MODEL HYBRID BERBASIS EDIT DISTANCE, PHONETIC, DAN FUZZY MATCHING DENGAN OPTIMASI BOBOT UNTUK VERIFIKASI NAMA INVOICE

Angga Rakhmansyah, Joko Riyanto
Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang,
Tangerang Selatan, Banten, Indonesia
anggafsn@gmail.com

ABSTRAK

Proses verifikasi kesesuaian nama antara faktur dan basis data induk di perusahaan jasa tenaga kerja asing menghadapi tantangan serius akibat keberagaman konvensi penamaan multi-etnis. Ketidakkonsistenan penulisan mulai dari variasi ejaan, perbedaan Romanisasi nama Asia Timur, penggunaan patronimik Melayu, hingga penyisipan informasi tambahan dalam kolom nama menyebabkan proses pencocokan manual menjadi rentan terhadap kesalahan dan tidak efisien secara operasional. Penelitian ini mengembangkan arsitektur Algoritma Pencocokan Nama Multi-Layer untuk mengotomasi proses verifikasi nama pada faktur di PT Fortuna Sada Nioga, sebuah perusahaan jasa legalitas tenaga kerja asing (TKA). Sistem ini memadukan delapan lapisan algoritma pencocokan string untuk menangani variasi nama multi-etnis dari konvensi penamaan Indonesia, Korea, Jepang, dan Tiongkok. Melalui eksperimen grid search terhadap 1,5 juta konfigurasi bobot pada 17.908 pasangan nama, konfigurasi optimal mencapai akurasi, presisi, dan recall 100%. Lapisan Token-based dan Advanced Hybrid masing-masing memperoleh bobot tertinggi (30%), sementara Fuzzy Logic mendapat bobot 0% yang mengindikasikan redundansinya dalam arsitektur ensemble. Sistem Keputusan Tiga Zona (*Auto-Pass, Confirmation, Auto-Reject*) diterapkan untuk keperluan operasional.

Kata kunci : *verifikasi nama, string matching, edit distance, fonetik, ensemble, grid search*

1. PENDAHULUAN

Transformasi digital dalam pengelolaan administrasi keuangan perusahaan telah mendorong adopsi sistem informasi akuntansi yang menurut Anriva [1] mampu meningkatkan efisiensi operasional, akurasi data, dan transparansi. Namun pada operasional tertentu, seperti proses verifikasi kesesuaian nama antara dokumen invoice dengan basis data induk, otomasi penuh masih sulit dicapai. Permasalahan ini termasuk dalam domain string matching yang memerlukan algoritma mampu mengukur kemiripan teks secara akurat [2].

PT Fortuna Sada Nioga bergerak di bidang jasa legalitas dan penyediaan tenaga kerja asing (TKA). Staf administrasi perusahaan masih mencocokkan nama karyawan dan anggota keluarga di draf invoice dengan basis data induk secara manual. Keragaman latar belakang linguistik—Indonesia, Mandarin, Jepang, Korea, dan Barat—menciptakan variasi penulisan yang kompleks, mulai dari perbedaan ejaan, ketidakkonsistenan Romanisasi, hingga penyisipan metadata administratif dalam nama.

Penelitian terdahulu menunjukkan bahwa setiap algoritma pencocokan string memiliki keterbatasan tersendiri dalam menghadapi variasi nama yang heterogen. Algoritma Edit Distance seperti Levenshtein efektif mendeteksi kesalahan ketik pada level karakter, namun menghasilkan jarak yang besar ketika urutan kata dalam nama tertukar meski secara substansi identik [3]. Algoritma fonetik seperti Soundex unggul menangani variasi bunyi, namun tidak sensitif terhadap variasi struktur token [4]. Pendekatan berbasis token seperti Jaccard Similarity efektif mendeteksi urutan kata yang tertukar, namun tidak peka terhadap kesalahan ketik pada level karakter [2]. Dengan demikian, tidak ada satu algoritma tunggal yang dapat menangani seluruh jenis variasi nama multi-etnis secara andal. Pendekatan ensemble multi-layer—yang mengintegrasikan kelebihan masing-masing algoritma—menjadi pilihan yang tepat untuk mengatasi kompleksitas tersebut.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan mengembangkan sistem NameMatcher yang mengintegrasikan delapan lapisan algoritma pencocokan string dalam satu arsitektur multi-layer. Kontribusi utama meliputi:
(1) analisis empiris variasi nama multi-etnis dari data operasional nyata;
(2) arsitektur 8-layer dengan optimasi bobot melalui grid search 1,5 juta iterasi; dan
(3) Sistem Keputusan Tiga Zona berbasis distribusi statistik skor untuk penerapan operasional.

