Jurnal Ilmu Komputer Agri-Informatika
Volume 7 Nomor 2 halaman 105 – 113
e ISSN: 2654-9735, p ISSN: 2089-6026

Tersedia secara *online* di:
http://journal.ipb.ac.id/index.

Penerjemahan Bahasa Indonesia ke Bahasa Minang dari *Optical Character Recognition* dengan Menggunakan Algoritme Edit Distance

Translating Indonesian into Minang Languages from Optical Character Recognition Using the Edit Distance Algorithm

MAYANDA MEGA SANTONI[1]*, NURUL CHAMIDAH[1], DESTA SANDYA PRASVITA[1], REZA AMARTA PRAYOGA[2], BAYU PERMANA SUKMA[2]

Abstrak

Tri Gatra Bangun Bahasa yaitu utamakan bahasa Indonesia, lestarikan bahasa daerah dan kuasai bahasa asing. Melalui ini, maka bahasa daerah sebagai salah satu kekayaan bangsa Indonesia perlu dilestarikan. Selain itu, bahasa daerah juga berfungsi sebagai pendukung bahasa nasional yakni bahasa Indonesia. Pemanfaatan teknologi dapat digunakan sebagai upaya dalam pelestarian bahasa daerah. Penelitian ini memanfaatkan teknologi kecerdasan buatan yakni mesin penerjemah yang menerjemahkan bahasa Indonesia ke bahasa daerah berbasiskan citra teks. Bahasa daerah yang digunakan yakni bahasa daerah Minang. Fokus penelitian ini pada proses penerjemahan hasil *Optical Character Recognition* (OCR) dari citra teks bahasa Indonesia menggunakan algoritme Edit Distance, yakni Hamming Distance, Leveinshtein Distance dan Jaro-Winkler. Hasil penelitian ini menunjukkan bahwa algoritme Edit Distance dapat memperbaiki hasil OCR dalam melakukan penerjemahan ke bahasa daerah. Hasil OCR pada citra teks memiliki akurasi awal yakni 50.72%. Setelah diterapkan algoritme Edit Distance, akurasi penerjemahan meningkat menjadi 68.34% untuk algoritme Hamming Distance, 70.5% untuk algoritme Leveinshtein Distance dan 70.2% untuk algoritme Jaro-Winkler. Dari ketiga algoritme ini, Leveinshtein Distance memiliki performasi akurasi penerjemahan paling tinggi.

Kata Kunci**:** penerjemahan, bahasa indonesia, bahasa minang, *hamming distance*, *leveinshtein distance*, jaro-winkler, *optical character recognition*

Abstract

Tri Gatra Bangun Bahasa, namely prioritizing Indonesian, preserving local languages and mastering foreign languages. Through this, the local language as one of the wealth of the Indonesian nation needs to be preserved. In addition, local languages also function as a supporter of the national language, namely Indonesian. The use of technology can be used as an effort to preserve local languages. This study utilizes artificial intelligence technology, namely machine translation that translates Indonesian into local languages based on text images. The local language used is the Minang language. The focus of this research is the process of translating the results of Optical Character Recognition from Indonesian text images using the edit distance algorithm, namely Hamming distance, Leveinshtein distance and Jaro-Winkler. The results of this study indicate that the edit distance algorithm can improve the OCR results in translating into local languages. OCR results on text images have an initial accuracy of 50.72%. After applying the edit distance algorithm, the translation accuracy increases to 68.34% for the Hamming distance algorithm, 70.5% for the Leveinshtein distance algorithm and 70.2% for the Jaro-Winkler algorithm. Of the three algorithms, Leveinshtein distance has the highest translation accuracy performance.

*Keywords**:** translation, indonesian language, minang language, hamming distance, leveinshtein distance, jaro-winkler, optical character recognition*

[1]Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jakarta;
[2]Badan Pengembangan dan Pembinaan Bahasa, Kementerian Pendidikan dan Kebudayaan;

PENDAHULUAN

Bahasa daerah merupakan alat komunikasi intraetnik. Selain itu, bahasa daerah juga berfungsi sebagai pendukung bahasa nasional, yakni bahasa Indonesia. Bahasa daerah juga merupakan salah satu kekayaan budaya yang dimiliki bangsa Indonesia. Oleh karena itu, pelestarian bahasa daerah harus terus digiatkan dan dikembangkan untuk dapat memperkokoh ketahanan budaya bangsa (Asrif 2010).

Kemajuan teknologi harus dapat dimanfaatkan dalam upaya melestarikan bahasa daerah. Teknologi kecerdasan buatan merupakan salah satu teknologi yang dapat menjawab tantangan tersebut. Kecerdasan buatan merupakan sebuah teknologi yang dapat mengadopsi kecerdasan yang dimiliki manusia untuk diimplementasikan ke dalam sebuah komputer. Mesin penerjemah merupakan salah satu teknologi kecerdasan buatan yang dapat digunakan.

