

Identifikasi Kuantitas Berbasis *Rule* Pada Masalah *Text-to-SQL*

Bramasta Albatio Haryono¹, Agung Prasetya²

^{1,2} Universitas Bhinneka PGRI Tulungagung, Tulungagung, Jawa Timur, 66221, Indonesia

e-mail: ¹bramastaaalbatioharyono@gmail.com, ²agung@ubhi.co.id

Diterima
15-07-2025

Direvisi
13-08-2025

Disetujui
10-09-2025

Abstract: This study focuses on developing a rule-based system designed to detect and extract quantity information from Indonesian sentences as part of the Text-to-SQL process. The proposed system consists of three main components: a user interface, a knowledge base containing IF-THEN rules, and a forward-chaining inference engine implemented using the Experta framework. The dataset comprises 1,000 sentences from various domains, including academic, inventory, accounting, and booking, with label distributions of 815 digit data, 164 word data, and 21 unknown data. Evaluation results demonstrate system performance with an Exact Match Ratio of 0.76 and a Jaccard Similarity of 0.75. For multilabel classification, the system achieves a Micro Precision of 0.98, Micro Recall of 0.66, and Micro F1-score of 0.79, while the Macro average reaches 0.52. With a Hamming Loss of 0.00, the system shows very low label errors. This research is expected to serve as a foundation for developing hybrid models combining rule-based methods and machine learning to improve accuracy, flexibility, and overall system performance in the future.

Keywords: Experta, Forward Chaining, Quantity Identification, Rule-based System, Text-to-SQL

Abstrak: Penelitian ini berfokus pada pengembangan sistem berbasis aturan (rule-based) untuk mendeteksi serta mengekstraksi informasi kuantitas dalam kalimat berbahasa Indonesia sebagai bagian dari proses Text-to-SQL. Sistem yang dirancang terdiri dari tiga komponen utama, yaitu antarmuka pengguna, basis pengetahuan berbentuk aturan IF-THEN, serta mesin inferensi berbasis forward chaining yang dibangun menggunakan kerangka kerja Experta. Dataset penelitian mencakup 1.000 kalimat dari berbagai domain, termasuk akademik, inventori, akuntansi, dan pemesanan, dengan distribusi label sebanyak 815 data digit, 164 data kata, dan 21 data unknown. Hasil evaluasi menunjukkan kinerja sistem dengan nilai Exact Match Ratio sebesar 0,76 dan Jaccard Similarity sebesar 0,75. Pada klasifikasi multilabel, diperoleh Micro Precision 0,98, Micro Recall 0,66, dan Micro F1-score 0,79, sedangkan rata-rata per label (Macro) sebesar 0,52. Dengan Hamming Loss 0,00, sistem menunjukkan kesalahan label yang sangat rendah. Penelitian ini diharapkan dapat menjadi dasar pengembangan model hybrid yang menggabungkan aturan dan pembelajaran mesin untuk meningkatkan akurasi, fleksibilitas, serta performa sistem di masa mendatang.

Kata kunci: Eksperta, Forward Chaining, Identifikasi Kuantitas, Rule-based System, Text-to-SQL

I. PENDAHULUAN

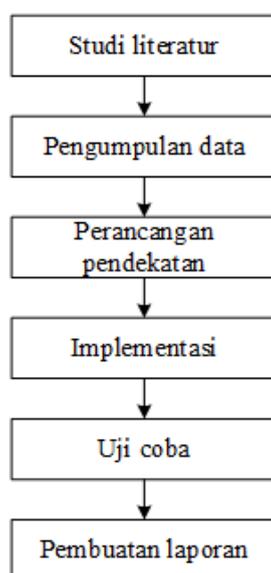
Text-to-SQL merupakan teknologi yang memungkinkan pengguna tanpa pengetahuan SQL untuk mengakses data melalui perintah dalam bahasa alami. Teknologi ini meningkatkan aksesibilitas terhadap database dan efisiensi dalam pengambilan informasi. Dalam bahasa Indonesia, penerapan Text-to-SQL masih menghadapi tantangan linguistik seperti struktur kalimat yang kompleks, variasi sinonim, dan perbedaan ekspresi kuantitas. Pendekatan berbasis aturan (*rule-based*) memungkinkan sistem untuk secara eksplisit mengenali pola-pola kuantitas seperti “berapa banyak” atau “jumlah”, dan mengonversinya menjadi fungsi SQL seperti *COUNT* dan *SUM*.

Beberapa studi sebelumnya telah mengembangkan model *Text-to-SQL* menggunakan pendekatan berbasis aturan maupun pendekatan pembelajaran mesin. Namun, studi-studi tersebut umumnya berfokus pada bahasa Inggris (Hu et al., 2020)(Yang et al., 2021), sehingga penerapannya pada bahasa Indonesia belum optimal. Pendekatan berbasis syntax parsing dan *sequence learning* pun belum mampu menangani seluruh variasi struktur kalimat secara akurat (Raifer et al., 2022). Selain itu, sistem *rule-based* bersifat eksplisit namun kaku, sehingga rentan gagal apabila pengguna menggunakan sinonim atau pola kalimat yang tidak dikenali (Qin et al., 2022). Kebaruan ilmiah dari penelitian ini terletak pada pengembangan sistem *Text-to-SQL* berbasis aturan yang dirancang khusus untuk menangani struktur kalimat bahasa Indonesia, serta mengidentifikasi kuantitas sebagai bagian dari komponen filter dalam *query SQL*. Penelitian ini juga menggabungkan *rule-based* dengan pendekatan pembelajaran mesin untuk memperluas fleksibilitas dalam memahami variasi bahasa. Berdasarkan hal tersebut, rumusan masalah yang diangkat dalam artikel ini adalah: bagaimana performa pendekatan berbasis *rule* dalam mengidentifikasi kuantitas pada permasalahan *Text-to-SQL* dalam bahasa Indonesia?

Tujuan dari kajian ini adalah mengevaluasi performa sistem berbasis aturan dalam mengenali kuantitas dari pertanyaan berbahasa Indonesia dan menerjemahkannya menjadi *query SQL* yang sesuai, serta mengusulkan perbaikan berbasis integrasi machine learning untuk meningkatkan fleksibilitas dan akurasi sistem.

II. METODE PERANCANGAN

Alur Metode perancangan digambarkan dalam alur kerja berikut.



