Answering Comparison in Indonesian Question Answering System with Database

Athia Saelan, Ayu Purwarianti, and Dwi Hendratmo Widyantoro

School of Electrical Engineering and Informatics,
Institut Teknologi Bandung, INDONESIA

Abstract: In this research, we build a question answering system that can answer comparative questions. Our system contains two components, question analysis and answer processing. In question analysis, we use information extraction method to extract entities, aspects, relations, and constants from the question. We also classify the question into five question types such as entity-mentioned, entity-other, entity-all, aspect, and yes/no. These processes in question analysis component are solved by machine learning technique. In answer processing component, we process the comparison by using word lists and rules. The result of comparison processing is used to find the answer by generating query to get relevant data from the database. The query generation process is performed by using rules based on the question type. After that, the data from database is also processed based on question type to generate the answer. Based on the experiment results, our proposed method for comparison question answering system is promising.

Keywords: question answering, comparison, comparative question, information extraction, database

1. Introduction

Comparison is an activity that human usually do before making decision. For example, if they want to decide which product to buy, they do some comparison between those products. This comparison process is not really a trivial task. This process includes searching for information of each product, comparing each aspect of those products, and making the decision. Question answering system is a system that answers question in natural language automatically. By using this system, user could get information with less effort. This also can be applied to comparison. In this research, we build a question answering system that can answer comparative question. By using such system, we don't have to search for each product information and compare each aspect manually. This also could be used for example in a chatbot for selling products. The customer may want to compare those products and the chatbot has to answer those questions.

There are many researches in question answering field, including those for Indonesian language [1-5]. To answer a question, a question answering system can get the answer from structured source like databases [6-8], graphs or semantic web [9-12], or from unstructured source like texts [1,3,4,13,14]. One type of question answering system that uses databases is called Natural Language Interface to Databases (NLIDB) [6-8]. This system usually handles comparison. However, since it is intended as an interface to database, the questions handled by NLIDB are not really natural. In this research, we used real world questions that are meant to be asked to human, so the questions are more natural and possibly not well structured. As far as we know, there is no existing question answering system that explicitly handles comparison with more natural question, especially for Indonesian language.

There are some researches that focus on comparative entity mining for opinion mining [15-18] and for collecting comparable entities [19]. Research in [19] handles comparative question, but their task is to collect comparable entities and not to answer them. Our research aims to answer those type of questions. Those researches also limit the questions that explicitly mention two or more entities. In our research, we handle more types of question. The question could

Received: November 14th, 2017. Accepted: December 20th, 2018

DOI: 10.15676/ijeei.2018.10.4.11

contains comparison between two or more entities (e.g. “which one is better, X or Y?”), only one entity (e.g. “Which one is better than X?”), or even no entity mentioned (e.g. “What is the best one?”).

In our previous work [20], we described the question analysis process. We used information extraction and question classification techniques for the question analysis. In this paper, we will explain how we handle the comparison, and how we create queries based on information extraction and question classification result. The query will be executed against the database, and we will also explain how to create the answer based on the query result.

The data we used are collected from gadget review sites, forum, and from respondent. There are 477 comparative questions, we use 382 questions as training data and 95 for testing and evaluation. The evaluation result shows that we have reached 32.6% of correct answer in the test data.

To the best of our knowledge, this is the first study in question answering for comparative question, and also the first study in comparative question for Indonesian language. This problem is interesting, because we have to combine question answering task with comparative entity mining task. We have to classify the question based on the question type and also maintain the comparison type. Our contributions include

1. Question types and important information types for answering comparative questions.
2. Rules for dealing with comparison and adjective in a comparative question.
3. Rules for getting the answer from database.

The rest of the paper is organized as follows. Section 2 discusses our previous work on question analysis and the data we use. Section 3 describes our system, how we handle comparison, and how we get and process the answer. We evaluate the system and discuss the result in section 4, followed by conclusion in Section 5.

2. Overview on Data and Question Analysis

A. Data

We collect 477 comparative question for training and testing data, including 150 question from our previous work [20]. Some of the questions from the previous work are not used because it is still ambiguous whether the questions are comparative or not. In the previous work, we collect the data from comments of some gadget review sites and a forum. The sites are *detekno.com*, *ulasgadget.com*, *hargahpxiaomi.com*, *ulashape.com*, *begawei.com*, and *hpsaja.com*, and the forum is *kaskus.co.id*. There are 3,183 comments collected from the sites, and only 145 considered as comparative questions. So we add 5 questions from the forum because there are really few amounts of data for some question type. To collect more data for this work, we also ask some respondents to provide more comparative questions. We split the data into 382 questions for training and 95 for testing.

B. Question types

Table 1. Data statistic for each question type

No	Question Type	Amount
1	Entity-mentioned	108
2	Entity-other	89
3	Entity-all	107
4	Aspect	90
5	Yes/no	83
Total		477

We group the questions into 5 types, like the previous research, but we renamed them. These are based on the Expected Answer Type (EAT). Actually, there are 3 EAT, which are entity, aspect, and yes/no. Entity answer type can be further divided into 3 types. The first one is entity that is mentioned in the question, the second one is entity other than entities mentioned, and the

last one is any entity. So we group the questions into 5 types. The amount of each data type is shown in Table 1.

