

PENERJEMAH TEKS BAHASA INDONESIA KE DALAM DML(DATA MANIPULATION LANGUAGE) DENGAN SUB-QUERY

Debby Meidi P. P.¹, Ken Kinanti Purnamasari²

^{1,2}Teknik Informatika – Universitas Komputer Indonesia

Jl. Dipatiukur 112-114 Bandung 40132

Email : debbymeidi13@gmail.com¹, ken.kinanti@email.unikom.ac.id²

ABSTRAK

Beberapa penelitian yang telah dilakukan mengenai penerjemahan bahasa alami ke dalam bahasa query, adalah SQL parser yang dilakukan oleh Nendi Isharmawan, dimana penelitian tersebut berfokus kepada pengolahan bahasa alami yang bertugas sebagai jembatan untuk mengakses data yang berada di dalam basis data. Penelitian lain yang telah dilakukan oleh Ihsan Faturohman, penelitian ini berfokus pada fungsi select, pada kedua penelitian ini perintah sub-query masih belum terdeteksi. Metode yang digunakan dalam penelitian ini adalah dengan metode rule-based. Proses utama terbagi menjadi dua tahap yaitu preprocessing dan translasi. Tahap Preprocessing terdiri dari case folding, filtering, tokenizing kata, stemming, dan stopword removal. Tahap Translasi terdiri dari deteksi kata kunci, deteksi tabel dan kolom, tokenizing perintah, identifikasi perintah DML, indentifikasi konten, dan penyusunan query. Berdasarkan hasil pengujian terhadap 30 kalimat perintah yang terdiri dari 6 kombinasi query dihasilkan akurasi sekitar 82,35%. Kesalahan yang terjadi diakibatkan belum tertanganinya kalimat pencarian tanggal dan juga sub-query dengan lebih dari dua tabel dan sub-query dengan jenis nested query. Hal tersebut dapat ditangani dengan menambahkan aturan-aturan pendeteksi nested sub-query dan menambahkan pendeteksian tanggal untuk pencarian.

Kata kunci : Translasi, Query, Data Manipulation Language, Basis Data, Pemrosesan Bahasa Alami

1. PENDAHULUAN

Basis Data atau lebih sering disebut Database adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap table yang ada[1], untuk mengakses database dijumpai oleh suatu sistem yang disebut dengan DBMS (Database Management System). SQL merupakan bahasa standar yang paling banyak digunakan dalam DBMS, dalam SQL sendiri terbagi kembali menjadi dua yaitu DDL(Data Definition Language) untuk mendefinisikan tabel dalam database dan DML(Data Manipulation Language) untuk memanipulasi data atau informasi yang berada

didalam database sehingga menjadi informasi yang berguna, Sub-query merupakan bagian dari DML yang jika diartikan adalah query dalam query, sehingga untuk mendapatkan informasi pada tabel ataupun dalam relasional tabel yang sangat kompleks, kita dapat menggunakan kondisi dengan sub permintaan tertentu. Sub-query sendiri dapat membuat perintah SQL yang relatif lebih simpel jika dibandingkan dengan fungsi join sehingga mempersingkat perintah serta dapat menggunakan fungsi logika yang lebih mudah dipahami daripada query biasa[1].

Penelitian yang telah dilakukan sebelumnya mengenai penerjemahan bahasa alami ke dalam bahasa query adalah penelitian yang dilakukan mengenai bagaimana menerjemahkan bahasa alami kedalam bahasa SQL dengan memperhatikan kata-kata dalam kalimat yang dimasukkan tanpa melihat struktur katanya, yang nantinya kata-kata tersebut akan dibandingkan dengan daftar keyword yang ada[2], namun dalam penelitian tersebut hanya dapat mendeteksi query biasa yang tanpa mengandung sub-query. Penelitian lain yang telah dilakukan adalah SQL parser yang dilakukan oleh Nendi Isharmawan[3], dimana penelitian tersebut berfokus kepada pengolahan bahasa alami yang bertugas sebagai jembatan untuk mengakses data yang berada di dalam basis data, dalam penelitian ini hasil terjemahan query belum ada yang menghasilkan sub-query. Penelitian lain yang telah dilakukan adalah penerjemahan bahasa alami ke bahasa SQL yang dilakukan oleh Ihsan Faturohman, penelitian ini berfokus pada fungsi select yang terbagi menjadi enam yaitu select dengan kondisi, tanpa kondisi, banyak kondisi, banyak tabel tanpa kondisi, banyak tabel dengan banyak kondisi dan table order[4], pada penelitian ini perintah sub-query masih belum terdeteksi.