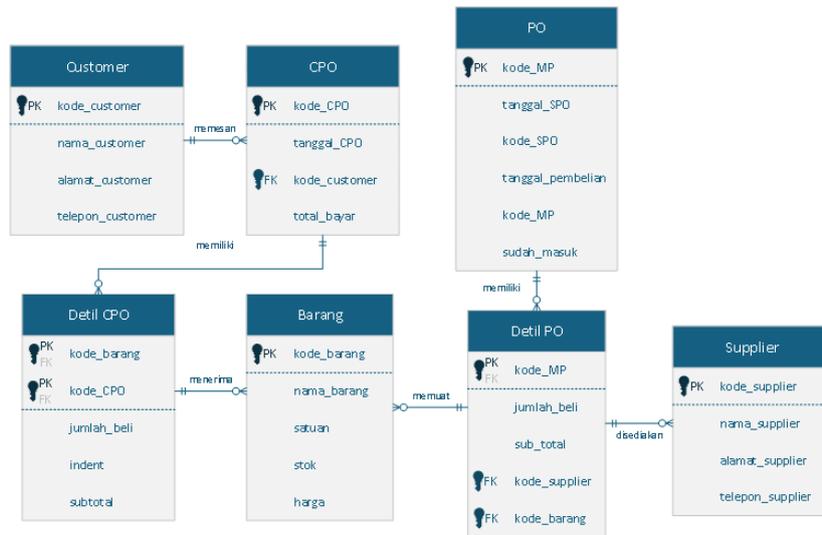
Gambar 1. Alur tahapan penelitian

1. Metode pengumpulan data

Tahap ini bertujuan untuk memperoleh data awal yang kemudian akan digunakan dalam proses akuisisi pengetahuan untuk membentuk sistem deteksi fungsi agregasi berbasis *rule-based*. Teknik yang digunakan dalam tahap ini adalah analisis dokumen, karena data berupa kalimat dan struktur basis data tidak selalu tersedia secara langsung melalui observasi, dan beberapa basis data tidak dapat dipublikasikan karena alasan privasi.

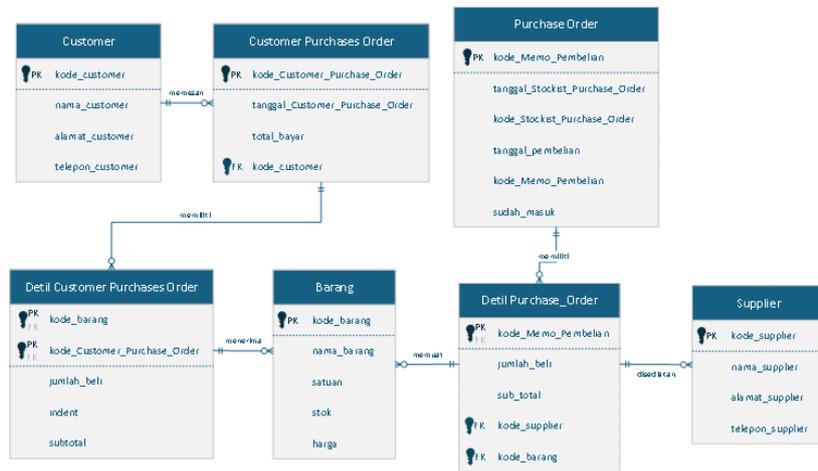
Jenis data pertama yang dikumpulkan dalam penelitian ini adalah *Entity Relationship Diagram* (ERD). ERD merupakan representasi visual dari struktur basis data yang terdiri atas entitas, atribut, dan relasi antar entitas (Pulungan et al., 2023). Dalam konteks penelitian ini, ERD digunakan sebagai acuan utama dalam membentuk kalimat-kalimat uji dan latih yang berkaitan dengan fungsi agregasi.

Setiap ERD yang ditemukan dalam dokumen sumber digambar ulang menggunakan notasi *Crow's Foot* untuk menyesuaikan dengan standar visualisasi yang umum digunakan.



Gambar 2. Tabel ERD sebelum perubahan

Proses penggambaran ulang melibatkan beberapa langkah penting, seperti menyempurnakan istilah pada nama tabel dan kolom. Singkatan yang tidak umum—seperti “PO”, “CPO”, atau “MP”—diubah menjadi bentuk lengkap berdasarkan konteks informasi dalam dokumen sumber, tanpa melibatkan pengetahuan eksternal.



Gambar 3. Tabel ERD Sesudah Perubahan

Selain itu, dilakukan penyederhanaan istilah dengan menghilangkan kata-kata yang tidak relevan seperti “tabel”, “tbl”, “kolom”, dan istilah teknis redundan lainnya. Nama tabel dan kolom juga diseragamkan menjadi bentuk tunggal untuk menghindari ambiguitas pada saat pemrosesan aturan. Dengan proses ini, ERD tidak hanya menjadi sumber struktur data, tetapi juga berfungsi sebagai dasar untuk menghubungkan kebutuhan sistem (fitur dan proses bisnis) (Eza, 2025) dengan konstruksi kalimat-kalimat kueri yang digunakan dalam eksperimen deteksi fungsi agregasi.

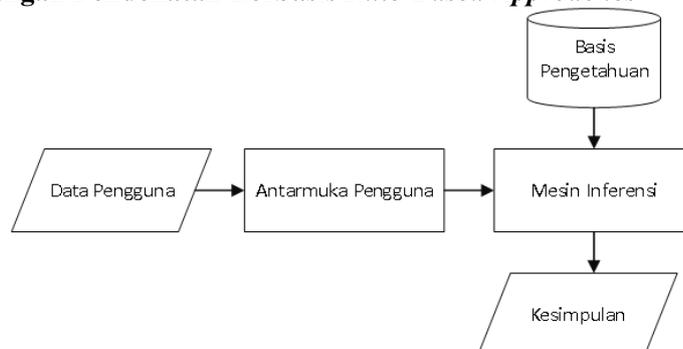
Jenis data kedua yang dikumpulkan adalah kalimat berbahasa alami yang mengandung potensi struktur kueri. Kalimat-kalimat ini diturunkan dari deskripsi fitur sistem, kebutuhan, dan proses bisnis yang terdapat dalam dokumen. Tidak digunakan template dalam pembentukan kalimat; semua kalimat ditulis secara alami untuk merepresentasikan pertanyaan pengguna yang realistis.

Tabel 1. Sample Hasil Pembentukan Kalimat

No	Kalimat	Fitur/Kebutuhan/Proses Bisnis
1.	Masukkan pesanan barang ke dalam database dengan data-data sebagai berikut: nama produk = ABC, jumlah pesanan = 5!	Administrasi melakukan input sekaligus mengecek daftar pesanan barang dari Database. Bila produk atau nutrisi ada, Administrasi akan mencetak <i>Customer Purchase Order</i> 3 rangkap. Rangkap 1 diarsipkan, rangkap 2 diserahkan pada <i>Customer</i> sambil melakukan pembayaran dan pengambilan produk atau nutrisi, rangkap 3 diserahkan ke Logistik untuk diarsipkan.
2.	Tampilkan daftar pesanan barang !	
3.	Berapa rata-rata barang terjual untuk semua produk hingga 20 Januari 2024?	
4.	Hapus 5 pesanan dengan status "dibatalkan" dari database setelah dikonfirmasi!	
5.	Tampilkan pesanan barang yang statusnya "belum selesai"!	Sistem menghapus 5 pesanan dengan status "dibatalkan" setelah dikonfirmasi bahwa pesanan tersebut
6.	Tambahkan pesanan baru untuk pelanggan dengan nama "Ahmad Nasrul".	Administrasi memeriksa status pesanan dan memastikan barang yang belum selesai diproses. Jika status berubah, admin memperbarui sistem untuk menghindari keterlambatan pengiriman.
7.	Berapa jumlah barang yang telah habis terjual hingga saat ini?	Administrasi menginput data pesanan baru dari pelanggan, mencatat detail barang, kuantitas, dan tanggal pesanan ke dalam sistem.
8.	Tampilkan barang yang stoknya kurang dari 10 unit!	Logistik memeriksa stok barang yang terjual dari laporan penjualan harian untuk analisis permintaan produk dan <i>restocking</i> .
9.	Berapa total pesanan pelanggan pada bulan Januari 2024?	Sistem membantu logistik memprioritaskan pengisian ulang barang yang hampir habis dengan memberi laporan secara otomatis.
		Sistem menghitung jumlah total pesanan berdasarkan periode waktu tertentu yang diminta oleh bagian administrasi atau keuangan untuk laporan bulanan.