In the entity-mentioned type, the answer is entities. Possible answer for this question type is only entities mentioned in the question. For example, "*bagusan mana A atau B?*" ("which one is better, A or B?"), the possible answer is limited into A or B, not the others.

Answer for entity-other type is also entity. But in this question type, the possible answer is only entities other than entities mentioned in the question. For example, "*Smartphone apa yang kualitasnya setara dengan X?*" ("Which smartphone has similar quality with X?"). In that question, the entity "X" is not a possible answer.

The third entity answer type, entity-all type's answer is not limited, whether the entity is mentioned or not. For example, "*Uang 2 juta bisa dapat HP seperti apa?*" ("What smartphone could I buy with 2 million rupiahs?"). From the data collected, we found that all questions in this group are comparison with constants. Actually, it is possible for a question to be in superlative form. But since it is not found in our data, we didn't handle it yet.

In aspect question type, the answer is aspects. For example, "*Apa bedanya A dengan B?*" ("What is the difference between A and B?"). The answer for this question is a list of aspects that are different between A and B, and maybe we need explanation in the answer, like which one is better for each aspect mentioned.

Finally, for yes/no question, the answer is yes or no. For example, "*Apakah A lebih baik daripada B?*" ("Is A better than B?").

C. Important Information

In a question, not all words are important. Some words are important, and some others are only addition. Additionally, each information has different role in a question. In the previous work, we used 4 types of information. In this work, we renamed "comparison" into "relation" to avoid ambiguities. Beside that, we found that constants also could be used for comparison with constants. We also remove the constraint information, since we don't use the information in this work. So there are also 4 types of information. The amount and the word count for each type is shown in Table 2.

Table 2. Data statistic for each information type

No	Information Type	Amount	Word count
1	Entity	596	1831
2	Aspect	349	493
3	Relation	482	600
4	Constant	74	153
5	Other		2401
Total			5478

For explanation, we used example as follows. "*Smartphone yang kualitasnya setara dengan X dan harganya max 2,5-2,8 jutaan tipe apa ya?*" ("Which smartphone have similar quality with X, and max price 2.5-2.8 million rupiahs?"). That question includes the 4 types of important information that are explained as follows.

- Entity, that is "X"
- Aspects, that are "quality" and "price"
- Relations, that are "similar" and "max"
- Constant, that is "2.5-2.8 million"

However, a question does not necessarily contain all 4 types of information. If a question has some information missing, there are 2 possibilities, that the information are implicit, or the information are not needed in that question.

D. Data Analysis

Table 3. Data statistic for relationship between question and information types

No	Question type	Amount	Amount of question that contains information			
			Entity	Aspect	Relation	Constant
1	Entity-mentioned	108	108	15	92	0
2	Entity-other	89	89	78	85	18
3	Entity-all	107	26	96	90	45
4	Aspect	90	90	4	71	0
5	Yes/no	83	83	60	81	3

After we define the question and information types, we can analyze about what information that is usually contained in each question type. This will help us to understand what the nature of each question type is, and also to determine some things like the default value of relation in each type. Table 3 shows the summary of relationship between question and information types.

All of the entity-mentioned question types contain more than one entity. This is obvious, because the question has to mention the entity, and the answer is one of the entities mentioned. Only 15 questions in this type contain aspect. If there is no aspect mentioned, we can say that the comparison is for all aspects. There are 92 questions that mention the relation explicitly. All of the relations mentioned are "better". So we will choose this relation as default relation if there is no relation mentioned in a question in this type.

In the entity-other question type, almost all of the information are often mentioned. The entity is always mentioned, same as entity-mentioned type. In this type, the relation is also mentioned in almost all of the question. An example that does not mention relation is "*selain A yang punya kapasitas internal storage 64gb apa lagi?*" ("What phone other than A that has internal storage capacity 64GB"). Although the relation is not mentioned, we can tell that this means we have to look for phones that have internal storage capacity equal to 64GB. So we choose the default relation is equality.

For the entity-all question type, relation and aspect are usually mentioned, but sometimes they are omitted. The aspect is usually price, and it is not mentioned because it is obvious, for example, the value "2 millions" is not likely to be reached by other aspects. The entire question in this type also contains comparison with constants.

All questions in the aspect question type have entity information. Some questions only contain one entity, for example "*kelebihanya X dimana gan??*" ("What is the advantages of X"). In this question, we should compare the entity with all other entities generally. Other information that is often found is relation. There are three types of relation found, that is difference, superiority, and inferiority. But there is also possibility that we found a similarity relation in this question type.

The last one is yes or no question type. In this type, all the questions contain entities, most question also mention relation, and some questions also contain aspects. Some of them contain two entities and one relation, and some only contain one entity. There are also few question that contains constant.

E. Preliminary Results on Question Analysis

The question analysis process consists of two tasks. Question classification process is performed to get the question type or expected answer type, as stated in Section 2-B, while information extraction process is done to get important information as stated in Section 2-C. Preliminary experiments have been conducted to find out which task order scenario is suitable for question analysis, between question classification first and information extraction first. Additionally, we also attempt to find out which algorithm is the best for each task.

The experimental setting is still the same as [20], but we used the data described in Section 2-A – 2-C. From the preliminary experiment, it is found that the information extraction first

scenario is better than question classification first. For the information extraction process, we do token classification with these features: the word lexical, 2 words before and after, and the previous label. For the question classification, we use bag of word and information extraction result as features. For the algorithm, SVM is found to be better in both tasks.

