TRANSLATION OF INDONESIAN LANGUAGE TEXT INTO DATA DEFINITION LANGUAGE (DDL) WITH COMPOUND SENTENCES HANDLING

Iqram Anwar[1], Ken Kinanti Purnamasari[2]

[1,2] Informatics Engineering- Indonesia Computer University
Jl. Dipatiukur 112-114 Bandung 40132
Email : iqramanwar@email.unikom.ac.id[1], ken.kinanti@email.unikom.ac.id[2]

ABSTRACT

Special knowledge is needed to be used with DBMS by understanding the language format of SQL (Structured Query Language). Another way to use DBMS is using Indonesian. Natural languages cannot be directly processed by DBMS, allowing a system that can be used to change the SQL language. One of the basic data is Data Definition Language (DDL) which is used to create, modify, and delete basic data and objects needed in the database. The method used to translate Word commands in Indonesian into DDL queries is the Rule-Based method. The process in the system that is built has two main stages, namely Preprocessing and Translation. Preprocessing stage consists of case folding, filtering, tokenizing words, stemming, and remove stopword. The translation phase consists of keywords, tokenizing commands, DDL identification commands, content identification, and query compilation. Based on the results of trials on 25 types of words with the total number of sentences as many as 93, the average results of the accuracy obtained were 72.04%.

Kata kunci : Translasi, *Query*, Bahasa Alami, *Structured Query Language, Data Definition Language*

1. PRELIMINARY

DBMS (Database Management System) is an intermediary medium for users with a database stored on storage media. How to interact between users and the database with a special language that has been arranged by the DBMS company. A Database Language can usually be divided into 2 forms, Data Definition Language (DDL) and Data Manipulation Language (DML) [1]. Data Definition Language (DDL) in the database is used to create, change, and delete databases and objects needed in the database [1]. Data Manipulation Language is a form of database language that is useful for manipulating and retrieving data on a database [1].

Several studies have been carried out in the use of databases with intermediaries in natural languages including English [2] [3] and Indonesian [4] [5] [6]. Research in English has been able to translate natural language into DDL and DML queries but the accuracy value is unknown. Research in Indonesian only translates natural language into DML which focuses on select queries in the case of retrieving data on a database. The method used to translate natural language is to look at the words in the natural language sentence without looking at the structure of the sentence then the words are compared to the keyword list to get the intended field [5]. The results of the study obtained an accuracy value of 86% [5]. The research that focuses on DDL obtained an accuracy of 91.57% [6]. Research in Indonesian can only translate input sentences into one query, because the system that has been built cannot detect input sentences that will generate more than one query.

Based on the description above, it is needed a method that can detect input sentences that will generate more than one query. In the translation process stage, it follows the previous research stage using the rule-based method.

2. RESEARCH CONTENT

In this chapter will discuss the theoretical basis, research methods, results and discussion which are the problems to be analyzed, a general description of the system, the stages of the system process and the results of the research.

2.1 Natural Language Processing

Natural Language Processing is often applied to written applications that often encounter difficulties because the writing of each person is not always standard in accordance with the grammar so that it is not in accordance with the applicable rules. One application of natural language processing is the natural language translator system such as the translator system from English to Indonesian. The translator not only can translate words per word (dictionary) but can also translate from the original language to the target language with understandable intentions [7].

2.2 Compound Sentences

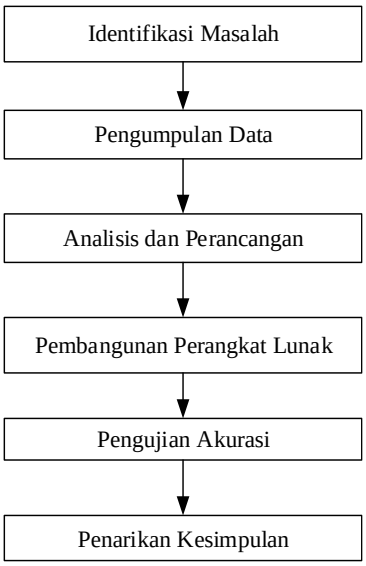
Compound sentences are sentences consisting of two or more clauses in which one clause with the other clause is interconnected [8]. Generally,

linguists divide relationships between clauses into two types, namely coordination and subordination.