Pada sampel diatas, beberapa kalimat diturunkan dari satu prosedur dalam proses bisnis. Seperti yang sudah dijelaskan, selain proses bisnis, penelitian ini juga mempertimbangkan fitur dan deskripsi kebutuhan dalam pembentukan kalimat.

2. Perancangan Pendekatan Berbasis *Rule-Based Approaches*



Gambar 4. Rancangan Pendekatan Berbasis *Rule-based Approaches*

Penelitian ini mengusulkan pendekatan berbasis aturan (*rule-based approach*) untuk mengidentifikasi elemen kuantitas dalam kalimat berbahasa Indonesia yang akan dikonversi ke dalam *query SQL*. Rancangan sistem yang dikembangkan terdiri atas beberapa komponen utama, yaitu: antarmuka pengguna, mesin inferensi, dan basis pengetahuan. Sistem ini menerima input

berupa kalimat pengguna, dan menghasilkan *output* berupa kesimpulan kuantitas berdasarkan penalaran logis terhadap basis pengetahuan.

Alur kerja sistem dimulai dari input berupa kalimat berbahasa Indonesia yang dimasukkan melalui antarmuka pengguna. Komponen ini bertugas melakukan praproses teks seperti segmentasi dan normalisasi agar kalimat dapat direpresentasikan dalam bentuk yang dapat dipahami oleh mesin inferensi. Mesin inferensi kemudian berperan dalam menalar informasi menggunakan aturan *IF-THEN* yang disimpan dalam basis pengetahuan, dengan strategi inferensi *forward chaining* untuk menghasilkan kesimpulan (Ramadhoni et al., 2023).

Basis pengetahuan dalam sistem ini berisi kumpulan aturan logika orde pertama (*First Order Logic*) yang dibangun melalui proses akuisisi pengetahuan bersama para pakar domain (Rahayu et al., 2025). Untuk mengatasi konflik antar aturan atau potensi ambiguitas dalam *reasoning*, digunakan strategi *first applicable*, yakni aturan pertama yang sesuai langsung diterapkan dalam proses inferensi.

Pendekatan ini dirancang tanpa subsistem penjelasan (*explanation subsystem*), mengingat implementasinya difokuskan pada integrasi dengan pendekatan *template-based* dalam *Text-to-SQL*. Oleh karena itu, keluaran sistem bersifat deterministik dan langsung dapat digunakan dalam proses konversi ke bentuk SQL. Deskripsi rinci dari masing-masing komponen dijabarkan pada bagian berikutnya.

a. Data Pengguna

Masukan utama pendekatan berbasis *rule-based approaches* dalam penelitian ini adalah kalimat dalam Bahasa Indonesia. Kalimat-kalimat ini terdiri dari tiga jenis: pertanyaan, perintah, dan pernyataan deklaratif. Berikut adalah contoh sampel data pengguna:

Tabel 2. Sample Data Pengguna

No	Kalimat	Jenis
1	Berapa jumlah total pesanan pada bulan Desember 2023?	Pertanyaan
2	Apa rata-rata barang yang terjual untuk produk merk "ABC" bulan ini?	Pertanyaan
3	Tampilkan semua data pelanggan yang melakukan transaksi pada Januari 2024.	Pertanyaan
4	Tambahkan pesanan baru untuk pelanggan bernama "Siti Aminah" dengan jumlah barang 3 unit!	Perintah
5	Hapus data pesanan dengan ID "P12345"!	Perintah
6	Update status pesanan ID "P12346" menjadi "Selesai"!	Perintah
7	Semua pesanan pelanggan pada bulan ini telah tercatat di sistem.	Deklaratif/pernyataan
8	Basis data kami mencatat bahwa stok barang "XYZ" telah habis.	Deklaratif/pernyataan
9	Transaksi yang dilakukan pada tanggal 1 Februari 2024 mencapai rekor tertinggi.	Deklaratif/pernyataan

Penelitian ini hanya mempertimbangkan masukan berupa kalimat lengkap dan tunggal. Kalimat harus memiliki struktur sintaksis Bahasa Indonesia yang utuh. Kalimat dalam bentuk paragraf atau unit yang lebih kecil seperti frasa tidak diproses oleh sistem.

b. Komponen antarmuka pengguna

Komponen antarmuka pengguna bertanggung jawab melakukan pemrosesan awal terhadap kalimat. Komponen ini terdiri atas beberapa subproses, yaitu filtering untuk memastikan input adalah kalimat valid dengan tanda baca dan panjang minimum tertentu, *text cleaning* yang menstandarisasi huruf menjadi huruf kecil serta menghapus tanda baca dan URL, tokenisasi untuk memisahkan kalimat menjadi kata berdasarkan spasi, dan stemming untuk mengubah kata berimbuhan menjadi bentuk dasarnya. Proses ini penting agar kalimat siap diproses oleh komponen mesin inferensi tanpa kehilangan makna semantik yang penting.

c. Komponen basis pengetahuan

Basis pengetahuan dalam penelitian ini berfungsi sebagai wadah penyimpanan representasi pengetahuan (*knowledge*) yang digunakan dalam proses penalaran (*reasoning*) oleh mesin inferensi. Pengetahuan direpresentasikan dalam bentuk himpunan *production rule* atau aturan *IF-THEN*. Setiap aturan terdiri atas bagian *antecedent* dan *conclusion*, dengan kemungkinan lebih dari satu *antecedent* dalam satu aturan (Fadila, 2024). Hubungan antar *antecedent* dapat berupa operator logika seperti AND, OR, NOT, dan XOR. Bentuk umum aturan dinyatakan sebagai:

IF [NOT] *antecedent* ([AND|OR|XOR] [NOT] *antecedent*) THEN *consequent**, dengan tanda kurung siku [] menyatakan bagian opsional, dan tanda kurung biasa (*) menunjukkan pengulangan yang mungkin lebih dari satu kali.

Pembentukan basis pengetahuan dilakukan melalui proses akuisisi pengetahuan (*knowledge acquisition*) yang terdiri atas dua fase utama. Fase pertama adalah perencanaan, yang mencakup penentuan pakar domain dan teknik akuisisi yang digunakan. Kriteria pakar domain dalam penelitian ini mencakup profesional dengan pengalaman minimal dua tahun dalam membuat atau menganalisis *query SQL*, atau akademisi yang mengajar topik terkait secara berkelanjutan selama dua tahun, memiliki ketersediaan waktu, komitmen, serta kemampuan kerja sama, mampu memahami teks berbahasa Indonesia, dan mampu mengomunikasikan hasil penalaran secara verbal maupun tertulis. Penelitian ini melibatkan 15 pakar domain, terdiri atas 5 akademisi dan 10 profesional. Pendekatan multi-pakar dipilih untuk meminimalkan bias, ketidakkonsistenan, dan ketidaklengkapan pengetahuan.