Berdasarkan penjelasan di atas terlihat bahwa belum adanya penelitian yang dapat menerjemahkan dari bahasa Indonesia ke bahasa SQL yang mengandung penggunaan sub-query didalamnya. Dalam proses penerjemahan ke dalam bahasa SQL akan dilakukan dengan menggunakan metode *rule-based* mengikuti dengan penelitian yang sudah dilakukan sebelumnya.

2. ISI PENELITIAN

2.1 NLP(Natural Language Proessing)

NLP merupakan bagian dari ilmu AI yang memiliki fokus pada pemrosesan bahasa alami, dimana bahasa alami adalah bahasa yang umum digunakan oleh manusia untuk saling berkomunikasi satu sama lain, namun bahasa yang diterima oleh komputer harus diproses terlebih dahulu dan dipahami agar maksud dari pengguna dapat dipahami[5].

2.2 Basis Data

Basis Data atau lebih sering disebut *Database* adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap table yang ada. Satu database menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi[6].

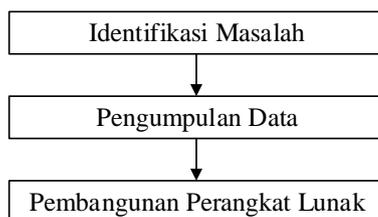
Database atau basis data sangat penting dalam sistem informasi karena basis data merupakan gudang penyimpanan data yang dapat diolah lebih lanjut. Peran basis data menjadi sangat penting karena dapat mengorganisir data dan juga menghindari duplikasi data atau lebih sering disebut dengan redundansi. Dalam basis data ada proses database management system atau disingkat DBMS yang merupakan suatu sistem yang memungkinkan admin database melakukan pengaksesan data, pengendalian dan juga pemeliharaan data secara efisien. Dalam penelitian ini basis data digunakan untuk menyimpan data pendukung sistem dan juga digunakan sebagai media pengetesan dari perintah hasil penerjemahan bahasa Indonesia ke bahasa SQL.

2.3 Sub-query

Jika diartikan sub-query sendiri adalah query dalam query. Jadi untuk mendapatkan informasi pada sebuah tabel ataupun dalam relasional tabel yang sangat kompleks, kita dapat menggunakan kondisi dengan sub permintaan tertentu[1].

2.4 Metode Penelitian

Metode penelitian yang akan digunakan adalah metode eksperimental dimana dalam metode ini dilakukan observasi di bawah kondisi buatan yang dibuat dan diatur, dimana dilakukan manipulasi terhadap objek yang diteliti sehingga adanya kontrol dalam penelitian tersebut[5]. Adapun gambaran dari alur penelitian yang dilakukan dapat dilihat pada Gambar 1.1 berikut.



Gambar 1. 1 Metode Penelitian

Berikut merupakan langkah-langkahnya.