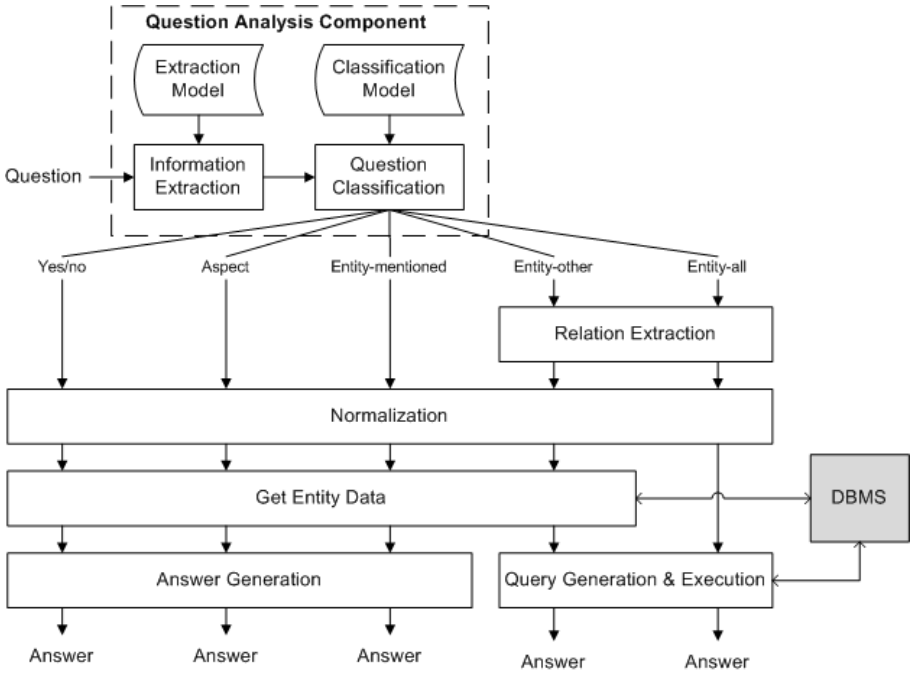

Figure 1. System Architecture

3. Question Answering System for Comparison

The architecture of the system is shown in Figure 1. The question analysis component consists of information extraction and question classification, as we discuss in the previous section, will be explained in Section 3-A. This component will classify the question into five types. Next process will depend on the question type. Before that, we will discuss about the database in Section 3-B. Next we will explain the method we use to extract relation, comparison, and aspect. The process for each question type will be explained in Section 3-D.

A. Question Analysis Process

Based on the results of preleminary experiments, we perform information extraction first, and then question classification. For both tasks, we use SVM as the classifier. Tables 4 and Table 5 show the detail features used in each process. Figure 2 shows example of question analysis process for a sentence.

Table 4. Features used in information extraction process

No	Feature	Explanation
1	word	The word lexical
2	word-1	The lexical of the previous word
3	word-2	The lexical of the two previous word
4	word+1	The lexical of the next word
5	word+2	The lexical of the two next word
6	Prevlabel	Extraction result of the previous word

Table 5. Features used in question classification process

No	Feature group	Explanation
1	Bag of words	0 if the word does not exist in the sentence, 1 if the word exists in the sentence. The word is lowercased, punctuation not included, and excluding the words that only appeared once in the dataset
2	Extraction result	The frequency of extraction result (0, 1, >1)