1. Coordination relationship (Equivalent Compound)
   Coordination relationships combine two or more clauses, each of which has an equal position in sentence structure [8]. The sentence formed from two or more clauses with the relationship between clauses formed coordinatively can be called equivalent compound sentences, so that the sentence consists of clauses that are arranged in coordination with the conjunctions in which each clause can stand alone into sentences.
2. Subordinate Relationships (Multi-Level Compound)
   Subordinate relationships show a hierarchical relationship, that is, combining two or more clauses in stages, some of which function as main clauses and some that function as subordinate clauses [8]. Sentences that are formed from several clauses that are related to subordinate conjunctions cause one clause to be part of another clause, meaning that one clause depends on another clause and cannot stand alone. This sentence is called multilevel compound sentences.

2.3 Research Methods

In this study the author will use descriptive research methods. This method is used because this study systematically describes the facts and characteristics of objects and subjects studied precisely [9]. The research phase consists of problem identification, data collection, analysis and design, software development, drawing conclusions.



Picture 1. 1 Research Flow

The software development method used is Sequential Linear Model [10]. The stages in the model are:

1. Analysis
   Analysis of input data is the command line of the DDL command in Indonesian to build the appropriate preprocessing stage and the determination of keywords for the translation stage.
2. Design
   At this stage, the first step taken is to make the overall system design, including Preprocessing and Translation schemes.
3. Implementation
   This stage is the stage of making an application with a design that has been made at the stage of designing and implementing the process of translating Indonesian commands into SQL syntax.
4. Testing
   At this stage, the translation of the translator DDL command uses the Indonesian language whether it is correct and can execute the command as expected.

2.4 Problem Analysis

The research that has been done related to the process of translating natural language into SQL queries with Indonesian language intermediaries that focus on DML SELECT queries obtained an accuracy of 86% [5], while for the translation case in DDL the average accuracy is 91.57% [6] . But in the previous research that has been mentioned, the research only translates sentences into one SQL query, so the existing system cannot translate sentences containing more than one query command.

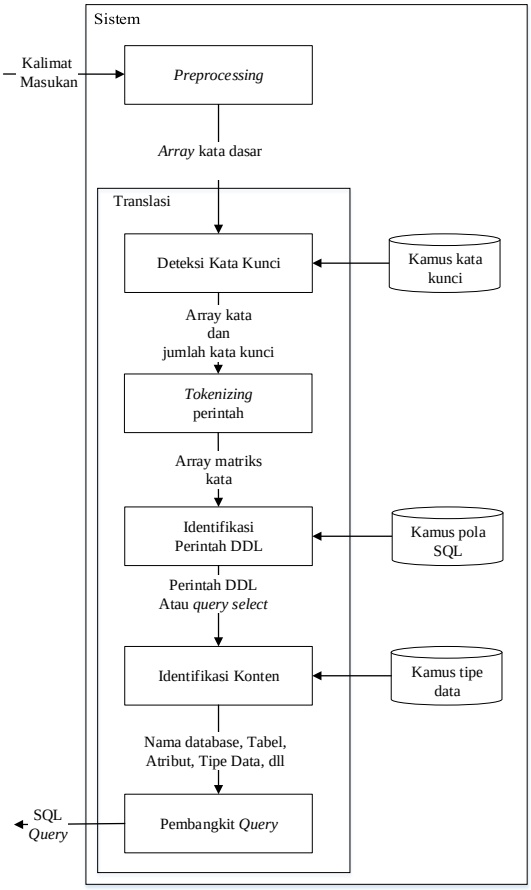
Table 1. 1 Example of Command in Indonesian

No	DDL command in Indonesian	Translation Results
1.	buatkan database dengan nama unikom kemudian hapus database dengan nama db_kampus	CREATE DATABASE unikom;
		DROP DATABASE db_kampus;

This study aims to add features so that input sentences that have more than one query command can be translated and produce more than one DDL query.

2.5 System Overview

This research has two main processes, namely preprocessing and translation. The preprocessing process has stages including case folding, filtering, tokenizing words, stemming, removing stopword and the translation process have stages including keyword detection, tokenizing commands, identification of DDL commands, content identification, query compilation.