1. Identifikasi masalah
Dalam tahapan ini dilakukan pengamatan terhadap masalah yang ada pada penelitian sebelumnya dan juga melihat contoh kasus perintah sub-query yang ada sehingga dapat membantu untuk mendefinisikan kebutuhan yang akan membantu dalam pencapaian penelitian yang dilakukan.
2. Pengumpulan data
Pada tahapan ini dilakukan pengumpulan data yang dapat mendukung penelitian seperti studi literatur dari buku elektronik, buku, jurnal dan contoh kasus yang ada untuk menentukan variabel-variabel apa saja yang dibutuhkan untuk mencapai penelitian yang dilakukan.
3. Pembangunan perangkat lunak
Metode yang digunakan dalam tahapan ini adalah metode prototype, metode ini digunakan karena dalam prosesnya dibuat purwarupa terlebih dahulu lalu diuji, ketika hasil uji yang didapat belum memenuhi kriteria yang diinginkan maka dapat dilakukan analisis ulang kembali.

2.5 Analisis Masalah

Data Manipulation Language (DML) merupakan suatu metode perintah SQL yang digunakan untuk proses pengolahan isi data di dalam tabel seperti menampilkan, memasukan, mengubah, menghapus isi data dan tidak terkait dengan perubahan struktur serta definisi tipe data dari objek database. Perintah SQL yang termasuk dalam DML antara lain select, update, dan delete. Pada penelitian ini, perintah DML yang digunakan hanya perintah *select*.

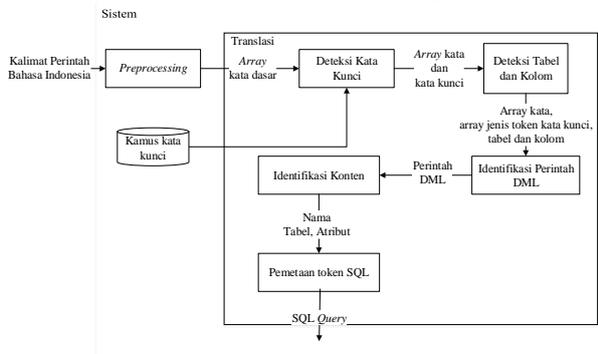
2.6 Gambaran Umum Sistem

Dalam penelitian ini akan dibangun sebuah sistem penerjemah bahasa Indonesia ke dalam query DML. Tahap sistem dimulai dari penerimaan input kalimat bahasa alami dalam bahasa Indonesia kemudian diproses pada tahap preprocessing dan tahap translasi. Pada tahap preprocessing terdapat beberapa tahapan diantaranya case folding, filtering, tokenizing kata, stemming, dan removing stopword, sedangkan pada tahap translasi terdapat tahap deteksi kata kunci, deteksi tabel dan kolom, identifikasi perintah DML, identifikasi konten, dan penyusunan query.

Setelah tahap preprocessing menghasilkan token - token kata dasar yang akan diproses pada tahap translasi dengan mendeteksi setiap token kata dengan kata kunci awal pada kamus, setelah mendeteksi kata kunci awal dilanjutkan dengan membandingkan setiap token kata dengan kamus kata kunci yang telah ditentukan.

Jika hasil perbandingan antara token kata dengan kata kunci, kemudian dilakukan perbandingan dengan data tabel dan kolom, maka

token kata teridentifikasi sebagai query DML, kemudian tahap identifikasi konten yaitu penelusuran dari sisa token kata untuk mengisi konten pada template query yang sudah teridentifikasi sebelumnya, dan pada tahap akhir yaitu penyusunan query, kemudian hasil query dari proses pemetaan token SQL akan ditampilkan.



Gambar 1. 2 Gambaran Sistem

2.7 Analisis Data Masukan

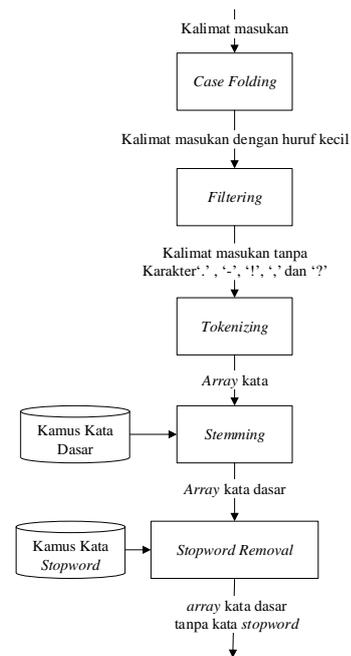
Penelitian ini memiliki data masukan berupa kalimat perintah dalam bahasa Indonesia yang dimasukan oleh pengguna dan data keluaran berupa query SQL. Adapun contoh dari kalimat masukan yang akan ditangani pada penelitian ini dapat dilihat pada tabel berikut.