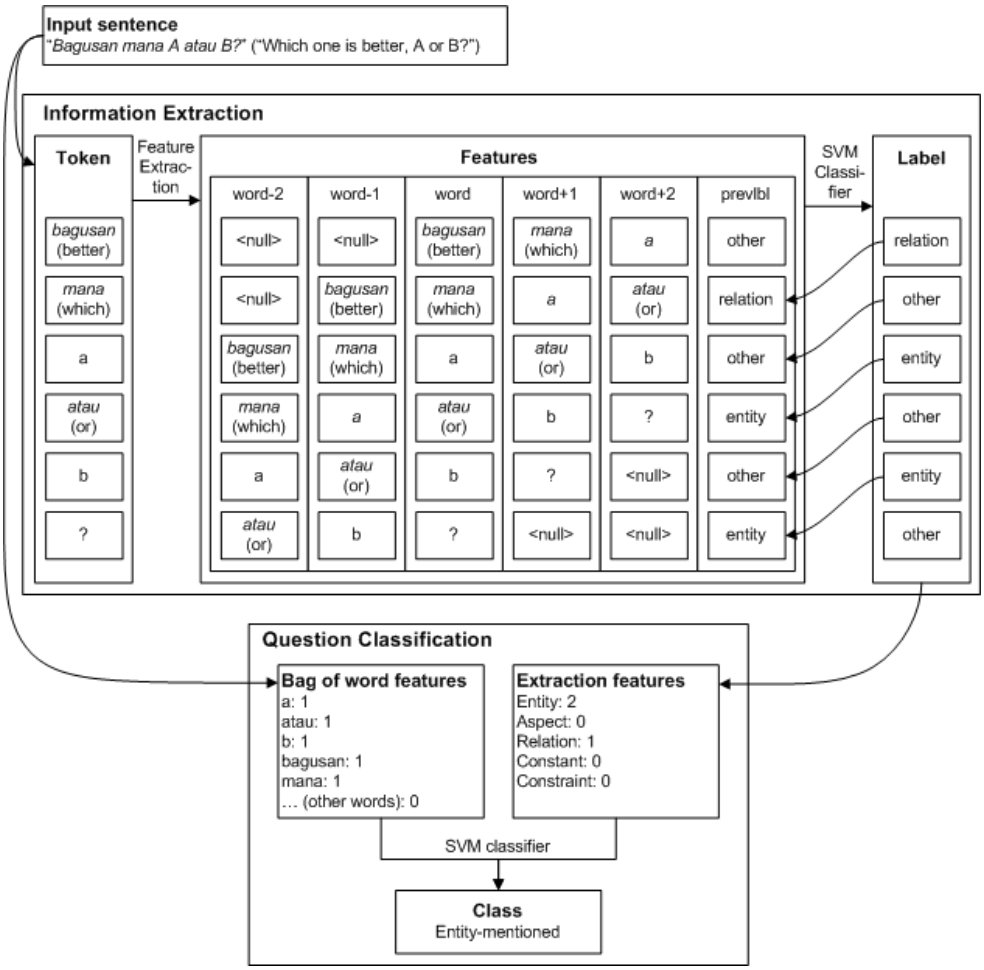

Figure 2. Example of question analysis process

B. Database Structure

In this work, we use tables as a source for answering the questions. The table contains entity name and list of aspects. We have implemented 11 aspects, including weight, battery, screen size, screen resolution, processor core, processor speed, RAM, storage, rear camera, front camera, and price. Thus, we have 12 columns in the table. For this research, we implement the table for 48 entities. Table 6 shows example of table we use as an answer source. But for doing comparison, we normalize the table in 1-10 scale.

Table 6. Example of table used as the answer source

Entity	Weight	Battery	Screen size	Screen res	Proc core	Proc speed	RAM	Storage	Rear cam	Front cam	Price
A	159	2800	5.3	2560	4	2.15	4	32	16	8	8700000
B	156	2800	5.3	2560	8	1.8	3	32	16	8	6435000
C	142	2800	5.3	1280	8	1.5	2	16	13	5	2070000
D	132	2100	4.5	854	4	1.1	1	8	5	2	1000000

C. Dealing with Comparison and Adjective

Information about comparison or relation from the extraction result is still very diverse. Besides, some questions do not mention the relation explicitly. So we need additional process to classify the relation into more formal form.

We employ word lists to classify the relation from the extraction result. We call them comparison type. There are nine comparison types, consisting of six comparatives, two adjectives, and one negation. Table 7 shows the word lists for each comparison type that we use. These lists are still limited to the words we found in the data.

Table 7. Word lists for comparison type

No	Name	Symbol	Word list	Examples in English
1	Equality	=	sama, setara, dekat, segitu, saing, sekitar, kisar	Same, similar, near, about
2	Difference	≠	Beda	Differ
3	More	>	Lebih	More
4	Less	<	Kurang	Less
5	More or equal	≥	Min	Min
6	Less or equal	≤	max, maks	Max
7	Positive	+	atas, tinggi, besar, mending, baik, bagus, rekomen, recommend, unggul, mantap, worth, menang, sip, plus, mahal	Above, high, big, good, win, ok, expensive
8	Negative	-	Bawah, rendah, kecil, lemah, jelek, buruk, minus, murah	Below, low, small, bad, cheap
9	Negation	!	tidak, gak	No, not

Comparison type – score
Equality : 0
Difference : 0
More : 0
Less : 0
More or equal : 0
Less or equal : 0
Positive : 1
Negative : 1
Negation : 0

Figure 3. Example output of comparison processing for the relation “keunggulan dan kelemahan”

We use partial matching to match each word that is classified as relation from the sentence. For each word that matches, we add score for that comparison type. So, every relation may have several comparison types with certain score. For example, the system has to process the relation “keunggulan dan kelemahan” (strength and weakness). First, for the word “keunggulan”

(strength), the word matches with the word "unggul" in the positive word list, so we have 1 score for the positive comparison type. Next the word "dan" (and) doesn't match any word in the lists. Lastly, the word "kelemahan" (weakness) matches "lemah" in the negative word list, and we add score 1 to the negative comparison type. Figure 3 illustrates the example output of comparison processing for this relation. After that, this output is processed based on the question type in the query generation process.

In case there is no matching word, the system will do some relaxations to the matching process. First, the system will remove the affixes, including prefixes "per-" and "ke-", and suffixes "-nya" and "-an", since those affixes are mostly found in the relation words. Second, the system will only match the consonant, since in Indonesian informal writing, people often abbreviate the words by omitting the vowels (e.g. "bagus" to "bgs"), or change the vowel into another vowel that sounds similar when pronounced (e.g. "mantap" to "mantep"). And third, the system allows only one difference between consonant in the relation word and the word list.

If there is still no matching word, or there is no detected relation at all, the system will use the default comparison types that are shown in Table 8. Most question types have an "equality" comparison type as a default, because it is considered fairer. The two exceptions are "entity-mentioned" and "aspect". For the type "entity-mentioned", the default is "more", since almost all questions in this type are asking for the better choices between entities mentioned, as we have discussed in Section 2-D. For the question type "aspect", the default is "difference", because most people ask the difference of aspects between entities rather than the equality.