2. TINJAUAN PUSTAKA

Bagian ini menguraikan landasan teori dan penelitian terdahulu yang menjadi dasar pengembangan arsitektur NameMatcher. Kajian difokuskan pada berbagai pendekatan pencocokan string, mulai dari metode pengukuran jarak pengeditan (Edit Distance), identifikasi kesamaan bunyi (Algoritma Fonetik), hingga pencocokan berbasis token dan N-gram. Selain itu, bagian ini juga meninjau penerapan ensemble learning dan optimasi parameter melalui grid search yang digunakan untuk menggabungkan performa dari berbagai algoritma tunggal tersebut.

2.1. Algoritma Edit Distance

Edit Distance mengukur jumlah minimum operasi penyuntingan yang diperlukan untuk mengubah satu string menjadi string lain. Darmanto [3] membuktikan bahwa algoritma Levenshtein Distance unggul dalam mendeteksi kesalahan penulisan pada teks berbahasa Indonesia, khususnya untuk karya ilmiah mahasiswa. Firmansyah & Deswana [5] menerapkan Levenshtein pada sistem knowledge management berbasis web dan memperoleh skor kemiripan rata-rata 88,2% dalam menangani ketidakcocokan string. Santoso & Yan [6]mengembangkan pendekatan hybrid yang mengombinasikan Levenshtein dengan metode empiris untuk koreksi typo dokumen berbahasa Indonesia, mencapai presisi 92% dengan F1-score 90,5%.

2.2. Algoritma Fonetik

Algoritma fonetik mengidentifikasi kesamaan bunyi antar string meskipun ejaannya berbeda. Soundex memetakan konsonan ke kode angka sehingga variasi ejaan seperti "Achmad" dan "Ahmad" menghasilkan kode yang sama. Metaphone dan NYSIIS memberikan akurasi lebih tinggi pada nama-nama dengan pola vokal tidak lazim. Suzanti dkk. [4] menerapkan Levenshtein Distance untuk koreksi ejaan artikel pariwisata berbahasa Indonesia dan memperoleh presisi sempurna (1,0) dengan recall 0,89.

2.3. Token-based dan N-gram Matching

Jaccard Similarity mengukur rasio irisan terhadap gabungan dua himpunan token, efektif untuk mendeteksi nama tertukar urutannya [2]. Koefisien Sørensen–Dice menghitung kemiripan berbasis N-gram (bigram/trigram) yang mampu menangkap kesamaan substruktur tanpa bergantung pada batas kata. Bintang dkk. [7] menerapkan String Matching dan Regular Expression pada aplikasi KBBI untuk pencarian kata meskipun terdapat variasi morfologis. Kokong dkk. [8] mengombinasikan Damerau-Levenshtein dan N-gram untuk koreksi ejaan bahasa Indonesia dengan tingkat deteksi 80–100% dan akurasi koreksi 96%.

2.4. Ensemble Learning dan Grid Search

Ensemble learning menggabungkan beberapa model untuk menghasilkan prediksi yang lebih baik dari model individual. Desiani dkk. [9] membuktikan bahwa weighted voting ensemble CNN menghasilkan akurasi 93,3% yang lebih tinggi dari model tunggal manapun. Rianti & Supono [2] secara komprehensif membandingkan Edit Distance, Levenshtein, Hamming, dan Jaccard Similarity dalam mendeteksi string matching, namun belum mengkaji kombinasi multi-algoritma secara simultan. Nugraha & Sasongko [10] membuktikan bahwa metode Grid Search dengan Cross Validation menghasilkan konfigurasi parameter yang lebih optimal dibandingkan pendekatan manual.

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif eksperimental. Tahapan meliputi pengumpulan data, analisis variasi nama, perancangan arsitektur algoritma, optimasi bobot melalui grid search, dan evaluasi kinerja.

3.1. Pengumpulan Data

Data primer diperoleh dari arsip operasional PT Fortuna Sada Nioga, terdiri dari 7.857 entri nama karyawan (worker) dan 995 entri nama keluarga (family). Data dikumpulkan dari arsip operasional periode 2011 hingga 2025, mencakup seluruh data nama karyawan TKA aktif beserta anggota keluarga yang tercatat dalam sistem invoice perusahaan. Setelah deduplikasi diperoleh 7.185 nama karyawan unik dan 873 nama keluarga unik. Spesifikasi dataset disajikan pada Tabel 1.