Tabel 1. 1 Contoh Perintah masukan

Contoh Perintah DML dalam Bahasa Indonesia
Tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota Bandung!

2.8 Preprocessing

Tahapan ini merupakan proses awal untuk mempersiapkan data masukan yang berupa kalimat teks bahasa Indonesia sebelum proses translasi. Dalam banyak penelitian NLP proses *preprocessing* sangat penting dilakukan untuk menghasilkan akurasi yang lebih baik[7]. Proses ini memiliki beberapa tahapan yaitu *case folding*, *filtering*, *tokenizing kata*, *stemming*, dan *removing stopwords*.



Gambar 1. 3 Proses Preprocessing

1. Case Folding

Proses *case folding* merupakan penyeragaman setiap huruf menjadi huruf kapital atau huruf kecil[8]. Dalam penelitian ini setiap huruf diseragamkan menjadi huruf kecil.

Tabel 1. 2 Contoh Proses Case Folding

Sebelum	Sesudah
Tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota Bandung!	tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota bandung!

2. Filtering

Filtering merupakan proses untuk memilah setiap karakter yang ada pada kalimat masukan yang memiliki tujuan untuk mengurangi karakter yang dianggap tidak diperlukan oleh proses selanjutnya, sehingga dapat mengurangi *noise*. Pada penelitian ini, karakter yang dibolehkan menjadi masukan hanya 'a' sampai 'z' dan '0' sampai '9'.

Tabel 1. 3 Contoh Proses *Filtering*

Sebelum	Sesudah
tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota bandung!	tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota bandung

3. *Tokenizing*

Tokenizing adalah proses untuk memisahkan kalimat menjadi kata – kata (token) dengan parameter pemisah berupa spasi (‘ ’), dengan parameter tersebut kalimat dapat diubah menjadi kumpulan token yang dimana token tersebut dapat berupa kata atau angka[7]. Proses ini dilakukan karena pada proses selanjutnya akan dilakukan pemrosesan setiap kata. Hasil dari tokezing kata yaitu berupa token – token kata. Dalam penelitian ini proses tokenizing diterapkan agar sistem dapat mendeteksi setiap kata yang telah dipisahkan menjadi kumpulan token.

Tabel 1. 4 Contoh Proses *Tokenizing*

Sebelum	Sesudah	
	Indeks Token	Token Kata
tampilkan data transaksi dengan kode_mitra yang mitranya berasal dari kota bandung	1	tampilkan
	2	data
	3	transaksi
	4	dengan
	5	kode_mitra
	6	yang
	7	mitranya
	8	berasal
	9	dari
	10	kota
	11	bandung

4. *Stemming*

Stemming merupakan proses untuk mengubah suatu kata yang memiliki imbuhan menjadi kata dasar. Pada penelitian ini, proses *stemming* menggunakan *library stemmer* untuk bahasa Indonesia dari sastrawi yang telah mengadopsi Algoritma Nazief & Andriani[9]. Dalam bahasa Indonesia awalan dan akhiran

disebut dengan imbuhan, contohnya adalah imbuhan “-nya’ seperti dalam kata “mitranya” yang memiliki kata dasar “mitra”. Proses *stemming* dilakukan karena kata yang akan dideteksi merupakan kata dasar sehingga perlu dilakukan proses penghapusan imbuhan.