Table 8. Default comparison type for each question type

Question type	Default comparison type
Entity-mentioned	More
Entity-other	Equality
Entity-all	Equality
Aspect	Difference
Yes/no	Equality

D. Getting and Generating Answer

After processing the comparison and adjectives, the system has to get the answer from the database. the system uses information extracted in the question analysis process. Before using them, the system will normalize extracted information. The relation has been processed in the comparison processing. For the aspect words, we remove the suffix "-nya". We also normalize the aspect using dictionary and ontology. For the constant, we remove all non-numerical parts, and get the numerical part as the value. If there are two numerical parts, we consider it as a value range. We also process the entity to match the most similar entity in the database.

After that, the system will do the process based on the question type. There are 5 types of information. In the following we will discuss the process for getting and processing answer for each question type.

Figure 4. Answer Processing for Entity-mentioned Question Type

1) Answer Processing for Entity-mentioned Question Type

Figure 4 depicts the process of obtaining answer for entity-mentioned question type. After performing the basic normalization, the system has to determine the comparison. In this question type, we need just one comparison, between positive (best) and negative (worst). This is determined by the result of comparison processing. If the positive score is higher than the negative one, the comparison is positive, and vice versa. If there is a negation or less comparison, the comparison would be the opposite. For example, if there is a word "good" in the question, it

is positive. But if we find the word “not good” or “less good”, it becomes negative. But if there is no comparison word found, the default is positive, because, in this case, people tend to find something best. Table 9 summarizes this rule.

Table 9. Rule for entity-mentioned question type

Condition	Comparison	Example
Positive > Negative	Positive	Bagus (good), lebih bagus (better)
Negative > Positive	Negative	Buruk (bad)
Positive + Negation/Less	Negative	Kurang bagus (less good)
Negative + Negation/Less	Positive	Tidak buruk (not bad)
No relation	Positive	

After determining the comparison, the system has to retrieve the data of all entities mentioned in the question. The system retrieves the data from the database by executing the following query. If there is no aspect mentioned, all the aspects in the table will be retrieved.

select entity_name, <list of aspects> from entity_table where entity_name in (<list of entities>)

Last, the system has to determine the selected entity. To do this, the system sums up the normalized value from all the aspects retrieved for each entity. If the comparison is positive, the entity with highest value will be selected. Otherwise, if the comparison is negative, the lowest value entity will be selected. The answer for this question type is the selected entity. It could be one entity or list of entities.

2) Answer Processing for Entity-other Question Type

Figure 5 shows the process of getting answer for entity-other question type. For this question type, first, we have to determine lists of aspect-relation-entity and aspect-relation-constant. To explain what is aspect-relation-entity and aspect-relation-constant, we will use this example. “*Smartphone yang kameranya setara dengan X dan harganya max 2.800.000 tipe apa ya?*” (“Which smartphone have similar camera with X and the max price 2,800,000 rupiahs?”). In this question, the extracted information are entity (“X”), aspects (“camera” and “price”), relations (“similar” and “max”), and constant (“2,800,000”). If we look further, there are relationships between aspect, relation, and entity or constant. For example, the aspect “camera” has “similar” relation with entity “X”. We call this “camera-similar-X” relationship aspect-relation-entity (“the camera is similar to entity X”). In addition, the aspect “price” has a relation “max” and a constant “2,800,000”. We call this “price-max-2,800,000” relationship aspect-relation-constant (“the max price is 2.8 million”).

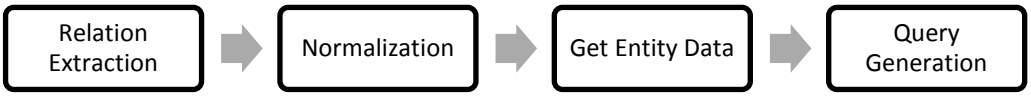

Figure 5. Answer Processing for Entity-other Question Type

To extract the relationship, we use the simple word distance method. First the system should find entities/constants. Then for each entity/constant, the system will find its relation and aspect. We simply select the nearest relation and aspect for each entity/constant. We also have to get all mentioned entities by executing the same query as in the entity-mentioned type.

select entity_name, <list of aspects> from entity_table where entity_name in (<list of entities>)

This is done to get the value of the aspects compared from each entity. So after getting the value, all data from aspect-relation-entity are moved to aspect-relation-constant by replacing the entity with the value of corresponding aspect of the entity. For example, from the aspect-relation-

entity “camera-similar-X”, we get the camera of X is 8 from the database. We can move this relationship to aspect-relation-constant “camera-similar-8”.

Finally, we have to get all the data based on aspect-relation-constant, by executing this query. The answer for the question is the result from the query execution. It could be one entity or list of entities.

select * from entity_table where <list of aspect-relation-constant>
and entity_name not in (<list of entities>)

From the previous example, we now have two aspect-relation-constant, “camera-similar-8” and “price-max-2,800,000”. According to Table 6, the “similar” relationship can be translated as “=”, and the “max” relationship can be translated as “≤”. From this, we can generate the query. This query should return entities that are suitable for answering the question.

select * from entity_table where camera = “8” and price <= “2800000” and entity_name not in (“X”)

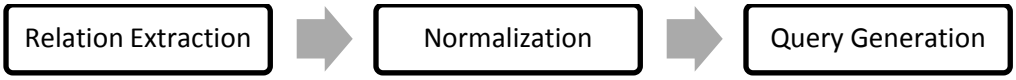

Figure 6. Answer Processing for Entity-all Question Type

3). Answer Processing for Entity-all Question Type

Figure 6 depicts the process of retrieving answer for entity-all question type. First we have to determine lists of and aspect-relation-constant, with the same procedure as in the entity-other type. Then we get all the data, by executing the same query as in the entity-other type. The answer for the question is the result from the query execution. It could be one entity or list of entities.