Tabel 1. Spesifikasi Dataset Penelitian

Parameter	Keterangan
Data Mentah Karyawan	7.857 entri (7.185 nama unik)
Data Mentah Keluarga	995 entri (873 nama unik)
Total Nama Unik	8.012 nama
Asal Etnis	Indonesia, Korea, Jepang, Tiongkok, Barat
Nama Di-sample untuk Uji	1.000 nama representatif
Variasi per Nama	15 jenis (typo, case, spasi, urutan, dll.)
Pasangan Positif (SAMA)	12.908 pasangan
Pasangan Negatif (BEDA)	5.000 pasangan (acak)
Total Pasangan Uji	17.908 pasangan

Pengambilan sampel sebanyak 1.000 nama dari total 8.012 nama unik didasarkan pada dua pertimbangan utama. Pertama, ukuran sampel ini dipastikan mencakup representasi proporsional dari seluruh kelompok etnis yang ada dalam dataset (Indonesia, Korea, Jepang, Tiongkok, dan Barat), sehingga evaluasi tidak bias pada satu kelompok. Kedua, dengan setiap nama menghasilkan hingga 15 variasi, total 17.908 pasangan uji yang dihasilkan memberikan ruang evaluasi yang cukup luas untuk grid search 1,5 juta iterasi, sekaligus masih dapat diproses dalam kapasitas komputasi yang tersedia. Dari 1.000 nama sampel, pasangan positif (SAMA) dibangun dengan membuat 15 jenis variasi penulisan secara sistematis. Setiap variasi merepresentasikan skenario kesalahan atau perbedaan yang lazim dijumpai dalam data operasional. Sebagai ilustrasi, dari nama asli "Son Heung Min" dihasilkan variasi-variasi berikut:

Tabel 2. Komposisi Dataset Pasangan Positif (SAMA)

Jenis Variasi	Contoh Hasil	Pasangan
Original	Son Heung Min	1.000
Lowercase	son heung min	1.000
Missing char	Son Heng Min	1.000
Extra char	Son Heunng Min	1.000
No space	SonHeungMin	1.000
Extra space	Son Heung Min	1.000
Reversed	Min Heung Son	1.000
Mixed case	sON HeUnG MIN	999
Typo char 2	Son Heung Nin	888
Typo char	Son Jeung Min	876
Doubled char	Son Heungg Min	874
Transposed	Son Heugn Min	707
Swapped words	Heung Son Min	534
Abbreviated	Son H Min	531
Uppercase	SON HEUNG MIN	499

Jumlah pasangan yang dihasilkan bervariasi antar jenis variasi karena adanya prasyarat tertentu. Variasi yang tidak memerlukan syarat khusus seperti lowercase, missing char, extra char, no space, extra space, dan reversed menghasilkan masing-masing 1.000 pasangan dari seluruh nama sampel. Sebaliknya, variasi uppercase hanya menghasilkan 499 pasangan karena nama yang sudah ditulis dalam huruf kapital seluruhnya akan menghasilkan string identik dengan aslinya sehingga di-skip oleh sistem.

Variasi swapped words (534) dan abbreviated (531) memerlukan minimal tiga kata dalam nama, sementara banyak nama dalam dataset hanya terdiri dari dua kata. Variasi transposed (707) memerlukan dua huruf alfabet bersebelahan yang bisa ditukar, dan variasi typo (876, 888) serta doubled char (874) bergantung pada posisi karakter yang valid untuk dimodifikasi. Total seluruh pasangan positif yang dihasilkan adalah 12.908, ditambah 5.000 pasangan negatif (nama berbeda orang yang dipilih secara acak), sehingga dataset uji keseluruhan berjumlah 17.908 pasangan.

Data mentah mengandung berbagai inkonsistensi, mulai dari perbedaan kapitalisasi (*All Caps vs Title Case*), entri terduplikasi, hingga penyisipan metadata ke dalam kolom nama. Oleh karena itu, sebelum masuk ke tahap pencocokan, setiap string nama melewati tahap pra-pemrosesan (*preprocessing*) yang meliputi langkah-langkah berikut:

a. Normalisasi Kapitalisasi
   seluruh karakter dikonversi ke huruf kecil (lowercase) agar perbandingan bersifat case-insensitive.
b. Pembersihan Metadata
   informasi dalam tanda kurung seperti "(Anak Mr. Kim)" dan "(*Wife Of* Kim)" dihapus secara otomatis menggunakan ekspresi reguler.
c. Normalisasi Spasi dan Karakter Khusus
   spasi ganda, tanda hubung, dan karakter non-alfabet dihilangkan atau distandarkan menjadi satu spasi tunggal.
d. Penghapusan Stopword Relasional
   token-token penanda hubungan keluarga seperti "bin", dan "binti", dihapus dari string sebelum pencocokan dilakukan.