Mesin penerjemah bahasa Indonesia ke bahasa daerah sudah banyak dikembangkan. Nuraini dan Firmansyah mengembangkan kamus terjemahan digital bahasa aceh ke Bahasa Indonesia berbasis web (Nuraini dan Firmansyah 2020). Januardi *et al.* juga mengembangkan aplikasi berbasis Android dalam melakukan prediksi pencarian kata dalam kamus bahasa Manggarai (Nusa Tenggara Timur), Indonesia dan Inggris (Januardi *et al.* 2019). Priyanto dan Ulinnuha juga melakukan perancangan aplikasi penerjemahan Bahasa Indonesia ke bahasa Jawa berbasis Android (Priyanto dan Ulinnuha 2017).

Dari ketiga penelitian ini, mesin penerjemah memiliki inputan berupa teks yang kemudian teks tersebut diterjemahkan sesuai dengan bahasa yang diinginkan. Selain dalam bentuk teks, mesin penerjemah juga bisa menerjemahkan teks yang terdapat dalam sebuah gambar. Teks yang terdapat di dalam gambar diekstrak terlebih dahulu menjadi teks atau biasa disebut dengan *Optical Character Recognition* (OCR). Hasil dari OCR berupa teks yang akan diterjemahkan ke dalam bahasa daerah yang diinginkan.

Algoritme Edit Distance banyak digunakan pada pemrosesan data teks khususnya pada mesin penerjemahan. Algoritme Edit Distance merupakan sebuah algoritme yang biasa digunakan untuk mendeteksi kesalahan ejaan. Jika terdapat dua buah kata, maka edit distance adalah jumlah operasi minimum untuk mengubah satu *string* ke *string* lainnya. Semakin sedikit jumlah operasinya maka semakin mirip dua kata terebut. Terdapat beberapa algoritme pada Edit Distance, yakni Hamming Distance, Levenshtein Distance dan Jaro-Winkler.

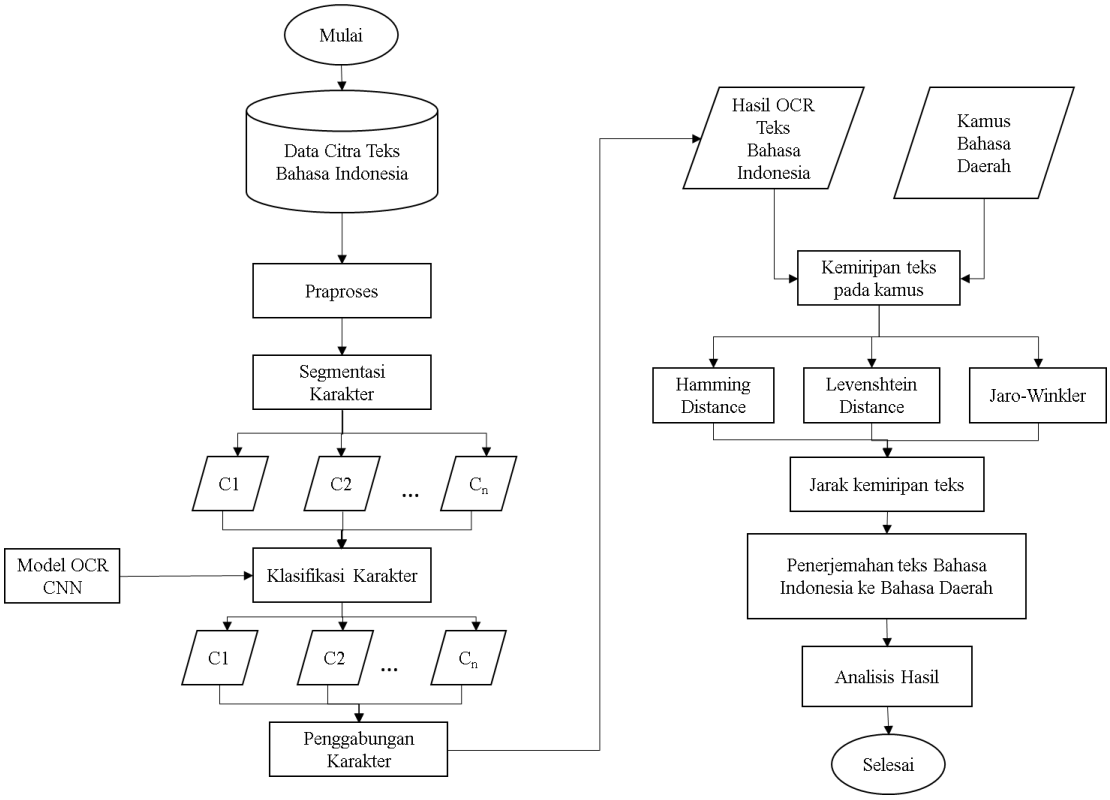
Przybocki *et al.* menerapkan algoritme Edit Distance dalam mengevaluasi sebuah mesin penerjemah (Przybocki *et al.* 2006). Pada penelitian Hu *et al.* (2015) digunakan algoritme Hamming Distance untuk memperkirakan kemiripan dalam pencarian teks. Hossain *et al.* menggunakan Levenshtein Distance dalam melakukan perbaikan otomatis pada hasil terjemahan bahasa Bengali ke bahasa Inggris dengan tingkat akurasi sebesar 78.13% (Hossain *et al.* 2019). Penelitian Wint *et al.* juga menggunakan Levenshtein Distance untuk melalukan perbaikan ejaan pada dataset sosial media dengan tingkat akurasi 90% (Wint *et al.* 2017). Levenshtein Distance juga digunakan untuk melakukan *autocomplete* dan *spell checking* dalam proses pencarian data perpustakaan (Yulianto *et al.* 2018). Cahyono dalam penelitiannya menggunakan algoritme Jaro-Winkler dan algoritme Paragraph Vector dalam melakukan perbandingan kemiripan dokumen pada dokumen ilmiah.