Picture 1. 2 Block Diagram of System Flow

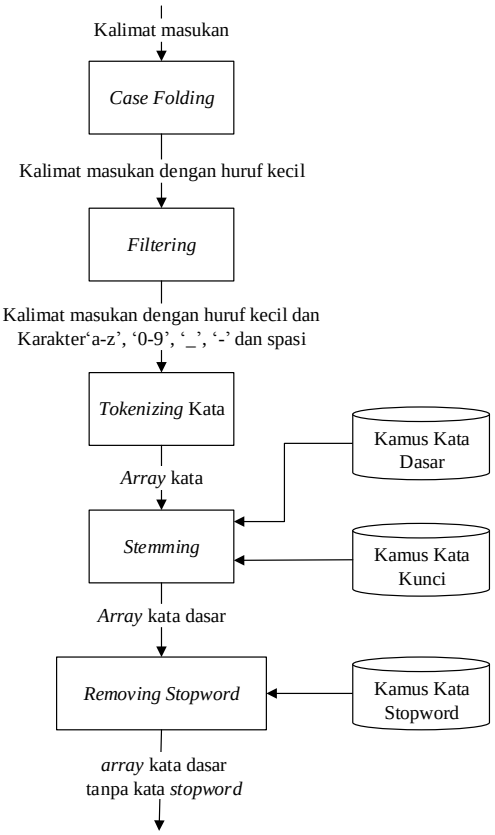
Before the preprocessing process is carried out, there are input sentences entered by the user.

Table 1. 2 Example of Input Sentence

Input Sentence	
Single Query	**Plural Queries**
Buatlah basisdata Unikom!	Tambahkan field pada Tabel user dengan tgllahir tipe data date kemudian ubah Field ‘alamat’ menjadi ‘kota’ pada tabel mahasiswa..!

2.6 *Preprocessing*

Preprocessing is the initial process for preparing input data in the form of Indonesian text sentences before entering the translation process. This process consists of several stages, namely case folding, filtering, tokenizing words, stemming, and removing stopword.



Picture 1. 3 Block Diagram of Preprocessing

1. Case Folding

   Case folding is the process of homogenizing each letter in the input sentence by changing all letters into capital (uppercase) or small (lowercase). In this study the case folding process homogenizes all the letters into small (lowercase) which is useful in the process of stemming, removing stopword, and translation because in the process the font size must be uniform.
2. Filtering

   Filtering is a stage to sort each character in the input sentence only a certain character. The purpose of the filtering process is to reduce characters that are not needed so as to reduce noise during the information retrieval process. In this study, characters that are input are only 'a' to 'z', '0' to '9', '_', ',' and spaces.
3. Word Tokenizing

   Word Tokenizing is a stage to cut sentences into words with separators in the form of spaces (''). Tokenizing of words is done because in the next stage, each word will be processed. The results of the word tokezining are in the form of word tokens.
4. Stemming

   Stemming is a stage to change a word that has particle affixes to be the basic word. The stemming process can be done using crawling techniques by comparing each word token with the basic words contained in the KBBI dictionary

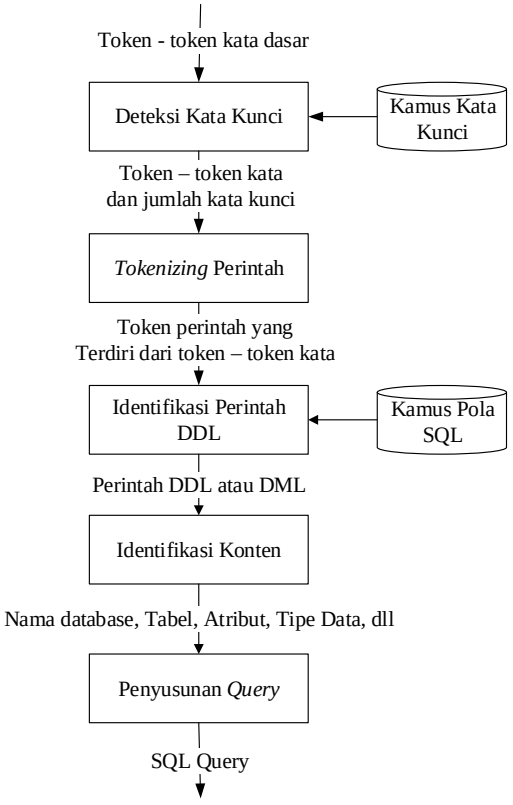
[11]. In this study, the stemming process uses the stemmer library for the Indonesian language from writers who have adopted the Nazief & Andriani Algorithm [12]. An example of using stemmer is to eliminate the affix 'to' the word 'make' so that it becomes the basic word that is 'make'. Words that are changed to basic words are only words that are found in the keyword dictionary.