Tabel 3. Contoh Hasil Preprocessing Nama

Kondisi	Nama Sebelum Preprocessing	Nama Sesudah Preprocessing	Langkah
Metadata dalam kurung	Chen Qihang (Anak Chen Zaixin)	chen qihang	b, a
Patronimik Melayu	FARIZ AMIN BIN FATHUR RAHMAN	fariz amin	a, d
Spasi tidak konsisten	Son Heung Min	son heung min	c, a
Huruf campuran	KiM JuNsEo	kim junseo	a
Tanda hubung	Al-Rashid Mohammed	al rashid mohammed	c, a

3.2. Analisis Variasi Nama Multi-Etnis

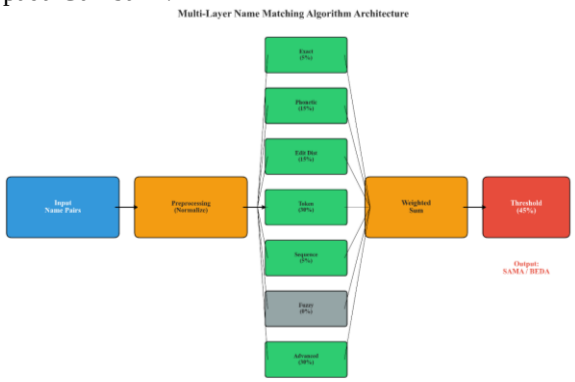
Dari hasil analisis terhadap dataset, ditemukan sejumlah pola variasi yang menjadi tantangan utama bagi sistem pencocokan. Variasi-variasi ini jauh lebih kompleks dibandingkan kesalahan ejaan konvensional yang dikaji oleh Darmanto [3]. dalam konteks karya ilmiah:

a. Injeksi Metadata
   Nama kerap disisipi informasi tambahan dalam tanda kurung, misalnya "Steven (Anak Michael)". Token-token tambahan ini menyebabkan jarak edit membengkak meskipun secara substansi nama yang dimaksud identik.
b. Variasi Struktur Nama Asia Timur
   "Son Heung Min" dan "Son Heungmin" secara fonetik identik namun secara ortografis berbeda. Demikian pula "Liang Shu" (marga di depan) versus "Shu Liang" (marga di belakang).
c. Patronimik Melayu
   Perbedaan dalam penggunaan "bin" atau "binti", misalnya "Faez Alhaq" dan "Faez Alhaq bin Fazlur Azmi", di mana nama pertama merupakan subset sempurna dari nama kedua.

3.3. Arsitektur Algoritma 8-Layer

Guna menangani keragaman variasi tersebut, sistem dirancang dengan delapan lapisan pencocokan. Setiap lapisan menghasilkan skor kesamaan dalam rentang 0 sampai 1, yang kemudian diagregasi di lapisan terakhir melalui penjumlahan terbobot. Pendekatan agregasi multi-komponen ini sejalan dengan konsep weighted voting yang diterapkan oleh Desiani dkk. [9] dalam kerangka ensemble learning, di mana kombinasi beberapa model melalui pembobotan terbukti menghasilkan akurasi 93,3% yang lebih tinggi

dari model individual. Arsitektur sistem ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur multi-layer name matching algorithm

Seperti diilustrasikan pada Gambar 1, sistem memproses pasangan nama mentah melalui tahap pra-pemrosesan kemudian menghitung skor dari tujuh lapisan pencocokan. Skor-skor tersebut diagregasi pada lapisan kedelapan menjadi skor akhir tunggal melalui penjumlahan terbobot. Rincian fungsi dan landasan matematis tiap lapisan diuraikan sebagai berikut.

Layer 1 (Exact Match) menggunakan Jaccard Index $J(A,B) = |A \cap B|/|A \cup B|$ dan Overlap Coefficient untuk mendeteksi kecocokan himpunan token.

Layer 2 (Phonetic) menggabungkan Soundex (bobot 0,4), Metaphone (0,4), dan NYSIIS (0,2).

Layer 3 (Edit Distance) menghitung skor Levenshtein ternormalisasi $S = 1 - lev(a,b)/\max(|a|,|b|)$ dan Jaro-Winkler.

Layer 4 (Token N-gram) menggunakan $Dice(A,B) = 2|A \cap B|/(|A|+|B|)$ pada himpunan bigram dan trigram.