This question type is processed similarly to the entity-other type for the constants, because this question is really similar to the entity-other question type. The only difference is that there is no entity mentioned in this type of question. The example from entity-other type could be changed into entity-all question type by changing the entity into constant, “*Smartphone yang kameranya setara 8MP dan harganya max 2.800.000 tipe apa ya?*” (“Which smartphone have 8MP camera and the max price 2,800,000 rupiahs?”). And so the process is also similar to the entity-other question type.

Figure 7. Answer Processing for Aspect Question Type

4). Answer Processing for Aspect Question Type

Figure 7 shows the process of obtaining answer for aspect question type. For this question type, first we have to determine the comparison. For this question type, we need just one comparison, between equality, difference, positive (superiority), and negative (inferiority). We choose the comparison based on the highest score of the result of comparison processing. It is similar to the rule for entity-mentioned type. The difference is that we have four types here, and if there is no comparison word found, the default comparison is difference. There is one exception, if there are “more” or “positive” comparison, and also “less” or “negative” comparison in the same question, the comparison is difference (eg. “*kelebihan dan kekurangan*” or “strength and weakness”). Table 10 shows this rule.

Table 10. Example for aspect question type

Condition	Comparison	Example
More/positive Negation + less/negative	Positive	*Kelebihan* (strength)
Less/negative Negation + more/positive	Negative	*Kelemahan* (weakness)
Equality	Equality	*Persamaan* (similarity)
Difference	Difference	*Perbedaan, bedanya* (difference)
More/positive + less/negative	Difference	*kelebihan dan kekurangan* (strength and weakness) *Baik buruk* (good and bad)
No relation word	Difference	

After that, we have to get all mentioned entities by executing the same query as in the entity-mentioned type. The main difference is that there is a possibility of only one entity mentioned (e.g. "*apa kekurangan dari X?*" or "what is the weakness of X?"). In this case, the value will be compared to 5 as a median of range 1-10.

The final step is to generate the answer. If the comparison is positive or negative, the first mentioned entity (subject) is compared to each of the other entities (objects). If it is positive, the answer is a list of aspects that the subject has better value than the object, and vice versa. For example, "*Kelemahannya A dibanding B apa ya?*" ("What is the weakness of A compared to B?"). In this case, A is the subject, B is the object, and the relation is negative. So we should answer this question with a list of aspect that B has better value than A.

If the comparison is equality, the answer is list of aspects that have the same value for all entities mentioned. And if the comparison is difference, the answer is also list of aspect, but it also added by the information about which entity is better for each aspect. For example, "*Apa bedanya A dengan B?*" ("What is the difference between A and B?"). We could answer this question with only the list of aspect, like "weight, RAM, and price". But this is not really informative, so we add which entity is better for each aspect, for example "the weight is better in B (153), the RAM is better in A (4), and the price is better in B (6,435,000)".

5) Answer Processing for Yes/no Question Type

Figure 8. Answer Processing for yes/no Question Type

Table 11. Rule for yes/no question type

Order	Comparison found in the question	Result	Negation
1	Positive	More	Less or equal
2	Positive + Less	Less	More or equal
3	Negative	Less	More or equal
4	Negative + Less	More	Less or equal
5	Less or equal	Less or equal	More
6	More or equal	More or equal	Less
7	Difference	Difference	Equality
8	Equality or no relation	Equality	Difference

Figure 8 shows the process of obtaining answer for yes/no question type. For this question type, first step is to determine the comparison, between equality, difference, less, or more. Table 11 lists the rule. If there is a negation, the result becomes the opposite.

Next step is to get all mentioned entities by executing the same query as in the entity-mentioned type. Finally, we have to compare the entities. If the comparison is not the same as the true comparison, the answer is no. We assume that there are only two entities in the question. Based on the comparison,

- Equality/difference: comparing all the aspects from the two entities. If there is any difference, it is different.
- Less/more: Selecting the "best" entity with the method explained in "entity-mentioned".

For example, "*Apakah A sama dengan B?*" ("Is A same with B?"). The comparison is equality, so we have to compare all aspects of A and B that are mentioned in the question. But since the question doesn't mention any aspect, we have to compare all aspects. If there is any difference, the answer is no. Otherwise, the answer is yes.

Another example, "*Apakah A lebih baik daripada B?*" ("Is A better than B?"). For this question, we have to select the best entity with the entity-mentioned method. If we find that A is the best, the answer is yes. But if B is the best, the answer is no.

4. Evaluation and Discussion

For evaluation data, we use 95 questions from the testing data. We assume that the questions have the objective answers from the database. We also assume that entities and aspects mentioned in the questions hava matching ones in the database. We do not include some aspects that are based on subjective opinions, like design. Statistic of testing data used for the evaluation is shown in Table 12.