Tabel 1. 5 Contoh Proses *Stemming*

Indeks Token	Sebelum	Sesudah
1	tampil <u>kan</u>	tampil
2	data	data
3	transaksi	transaksi
4	dengan	dengan
5	kode_mitra	kode_mitra
6	yang	yang
7	mitr <u>nya</u>	mitra
8	<u>ber</u> asal	asal
9	dari	dari
10	kota	kota
11	bandung	bandung

5. *Stopword Removal*

Stopword removal adalah proses pengilangan kata yang sering muncul dan dianggap tidak penting, dalam suatu bahasa *stopword* membantu untuk membangun struktur kalimat tetapi tidak mewakili konten apapun dari isi kalimat tersebut[10], beberapa contohnya adalah kata ‘yang’, ‘dan’, ‘dengan’, ‘atau’ dan lain – lain. Proses *stopword removal* dilakukan untuk menghapuskan kalimat-kalimat yang dianggap tidak penting yang dapat menimbulkan hasil dari proses translasi menjadi kurang baik atau keliru.

Tabel 1. 6 Contoh Proses *Stopword Removal*

Indeks Token	Sebelum	Sesudah
1	tampil	tampil
2	<u>data</u>	
3	transaksi	transaksi
4	<u>dengan</u>	
5	kode_mitra	kode_mitra
6	<u>Yang</u>	
7	mitra	mitra

Tabel 1.6 Contoh Proses *Stopword Removal* (Lanjutan)

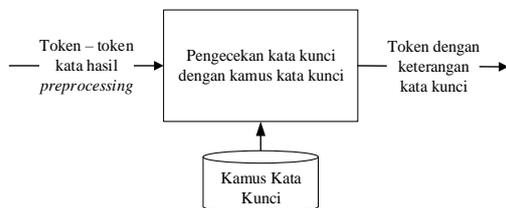
Indeks Token	Sebelum	Sesudah
8	asal	asal
9	<u>Dari</u>	
10	Kota	kota
11	bandung	bandung

2.9 Translasi

pada proses translasi dilakukan penerjemahan dari token – token kata hasil *preprocessing* ke dalam bentuk perintah *query* DML. kalimat masukan yang telah melalui tahap *preprocessing* ke dalam bentuk *query* DML. Pada penelitian ini, proses translasi terbagi menjadi lima tahap yaitu deteksi kata kunci, deteksi tabel dan kolom, identifikasi perintah DML, identifikasi konten dan pemetaan token SQL.

1. Deteksi Kata Kunci

dalam proses deteksi kata kunci, setiap token kata dasar yang telah didapatkan dari hasil tahap *preprocessing* dilakukan proses pendeteksian kata kunci untuk menentukan jenis dari setiap token yang akan digunakan dalam proses selanjutnya.



Gambar 1. 4 Proses Pendeteksian Kata Kunci

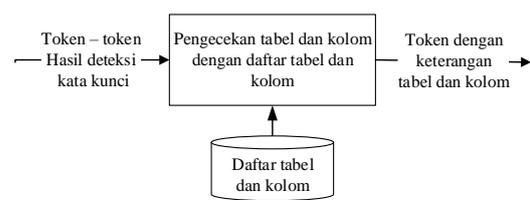
Dari proses diatas dapat dihasilkan kata kunci yang terdeteksi seperti berikut.

Tabel 1. 7 Hasil Proses Deteksi Kata Kunci

Posisi Token	Token Kata	Jenis Token
1	tampil	Perintah
2	asal	Kondisi

2. Deteksi Tabel dan Kolom

Pada proses ini, setiap token kata dasar yang tidak termasuk dalam kata kunci akan melalui proses pendeteksian tabel dan kolom untuk menentukan jenis dari setiap token yang akan digunakan dalam proses selanjutnya.



