Study and Implementation of Monolingual Approach on Indonesian Question Answering for Factoid and Non-Factoid Question *

Alvin Andhika Zulen[a] and Ayu Purwarianti[b]

School of Electrical Engineering and Informatics, Bandung Institute of Technology,
Ganesha 10, Bandung 40135, West Java, Indonesia
[a] alvin.andhika@gmail.com
[b] ayu@stei.itb.ac.id

Abstract. We developed an open domain QA system that can handle factoid and non-factoid questions in Indonesian language by using monolingual approaches. EAT classification is done by identifying question word and clue words. Keyword extraction from question is done by looking at POS information of each word in question, eliminating stop words, and stemming. We use articles from Indonesian Wikipedia as corpus and Lucene framework as the base for passage retriever component, with three additional processing: query expansion, boost EAT, and boost term. For factoid questions, answer finding is done by using Named Entity Recognition. Answer scoring is done by calculating keyword occurrences and answer-keywords distance (MRR = 0.6191). For non-factoid questions, answer finding is done by identifying sentence pattern and clue words. Answer scoring is done by considering pattern priority and keyword occurrences (MRR = 0.8079).

Keywords: monolingual, open-domain, Indonesian language, factoid, non-factoid.

1 Introduction

Question Answering (QA) is a task in Natural Language Processing (NLP) that will automatically provide answers to questions posed in natural language. QA system can use a database or document collection (local or web) as the sources of the answer.

A QA system usually consists of three main components (Harabagiu *et al.*, 2000): question analyzer, passage retriever, and answer finder. Question analyzer component aims to classify the question according to the Expected Answer Type (EAT) as well as extract the keywords in question. These keywords will be used as input query in passage retriever component to get candidate documents/paragraphs that contain the answer. Answer finder component searches for candidate answers from documents/paragraphs that have been found previously. Each candidate answer will be given a score based on its compliance with the question. Some candidate answers will be selected as the best answers to the question. Each component can use various methods based on the language, question domain, question type, and available tools.

QA system for Indonesian language that has been built: handle factoid questions only (Purwarianti *et al.*, 2007; Wisudawan, 2010), handle non-factoid questions only (Yusliani, 2010), the domain of questions is limited (Mahendra *et al.*, 2008), cross lingual and use NLP tools for English (Wijono *et al.*, 2006; Wisudawan, 2010). There are also researches in NLP for Indonesian language that have been done, including parser, stemmer (Adriani *et al.*, 2007), Part-of-Speech (POS) tagger (Wicaksono and Purwarianti, 2010), and Named-Entity (NE) Tagger (Budi *et al.*, 2005).

From these studies, we obtained some things that can be explored further and improved to produce a QA system for Indonesian language using monolingual approaches with wider scope

---

of questions that can be handled. This QA system is expected to handle factoid (Person, Organization, Location, Datetime, Quantity) and non-factoid questions (Definition, Reason, Method). Moreover, the approach to be used in this QA system is monolingual approach, with expectation to produce a QA system with better accuracy.

The rest of this paper is organized into discussion of methods that are used in each component (question analyzer, passage retriever, answer finder), experiments, and conclusions.

2   Question Analyzer

2.1   Question EAT Classification

There are several methods that commonly used in question analyzer component: question pattern matching method (Fukumoto *et al.*, 2007; Ren *et al.*, 2008; Zhang *et al.*, 2010) machine learning method (Purwarianti *et al.*, 2007), and semantic analysis method (Mahendra *et al.*, 2008). From these alternatives, this QA system uses question pattern matching method, because it doesn't need any additional tools and resources, easy to implement, and is expected to classify questions with high accuracy.

EAT classification is done by using rules which consider question words and clue words. We use clue words in the rules because question word is not enough to determine the EAT of a question. A question word can be used to ask questions with different answer types, such as "apa" (*what*) and "berapa" (*how*). Rules that we use in are listed in Table 1.

Table 1: EAT Classification Rules

EAT	Question Words	Clue Words
Factoid		
Person	Siapa , Siapakah	-
Location	Dimana , Dimanakah	-
	Kemana , Kemanakah	-
	Darimana , Darimanakah	-
Date/Time	Kapan , Kapankah	-
	Berapa , Berapakah	tanggal, bulan, tahun, abad, jam, menit, detik
Organization	Apa , Apakah	organisasi, perusahaan, badan, institusi, lembaga, partai, komisi, sekolah, komite, universitas
Quantity	Berapa , Berapakah	-
Non-factoid		
Definition	Apa , Apakah	definisi, yang dimaksud, pengertian, arti
Reason	Mengapa , Kenapa	-
	Apa , Apakah	penyebab, menyebabkan
Method	Bagaimana , Bagaimanakah	-
		-

2.2   Keyword Extraction

Keywords are words that can be used to describe the content of a question. Keywords can be the words in the question or other words related to the words in the question. Keywords will be used to retrieve documents and paragraphs that are estimated to contain answer to the question. In other words, wrong keywords can result in not retrieving document that contains the answer.