Kedua, teknik akuisisi pengetahuan yang digunakan adalah *Consensus Decision-Making* (teknik voting). Dalam tahap ini, *knowledge engineer* menyusun sekumpulan kalimat dari data yang dikumpulkan dan mengkonversinya menjadi daftar pertanyaan yang harus dijawab oleh para pakar. Pakar diminta untuk mengidentifikasi elemen kuantitas pada setiap kalimat dan memberikan indikator atau alasan atas klasifikasinya. Contohnya adalah pengenalan angka numerik, pecahan, atau tanggal sebagai bentuk kuantitas. Penilaian kuantitas dilakukan secara kolektif oleh semua pakar, dan jika terjadi ketidaksepakatan, keputusan akhir diambil berdasarkan suara terbanyak. Karena jumlah pakar ganjil, proses *voting* dapat berlangsung secara efektif tanpa memerlukan bobot tambahan.

Teknik voting yang digunakan dalam penelitian ini diterapkan melalui tahapan konkret berupa pembentukan sekumpulan pertanyaan oleh *knowledge engineer*. Pertanyaan-pertanyaan tersebut disusun dengan cara mengolah kalimat-kalimat yang diperoleh pada tahap pengumpulan data. Setiap kalimat kemudian dianalisis oleh pakar domain untuk mengidentifikasi keberadaan elemen kuantitas yang terkandung di dalamnya, beserta indikator yang menjadi dasar klasifikasi. Contoh elemen kuantitas yang dikenali antara lain angka numerik, pecahan, dan penanda waktu seperti tanggal. Hasil anotasi pakar disusun dalam sebuah tabel yang mencatat kalimat, elemen kuantitas yang teridentifikasi, serta alasan atau indikator klasifikasinya. Jika suatu kalimat tidak relevan dengan domain basis data yang sedang dianalisis, maka meskipun mengandung kata atau angka yang menyerupai kuantitas, elemen tersebut tidak diklasifikasikan sebagai kuantitas.

Setiap pakar domain mengisi bagian identifikasi kuantitas dan indikatornya, sementara bagian informasi awal seperti ID kalimat dan domain basis data diisi oleh *knowledge engineer*. Pakar juga diberikan ruang untuk memberikan catatan tambahan sebagai justifikasi jika suatu kata tidak dimasukkan sebagai kuantitas. Catatan ini sangat penting karena dapat menjadi dasar dalam penyusunan aturan (*rule*) untuk mendeteksi elemen non-kuantitas dalam teks.

Keputusan akhir terhadap klasifikasi dilakukan melalui mekanisme *voting*, yaitu pengambilan keputusan berdasarkan jumlah suara terbanyak jika terjadi perbedaan pendapat antar pakar. Setiap suara pakar memiliki bobot yang sama, dan karena jumlah pakar ganjil, maka proses pengambilan keputusan dapat dilakukan secara efisien tanpa memerlukan sistem pembobotan tambahan.

Untuk menunjang efisiensi proses anotasi, formulir pengisian disediakan secara daring melalui aplikasi berbasis web. Selain itu, setiap pakar juga diberi akses terhadap Entity Relationship Diagram (ERD) dari basis data yang relevan dengan domain masing-masing. Akses ini bertujuan

agar pakar dapat melakukan validasi konteks dan memastikan bahwa kalimat yang dianalisis memang berhubungan dengan domain basis data tertentu. Kalimat yang tidak relevan akan dieliminasi dari proses anotasi. Hasil ekstraksi pengetahuan ini menjadi dasar dalam pembentukan aturan-aturan sistem yang secara otomatis mampu mengenali baik elemen kuantitas maupun non-kuantitas dalam proses konversi teks ke *SQL (Text-to-SQL)*.

Untuk mendukung kemudahan input, form pengisian pertanyaan disediakan secara daring melalui aplikasi web. Dalam proses ini, pakar juga diberi akses pada *ERD (Entity Relationship Diagram)* dari basis data terkait agar dapat mengevaluasi relevansi setiap kalimat dengan domain basis data. Kalimat yang tidak relevan akan diabaikan dalam penentuan kuantitas. Hasil ekstraksi pengetahuan ini kemudian digunakan untuk membentuk *rule-rule* sistem yang mampu mengenali baik kuantitas maupun non-kuantitas secara otomatis dalam proses *Text-to-SQL*.

Analisis Pengetahuan fase ini terdiri dari satu proses yaitu menerjemahkan hasil ekstraksi pengetahuan ke representasi pengetahuan. Seperti yang telah diuraikan sebelumnya, penelitian ini menggunakan kumpulan *IF-THEN rule* sebagai representasi pengetahuan. Setiap rule ditulis dalam *First Order Logic (FOL)*. Berikut ini adalah sampel rule yang diperoleh pada saat analisis pengetahuan.

Tabel 3. Sample Rule-based Approaches

Id Rule	Rule Based Approaches	First Order Logic (FOL)
R1	IF kata adalah angka literal (regex $\backslash d+$) THEN kelas kata adalah bilangan bulat	$\forall x(\text{kata}(x) \wedge \text{regexDigit}(x) \rightarrow \text{BIL_BULAT}(x))$
R2	IF kata adalah satuan (misal: unit, kg, liter, dsb.) THEN kelas kata adalah satuan	$\forall x(\text{kata}(x) \wedge \text{termasuk}(x, \text{SATUAN_SET}) \rightarrow \text{SATUAN}(x))$
R3	IF kata adalah bulan (misal: april, mei) THEN kelas kata adalah bulan	$\forall x(\text{kata}(x) \wedge \text{termasuk}(x, \text{BULAN_SET}) \rightarrow \text{BULAN}(x))$
R4	IF kata adalah hari (misal: senin, selasa) THEN kelas kata adalah hari	$\forall x(\text{kata}(x) \wedge \text{termasuk}(x, \text{HARI_SET}) \rightarrow \text{HARI}(x))$
R5	IF kata adalah presentase (regex $\backslash d+\%$) THEN kuantitas	$\forall x(\text{kata}(x) \wedge \text{regexPersen}(x) \rightarrow \text{KUANTITAS}(x))$
R6	IF kata adalah bilangan pecahan (regex $\backslash d+[.,]\backslash d+$) THEN kelas kata adalah bilangan pecahan	$\forall x(\text{kata}(x) \wedge \text{regexDesimal}(x) \rightarrow \text{BIL_PECAHAN}(x))$
R7	IF kata adalah bilangan kata (misal: satu, dua) THEN kelas kata bilangan bulat	$\forall x(\text{kata}(x) \wedge \text{regexDesimal}(x) \rightarrow \text{BIL_PECAHAN}(x))$
R8	IF kata adalah awalan ke- dan bilangan THEN kelas kata urutan	$\forall x(\text{kata}(x) \wedge \text{regexKeBilangan}(x) \rightarrow \text{URUTAN}(x))$
R9	IF kata1 bilangan dan kata2 satuan dan posisi berurutan THEN kuantitas	$\forall x \forall y(\text{posisiSetelah}(y, x) \wedge \text{BIL_BULAT}(x) \wedge \text{SATUAN}(y) \rightarrow \text{KUANTITAS}(x, y))$
R10	IF kata adalah harga (regex $\text{rp}\backslash d+, \backslash d+$) THEN kuantitas	$\forall x(\text{kata}(x) \wedge \text{regexHarga}(x) \rightarrow \text{KUANTITAS}(x))$
R11	IF tiga kata berurutan (bilangan bulat + bulan + bilangan bulat) THEN kuantitas	$\forall x,y,z(\text{posisiSetelah}(y,x) \wedge \text{posisiSetelah}(z,y) \wedge \text{BIL_BULAT}(x) \wedge \text{BULAN}(y) \wedge \text{BIL_BULAT}(z) \rightarrow \text{KUANTITAS}(x,y,z))$
R12	IF dua kata berurutan (bilangan bulat + bulan) THEN kuantitas	$\forall x,y(\text{posisiSetelah}(y,x) \wedge \text{BIL_BULAT}(x) \wedge \text{BULAN}(y) \rightarrow \text{KUANTITAS}(x,y))$
R13	IF dua kata berurutan (bulan + bilangan bulat) THEN kuantitas	$\forall x,y(\text{posisiSetelah}(y,x) \wedge \text{BULAN}(x) \wedge \text{BIL_BULAT}(y) \rightarrow \text{KUANTITAS}(x,y))$