Based on 95 evaluation data, 32.6% of the questions can be answered correctly. We define correct answer as the answer that has partial match with our reference answer. For example, if the reference answer is "A, B, C", the answer "A" is considered considered correct. We also define information extraction is correct if the information extracted is enough to answer the question correctly. For example, if the word in the sentence is "screen resolution" but the extracted is only "resolution" part, it is also considered correct since the system will associate "resolution" with "screen resolution". Table 13 shows the accuracy of information extraction, question classification, and answer processing for each question type.

Table 12. Statistics of Evaluation Data

No	Question Type	Amount
1	Entity-mentioned	21
2	Entity-other	18
3	Entity-all	21
4	Aspect	18
5	Yes/no	17
Total		95

Table 13. System evaluation results.

No	Question Type	Accuracy		
		Information Extraction	Question Classification	Answer Processing
1	Entity-mentioned	5 (23.8%)	10 (47.6%)	5 (23.8%)
2	Entity-other	8 (44.4%)	11 (61.1%)	4 (22.2%)
3	Entity-all	9 (42.9%)	21 (100.0%)	4 (19.1%)
4	Aspect	14 (77.8%)	17 (94.4%)	15 (83.3%)
5	Yes/no	4 (23.5%)	6 (35.3%)	3 (17.7%)
Total		40 (42.1%)	65 (68.4%)	31 (32.6%)

For the overall system, there are 24 questions that complete all the tasks (information extraction, question classification and answer processing) correctly. There are 7 questions that can be answered correctly, although the result of information extraction is incorrect. The question type classification is one of the most important part since incorrect question classification cannot provide the right answer.

The aspect type seems to be the most easy to answer question. It is answered correctly in 15 out of 18 questions. The 3 questions incorrectly answered also has wrong answer in the information extraction or classification task. It also can handle various combination in this question type, one entity or more, with various relation. However, it seems that there are no question with aspect mentioned in this question type.

Error in question classification occurs in 30 out of 95 sentences. Table 14 depicts the detail result of the question classification. The question type having the lowest recall is yes/no, which is only 6 out of 17 questions. This is because the question structure is similar to other question types such as aspect type that usually mentions the compared entities. A question that mentions more than 1 entity is usually misclassified. There is also yes/no question that contains constants, that is really rare in our dataset, that also tend to misclassified.

Table 14. Question Classification Prediction.

		Classification Results				
		Entity-mentioned	Entity-other	Entity-all	Aspect	Yes/no
Class	Entity-mentioned	10	2	5	1	3
	Entity-other		11	6	1	
	Entity-all			21		
	Aspect	1			17	
	Yes/no		1	3	7	6

There are 3 questions that can be answered correctly in the yes/no type. However, 2 of the questions actually have the wrong result of information extraction. There are also 2 questions that actually give correct detail answer, but unfortunately select the wrong yes/no. It seems that it is caused by the information extraction that failed to extract the relation word.

The entity-mentioned obtains 10 out of 21 questions classified correctly. We conjecture that this could be caused by 2 factors. The first possibility is because the information extraction result is wrong, since we also have lowest performance for the information extraction in this question type. This will affect the question classification performance. The second possibility is because most of the question in this type is somewhat subjective. The typical question is, “which one is better?” that really depends on personal choice. For the objectivity of our evaluation, we choose some question that are objective, or even we add some aspect to the question to make it objective, for example “Which one has better camera”. This could make the evaluation data different from the question in the training data. This also can be the reason the information extraction performance is pretty low in this question type.

Another interesting thing is in the entity-all question type. The question classification accuracy reaches 100%, but the answer accuracy is pretty low. This is because we haven’t handle the entity-all question type that has relation with no constant, for example “what phone has good camera?” instead of “what phone has camera more than 8MP”. It turns out that 13 out of 21 questions in the evaluation data have relation with no constant. But 1 question is answered correctly because it also has comparison with constant.

5. Conclusions

In this research, we built a question answering system for answering questions containing comparison. The questions are grouped into 5 types based on expected answer type, that are entity-mentioned, entity-other, entity-all, aspect, and yes/no. Information to be extracted from a question are entity, relation, aspect, and constant. Question classification and information extraction tasks are performed by statistical method. Based on experiment, extraction first is better than classification first in the order of task execution. We process the comparison using word lists and rules. For the answer processing, we use rules based on the question type and information contained in the question. In this work, we still make assumption on what is better for each aspect. But for some aspects, different person may have different preference, so we may need some personalization here. A limitation of this research is that we still only handle quantitative type of comparison. We do not handle anything other than comparison. For the future work, we will consider to use data from other domains. It is also considerable to add real world knowledge and personalization, and to implement system with more unstructured answer source.

6. Acknowledgement

This research is partially supported by "*Penelitian Pendidikan Magister menuju Doktor untuk Sarjana Unggul*" (Master's Program toward Doctoral Degree for Excellent Graduate) from Kemenristekdikti (Indonesian ministry of research, technology, and higher education) within research "*Pengembangan Chatbot Berbahasa Indonesia*" (Indonesian Chatbot Development).

7. References

[1]. A. Purwarianti, M. Tsuchiya, and S. Nakagawa, “A Machine Learning Approach for Indonesian Question Answering System,” *Artificial Intelligence and Application, 2007*.