Keywords extraction is done by looking at POS information of each word in the question and taking words with certain POS tags as keywords (Purwarianti *et al.*, 2007). For Indonesian language, POS tagger that we used is IPOSTagger (Wicaksono and Purwarianti, 2010). POS tags that will be taken as keywords can be seen in Table 2. In the process of extracting the keywords, we also remove the stop words in the question. Stop word list that is used in this system obtained from Wibisono (2007) that has been modified by removing some of the words that should not be eliminated. Stemming process is also done to get the query in the basic word form. We use Nazief-Adriani algorithm (Adriani *et al.*, 2007) as stemming algorithm.

Table 2: Keyword POS

POS	POS Name	POS	POS Name
NN	*Common Noun*	CDO	*Ordinal Numerals*
NNP	*Proper Noun*	CDC	*Collective Numerals*
NNG	*Genitive Noun*	CDP	*Primary Numerals*
VBI	*Intransitive Verb*	JJ	*Adjective*
VBT	*Transitive Verb*	FW	*Foreign Words*

3 Passage Retriever

3.1 Corpus

Corpus in the QA system is used as a source for searching answer to the question given. The corpus must be documents in Indonesian language and should not include only a single topic/domain. There are two main alternatives of Indonesian language corpus that can be used in terms of accessing methods:

1. Offline corpus can be a collection of Indonesian articles, e.g. articles from electronic media, electronic books, and database dump of Indonesian Wikipedia articles that can be obtained at Wikimedia (2011). This corpus can be stored in text files or in databases.
2. Online corpus can be a collection of articles on the web. We must consider how to access the articles and parts of the article in using this corpus.

From these alternatives, we use offline corpus obtained from Indonesian Wikipedia, because:

- Ease of access compared to online corpus.
- More comprehensive.
- Integrated in one file, which is more efficient than storing each article in a text file.
- Java Wikipedia Library (JWPL) (Zesch *et al.*, 2011) has provided mechanisms in processing the Wikipedia dump, accessing the data that has been stored in the database, and parsing the articles in Wikipedia format into plain text format. However, there are still some weaknesses in the parser, which does not remove tags that should be removed and remove parts that should not be removed. Therefore, we made a preprocessing parser to handle the lack of JWPL parser.

3.2 Searching Technique

Searching documents and paragraphs that are considered relevant from the corpus is done through searching keywords generated from question analyzer. Searching is performed by finding documents from corpus that contain the keywords and then look for paragraphs from those documents that contain the keywords, which will be used as input to answer finder.

There are three alternatives that can be used to implement passage retriever: implementing IR systems from scratch; using search engine APIs, such as Yahoo and Google; or using IR system framework, such as Lucene and Lemur. From these alternatives, this QA system uses Lucene framework in passage retriever component, because:

- Not feasible to implement component from scratch that can handle about 150,000 documents, in terms of storage and time.
- When utilizing the search engine API, what we can do is just search for documents. Paragraph searching cannot be done. We also can't use specific document processing for Indonesian language, such as stemming and stop word elimination.

Passage retriever component is implemented by modifying Lucene framework. We applied lowercasing, symbol removal, stop word elimination, and stemming in processing of each token in the documents. We also made some modifications in passage retriever component to see the effect of these additional methods to the accuracy of the system:

1. Query expansion by adding expansions of abbreviated words, which are obtained from **Kateglo** (Lanin and Hardiyanto, 2011) and **Kamus Besar Bahasa Indonesia** (KBBI).

2. Boost paragraphs that contain the EAT of the question, with boost factor of 2.0. These paragraphs would be at a higher rank than the paragraphs that do not contain the EAT. EAT checking in paragraph is done by NE searching for factoid question and clue words searching for non-factoid question.
3. Boost term that has POS: **NN**, **NNP**, **NNG**, **VBT**, and **VBI**, with boost factor of 1.5. Verb and noun words tend to be more important as keywords than other types of words. Documents that contain these types of words will be considered more relevant.