R14	IF frase numerik (lebih dari 2 kata bilangan/kepilatan) + satuan kuantitas THEN	$\forall x,y(\text{posisiSetelah}(y,x) \wedge \text{FRASE_NUMERIK}(x) \wedge \text{SATUAN}(y) \rightarrow \text{KUANTITAS}(x,y))$
R15	IF numerik diikuti oleh bilangan pecahan atau satuan THEN kuantitas	$\forall x,y(\text{NUMERIK}(x) \wedge (\text{PECAHAN}(y) \cup \text{SATUAN}(y)) \wedge \text{posisiSetelah}(y,x) \rightarrow \text{KUANTITAS}(x,y))$
R16	IF numerik diikuti simbol matematika (>, <, =, >=, <=) THEN kuantitas	$\forall x(\text{NUMERIK}(x) \wedge \text{SIMBOL}(x) \rightarrow \text{KUANTITAS}(x))$
R17	IF numerik berdiri sendiri dalam konteks pengukuran atau pembandingan THEN kuantitas	$\forall x(\text{NUMERIK}(x) \wedge \text{CONTEXT_MENGUKUR}(x) \rightarrow \text{KUANTITAS}(x))$

Seperti yang ditunjukkan pada sampel diatas, setiap *rule* yang berbentuk FOL terdiri dari *quantifier*, *predicate*, penghubung logis, variabel, konstanta. Pada penelitian ini, jenis DML menjadi *predicate*. Alasan penggunaan FOL pada penelitian adalah memudahkan dalam proses verifikasi basis pengetahuan karena FOL sudah berbentuk baku sehingga mengurangi adanya ambiguitas.

d. Verifikasi basis pengetahuan

Verifikasi basis pengetahuan dilakukan untuk memastikan bahwa seluruh aturan yang disusun bersifat konsisten dan lengkap. Proses ini terdiri atas dua tahap utama, yaitu verifikasi konsistensi dan verifikasi keterlengkapan (*completeness*). Pada tahap pertama, pemeriksaan konsistensi bertujuan mengidentifikasi potensi masalah seperti *redundant rules*, *conflicting rules*, *subsumed rules*, *unnecessary IF conditions*, dan *circular rules*. Untuk mendukung proses ini, digunakan *Matrix Element-Rule*, yaitu matriks yang merepresentasikan hubungan antara elemen-elemen pengetahuan dan aturan (*rule*) yang mengandungnya. Dalam matriks ini, baris menunjukkan elemen-elemen pengetahuan, sementara kolom menunjukkan aturan. Setiap sel dalam matriks diisi dengan label A jika elemen bertindak sebagai *antecedent*, atau C jika elemen bertindak sebagai *conclusion*. Struktur ini memudahkan identifikasi aturan-aturan yang saling tumpang tindih, tidak relevan, atau saling bertentangan dalam proses penalaran.

Tahap kedua adalah verifikasi keterlengkapan, yang bertujuan mendeteksi kondisi seperti *unreferenced attribute values*, *illegal attribute values*, *unreachable conclusions*, *dead-end IF conditions*, dan *dead-end goals*. Untuk mendukung analisis ini, digunakan *Matrix Element-Element*, yaitu matriks yang merepresentasikan hubungan ketergantungan (*dependency*) antar elemen pengetahuan. Dalam matriks ini, baris menunjukkan elemen yang berperan sebagai *antecedent*, sedangkan kolom menunjukkan elemen yang menjadi *conclusion*. Setiap sel matriks diisi dengan ID aturan yang menunjukkan adanya hubungan kausal. Tidak seperti *Matrix Element-Rule*, matriks ini bersifat tidak simetris karena mengandung arah hubungan sebab-akibat.

Analisis terhadap kedua jenis matriks ini dilakukan secara manual oleh *knowledge engineer* untuk memastikan bahwa seluruh elemen memiliki peran yang valid dalam sistem dan tidak terdapat aturan yang terisolasi atau tidak dapat dijangkau oleh proses penalaran. Dengan demikian, proses verifikasi ini memastikan bahwa basis pengetahuan yang dibangun memiliki integritas logis dan fungsional dalam mendukung sistem *reasoning* berbasis *rule*.

e. Komponen mesin inferensi

Komponen mesin inferensi bertugas menyimpulkan hasil dari kalimat masukan dengan menerapkan aturan-aturan yang tersedia. Mesin inferensi menggunakan strategi *forward chaining*, yaitu menarik kesimpulan dari fakta awal ke tujuan akhir. Strategi *first applicable* diterapkan untuk memilih aturan pertama yang cocok dan langsung digunakan, sehingga sistem dapat bekerja secara efisien walaupun jumlah aturan terbatas.

3. Rancangan Ujicoba

Uji coba dilakukan untuk mengukur performa dari pendekatan berbasis *rule-based approaches* yang diusulkan dalam penelitian ini. Terdapat dua aspek utama dalam proses pengujian, yaitu skenario uji coba dan evaluasi performa sistem.

a. Skenario ujicoba

Data yang diperoleh dari tahap pengumpulan data dibagi menjadi dua bagian, yaitu 80% sebagai data latih dan 20% sebagai data uji. Pembagian ini mengikuti praktik umum dalam penelitian

pemrosesan bahasa alami (*Natural Language Processing*) dan bertujuan untuk memastikan bahwa sistem diuji secara objektif dengan data yang belum pernah digunakan selama proses pelatihan.