Layer 5 (Sequence) menghitung LCS ternormalisasi $S = |LCS(a,b)|/\max(|a|,|b|)$.

Layer 6 (Fuzzy) menerapkan logika fuzzy generik.

Layer 7 (Advanced Hybrid) adalah lapisan heuristik untuk pola transliterasi kompleks.

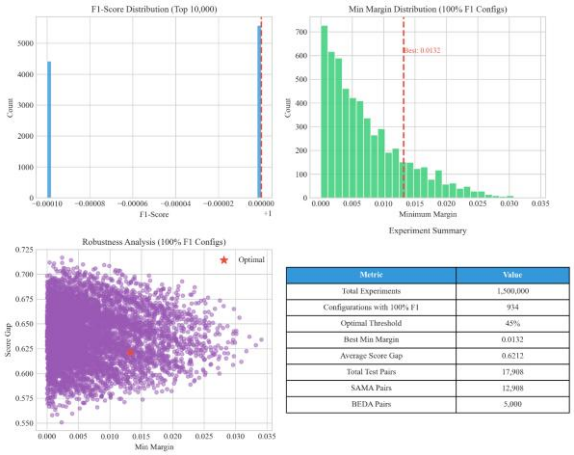
Layer 8 (Ensemble) mengagregasi $S_{final} = \Sigma(w_i \times S_i)$, $i=1..7$.

3.4. Desain Eksperimen Grid Search

Untuk menentukan bobot optimal tiap lapisan, dilakukan grid search terhadap 1.500.000 kombinasi bobot. Setiap bobot merupakan kelipatan 5% dengan total 100%, menghasilkan ruang pencarian yang besar namun terbatas. Metrik evaluasi yang digunakan meliputi Akurasi, Presisi, Recall, dan F1-Score. Seperti dibuktikan oleh Nugraha & Sasongko [10], pendekatan grid search menghasilkan konfigurasi lebih optimal dibanding penentuan manual.

4. HASIL DAN PEMBAHASAN

Sistem telah menguji 1.500.000 kombinasi bobot terhadap 17.908 pasangan nama (12.908 pasangan SAMA dan 5.000 pasangan BEDA). Dari seluruh ruang pencarian, ditemukan 934 konfigurasi yang berhasil memisahkan kedua kelas secara sempurna dengan F1-Score 100%. Visualisasi ringkasan eksperimen disajikan pada Gambar 2.



Gambar 2. Visualisasi distribusi performa grid search

Pada gambar tersebut mengilustrasikan distribusi performa dari 1,5 juta konfigurasi bobot yang diuji. Grafik tersebut menunjukkan bagaimana sebaran nilai *F1-Score* memusat, di mana hanya sebagian kecil kompisisi bobot (934 konfigurasi) yang berhasil mencapai tingkat akurasi sempurna (100%). Visualisasi ini juga memperlihatkan rentang margin pemisahan antar kelas, yang mengonfirmasi bahwa pendekatan *grid search* tidak hanya menemukan bobot yang akurat, tetapi juga mencari kombinasi yang memberikan batas keputusan (*decision boundary*) paling tegas antara pasangan nama yang SAMA dan BEDA

4.1 Konfigurasi Bobot Optimal

Dari 934 konfigurasi yang mencapai akurasi sempurna, dipilih satu konfigurasi unggulan ("Rank 1") berdasarkan margin pemisahan terbesar. Distribusi bobot optimal disajikan dalam Tabel 4.

Tabel 4. Distribusi Bobot Konfigurasi Optimal

Komponen Layer	Bobot (%)	Analisis Peran
Token-based	30%	Menangani pengurutan ulang nama dan variasi struktur
Advanced Hybrid	30%	Menangkap pola semantik dan heuristik kompleks
Phonetic	15%	Menangani variasi ejaan Romanisasi nama Asia Timur
Edit Distance	15%	Menangani kesalahan ketik karakter minor
Exact Match	5%	Dorongan skor untuk kecocokan sempurna
Sequence	5%	Pelengkap alinyemen karakter yang terputus
Fuzzy	0%	Redundan; fungsinya tercakup oleh Token dan Advanced

Dominasi Token-based (30%) mengonfirmasi bahwa variasi struktural—bukan sekadar typo karakter—adalah permasalahan dominan dataset ini. Temuan kontraintuitif berupa bobot 0% pada Fuzzy Logic menunjukkan bahwa dalam arsitektur ensemble berbasis weighted voting [9], algoritma fuzzy generik justru redundan ketika dikombinasikan dengan metrik lain yang sudah terkalibrasi.