5. Removing Stopword

   Stopword removal is a step to eliminate the words in the input sentence contained in the word stopword dictionary. In this study, the stopword determination is adjusted to the case being studied, usually the stopword selection depends on words that often appear in a text, for example the words 'which', 'and', 'with', 'or' and others.

2.7 Translation

Translation is the process of translating tokens input sentence word tokens from preprocessing results to DDL queries.



Picture 1. 4 Block Diagram of Translation

In this study, the translation process has 5 stages there are keyword detection, tokenizing commands, identification of ddl commands, content identification, query compilation.

1. Keyword Detection

   In this process, the basic tokens obtained from the preprocessing result will be detected by initial keyword by checking each token in the keyword dictionary. The purpose of checking keywords is to determine the number of command words that will generate a query.

Table 1. 3 Keyword Detection Result

Input Sentence	Keyword Detected	Token Position	Number of Keywords
Single Query	buat	1	1
Plural Queries	tambah	1	2
	ubah	9	

2. Command Tokenizing

   In this process the process of separating tokens will be carried out based on keyword detection data and array position as a result of the keyword detection process. The purpose of the tokenizing process is to separate each complete command sentence, so that each sentence can be translated.

3. Identification of DDL Command

   This process is a stage to identify each token - the word token in each command sentence to get a SQL DDL pattern. Each token in each command sentence is compared to the keyword dictionary to get the DDL command. The purpose of the DDL identification process is to get the SQL pattern template for each command sentence. The word token in each command sentence that has been identified as a keyword will be compared to the SQL pattern dictionary, so that the SQL pattern template will be identified.

4. Content Identification

   The handlebar template check is done by searching for characters in the SQL pattern which starts with "{{" and ends with "}}". In the SQL pattern, the DDL command identification results found the handlebar template {{nama_db}}, {{table_name}}, {{nama_kolom}}, {{nama_kolom_baru}}, and {{column_ipe_ops}}, this process is to determine the table name, column name, new column name, and column name that will be created by searching each word token. The purpose of this process is to determine the contents of all handlebar template content from the rest of the tokens that are identified by the DDL command.

5. Query Preparation

   in this process, the preparation of queries is carried out according to the results obtained from the identification process of the DDL command and the content identification process. The purpose of the query preparation process is to

map each of the result identification tokens into the template handlebar.

Table 1. 4 Proces of Query Preparation

Input Sentence	Before	After
	SQL Pattern	**SQL Pattern**
Single Query	CREATE DATABASE {{nama_db}};	CREATE DATABASE unikom;
Plural Queries	ALTER TABLE {{nama_tabel}} ADD {{kolom_tipe_opsi}};	ALTER TABLE user ADD tgllahir date;
	ALTER TABLE {{nama_tabel}} CHANGE {{nama_kolom}} {{nama_kolom_baru}};	ALTER TABLE mahasiswa CHANGE alamat kota;

2.8 Test Result

The accuracy testing phase in this study was conducted to measure the level of success in the process of translating input sentences into SQL DDL queries. The DDL queries tested include the CREATE, ALTER, RENAME, TRUNCATE, and DROP functions. Input sentence data which is one command sentence in Indonesian containing two DDL queries. The test is done by translating input sentences and the results of the query from the translation process by the system will be compared to querying the predetermined expectations. One example of the input sentence used in the test is in the following table.

Table 1. 5 Sample Test Sentence

Test Data
Buatlah database dengan nama db_kampus dan buat tabel mahasiswa dengan field nim tipe data char (7), nama tipe data varchar (50), alamat tipe data varchar (100), tahun_msk tipe data integer

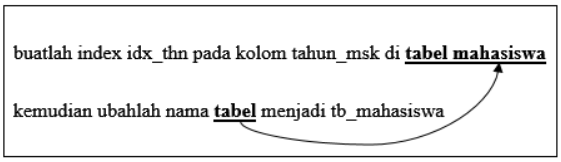
the results are true if the query generated by the system is the same as querying the specified expectations. Accuracy testing includes 25 input sentences with 93 command sentences which are a combination of each DDL function.