Pada tahap pelatihan, data latih digunakan dalam proses pembentukan basis pengetahuan. Setiap kalimat dianalisis untuk mengidentifikasi keberadaan kuantitas, kemudian diberikan anotasi berupa label jenis kuantitas (misalnya angka numerik, pecahan, tanggal, harga, satuan, dan sebagainya), serta penandaan kata atau frasa yang bertindak sebagai indikator kuantitas. Informasi tersebut digunakan oleh *knowledge engineer* untuk menyusun aturan berbasis *IF-THEN* yang menjadi inti dari sistem berbasis aturan.

Sementara itu, data uji hanya digunakan untuk mengevaluasi performa sistem dalam mendeteksi kuantitas secara otomatis. Data ini tidak digunakan dalam proses pembentukan aturan agar hasil evaluasi tidak terpengaruh oleh bias pelatihan dan tetap menjaga objektivitas sistem. Dalam data uji, hanya disediakan label jenis kuantitas tanpa penandaan indikator, sehingga sistem diuji sepenuhnya berdasarkan kemampuan reasoning-nya terhadap teks alami.

Pemilihan data latih dan data uji dilakukan secara acak (*randomized*), dengan mempertimbangkan distribusi kuantitas dari berbagai jenis agar tetap seimbang. Strategi ini diterapkan untuk menghindari dominasi satu jenis kuantitas tertentu yang dapat memengaruhi performa sistem secara tidak adil. Evaluasi sistem dilakukan dengan satu skenario pengujian, yaitu menggunakan strategi inferensi *forward chaining* dan pendekatan *reasoning first applicable*. Strategi ini dipilih karena sesuai dengan karakteristik sistem berbasis aturan yang dikembangkan, serta mempertimbangkan ukuran basis pengetahuan yang relatif masih terbatas dan dapat diproses secara efisien.

b. Evaluasi

Evaluasi performa pendekatan dilakukan untuk mengukur kemampuan sistem dalam mengidentifikasi elemen kuantitas secara akurat dari kalimat berbahasa Indonesia. Evaluasi ini menggunakan metrik standar dalam klasifikasi, yaitu akurasi (*accuracy*), presisi (*precision*), recall, dan *F1-Score*, yang dihitung berdasarkan *Confusion Matrix* (Putra et al., 2024).

Dalam penelitian ini, *Confusion Matrix* disusun sesuai dengan jumlah label kuantitas yang diidentifikasi, seperti bilangan bulat, bilangan pecahan, tanggal, persentase, satuan, dan non-kuantitas. Setiap baris pada matriks mewakili hasil prediksi sistem, sedangkan kolom menunjukkan label sebenarnya. Nilai pada diagonal utama matriks menunjukkan jumlah prediksi yang benar untuk setiap kelas (*True Positive/TP*), sementara nilai di luar diagonal menggambarkan kesalahan prediksi antar kelas (*False Positive dan False Negative*).

Tabel *Confusion Matrix* memberikan gambaran distribusi performa sistem terhadap masing-masing kategori kuantitas. Misalnya, jika sistem memprediksi elemen sebagai bilangan bulat tetapi label sebenarnya adalah tanggal, maka akan tercatat sebagai kesalahan prediksi pada posisi baris bilangan bulat dan kolom tanggal.

Evaluasi dilakukan menggunakan pendekatan *macro-average*, di mana setiap kelas kuantitas diberi bobot yang sama, tanpa memperhatikan distribusi jumlah data pada tiap kelas. Pendekatan ini dipilih untuk menilai performa sistem secara adil pada semua kategori, termasuk kelas minoritas yang jumlah datanya lebih sedikit.

Tabel 4. Rumus Evaluasi

Metrik	Rumus
Recall Kelas	$recall\ kelas\ k = \frac{TP\ kelas\ k}{TP\ kelas\ k + FN\ kelas\ k}$
Recall Keseluruhan	$recall\ keseluruhan = \frac{1}{K} \sum_{i=1}^K recall_i$
Precision Kelas	$precision\ kelas\ k = \frac{TP\ kelas\ k}{TP\ kelas\ k + FP\ kelas\ k}$
Precision Keseluruhan	$precision\ keseluruhan = \frac{1}{K} \sum_{i=1}^K precision_i$

$$F1\text{-Score} = 2 \frac{\text{precision keseluruhan} * \text{recall keseluruhan}}{\text{precision keseluruhan} + \text{recall keseluruhan}}$$

Keterangan :

TP(i): *True Positive* pada kelas ke-i

FN(i): *False Negative* pada kelas ke-i

FP(i): *False Positive* pada kelas ke-i

K: Jumlah kelas kuantitas

Evaluasi ini memberikan gambaran objektif mengenai kinerja pendekatan berbasis aturan dalam mengenali berbagai bentuk kuantitas. Hasil evaluasi menjadi indikator penting dalam menilai efektivitas pendekatan yang diusulkan, serta dasar untuk pengembangan lebih lanjut dalam sistem *Text-to-SQL* berbasis *rule*.

III.HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan sebuah sistem identifikasi kuantitas berbasis pendekatan aturan (*rule-based*) dalam konteks *Text-to-SQL* berbahasa Indonesia. Sistem dikembangkan menggunakan *framework Experta* berbasis *forward chaining* di *Python*, yang memanfaatkan aturan *IF-THEN* untuk mendeteksi dan mengekstraksi entitas numerik, baik dalam bentuk digit maupun bilangan kata, dengan fokus pada pola linguistik kuantitatif yang umum dijumpai dalam kalimat interogatif terhadap basis data.

1. Gambaran Umum Pengujian

Penelitian ini menghasilkan sebuah sistem identifikasi kuantitas berbasis pendekatan aturan (*rule-based*) dalam konteks *Text-to-SQL*. Sistem dikembangkan menggunakan *framework Experta* berbasis *forward chaining* di *Python*, yang memungkinkan pemisahan antara basis pengetahuan (aturan *IF-THEN*) dan fakta kalimat sebagai data uji. Tujuan utama sistem adalah mendeteksi dan mengekstraksi entitas numerik, baik dalam bentuk digit maupun bilangan kata.

Pengujian dilakukan terhadap 1.000 kalimat uji fiktif yang dirancang sesuai kaidah bahasa Indonesia (EYD). Kalimat-kalimat tersebut disusun untuk mencerminkan berbagai struktur pertanyaan umum dalam domain basis data, seperti “Tampilkan...”, “Hitung...”, dan “Berapa...”. Setiap kalimat telah dilabeli secara manual (*ground-truth*) untuk tiga kategori utama kuantitas.

Komposisi data ini dipilih untuk memastikan sistem diuji terhadap berbagai pola linguistik dan ekspresi kuantitatif yang beragam, baik eksplisit maupun implisit.

Tabel 5. Tabel Distribusi Data Uji

Jenis Kuantitas	Total Kalimat
Kuantitas Digit	815 Kalimat
Kuantitas Kata	164 Kalimat
Non-Kuantitas	21 Kalimat

2. Hasil deteksi identifikasi kuantitas

Proses deteksi kuantitas terdiri dari empat tahap utama, yaitu tokenisasi dan normalisasi, penyusunan fakta awal (*initial fact assertion*), penerapan aturan deteksi berbasis *IF-THEN*, serta pemberian label *NO_QUANTITY* apabila tidak ada aturan yang terpenuhi, yang hasilnya kemudian dibandingkan dengan label *ground-truth* untuk mengevaluasi kinerja sistem.