Selain konfigurasi peringkat pertama, akurasi 100% juga dicapai oleh sejumlah variasi bobot lain. Tabel 5-7 menyajikan konfigurasi teratas yang menunjukkan stabilitas arsitektur yang dibangun.

Tabel 5. Konfigurasi Bobot Teratas 2-4

Rank	2	3	4
Exact	0%	5%	5%
Phonetic	10%	15%	15%
Edit Dist.	25%	5%	10%
Token	20%	25%	30%
Sequence	10%	25%	15%
Fuzzy	0%	0%	0%
Advanced	35%	25%	25%
Threshold	50%	45%	45%
Margin	0,012	0,012	0,012

Tabel 6. Konfigurasi Bobot Teratas 5-7

Rank	5	6	7
Exact	5%	0%	5%
Phonetic	15%	10%	20%
Edit Dist.	10%	30%	5%
Token	25%	25%	25%
Sequence	10%	0%	0%
Fuzzy	5%	0%	25%
Advanced	30%	35%	20%
Threshold	45%	50%	45%
Margin	0,012	0,011	0,011

Tabel 7. Konfigurasi Bobot Teratas 8-10

Rank	8	9	10
Exact	5%	5%	5%
Phonetic	10%	15%	15%
Edit Dist.	20%	15%	15%
Token	10%	35%	30%
Sequence	25%	5%	0%
Fuzzy	0%	0%	5%
Advanced	30%	25%	30%
Threshold	50%	45%	45%
Margin	0,011	0,011	0,011

Keberadaan variasi pada Tabel 4 membuktikan stabilitas arsitektur yang diusulkan. Meskipun ada pergeseran bobot antar konfigurasi layer Phonetic berkisar 10–20% dan Edit Distance 5–30% sistem tetap mempertahankan akurasi sempurna selama lapisan Token dan Advanced mendapat porsi yang cukup besar. Hal ini menandakan bahwa arsitektur tidak mengalami overfitting pada satu set parameter tunggal.

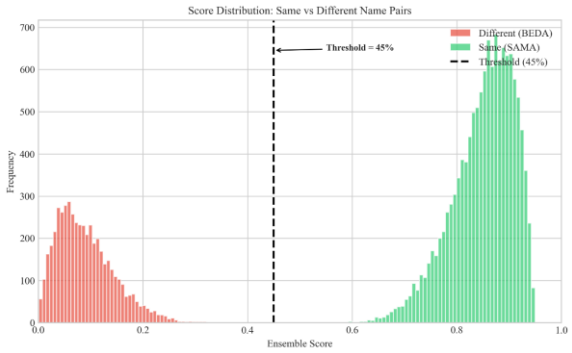
4.2 Analisis Statistik Distribusi Skor

Kinerja pemisahan kelas oleh sistem NameMatcher dirangkum dalam Tabel 8.

Tabel 8. Statistik Distribusi Skor Ensemble

Metrik Statistik	Kelas SAMA	Kelas BEDA
Jumlah Sampel	12.908	5.000
Rata-rata Skor	0,796	0,1244
Standar Deviasi	0,1422	0,0398
Skor Minimum	0,4625	0,0142
Skor Maksimum	0,9667	0,4388
Ambang Batas Kritis	P5: 0,5598	P95: 0,1896

Selisih rata-rata skor kedua kelas mencapai 0,6716. Yang lebih penting, skor tertinggi pada kelas BEDA (0,4388) masih berada di bawah skor terendah kelas SAMA (0,4625). Meskipun jaraknya relatif tipis (~0,02), penggunaan ambang batas berbasis persentil memberikan buffer keamanan yang lebih longgar untuk keperluan operasional. Distribusi skor divisualisasikan pada Gambar 3.



Gambar 3. Distribusi skor pasangan sama vs beda

Pada Gambar 3 tampak celah yang signifikan antara ekor distribusi kelas BEDA (merah) dan kepala distribusi kelas SAMA (hijau). Garis putus-putus menandai ambang batas 45% yang memisahkan kedua populasi tanpa irisan.

4.3 Matriks Konfusi dan Akurasi

Evaluasi kinerja menggunakan matriks konfusi menunjukkan hasil sempurna pada dataset uji, sebagaimana disajikan pada Tabel 9.