Berdasarkan penelitian-penelitian terdahulu, penulis menggunakan algoritme Edit Distance yakni Hamming Distance, Levenshtein Distance dan Jaro-Winkler dalam menerjemahkan teks bahasa Indonesia ke bahasa daerah dari hasil *Optical Character Recognition*.

METODE

Metode penelitian ini secara garis besar diilustrasikan pada Gambar 1. Fokus pembahasan pada tulisan ini pada proses penerjemahan hasil OCR teks bahasa Indonesia ke bahasa daerah. Proses OCR menggunakan model OCR *Convolutional Neural Networks* (CNN). Setelah

diperoleh hasil OCR, maka didapatkan teks bahasa Indonesia yang kemudian akan dilakukan pencarian kemiripan teks pada kamus bahasa daerah. Kamus bahasa daerah yang digunakan sebagai sampel penelitian adalah kamus bahasa Minang. Jumlah kata yang digunakan sebanyak 200 kata yang berasal dari daftar kata swadesh. Daftar kata ini merupakan daftar kosa kata dasar yang sering ditemukan di hampir semua bahasa (Maslakhah 2019). Daftar kata ini digunakan sebagai acuan pemilihan kata pada proses penerjemahan kata bahasa Indonesia ke bahasa daerah.



Gambar 11 Metode penelitian.

Data citra teks diperoleh dengan pengambilan gambar menggunakan lima jenis perangkat/*device* yang berbeda, yakni *Scanner*, Samsung Galaxy J5, iPhone 4, iPhone 5s, dan iPhone 7. Setiap citra teks menggunakan lima fon yang berbeda yakni Arial (fon 1), Bodoni (fon 2), Calibri (fon 3), Helvetica (fon 4) dan Times New Roman (fon 5). Spesifikasi perangkat dapat dilihat pada Tabel 1.

Tabel 1 Tingkat akurasi terjemahan pada *device* 1

Jenis Perangkat (Kode)	**Megapixel/ Scanner Type**	**Bukaan (f/)/ Sensor Type**	**Ukuran piksel / Optical Resolution**
Scanner EPSON (D1)	600 x 1200 dpi	CIS	Flatbed Colour Image Scanner
Samsung Galaxy J5 (D2)	13 MP	f/1.9	1.15 μm
Iphone 4 (D3)	15 MP	f/2.8	1.5 μm
Iphone 5S (D4)	8 MP	f/2.2	1.5 μm
Iphone 7 (D5)	12 MP	f/1.8	1.22 μm

Setiap citra teks dilakukan praproses berupa *thresholding* untuk mengubah citra RGB menjadi citra biner. Citra biner selanjutnya dilakukan segmentasi yakni memisahkan setiap huruf yang terdapat pada citra teks. Setiap huruf dari citra teks akan diklasifikasi dengan menggunakan model OCR CNN. Hasil prediksi dari setiap huruf akan digabungkan kembali, sehingga menghasilkan teks bahasa Indonesia.

Selanjutnya adalah tahapan penerjemahan. Setiap teks bahasa Indonesia hasil prediksi OCR diterjemahkan ke dalam bahasa daerah Minang dengan cara menghitung nilai *similarity* (kemiripan) teks bahasa Indonesia dari hasil OCR dengan teks bahasa Indonesia yang terdapat dalam kamus bahasa daerah minang. Perhitungan nilai *similarity* menggunakan tiga metode yakni Hamming Distance, Leveinshtein Distance dan Jaro-Winkler.

Hamming Distance

Hamming Distance merupakan sebuah algoritme jarak yang dapat menghitung kemiripan dari dua buah kata. Algoritme ini akan membandingkan dua buah kata yang sama dengan membandingkan setiap karakter pada posisi yang sama. Nilai dari hamming distance adalah jumlah karakter yang berbeda pada dua buah kata yang dibandingkan. Operasi yang dilakukan pada hamming distance hanya operasi substitusi yakni operasi menukar huruf. Ilustrasi dari algoritme Hamming Distance dapat dilihat pada Gambar 2.

	[0]	[1]	[2]	[3]	[4]	[5]
S1	b	a	h	a	s	a
S2	b	a	t	o	s	a
Hamming distance	-	-	h	a	-	-
Nilai jarak	0	0	1	1	0	0

Gambar 12 Ilustrasi algoritme Hamming Distance.