Gambar 1. 5 Proses Deteksi Tabel dan Kolom

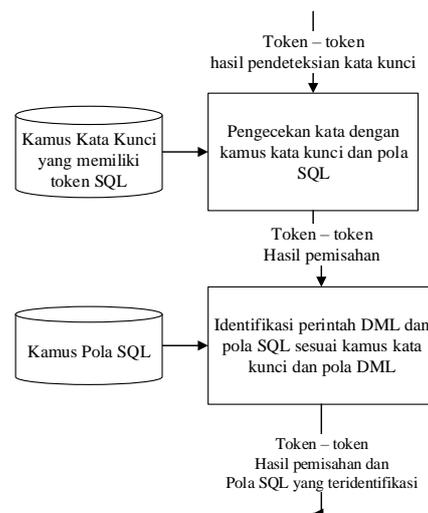
Dari proses pendeteksian tabel dan kolom di atas maka dihasilkan *array* yang sudah memiliki jenis token seperti berikut.

Tabel 1. 8 Hasil Pendeteksian Tabel dan Kolom

Posisi Token	Token Kata	Jenis Token
1	Tampil	Perintah
2	transaksi	Tabel
3	kode_mitra	Kolom
4	mitra	Tabel
5	Asal	Kondisi
6	kota	Kolom
7	Bandung	-

3. Identifikasi Perintah DML

Pada proses ini dilakukan pengidentifikasian semua token yang memiliki jenis token yang termasuk kata kunci seperti jenis token perintah dan jenis konten kondisi yang dibandingkan dengan data yang berada dalam database sehingga dapat menentukan perintah DML yang sesuai.



Gambar 1. 6 Proses Identifikasi Perintah DML

Dari proses yang perbandingan di atas diperoleh hasil seperti berikut.

Tabel 1. 9 Contoh Hasil Identifikasi Perintah DML

Posisi Token	Token	Jenis Token	Token Dalam DML
1	tampil	<u>Perintah</u>	<u>SELECT</u>
2	transaksi	Tabel	transaksi
3	kode_mitra	Kolom	kode_mitra
4	mitra	Tabel	mitra
5	Asal	<u>Kondisi</u>	<u>IN</u>
6	kota	Kolom	kota
7	Bandung	-	bandung

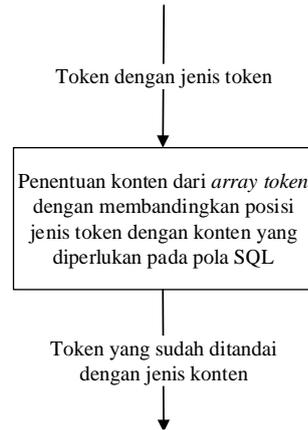
Setelah didapatkan perintah DML yang sesuai selanjutnya dilakukan proses pencarian pola SQL dengan cara membandingkan susunan jenis token dengan kamus data yang berada dalam database sehingga didapatkan pola SQL yang sesuai, dalam proses ini pola SQL yang ditemukan adalah 'Perintah + Tabel + Kolom + Tabel + Kondisi + Kolom'.

Tabel 1. 10 Contoh Hasil Pencarian Pola SQL

Pola SQL Teridentifikasi
SELECT * FROM [TABEL1] WHERE [KOLOM1] [KONDISI1] (SELECT [KOLOM1] FROM [TABEL2] WHERE [KOLOM2] = [KATA_KUNCI_PENCARIAN])

4. Identifikasi Konten

Dalam tahap ini dilakukan proses untuk menentukan konten dari pola pada proses sebelumnya. Pengecekan konten dilakukan dengan cara melakukan pengecekan pada setiap token pada susunan array token yang sudah diberi tanda jenis token, tabel, kolom dan kondisi. Pada proses sebelumnya ditemukan pola yang mengandung [tabel1], [tabel2], [kolom1], [kolom2], [kondisi] dan [kata_kunci_pencarian], maka proses ini dilakukan untuk menentukan tabel1, tabel2, kolom1, kolom2, kondisi dan kata_kunci_pencarian, yang akan dibuat.



Gambar 1. 7 Proses Identifikasi Konten

Penjelasan dari proses di atas dapat dilihat sebagai berikut.