Tabel 9. Matriks Konfusi Hasil Pengujian

	Prediksi SAMA	Prediksi BEDA
Aktual SAMA	12.908 (TP)	0 (FN)
Aktual BEDA	0 (FP)	5.000 (TN)

Hasil pada Tabel 9 menunjukkan kinerja klasifikasi yang sempurna. Nilai True Positive (TP) sebesar 12.908 berarti seluruh pasangan nama yang sesungguhnya SAMA berhasil diidentifikasi dengan benar oleh sistem. Nilai True Negative (TN) sebesar 5.000 menunjukkan seluruh pasangan nama BEDA ditolak dengan tepat. Yang paling krusial dari perspektif operasional adalah nilai False Positive (FP) = 0 dan False Negative (FN) = 0.

Tidak adanya FP berarti sistem tidak pernah menerima pasangan nama yang sebenarnya berbeda sebagai cocok—dalam konteks bisnis, ini mencegah risiko pembayaran atau persetujuan faktur kepada

entitas yang salah. Tidak adanya FN berarti sistem tidak pernah menolak pasangan yang sebenarnya sama, sehingga tidak ada transaksi yang tertahan secara tidak perlu.

Analisis per jenis variasi menunjukkan akurasi 100% pada seluruh 15 jenis variasi yang diuji, termasuk variasi paling menantang seperti swapped_words (534 pasangan) dan abbreviated (531 pasangan). Hal ini mengkonfirmasi bahwa kinerja sempurna bukan hanya dihasilkan oleh variasi mudah seperti extra_space atau lowercase, melainkan merata di seluruh jenis variasi.

Perlu dicatat bahwa akurasi 100% ini dicapai pada dataset yang dibangun secara sistematis dari 1.000 nama sampel. Celah antara skor terendah kelas SAMA (0,4625) dan skor tertinggi kelas BEDA (0,4388) sebesar 0,0237 menunjukkan bahwa pemisahan kelas, meskipun sempurna, memiliki margin yang relatif tipis. Oleh karena itu, penerapan Sistem Keputusan Tiga Zona menjadi penting sebagai lapisan keamanan tambahan dalam kondisi operasional nyata yang mungkin mengandung variasi di luar cakupan dataset uji.

4.4 Sistem Keputusan Tiga Zona

Agar hasil eksperimen dapat diterapkan dalam operasional sehari-hari, dikembangkan mekanisme keputusan tiga zona yang mengutamakan keamanan finansial:

a. Zona Hijau (Auto-Pass), Skor ≥ 0,5598 Sistem secara otomatis menyetujui kesesuaian nama. Batas bawah mengacu pada persentil ke-5 distribusi skor kelas SAMA.
b. Zona Abu-abu (Confirmation), 0,1896 ≤ Skor < 0,5598 Sistem menahan proses dan meminta konfirmasi manual. Zona ini menjadi penyangga antara area yang jelas cocok dan jelas tidak cocok.
c. Zona Merah (Auto-Reject), Skor < 0,1896 Sistem langsung menolak entri. Batas atas mengacu pada persentil ke-95 distribusi skor kelas BEDA.

4.5 Analisis Kasus Spesifik Etnis

Keberhasilan sistem ini ditopang oleh kemampuan lapisan-lapisan tertentu dalam menangani variasi etnis spesifik:

a. Nama Korea (variasi spasi)
   Pasangan "Son Heung Min" dan "Son Heungmin" gagal pada Exact Match, namun layer Phonetic (15%) memberikan skor tinggi karena keduanya identik secara bunyi. Penalti kecil pada layer Token (3 token vs 2 token) dikompensasi oleh bobot tinggi pada layer Advanced.
b. Nama Tionghoa (metadata)
   Pada entri "Li Yi Feng (Anak Mr. Li)", layer Advanced Hybrid (30%) memainkan peran utama dengan mendeteksi dan menurunkan bobot token metadata seperti "Anak", sehingga pencocokan tetap berfokus pada token inti "Li Yi Feng".
c. Nama Melayu (patronimik)
   Untuk pasangan "Fariz Amin" dan "Fariz Amin bin Fathur Rahman", metrik Overlap Coefficient pada Layer 1 dan analisis Token pada Layer 4 mendeteksi bahwa nama pendek merupakan subset dari nama panjang.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil membuktikan bahwa arsitektur Multi-Layer Name Matching 8-layer mampu menangani variasi nama multi-etnis secara andal. Dari 1,5 juta kombinasi bobot yang diuji melalui grid search, ditemukan 934 konfigurasi yang mencapai akurasi, presisi, dan recall 100% pada 17.908 pasangan nama uji. Konfigurasi optimal (Token 30%, Advanced Hybrid 30%, Phonetic 15%, Edit Distance 15%) berhasil memisahkan kelas SAMA dan BEDA dengan margin 0,0132—cukup untuk menjamin threshold keputusan 45% bekerja secara andal. Temuan bahwa lapisan Fuzzy Logic mendapat bobot 0% dalam konfigurasi optimal mengkonfirmasi bahwa optimasi bobot otomatis dapat mengeliminasi redundansi algoritma yang tidak selalu terdeteksi secara intuitif. Sistem Keputusan Tiga Zona yang dikembangkan memberikan kerangka operasional yang aman: pasangan dengan skor ≥ 0,5598 disetujui otomatis, kasus ambigu (0,1896–0,5598) diteruskan ke verifikasi manual, dan pasangan dengan skor < 0,1896 ditolak otomatis. Untuk pengembangan selanjutnya, tiga prioritas disarankan berdasarkan temuan penelitian ini. Pertama, memperkuat Layer 7 (Advanced Hybrid) yang mendapat bobot tertinggi namun masih berbasis aturan heuristik manual—integrasi model bahasa ringan atau embedding semantik dapat meningkatkan kemampuannya menangani transliterasi nama baru. Kedua, membangun mekanisme feedback loop dari zona konfirmasi manual, di mana setiap keputusan operator dimanfaatkan untuk memperluas dataset dan mengkalibrasi ulang bobot secara berkala. Ketiga, mengevaluasi ketahanan sistem terhadap nama dari etnis yang belum tercakup dalam dataset, seperti nama dari kawasan Asia Selatan atau Timur Tengah.