Leveinshtein Distance

Jarak dua buah kata pada algoritme Leveinshtein Distance adalah dengan menghitung jumlah transformasi kata yang diperlukan untuk mengubah kata 1 menjadi kata 2. Operasi yang digunakan yakni penghapusan, penyisipan dan penggantian (Hossain *et al.* 2019). Ilustrasi algoritme Leveinshtein Distance dapat dilihat pada Tabel 2.

Tabel 2 Ilustrasi algoritme Leveinshtein Distance

Operasi	**S_{sumber} = bakhosa**	**S_{traget} = bahasa**
Substitusi k dengan h	bahhosa	bahasa
Substitusi h dengan a	bahaosa	bahasa
Substitusi o dengan s	bahassa	bahasa
Substitusi s dengan a	bahasaa	bahasa
Meghapus a	bahasa	bahasa
Total transformasi	**5**	

Jaro-Winkler

Berbeda dengan Hamming Distance dan Leveinshtein Distance, dimana semakin banyak operasi yang dibutuhkan untuk mentranformasikan kata 1 dengan kata 2, maka semakin berbeda dua buah kata tersebut. Pada Jaro-Winkler, semakin tinggi nilainya, maka akan semakin mirip dua buah kata tersebut. Terdapat tiga tahapan dalam algoritme Jaro-Winkler: 1) Menghitung panjang kata; 2) Menghitung jumlah karakter yang sama dalam dua buah kata; dan 3). Menghitung jumlah tansposisi (Kornain *et al.* 2014). Rumus menghitung jarak pada algoritme Jaro-Winkler dapat dilihat pada Persamaan 1.

$$d_{jw} = d_j + (l \,.\, p\,(1\text{-}d_j), \text{ dengan } d_j = \frac{1}{3} + \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m\text{-}t}{m}\right) \qquad (1)$$

dengan d_{jw} adalah jarak Jaro-Winkler, l adalah panjang karakter yang sama sampai ditemukan ketidaksamaan, p adalah konstanta *scaling factor*. d_j adalah jarak Jaro untuk kata 1 (s_1) dan kata 2 (s_2), m adalah jumlah karakter yang sama dan t adalah jumlah transposisi.

Evaluasi

Untuk mengevaluasi performa masing-masing metode, pengujian akan dilakukan dengan menggunakan nilai akurasi. Nilai evaluasi ini digunakan untuk mengetahui performa metode Edit Distance dalam menerjemahkan hasil OCR teks bahasa Indonesia ke bahasa daerah. Untuk menghitung nilai akurasi menggunakan persamaan di bawah ini:

$$\text{Akurasi} = \frac{\text{jumlah kata diterjemahkan benar}}{\text{total teks hasil OCR}} \times 100\% \quad (2)$$

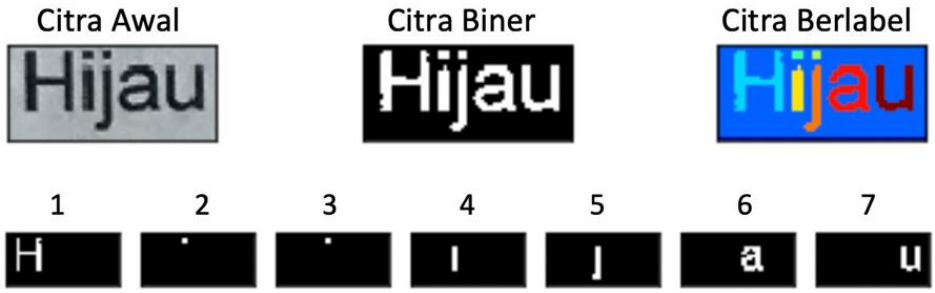
HASIL DAN PEMBAHASAN

Jumlah citra teks hasil OCR diperoleh sebanyak 200 kata. Daftar kata diambil dari daftar data swadesh yang diperoleh dari Laboratorium Kebinekaan Bahasa dan Sastra, Kementerian Pendidikan dan Kebudayaan. Kamus bahasa daerah yang digunakan merupakan kamus daerah bahasa Minang yang berasal dari Sumatera Barat. Daftar kata dapat dilihat pada Tabel 3.