Tabel 6. Tabel Distribusi Hasil Deteksi Kuantitas

Jenis Kuantitas	Jumlah Kalimat	Terdeteksi Benar	Gagal Deteksi	Salah Label
Digit	815	762	35	18
Kata	164	160	2	2
Non-Kuantitas	21	21	37	0
Total	1000	943	37	20

Sistem berhasil mendeteksi kuantitas secara akurat pada 94,3% dari total data uji, yang menunjukkan bahwa pendekatan *rule-based* yang dikembangkan cukup andal dalam mengidentifikasi kuantitas eksplisit dalam teks, sebagaimana ditunjukkan pada contoh “Tampilkan 5 produk terlaris bulan April 2025” yang menghasilkan keluaran ['5 produk', 'april 2025'] dengan nilai terdeteksi ['5', 'april 2025'], serta “List produk yang terjual antara 20 sampai 50 unit per hari” yang menghasilkan ['20', '50 unit'] dengan nilai terdeteksi ['20', '50'].

3. Evaluasi Performa Sistem

Evaluasi kinerja dilakukan menggunakan metrik klasifikasi seperti *Precision*, *Recall*, *F1-Score*, dan Akurasi, yang dihitung berdasarkan perbandingan prediksi sistem terhadap label *ground truth*.

Tabel 7. Tabel Distribusi Evaluasi Performa Sistem Berdasarkan Identifikasi Kuantitas

Jenis Agregasi	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
Digit	0.97	0.95	0.96
Kata	0.96	0.94	0.95
Non-Kuantitas	1.00	1.00	1.00
Rata-rata	0.98	0.96	0.97

Selain itu, evaluasi dengan metrik multilabel tambahan menunjukkan performa sistem yang sangat tinggi dan konsisten, dengan nilai *Exact Match Ratio* sebesar 0,76 (760 dari 1.000 kalimat seluruh labelnya sesuai), *Jaccard Similarity* sebesar 0,75 (menunjukkan kesamaan tinggi antara label prediksi dan *ground-truth*), serta *Hamming Loss* sebesar 0,00 yang mencerminkan tingkat kesalahan label per *instance* yang sangat kecil. Hasil evaluasi menunjukkan sistem memiliki performa yang sangat tinggi dan konsisten, baik dalam hal ketepatan prediksi maupun cakupan deteksi.

4. Analisis kesalahan dan kelemahan sistem

Meskipun sistem menunjukkan tingkat akurasi yang tinggi dalam mendeteksi kuantitas pada kalimat berbahasa Indonesia, terdapat beberapa kasus kesalahan yang memengaruhi performa keseluruhan, baik dalam bentuk gagal deteksi (*false negative*) maupun deteksi yang salah (*false positive*). Kesalahan ini terutama disebabkan oleh bentuk bahasa yang tidak formal, ambiguitas struktur kalimat, dan keterbatasan cakupan aturan dalam basis pengetahuan.

Tabel 8. Tabel Contoh Gagal Deteksi dan Deteksi Salah

Kalimat	Label Sebenarnya	Prediksi Sistem	Jenis Kesalahan
"Tampilkan detail beli dengan jumlah antara 20 sampe 50 unit."	['antara 20 sampe 50 unit']	[]	Gagal Deteksi
"Berapa pembelian dari tanggal 1 sd 15 februari."	['1 sd 15 februari']	[1 sd 15]	Salah Deteksi
"Tampilkan 7 laporan pertama di database."	['7 laporan', 'pertama']	['7', 'Pertama']	Salah Deteksi

5. Analisis Penyebab Kesalahan

Beberapa faktor yang menyebabkan kegagalan sistem dalam mendeteksi kuantitas secara tepat di antaranya adalah ketidaksesuaian bentuk bahasa informal, seperti pada frasa “sampe” yang tidak dikenali sebagai bentuk alternatif dari “sampai”, sehingga aturan rentang numerik gagal tereksekusi; kesalahan dalam parsing angka besar, seperti “100” yang dipecah menjadi token terpisah “10” dan “0” karena interpretasi yang keliru terhadap pemisah ribuan; deteksi terfragmentasi, misalnya pada frasa “7 laporan pertama” yang dipisah menjadi dua entitas terpisah meskipun secara semantik merupakan satu kesatuan; serta ketidak lengkapan dalam mendeteksi rentang waktu, seperti pada frasa “1 sd 15 februari” yang hanya sebagian teridentifikasi karena sistem belum mengantisipasi satuan waktu yang terpisah dari angka. Secara umum, penyebab utama dari kesalahan ini adalah keterbatasan aturan yang terlalu spesifik atau terlalu umum, kurangnya cakupan representasi satuan dalam basis pengetahuan, serta belum adanya pemrosesan semantik mendalam untuk memahami

konteks ekspresi kuantitatif. Hal ini menunjukkan tantangan inheren dalam sistem berbasis aturan yang sangat bergantung pada eksplisitnya pola linguistik yang didefinisikan melalui aturan *IF-THEN*.

IV. PEMBAHASAN

Berdasarkan hasil pengujian sistem Identifikasi Kuantitas Berbasis *rule-based* yang dikembangkan dalam penelitian ini, dapat disimpulkan bahwa pendekatan *rule-based* mampu mendeteksi kuantitas eksplisit baik dalam bentuk digit maupun kata bilangan dengan tingkat keberhasilan yang tinggi pada struktur yang konsisten dan eksplisit. Sistem ini dirancang untuk mengekstraksi entitas numerik dalam kalimat Bahasa Indonesia dan diimplementasikan menggunakan kerangka kerja *Experta* dengan strategi inferensi *forward chaining* dan resolusi konflik *first applicable*.

Tingginya jumlah kuantitas yang berhasil terdeteksi secara benar, yaitu 943 dari 1000 kalimat (94.3%), mencerminkan kekuatan sistem dalam mengenali pola numerik yang umum dan eksplisit. Sebagai contoh, sistem mampu mengenali frasa “5 produk” dan “50 unit” sebagai kuantitas yang valid. Selain itu, pada kategori kuantitas kata seperti “dua puluh” atau “seratus lima puluh”, deteksi juga berjalan efektif karena adanya aturan khusus yang mencocokkan bentuk kata dengan konversi numerik lain.

Evaluasi performa menggunakan metrik klasifikasi *multilabel* menunjukkan bahwa sistem memperoleh nilai *Micro Precision* sebesar 0.98, namun *Micro Recall* hanya mencapai 0.66. Hal ini menandakan bahwa prediksi yang dilakukan sistem cenderung benar ketika berhasil mendeteksi, tetapi masih banyak kasus yang terlewat. Nilai *Macro* yang lebih rendah (*Precision*, *Recall*, dan *F1* sebesar 0.52) memperkuat bahwa performa sistem sangat bergantung pada jenis label tertentu yang lebih mudah terdeteksi, seperti digit.