Table 1. 6 Accuracy Testing Results

No	Command Type	Sentence	Clarification Result	
			True	False
1	CREATE-CREATE	9	8	1
2	CREATE-ALTER	4	2	2
3	CREATE-RENAME	4	3	1
4	CREATE-TRUNCATE	4	2	2
5	CREATE-DROP	4	4	0
6	ALTER-CREATE	4	3	1
7	ALTER-ALTER	4	3	1
8	ALTER-RENAME	3	2	1
9	ALTER-TRUNCATE	3	2	1
10	ALTER-DROP	3	3	0
11	RENAME-CREATE	3	2	1
12	RENAME-ALTER	4	3	1
13	RENAME-RENAME	3	2	1
14	RENAME-TRUNCATE	3	1	2
15	RENAME-DROP	3	2	1
16	TRUNCATE-CREATE	3	2	1
17	TRUNCATE-ALTER	3	1	2
18	TRUNCATE-RENAME	3	2	1
19	TRUNCATE-TRUNCATE	3	3	0
20	TRUNCATE-DROP	3	2	1
21	DROP-CREATE	3	2	1
22	DROP-ALTER	5	4	1
23	DROP-RENAME	3	2	1
24	DROP-TRUNCATE	5	3	2
25	DROP-DROP	4	4	0
Total		93	67	26
Average Accuracy = 72,04%				

Based on the results of black box testing and accuracy of the system that has been built, the conclusions obtained are as follows.

1. System functionality works well, because it is marked by the success of every function in the system.
2. The average accuracy obtained was 72.04%. The low average accuracy value is caused because the system cannot handle sentences that have words that refer to the words in the previous clause. Examples of sentences can be seen in the picture.



Picture 1. 5 Examples of Sentences Containing Reference Words

In the example in Figure 1.5, the table name 'mahasiswa' is only mentioned once, so the table name in the second clause does not specify because the word table in the second clause refers to the table name in the first clause.

3. Closing

3.1 Conclusion

Based on the results of the implementation and testing of the research that has been done, it can be obtained information that the system built can translate DDL command sentences in Indonesian containing multiple queries and generate more than one DDL query. This research succeeded in translating command sentences containing plural DDL queries in Indonesian to DDL queries with 72.04% accuracy.

3.2 Suggestion

Based on the analysis and testing that has been carried out, there is a suggestion that this research will get better results, including the following.

1. Adding co-references in knowing pronouns for words that refer to the previous clause.
2. Add typo-checking word correction features.
3. Adding rules that can read sentences that produce queries with null or not null, foreign key, and primary key format attributes.

BIBLIOGRAPHY

[1] Fathansyah, *Basis Data*. Bandung: Informatika, 2004.

[2] S. Phadtare, "SQL Query Generation Using Human Language Processing," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 4, no. 4, hal. 696–698, 2016.

[3] S. Kaur dan R. S. Bali, "SQL generation and execution from natural language processing," *Int. J. Comput. Bus. Res. ISSN*, no. 1972, hal. 2229–6166, 2012.

[4] S. Wibisono, "Aplikasi Pengolah Bahasa Alami untuk Query Basisdata Akademik dengan Format Data Xml," *J. Teknol. Inf. Din.*, vol. 18, no. 1, hal. 65–79, 2013.

[5] Kuspriyanto, H. Sujani, H. Tjahjono, dan S. Kusuma, "Perancangan Translator Bahasa Alami ke dalam Format SQL (Structured Query Language)," *Teknoin*, vol. 10, no. 3, hal. 225–236, 2005.

[6] D. M. Aminuddin, "Data Definition Language (DDL) Pada Structured Query Language (SQL) Menggunakan Bahasa Indonesia," Universitas Komputer Indonesia, 2018.

[7] D. Anita dan M. Arhami, *Konsep Kecerdasan Buatan*. Yogyakarta: Andi Offset.

[8] K. Miftahul dan S. Ridwan, *SINTAKSIS Memahami Satuan Kalimat Perspektif Fungsi*. Jakarta: Bumi Aksara, 2014.

[9] Sugiyono, *Metode Penelitian Kuantitatif*. Bandung: Alfabeta, 2014.

[10] R. S. Pressman, *Software Engineering A Practitioner's Approach 7th Ed*. New York: McGraw-Hill, 2010.

[11] R. H. Fauzi, "Pembangunan Kamus Bahasa Indonesia sebagai Sumber Daya Natural Language Processing," *Pros. Semin. Nas. Komput. dan Inform.*, vol. 2017, hal. 193–200, 2017.

[12] M. Adriani, J. Asian, B. Nazief, dan E. Williams, "Stemming Indonesian : A Confi x-Stripping Approach," *ACM Trans. Asian Lang. Inf. Process*, vol. 6, no. 4, hal. 1–33, 2007.