Tabel 3 Kamus Bahasa Indonesia dan Bahasa Minang

Daftar kata Swadesh							
Bahasa Indonesia	**Bahasa Minang**	**Bahasa Indonesia**	**Bahasa Minang**	**Bahasa Indonesia**	**Bahasa Minang**	**Bahasa Indonesia**	**Bahasa Minang**
Abu	Abu	Danau	Danau	Kanan	Kanan	Pendek	Pendek
Air	Aia	Darah	Darah	Karena	Karano	Peras	Pareh
Akar	Aka	Datang	Tibo	Kecil	Ketek	Perempuan	Padusi
Anak	Anak	Daun	Daun	Kepala	Kapalo	Pergi	Pai
Angin	Angin	Debu	Debu	Kering	Kariang	Perut	Paruik
Anjing	Anjiang	Dekat	Dakek	Kiri	Kida	Pikir	Pikia
Apa	Apo	Dengan	Jo	Kotor	Kumuah	Pohon	Pohon
Api	Api	Dengar	Danga	Kuku	Kuku	Potong	Potong
Asap	Asok	Dingin	Dingin	Kulit	Kulik	Punggung	Pungguang
Awan	Awan	Dorong	Dorong	Kuning	Kuniang	Pusar	Pusek
Ayah	Apak	Dua	Duo	Kutu	Kutu	Putih	Putiah
Bagaimana	Baa	Duduk	Duduak	Lain	Lain	Rambut	Rambuik
Bagus	Rancak	Ekor	Ikua	Langit	Langik	Rumput	Rumpuik
Baik	Elok	Empat	Ampek	Laut	Lauik	Sakit	Sakik
Bakar	Baka	Engkau	Waang	Lebar	Leba	Satu	Ciek
Balik	Baliak	Gali	Galia	Leher	Lihia	Saya	Awak
Banyak	Banyak	Garam	Garam	Lelaki	Lelaki	Sayap	Sayok
Baring	Golek	Garuk	Garuak	Lempar	Gado	Sedikit	Saketek
Baru	Baru	Gemuk	Gamuak	Licin	Licin	Sempit	Sampiak
Basah	Babiyak	Gigi	Gigik	Lidah	Lidah	Semua	Kasado
Batu	Batu	Gigit	Gigik	Lihat	Caliak	Siang	Siang
Beberapa	Bara	Gosok	Gosok	Lima	Limo	Siapa	Sia
Benar	Bana	Gunung	Gunuang	Ludah	Ludah	Suami	Suami
Bengkak	Bangkek	Hantam	Antam	Lurus	Luruih	Sungai	Batang Aia
Benih	Baniah	Hapus	Apuih	Lutut	Lutuik	Tahu	Tau
Berat	Barek	Hati	Ati	Main	Main	Tahun	Tahun
Berburu	Baburu	Hidung	Hiduang	Makan	Makan	Tajam	Tajam
Berdiri	Tagak	Hidup	Hiduik	Malam	Malam	Takut	Takuik
Berenang	Baranang	Hijau	Hijau	Mata	Mato	Tali	Tali
Beri	Agiah	Hisap	Hisap	Matahari	Matoari	Tanah	Tanah
Berjalan	Berjalan	Hitam	Hitam	Mati	Mati	Tangan	Tangan
Berkata	Bakecek	Hitung	Hituang	Membelah	Mambalah	Tarik	Tariak
Berkelahi	Bacakak	Hujan	Ujan	Mengalir	Mangalia	Tebal	Taba
Besar	Gadang	Hutan	Utan	Mengapung	Maapuang	Telinga	Talingo
Bilamana	Bilo	Ia	Inyo	Menikam	Manikam	Telur	Talua
Binatang	Binatang	Ibu	Bundo	Merah	Merah	Terbang	Tabang
Bintang	Bintang	Ikan	Ikan	Mereka	Mereka	Tertawa	Galak
Buah	Buah	Ikat	Ikek	Minum	Minum	Tetek	Tetek
Bulan	Bulan	Ini	Iko	Mulut	Muluik	Tidak	Indak
Bulu	Bulu	Isteri	Bini	Muntah	Muntah	Tidur	Lalok
Bunga	Bungo	Itu	Tu	Nama	Namo	Tiga	Tigo
Bunuh	Bunuah	Jahit	Jaik	Nanti	Beko	Tipis	Tipih
Buruk	Buruak	Jantung	Jantuang	Napas	Napas	Tiup	Tiup
Burung	Buruang	Jatuh	Jatuah	Nyanyi	Nyanyi	Tongkat	Tungkek
Busuk	Busuak	Jauh	Jauah	Orang	Urang	Tua	Tuo
Cacing	Caciang	Kabut	Kabut	Pada	Pado	Tulang	Tulang
Cium	Ineh	Kaki	Kaki	Panas	Paneh	Tumpul	Tumpua
Cuci	Basuah	Kalau	Kalau	Panjang	Panjang	Uang	Pitiah
Daging	Dagiang	Kami	Kami	Pasir	Pasia	Ular	Ula
Dan	Dan	Kamu	Waang	Pegang	Pacik	Usus	Usus

Citra teks hasil OCR banyak mengalami kegagalan dalam memprediksi huruf, khususnya untuk huruf i dan j. Pada dua huruf tersebut akan tersegmentasi menjadi dua buah objek dikarenakan terdapatnya objek titik. Contoh kegagalan segmentasi dapat dilihat pada Gambar 3.



Gambar 13 Kesalahan pelabelan objek bukan huruf.

Tabel 4 Akurasi klasifikasi kata berdasarkan *device* dan *font*

Font/ Perangkat	Akurasi (%)					
	D1	D2	D3	D4	D5	Rata-rata
F1	85.0	81.0	25.0	82.5	96.0	73.9
F2	13.0	5.0	1.0	7.5	17.5	8.8
F3	50.5	80.0	4.0	48.0	89.5	54.4
F4	85.5	88.5	46.0	85.5	96.0	80.3
F5	43.5	45.5	4.5	39.5	48.0	36.2
Rata-rata	55.5	60.0	16.1	52.6	69.4	50.72