Namun demikian, terdapat pula 37 kasus gagal deteksi (*false negative*) dan 20 kasus salah label (*false positive*) yang menunjukkan bahwa sistem belum sepenuhnya mampu menangani ekspresi numerik dengan struktur linguistik yang bervariasi atau konteks yang ambigu. Misalnya, sistem gagal menyertakan unsur waktu seperti “Februari” dalam frasa “1 sd 15 Februari” karena tidak ada aturan yang menggabungkan angka rentang dengan nama bulan secara eksplisit.

Dibandingkan dengan pendekatan berbasis *machine learning*, sistem ini memiliki keunggulan dalam transparansi dan *explainability*. Setiap prediksi dapat ditelusuri ke aturan yang aktif dalam proses inferensi. Hal ini sangat bermanfaat untuk aplikasi yang memerlukan jejak penalaran seperti sistem edukasi atau rekomendasi berbasis pengetahuan. Namun, sistem masih kurang fleksibel dalam mengakomodasi variasi bahasa yang luas atau konteks yang tidak eksplisit.

Penelitian-penelitian sebelumnya yang juga menggunakan pendekatan *rule-based* menunjukkan pola hasil yang serupa, yaitu sangat baik untuk ekspresi eksplisit tetapi lemah pada struktur kalimat yang kompleks. Dengan demikian, hasil penelitian ini memperkuat temuan bahwa pendekatan *rule-based* cocok digunakan pada skenario dimana interpretabilitas lebih diutamakan daripada adaptabilitas mode.

Dengan akurasi rata-rata sekitar 0.76 (*Exact Match*), pendekatan ini tetap relevan dan menjanjikan untuk diterapkan di domain yang menekankan pada penelusuran logika sistem. Untuk meningkatkan fleksibilitas dan mengurangi kesalahan klasifikasi, pengembangan selanjutnya dapat diarahkan pada pendekatan hibrida yang menggabungkan sistem *rule-based* dengan model pembelajaran mesin untuk menangani ekspresi yang tidak terjangkau oleh aturan statis.

V. KESIMPULAN

Penelitian ini telah menghasilkan sistem identifikasi kuantitas berbasis pendekatan *rule-based* dalam konteks *Text-to-SQL* untuk kalimat berbahasa Indonesia. Sistem dikembangkan menggunakan *framework* *Experta* dengan aturan logika *IF-THEN*, serta menerapkan strategi inferensi *forward chaining* dan resolusi konflik *first applicable*. Sistem dirancang untuk mengenali kuantitas dalam bentuk digit maupun kata bilangan melalui pemindaian pola linguistik eksplisit.

Hasil evaluasi menunjukkan bahwa sistem ini mampu mencapai *Micro Precision* sebesar 0.98, *Micro Recall* sebesar 0.66, dan *Micro F1-score* sebesar 0.79. Selain itu, sistem memperoleh nilai

Exact Match Ratio sebesar 0.76, yang mengindikasikan bahwa sekitar tiga perempat kalimat uji memiliki prediksi kuantitas yang benar seluruhnya. Namun, perbedaan antara *Micro F1* dan *Macro F1* (hanya 0.52) menunjukkan adanya ketimpangan performa antar label, di mana sistem lebih unggul dalam mendeteksi label mayoritas seperti kuantitas digit, dan kurang optimal dalam menangani label minoritas seperti bilangan kata atau bentuk ekspresi yang tidak umum

Selain dari segi akurasi, sistem juga menunjukkan efisiensi tinggi dengan rata-rata waktu eksekusi sekitar 0,009 detik per kalimat, menjadikannya layak untuk skenario penggunaan real-time. Efisiensi ini diperoleh berkat penggunaan teknik pencocokan pola yang ringan seperti regex, tanpa melibatkan komputasi kompleks.

Keseluruhan hasil ini menunjukkan bahwa pendekatan *rule-based* yang dikembangkan memiliki potensi besar dalam tugas identifikasi kuantitas, khususnya pada struktur kalimat eksplisit dan terstruktur. Sistem ini juga menawarkan transparansi dalam proses inferensi, sehingga cocok diterapkan pada aplikasi yang membutuhkan *explainability*, seperti pendidikan, sistem rekomendasi, dan audit data.

Sebagai saran untuk pengembangan selanjutnya, pendekatan ini dapat diperkuat dengan integrasi teknik machine learning untuk mengatasi keterbatasan dalam mengenali ekspresi kuantitas yang lebih kompleks atau tidak langsung. Selain itu, pengayaan aturan dan penyesuaian terhadap keragaman linguistik dapat membantu meningkatkan *recall* dan generalisasi sistem.

REFERENSI

- Eza, P. E. P. (2025). PERANCANGAN DATA BASE SISTEM PEMBELAJARAN SEKOLAH DASAR MENGGUNAKAN ERD. *TEKNOFILE: Jurnal Sistem Informasi*, 3(5), 314–326.
- Fadila, M. D. (2024). *Evaluasi program bimbingan konseling menggunakan model countenance stake di smp al fath cirendeu*. Jakarta: FITK UIN Syarif Hidayatullah Jakarta.
- Hu, L., Xu, S., Li, C., Yang, C., Shi, C., Duan, N., Xie, X., & Zhou, M. (2020). Graph Neural News Recommendation with Unsupervised Preference Disentanglement. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 4255–4264). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.acl-main.392>
- Pulungan, S. M., Febrianti, R., Lestari, T., Gurning, N., & Fitriana, N. (2023). Analisis teknik entity-relationship diagram dalam perancangan database. *Jurnal Ekonomi Manajemen Dan Bisnis (JEMB)*, 1(2).
- Putra, F., Ihsan, R. M., Tahiyat, H. F., & Efrizoni, L. (2024). Evaluasi Performa Aplikasi Gojek Melalui Klasifikasi Kata Ulasan Pengguna Dengan Metode SVM. *Techno. Com*, 23(3).
- Qin, B., Hui, B., Wang, L., Yang, M., Li, J., Li, B., Geng, R., Cao, R., Sun, J., Si, L., Huang, F., & Li, Y. (2022). *A Survey on Text-to-SQL Parsing: Concepts, Methods, and Future Directions*. <https://doi.org/10.48550/arXiv.2208.13629>
- Rahayu, P. W., Judijanto, L., Syaddad, H. N., Apriyanto, A., & Adawiyah, R. (2025). *Pengantar Sistem Pakar: Teori dan Implementasi*. PT. Green Pustaka Indonesia.
- Raifer, M., Rotman, G., Apel, R., Tennenholtz, M., & Reichart, R. (2022). Designing an Automatic Agent for Repeated Language-based Persuasion Games. *Transactions of the Association for Computational Linguistics*, 10, 307–324. https://doi.org/10.1162/tacl_a_00462
- Ramadhoni, D. F., Abadi, L. P., & Suaedah, S. (2023). Implementasi metode forward chaining pada sistem pakar dalam mendiagnosa penyakit kucing. *Jurnal Rekayasa Komputasi Terapan*, 3(03), 111–117.

Yang, S., Yang, Z., Chen, X., Zhao, J., & Ma, Y. (2021). Distributed aggregation-based attributed graph summarization for summary-based approximate attributed graph queries. *Expert Systems with Applications*, 176, 114921. <https://doi.org/https://doi.org/10.1016/j.eswa.2021.114921>