- a. Menentukan konten tabel untuk [tabel1] dan [tabel2], kolom untuk [kolom1] dan [kolom2]

Tabel 1. 11 Contoh Hasil Identifikasi Konten Tabel dan Kolom

Posisi Token	Token	Jenis Token	Jenis Konten
1	tampil	Perintah	-
2	transaksi	<u>Tabel</u>	<u>Tabel1</u>
3	kode_mitra	<u>Kolom</u>	<u>Kolom1</u>
4	mitra	<u>Tabel</u>	<u>Tabel2</u>
5	IN	Kondisi	-
6	kota	<u>Kolom</u>	<u>Kolom2</u>
7	bandung	-	-

- b. Menentukan konten kondisi untuk [kondisi]

Tabel 1. 12 Contoh Hasil Identifikasi Konten Kondisi

Posisi Token	Token	Jenis Token	Jenis Konten
1	tampil	Perintah	-
2	transaksi	Tabel	Tabel1
3	kode_mitra	Kolom	Kolom1
4	mitra	Tabel	Tabel2
5	IN	<u>Kondisi</u>	<u>Kondisi</u>
6	kota	Kolom	Kolom2
7	bandung	-	-

- c. Menentukan konten [kata_kunci_pencarian]

Tabel 1. 13 Contoh Hasil Identifikasi Konten Kata Kunci Pencarian

Posisi Token	Token	Jenis Token	Jenis Konten
1	tampil	Perintah	-
2	transaksi	Tabel	Tabel1
3	kode_mitra	Kolom	Kolom1
4	mitra	Tabel	Tabel2
5	IN	Kondisi	Kondisi
6	kota	Kolom	Kolom2
7	bandung	-	<u>kata kunci pencarian</u>

5. Pemetaan Token SQL

Pada tahap ini dilakukan proses pemetaan token yang sesuai dengan hasil dari proses identifikasi perintah DML dan identifikasi konten.

Tabel 1. 14 Contoh Hasil Pemetaan Token SQL

Jenis Konten	Konten	Sebelum	Sesudah
		Pola SQL	Pola SQL
Tabel1	transaksi	SELECT *	SELECT
Kolom1	kode_mitra	FROM	* FROM
Tabel2	mitra	[TABEL1]	transaksi
Kolom2	kota	WHERE	WHERE
Kondisi 1	IN	[KOLOM1]	kode_mitra
kata_kunci_pencarian	bandung	[KONDISI1]	ra IN (
		(SELECT	SELECT
		[KOLOM1]	kode_mitra
		FROM	ra FROM
		[TABEL2]	mitra
		WHERE	WHERE
		[KOLOM2]	kota =
		=	'bandung
		[kata_kunci_pencarian])	')

2.10 Hasil Pengujian

Pengujian akurasi yang dilakukan pada penelitian ini bertujuan untuk mengetahui keberhasilan dari sistem menerjemahkan kalimat bahasa Indonesia ke dalam DML dengan sub-query. Adapun fungsi DML yang diuji adalah fungsi *select* dengan kondisi "=", IN dan NOT IN. Data kalimat masukan berupa kalimat bahasa Indonesia baku yang mengandung perintah DML dengan sub-query. Pengujian dilakukan dengan cara membandingkan hasil translasi dari sistem dengan query harapan yang telah ditentukan sebelumnya.

Tabel 1. 15 Hasil Pengujian

No.	Jenis Perintah	Jumlah kalimat	Hasil Pengujian	
			Benar	Salah
1	SELECT '=' MAX	4	3	1
2	SELECT '=' MIN	4	4	0
3	SELECT '=' AVG	3	2	1
5	SELECT '>' MIN	3	3	0
6	SELECT '>' AVG	3	3	0
7	SELECT '<' AVG	5	4	1
8	SELECT IN	6	4	2
9	SELECT NOT IN	6	5	1
Total		34	28	6

Cara perhitungan mengacu kepada *accuracy measure* karangan Abidin[11] untuk nilai akurasi keseluruhan dari data masukan pada sistem penerjemahan kalimat bahasa Indonesia ke dalam bahasa SQL yang mengandung sub-query.