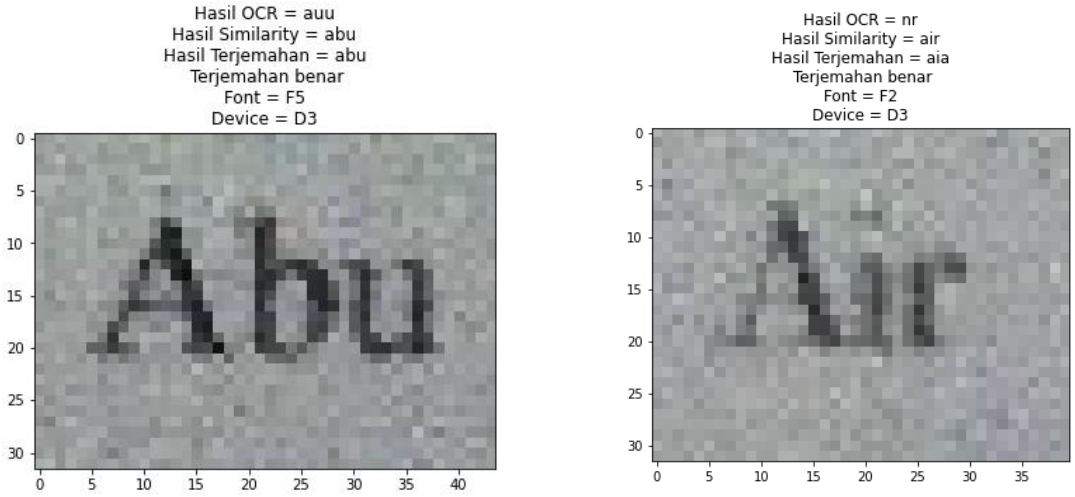
Hasil klasifikasi 200 citra teks dapat dilihat pada Tabel 4. Akurasi rata-rata untuk semua jenis perangkat dan fon adalah 50.72%. Akurasi terendah, jenis perangkat 3 (Iphone 4) dan tertinggi pada jenis perangkat 5 (Iphone 7). Sementara itu, untuk jenis fon, akurasi terendah pada fon 2 (Bodoni) dan akurasi tertinggi pada fon 4 (Helvetica).

Selanjutnya, hasil OCR dilakukan terjemahan menggunakan algoritme Hamming Distance, Leveinshtein Distance dan Jaro-Winkler. Akurasi terjemahan kata menggunakan algoritme hamming distance dapat dilihat pada Tabel 5. Akurasi rata-rata terjemahan kata menggunakan algoritme Hamming Distance adalah 68.34%.

Tabel 5 Akurasi terjemahan kata menggunakan algoritme Hamming Distance

Fon/ Perangkat	Akurasi (%)					
	D1	D2	D3	D4	D5	Rata-rata
F1	95.0	89.5	**68.5**	**95.0**	95.0	88,6
F2	35.0	19.5	5.5	25.0	29.0	22.8
F3	**91.0**	**95.5**	43.5	**92.5**	95.5	83.6
F4	**95.5**	94.0	**77.0**	94.5	95.5	91.3
F5	66.0	62.0	25.0	64.5	59.5	55.4
Rata-rata	76.5	72.1	**43.9**	74.3	74.9	68.34

Contoh keberhasilan algoritme Hamming Distance dalam melakukan terjemahan teks bahasa Indonesia hasil OCR dapat dilihat pada Gambar 4. Hasil OCR yang tadinya terjadi kegagalan yakni teks “abu” terbaca “auu”. Namun setelah dilakukan Perhitungan Hamming Distance pada kamus bahasa daerah diperoleh kemiripan kata dengan kata abu, sehingga hasil terjemahan yang dihasilkan benar.



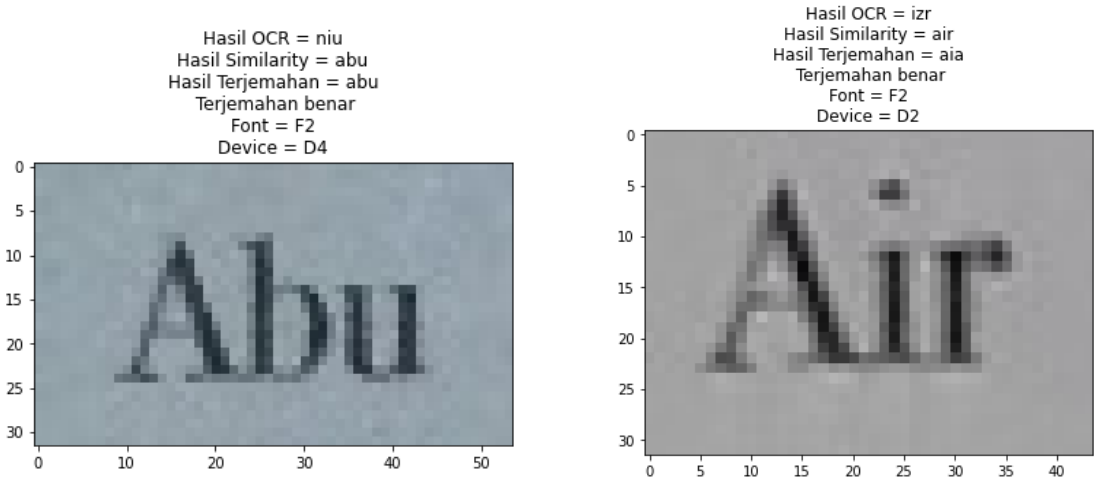
Gambar 14 Kasus berhasil pada algoritme Hamming Distance.