[2]. R. Mahendra, S. D. Larasati, and R. Manurung, “Extending an Indonesian Semantic Analysis-based Question Answering System with Linguistic and World Knowledge Axioms,” *22nd Pacific Asia Conference on Language, Information and Computation (PACLIC), pp. 262-271, Cebu, Philippines, 2008*.

[3]. M. I. Faruqi and A. Purwarianti, “An Indonesioan Question Analyzer to Enhance Performance of Indonesian-English CLQA,” *International Conference on Electrical Engineering and Informatics, Bandung, Indonesia, 2011.*

[4]. A.A. Zulen and A. Purwarianti, “Using Phrase-Based Approach in Machine Learning Based Factoid Indonesian Question Answering,” *in Proc. The 6th Conference of Indonesian Students Association in South Korea (CISAK), Daejeon, Korea,2013.*

[5]. S. Basuki and A. Purwarianti, “Statistical-based Approach for Indonesian Complex Factoid Question Decomposition,” *International Journal on Electrical Engineering and Informatics, vol. 8(2), pp. 356, 2016.*

[6]. A. M. Popescu, O. Etzioni, and H. Kautz, “Towards a theory of natural language interfaces to databases,” *In Proc. 8th international conference on Intelligent user interfaces, pp. 149-157, 2004*.

[7]. F. Li and H. V. Jagadish. "NaLIR: an interactive natural language interface for querying relational databases." *In Proceedings of the 2014 ACM SIGMOD international conference on Management of data, pp. 709-712. ACM, 2014.*

[8]. D. Saha, A. Floratou, K. Sankaranarayanan, U. F. Minhas, A. R. Mittal, and F. Özcan. "Athena: An ontology-driven system for natural language querying over relational data stores." *Proceedings of the VLDB Endowment 9, no. 12, pp. 1209-1220, 2016.*

[9]. J. D. Kim and K. B. Cohen. "Natural language query processing for SPARQL generation: A prototype system for SNOMED CT." *In Proceedings of biolink, pp. 32-38. 2013.*

[10]. X. Yao and B. Van Durme, “Information extraction over structured data: Question answering with freebase”, *In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp. 956–966, 2014.*

[11]. W. T. Yih, M. W. Chang, X. He, and J. Gao. "Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base." *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), vol. 1, pp. 1321-1331. 2015.*

[12]. M. Tatu, S. Werner, M. Balakrishna, T. Erekhinskaya, D. Moldovan, "Semantic question answering on big data", *In Proceedings of the International Workshop on Semantic Big Data, p. 10, ACM, 2016.*

[13]. N. Schlaefer, J. Ko, J. Betteridge, M. A. Pathak, E. Nyberg, and G. Sautter. "Semantic Extensions of the Ephyra QA System for TREC 2007", *In TREC, vol. 1, no. 1, p. 2. 2007.*

[14]. M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher, and H. Daumé III. "A Neural Network for Factoid Question Answering over Paragraphs", *In Empirical Methods on Natural Language Processing (EMNLP), pp. 633-644, 2014.*

[15]. N. Jindal, and B. Liu, "Identifying comparative sentences in text documents", *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 244-251, ACM, 2006.*

[16]. N. Jindal, and B. Liu, "Mining comparative sentences and relations", *Association for the Advancement of Artificial Intelligence (AAAI), vol. 22, pp. 1331-1336, 2006.*

[17]. M. Ganapathibhotla and Bing Liu, "Mining opinions in comparative sentences", *In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, pp. 241-248, Association for Computational Linguistics, 2008.*

[18]. S. Yang and Y. Ko, "Extracting comparative entities and predicates from texts using comparative type classification," *In Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, pp. 1636-1644, 2011.*

[19]. S. Li, C. Y. Lin, Y. I. Song, and Z. Li, "Comparable entity mining from comparative questions", *IEEE Transactions On Knowledge And Data Engineering, vol. 25(7), pp. 1498-1509, 2013.*

[20]. A. Saelan, A. Purwarianti, and D. H. Widyantoro, "Question analysis for Indonesian comparative question", *In Proceedings of the 1st International Conference on Computing and Applied Informatics (ICCAI), 2016.*

Athia Saelan is currently a doctoral student in Institut Teknologi Bandung. She was graduated from her bachelor and master degrees at Informatics Program, Institut Teknologi Bandung on 2012 and 2017. Her research interest include dialogue system and natural language processing.

Ayu Purwarianti is a lecturer at Institut Teknologi Bandung, Indonesia since 2008. She was graduated from Institut Teknologi Bandung for her undergraduate and master degree. She received his doctoral degree at Toyohashi University of Technology in December 2007 with research topics of cross language question answering. She has interest in computational linguistics, especially for Indonesian language. She has written several publications in conferences and journals related with computational linguistics for Indonesian language. She also provides Indonesian natural language processing tools to be used by other researchers.

Dwi Hendratmo Widyantoro is currently the Vice Dean of Academic Affairs at the School of Electrical Engineering & Informatics, Institute of Technology Bandung. He also serves as Chief Editor in the Journal of ICT Research and Application as well as Associate Editor in the International Journal of Electrical Engineering and Informatics. He received the MS and PhD degrees in computer science from Texas A&M University in 1999 and 2003, respectively. His research interests include machine learning, information summarization, information extraction, information classification as well as pattern recognition.