Berdasarkan pengujian yang telah dilakukan dengan menggunakan pengujian black box dan pengujian akurasi didapatkan kesimpulan seagai berikut.

1. Fungsionalitas sistem berjalan dengan semestinya
2. Akurasi yang didapatkan cukup baik namun hasil translasi masih ada yang belum sesuai dikarenakan beberapa hal yaitu:
 - a. Kolom alias yang isinya sama pada lebih dari satu tabel belum dapat terdeteksi alias yang dimaksudkan oleh pengguna untuk tabel yang mana.
 - b. Pencarian dengan menggunakan tanggal belum terformat sehingga dalam pengujian query pada database tidak akurat

3. PENUTUP

3.1 Kesimpulan

Berdasarkan kepada hasil dari implementasi dan pengujian terhadap penelitian yang telah dilakukan menghasilkan kesimpulan bahwa sistem dapat menerjemahkan kalimat perintah DML berbahasa Indonesia yang mengandung sub-query dan menghasilkan query DML dengan sub-query. Penelitian ini dapat menerjemahkan kalimat bahasa Indonesia ke dalam perintah DML dengan sub-query dengan akurasi 82,35%.

3.2 Saran

Berdasarkan kepada hasil analisis dan pengujian yang telah dilakukan didapatkan saran-saran agar penelitian ini dapat menghasilkan penerjemahan yang lebih baik, diantaranya adalah sebagai berikut.

1. Menambahkan aturan yang dapat sub-query bersarang atau nested sub-query dan dapat menangani lebih dari dua tabel.
2. Menambahkan pemeriksaan dan penanganan untuk ejaan kata yang salah.
3. Menambahkan aturan untuk pendeteksian penulisan tanggal lengkap, bulan – tanggal, dan bulan saja.

DAFTAR PUSTAKA

- [1] B. Nugroho, *Database Relasional Dengan Mysql*. Yogyakarta: Andi, 2005.
- [2] Kuspriyanto, H. Sujani, H. Tjahjono, and S. Kusuma, "Perancangan Translator Bahasa Alami Ke Dalam Format SQL (Structured Query Language)," vol. 10, no. 2005, p. 12, 2005.
- [3] N. Isharmawan, "Ekstraksi Informasi Dan SQL Parser Untuk Query Sql Dari Teks Berbahasa Indonesia," Universitas Komputer Indonesia, 2017.
- [4] I. Faturhman, "Perancangan Translator Bahasa Alami Ke Dalam Format SQL (Structured Query Language) Dengan Fokus Pada Kasus Retrieval Informasi," Universitas Komputer Indonesia, 2018.
- [5] W. Budiharto and D. Suhartono, *Artificial Intelegece Konsep dan Penerapannya*. 2014.
- [6] A. Fadilsyah, *Pemrograman Database (Konsep dan Implementasi)*. Yogyakarta: Graha Ilmu, 2008.
- [7] K. K. Purnamasari and I. S. Suwardi, "Rule-based Part of Speech Tagger for Indonesian Language," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 407, Sep. 2018.
- [8] I. H. Witten, "Text mining," 2002.
- [9] M. Adriani, J. Asian, B. Nazief, and E. Williams, "Stemming Indonesian : A Confi x-Stripping Approach," *ACM Trans. Asian Lang. Inf. Process*, vol. 6, no. 4, pp. 1–33, 2007.
- [10] B. Liu, *Web Data Mining Second Edition*, 2nd ed. Springer-Verlag Berlin Heidelberg, 2011.
- [11] Abidin, "Accuracy Measure," *Bahan kuliah data mining, Progr. Stud. Tek. Inform. FMIPA Univ. Syiah Kuala*, 2012.