Akurasi terjemahan kata menggunakan algoritme Leveinshtein Distance dapat dilihat pada Tabel 6. Akurasi rata-rata terjemahan kata menggunakan algoritme ini adalah 70.5%.

Tabel 6 Akurasi terjemahan kata menggunakan algoritme Leveinshtein Distance

Fon/ Perangkat	Akurasi (%)					
	D1	**D2**	**D3**	**D4**	**D5**	**Rata-rata**
F1	**95.5**	**91.0**	65.0	**95.0**	**95.5**	88.4
F2	42.0	**23.5**	7.0	**31.0**	43.5	29.3
F3	89.5	**95.5**	39.5	89.5	**96.0**	82.0
F4	**95.5**	**95.0**	**77.0**	94.5	**96.0**	91.6
F5	70.5	70.5	**27.0**	68.5	69.5	61.2
Rata-rata	**78.6**	**75.1**	43.1	**75.7**	80.1	70.5

Contoh keberhasilan algoritme Leveinsthein Distance dalam melakukan terjemahan teks bahasa Indonesia hasil OCR dapat dilihat pada Gambar 5. Hasil OCR yang tadinya terjadi kegagalan yakni teks “abu” terbaca “niu”. Namun setelah dilakukan perhitungan Leveinshtein Distance pada kamus bahasa daerah diperoleh kemiripan kata dengan kata abu sehingga hasil terjemahan yang dihasilkan benar.



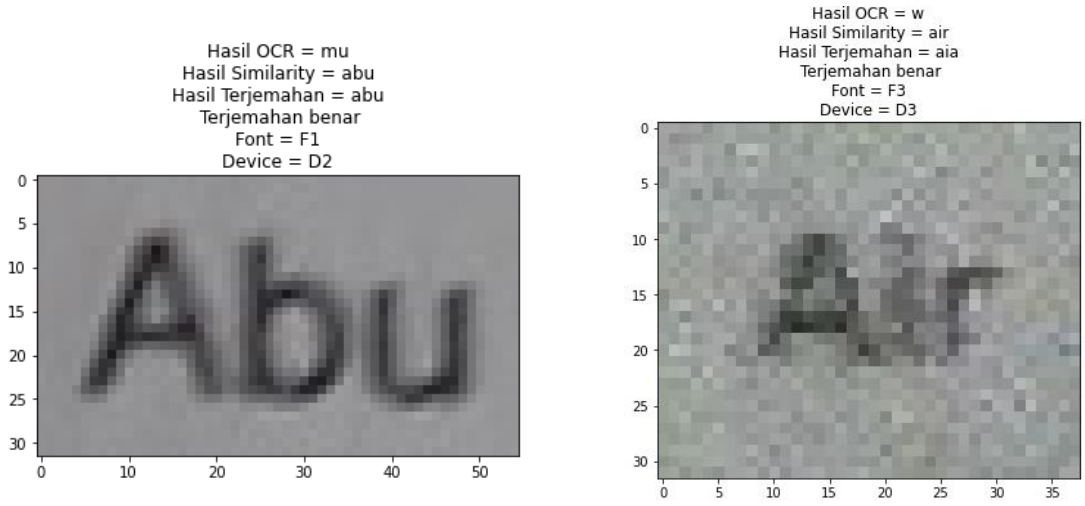
Gambar 15 Kasus berhasil pada algoritme Leveinsthein Distance.

Akurasi terjemahan kata menggunakan algoritme Jaro-Winkler dapat dilihat pada Tabel 7. Akurasi rata-rata terjemahan kata menggunakan algoritme Jaro-Winkler adalah 70.2%.

Tabel 7 Akurasi Terjemahan Kata Menggunakan Algoritme Leveinshtein Distance

Fon/ Perangkat	Akurasi (%)					
	D1	**D2**	**D3**	**D4**	**D5**	**Rata-rata**
F1	95.0%	89.0%	63.0%	94.5%	95.0%	87.3
F2	**42.5%**	21.0%	**10.5%**	29.0%	**50.0%**	30.6
F3	84.5%	95.0%	32.0%	83.5%	95.0%	78.0
F4	94.0%	93.5%	76.5%	**95.0%**	94.5%	90.7
F5	**76.0%**	**72.0%**	22.0%	**73.5%**	**77.5%**	64.2
Rata-rata	78.4%	74.1%	40.8%	75.1%	82.4%	70.2

Contoh keberhasilan algoritme Jaro-Winkler distance dalam melakukan terjemahan teks bahasa Indonesia hasil OCR dapat dilihat pada Gambar 6. Hasil OCR yang tadinya terjadi kegagalan yakni teks “abu” terbaca “niu”. Namun setelah dilakukan perhitungan Leveinshtein Distance pada kamus bahasa daerah diperoleh kemiripan kata dengan kata abu sehingga hasil terjemahan yang dihasilkan benar.



