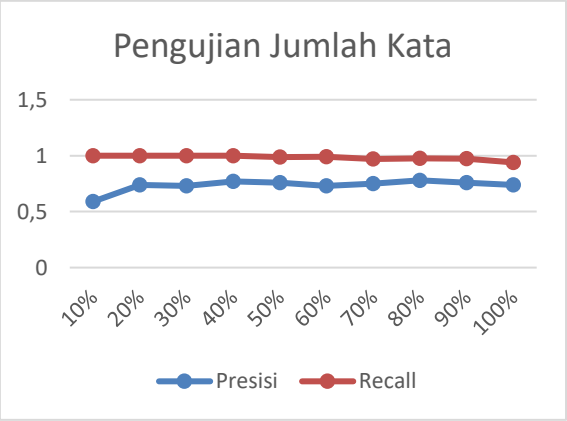
Gambar 2. Grafik Hasil Uji Coba Jumlah Kesalahan Ejaan Kata

Berdasarkan grafik hasil pengujian pada Gambar 2 menunjukkan bahwa nilai *recall* cenderung menurun dengan nilai tertinggi sebesar 0.99, sedangkan nilai presisi cenderung stabil dengan nilai tertinggi yaitu 0.76 pada

dokumen dengan jumlah kesalahan ejaan kata sebesar 60%. Berdasarkan hasil tersebut, tingginya nilai *recall* dibandingkan nilai presisi menandakan bahwa setiap kata yang relevan atau benar berhasil diambil oleh sistem sebagai hasil koreksi kata. Nilai *recall* yang tidak mencapai nilai optimal dikarenakan terdapat hasil koreksi kata sistem yang salah atau tidak relevan terhadap koreksi kata yang tepat. Sedangkan nilai presisi yang kurang tinggi disebabkan hasil koreksi sistem yang tidak hanya menampilkan koreksi kata yang tepat sesuai dengan nilai aktual namun juga memberikan kandidat koreksi kata lain dengan nilai jarak *edit* minimum yang sama.

4.2. Pengujian Jumlah Kata

Pengujian jumlah kata digunakan untuk mengetahui pengaruh jumlah kata dalam dokumen terhadap nilai presisi dan *recall* proses koreksi sistem. Pada pengujian ini menggunakan data yang sama dengan pengujian skenario pertama. Proses pengujian dilakukan dengan menentukan variasi jumlah dokumen yaitu 10%, 20%, 30%, … , 100% dari seluruh kata dalam setiap dokumen. Gambar 3 merupakan grafik hasil pengujian jumlah kata.



Gambar 3. Grafik Hasil Uji Coba Jumlah Kata

Berdasarkan grafik hasil pengujian pada Gambar 3, didapatkan bahwa bahwa nilai *recall* cenderung menurun dengan nilai tertinggi sebesar 1 pada dokumen dengan jumlah kata 10%, 20%, 30%, dan 40%, sedangkan nilai presisi cenderung stabil dengan nilai tertinggi yaitu 0.78 pada dokumen dengan jumlah kata 80% dari seluruh kata. Secara keseluruhan nilai *recall* lebih tinggi terhadap nilai presisi dengan beberapa hasil *recall* mencapai nilai optimal. Berdasarkan hasil tersebut dapat diambil kesimpulan bahwa hampir semua kata yang benar sesuai kondisi aktual berhasil diambil oleh sistem sebagai hasil koreksi sistem. Nilai *recall* yang menurun pada pengujian ini disebabkan terdapat hasil koreksi sistem yang salah atau tidak sama dengan kata yang diharapkan. Hal tersebut dapat terjadi dikarenakan koreksi kata yang tepat bukanlah kata dengan nilai jarak *edit* minimum terhadap kata yang salah. Sedangkan nilai presisi yang lebih rendah dibandingkan nilai *recall* disebabkan terdapat hasil koreksi sistem yang lebih dari 1 kandidat kata dengan jarak *edit* minimum. Nilai presisi dapat mencapai optimal pada setiap titik *recall* jika hasil koreksi sistem memberikan koreksi kata yang tepat sesuai yang diharapkan (Sutisna & Adisantoso 2010).

5. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini yaitu Metode *Dictionary Lookup* dan *Damerau-Levenshtein Distance* dapat diimplementasikan dengan baik pada proses deteksi dan koreksi kesalahan ejaan kata pada jurnal JTIIK. Pada skenario pengujian jumlah kesalahan ejaan kata didapatkan nilai presisi dan *recall* terbaik sebesar 0.76 dan 0.99. sedangkan nilai presisi dan *recall* terbaik sebesar 0.78 dan 1. Berdasarkan hasil pada kedua skenario menunjukkan bahwa nilai *recall* lebih tinggi daripada nilai presisi. Hasil tersebut menunjukkan bahwa semua koreksi kata yang diharapkan berhasil diambil oleh sistem sebagai hasil koreksi kata. Selain itu berdasarkan hasil pengujian yang didapatkan, jumlah kesalahan ejaan kata dan jumlah kata dalam dokumen tidak terlalu berpengaruh secara signifikan terhadap kinerja koreksi sistem. Hasil koreksi kesalahan kata lebih dipengaruhi oleh kelengkapan kata pada kamus sebagai acuan kandidat koreksi kata dan tipe kesalahan ejaan kata.

Aplikasi Deteksi Kesalahan Ejaan dan Penentuan Rekomendasi Koreksi Kata yang Tepat Pada Dokumen Jurnal JTIIK Menggunakan Dictionary Lookup dan Damerau-Levenshtein Distance dapat dikembangkan lagi untuk mendapatkan hasil yang lebih baik dengan penggunaan kamus Bahasa Indonesia yang lengkap dan terbaru serta dapat juga ditambahkan kamus khusus yang berisi istilah-istilah dalam penelitian bidang komputer. Metode yang digunakan juga dapat dikembangkan lagi sehingga dapat mendeteksi nama orang, nama tempat, dan istilah-istilah lain dalam topik tertentu. Selain itu dapat juga ditambahkan metode lain untuk memberikan

peringkat pada hasil penentuan kandidat koreksi dengan jarak minimum sehingga dapat diambil 1 kata terbaik. Hasil juga dapat lebih optimal dengan mempertimbangkan kata sebelum dan sesudah kata yang salah, kalimat sebelum dan kalimat sesudah, serta paragraf sebelum dan sesudah untuk menentukan topik bahasan dari dokumen. Topik bahasan yang didapatkan berguna untuk menentukan kata mana yang tepat untuk dijadikan koreksi kata dari seluruh kandidat koreksi sistem dengan nilai jarak edit terkecil.

DAFTAR PUSTAKA

Expertsystem, 2016. *Natural Language Processing and Text Mining.* [online] Tersedia di: <http://www.expertsystem.com/natural-language-processing-and-text-mining> [Diakses 26 September 2016]

Jupin, J., Shi, J.Y. & Obradovic, Z., 2013. Understanding Cloud Data Using Approximate String Matching and Edit Distance. , pp.1234–1243.

Liyana, N., Shuib, M. & Abdullah, N., 2010. The Use of Information Retrieval Tools : a Study of Computer Science Postgraduate Students. , (Cssr), pp.379–384.

Manning, C.D., Raghavan, P. & Schütze, H., 2009. *An Introduction to Information Retrieval* Online Edi., Cambridge University Press. Available at: http://www.informationretrieval.org/.

Mishra, R. & Kaur, N., 2013. A Survey of Spelling Error Detection and Correction Techniques. , 4, pp.372–374.

Muradmaulana, 2014. Sejarah Tradisi Tulis: Dari Masa ke Masa. [online] Tersedia di: <http://www.muradmaulana.com/2014/06/sejarah-tradisi-tulis-menulis-dari-masa.html> [Diakses 25 September 2016]

N, A.R. et al., 2011. Application of Document Spelling Checker for Bahasa Indonesia. , pp.978–979.

Setiadi, I., 2013. Damerau-Levenshtein Algorithm and Bayes Theorem for Spell Checker Optimization Damerau-Levenshtein Algorithm and Bayes Theorem for Spell Checker Optimization. , (November).

Soleh, M.Y. & Purwarianti, A., 2011. A Non Word Error Spell Checker for Indonesian using Morphologically Analyzer and HMM. , (July).

Sutisna, U. & Adisantoso, J., 2010. Koreksi Ejaan Query Bahasa Indonesia Menggunakan Algoritme Damerau Levenshtein. , 15(2), pp.25–29.

Thang, D.Q. & Huy, P.T., 2010. Determining restricted Damerau-Levenshtein edit-distance of two languages by extended automata.

Witten, I.H., 2002. Text mining.