.

DAFTAR PUSTAKA

[1] D. H. Anriva, "Tantangan dan Solusi Penerapan Sistem Informasi Akuntansi di Indonesia: Sebuah Analisis Tematik," *Jurnal Akuntansi*, vol. 13, no. 2, pp. 97–109, 2024, doi: 10.46806/ja.v13i2.1182.
[2] S. Rianti and R. A. Supono, "Perbandingan Algoritma Edit Distance, Levenshtein Distance, Hamming Distance, Jaccard Similarity Dalam Mendeteksi String Matching," *JSI Universitas Suryadarma*, vol. 10, no. 1, pp. 305–314, 2023.
[3] Darmanto, "Perbandingan Kinerja Algoritma Jaro-Winkler dan Levenshtein Distance untuk Deteksi Kesalahan Penulisan Bahasa Indonesia

pada Karya Ilmiah Mahasiswa," *JURTI*, vol. 9, no. 2, 2024, doi: 10.30872/jurti.v9i2.19134.

[4] I. O. Suzanti, Y. D. P. Negara, A. Radiah, A. D. Cahyani, and H. Husni, "Koreksi Kesalahan Ejaan pada Pencarian Artikel Pariwisata Berbahasa Indonesia Menggunakan Levenshtein Distance," *Jurnal Simantec*, vol. 13, no. 1, pp. 81–90, 2024, doi: 10.21107/simantec.v13i1.28186.

[5] M. Firmansyah and S. Deswana, "Application of the Levenshtein Algorithm for Optimizing Search Accuracy in a Web-Based Knowledge Management System," *JUSIFO*, vol. 10, no. 2, pp. 99–106, 2024, doi: 10.19109/jusifo.v10i2.21951.

[6] J. T. Santoso and S. Yan, "A Hybrid Approach to Typo Correction in Indonesian Documents Using Levenshtein Distance," *JTIE*, vol. 3, no. 2, pp. 151–168, 2024, doi: 10.51903/jtie.v3i2.184.

[7] J. M. Bintang, M. F. Ashshidiq, and H. F. Dzakwan, "Penerapan Algoritma String Matching dan Regular Expression pada Aplikasi Kamus Besar Bahasa Indonesia (KBBI)," *Bios*, vol. 4, no. 1, pp. 34–41, 2023, doi: 10.37148/bios.v4i1.57.

[8] D. A. R. S. Kokong, B. Irmawati, and R. Dwiyansaputra, "Spelling Error Correction in Indonesian Using Damerau-Levenshtein Distance dan N-Gram," *JTIKA*, vol. 6, no. 1, pp. 257–263, 2024, doi: 10.29303/jtika.v6i1.169.

[9] A. Desiani, "Weighted Voting Ensemble Learning of CNN Architectures for Diabetic Retinopathy Classification," *Jurnal INFOTEL*, vol. 16, no. 1, pp. 136–155, 2024, doi: 10.20895/infotel.v16i1.999.

[10] W. Nugraha and A. Sasongko, "Hyperparameter Tuning on Classification Algorithm with Grid Search," *Sistemasi*, vol. 11, no. 2, pp. 391–401, 2022, doi: 10.32520/stmsi.v11i2.1750.