Gambar 16 Kasus berhasil pada algoritme Jaro-Winkler/

Algoritme Hamming Distance, Leveinshtein Distance dan Jaro-Winkler dapat meningkatkan hasil pengenalan teks hasil OCR. Pada pengenalan teks hasil OCR, akurasi rata-rata diperoleh sebesar 50.72%. Hasil ini meningkat dengan menggunakan Algoritme Hamming Distance dengan akurasi rata-rata 68.34%, Leveinshtein Distance dengan akurasi rata-rata 70.5% dan Jaro-Winkler dengan akurasi rata-rata 70.2%. Dari ketiga algoritme tersebut, algoritme Leveinshtein Distance memiliki tingkat keberhasilan paling tinggi.

SIMPULAN

Pada penelitian ini telah dilakukan penerjemahan teks bahasa Indonesia ke bahasa daerah dari hasil *Optical Character Recognition* (OCR) pada citra teks bahasa Indonesia. Penerjemahan dilakukan menggunakan algoritme *similarity* yakni Edit Distance. Algoritme-algoritme Edit Distance yang digunakan adalah Hamming Distance, Leveinshtein Distance dan Jaro-Winkler. Algoritme Edit Distance dapat meningkatkan hasil akurasi terjemahan dari klasifikasi kata pada citra teks. Sebelum dilakukan terjemahan akurasi klasifikasi kata yakni 50.72%. Akurasi ini meningkat setelah diterapkan algoritme Edit Distance untuk mencari kemiripan kata pada kamus bahasa daerah. Dari ketiga algoritme Edit Distance, Leveinshtein Distance memiliki akurasi paling tinggi yakni 70.5%.

UCAPAN TERIMA KASIH

Terima kasih kepada Universitas Pembangunan Nasional Veteran Jakarta yang telah mendanai penelitian ini pada hibah penelitian internal Riset Dosen Pemula tahun 2020.

DAFTAR PUSTAKA

Asrif. 2010. Pembinaan dan Pengembangan Bahasa Daerah dalam Memantapkan Kedudukan dan Fungsi Bahasa Indonesia. *Mabasan* 4(1): 11–23.

Hossain MM, Labib MF, Rifat AS, Das AK, Mukta M. 2019. Auto-correction of English to Bengali Transliteration System using Levenshtein Distance. *2019 7th International Conference on Smart Computing and Communications, ICSCC 2019*. IEEE. hlm 1–5. doi: 10.1109/ICSCC.2019.8843613.

Hu H, Zhang L, Wu J. 2015. Hamming distance based approximate similarity text search algorithm. *2015 7th International Conference on Advanced Computational Intelligence, ICACI 2015*. hlm 1–6. doi: 10.1109/ICACI.2015.7184772.

Januardi DO, Budianto AE, S MPT. 2019. Prediksi Pencarian Kata dengan Algoritme Levenshtein Distance di dalam Kamus Bahasa Manggarai , Indonesia dan Inggris Berbasis Android. hlm. 45–51.

Kornain A, Yansen F, Tinaliah T. 2014. Penerapan Algoritme Jaro-Winkler Distance Untuk Sistem Pendeteksi Plagiarisme Pada Dokumen Teks Berbahasa Indonesia. *Program Studi Teknik Informatika STMIK GI MDP*. hlm 1–10. Tersedia pada: http://eprints.mdp.ac.id/1068/.

Maslakhah S. 2019. Penerapan metode. *Jurnal Ilmiah Bahasa, Sastra, dan Pengajarannya* 27(2): 159–167. Tersedia pada: https://journal.uny.ac.id/index.php/diksi/article/view/23098.

Nuraini, Firmansyah B. 2020. Implementasi Algoritme Knuth Morris Prath Untuk Kamus Terjemahan Digital Aceh – Bahasa Indonesia Berbasis Web. *Jurnal Nasional Informatika*. 1(1): 66–75. Tersedia pada: https://journal.uny.ac.id/index.php/diksi/article/view/23098.

Priyanto A, Ulinnuha F. 2017. Perancangan Aplikasi Penerjemah Bahasa Indonesia Ke Bahasa Jawa Untuk Media Bantu Belajar Siswa SMK Salafiyah Berbasis Android. *Indonesian Journal of Network & Security* 6(4): 39–46. Tersedia pada: http://ijns.org/journal/index.php/ijns/article/view/1473.

Przybocki M, Sanders G, Le A. 2006. Edit distance: A metric for machine translation evaluation. *Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC 2006*. hlm 2038–2043.

Wint ZZ, Ducros T, Aritsugi M. 2017. Spell corrector to social media datasets in message filtering systems. *2017 12th International Conference on Digital Information Management, ICDIM 2017*, 2018-Janua(Icdim). hlm 209–215. doi: 10.1109/ICDIM.2017.8244677.

Yulianto MM, Arifudin R, Alamsyah A. 2018. Autocomplete and Spell Checking Levenshtein Distance Algorithm To Getting Text Suggest Error Data Searching In Library. *Scientific Journal of Informatics* 5(1): 75. doi: 10.15294/sji.v5i1.14148.