4 Answer Finder

4.1 Factoid Question

For factoid questions, the methods that can be used include machine learning (Purwarianti *et al.*, 2007) and Named Entity Recognition (NER) (Zhang *et al.*, 2010). In machine learning methods, answer finder component uses machine learning algorithms to classify each word in the document if it is part of the answer or not. Attributes that can be used in classification such as keyword occurrences, bi-gram frequency, EAT, and POS. This method requires training data for the learning process. In NER methods, NER is used to find candidate answers by extracting NEs of the documents or paragraphs to get the candidate answers which NEs are appropriate with the EAT of the question. This method requires NE tagger tool for Indonesian language.

In this QA system, we use NER method to answer factoid question. This method was chosen because it is one of basic monolingual method to seek answers of factoid questions and has never been used in previous works. Factoid question EATs that can be handled are Person, Organization, Location, Datetime, and Quantity, which can be extracted using NE tagger.

To use this approach, we need Named Entity Tagger for Indonesian language. In this work, we implemented the NE tagger for Indonesian language by using a modified approach of Budi *et al.* (2005), with addition of feature details and NE classification rules. To extract NEs in a sentence, we identify word (token) feature, literal type feature, contextual feature, morphological feature, and POS feature. From these features, we make several rules that will classify each word in the sentence according to its NE (see example below). Sequential words with the same NE will be considered as one entity.

```
IF Token[i].Kind == "WORD" &&
   Token[i].Contextual == "Person Prefix" &&
   Token[i+1].Kind == "WORD" &&
   Token[i+1].Morphological == "TitleCase"
THEN Token[i+1].NE = "PERSON"
```

Each candidate answer for factoid questions is a NE. Steps performed on the answer finder to find candidate answers of factoid questions are:
1. Each paragraph from passage retriever component is separated into sentences.
2. Count the occurrence of keywords (stemmed and non-stemmed) in each sentence.
3. Perform NE tagging to each sentence.
4. For each sentence, take all NEs that appropriate with the EAT as candidate answers. Candidate answers that only contain keywords from question will not be included.
5. For each candidate answer, calculate the distance between the candidate answer to all the keywords from the question in sentence answer. Distance is calculated by counting the number of words between the candidate answer and the keyword.
6. Sort the candidate answers based on the number of keyword occurrences in the sentence and the distance between the candidate answers with the keywords in the sentence.

4.2 Non-Factoid Question

For non-factoid questions, the methods that can be used include pattern matching (Ren *et al.*, 2008; Fukumoto *et al.*, 2007; Yusliani, 2010; Zhang *et al.*, 2010) and semantic analysis (Niu,

2007). In pattern matching method, answer finder component uses rules that consider surface expression (sentence patterns) and linguistic clue (clue words) from the sentence. This method is the simplest monolingual method to find answers of non-factoid question and doesn't require any additional tools and resources. In semantic analysis method, questions and documents are represented in the semantic representation. Answer is obtained by uniting the representation of question with the known facts. This method requires a semantic parser for Indonesian to produce a semantic representation of Indonesian sentences. Until now, semantic parser for Indonesian has not publicly available.

In this QA system, we use pattern matching with surface expression and linguistic clue to answer non-factoid questions. Sentence answer for each category of questions has similar pattern to each other. The difference lies in the clue words that mark the answer for each question category that usually appear in the sentence answers of each question category.

We defined sentence patterns that covered all possibilities of keyword and clue word occurrences in the sentence (Table 3). List of clue words that is used can be seen in Table 4. Each pattern has a priority value. If a sentence has higher priority, the more likely it will be the answer of the question. Sentence patterns and clue words on this QA system is obtained from Yusliani (2010) with some modifications. Pattern 10-11 are additional patterns that are used to handle sentence answers that don't contain clue words.

Table 3: Answer Sentence Patterns for Non-Factoid Question

Priority	Patterns
1	All *non-stemmed* keywords + clue word + ...
1	... + clue word + all *non-stemmed* keywords
1	Sentence with all *non-stemmed* keywords. Sentence with clue word + ...
2	One or more *non-stemmed* keywords + clue word + ...
2	... + clue word + one or more *non-stemmed* keywords
2	Sentence with one or more *non-stemmed* keywords. Sentence with clue word + ...
3	One or more *stemmed* keywords + clue word + …
3	... + clue word + one or more *stemmed* keywords
3	Sentence with one or more *stemmed* keywords. Sentence with clue word + ...
3	Sentence with all *non-stemmed* keywords.
4	Sentence with one or more *non-stemmed* keywords.
0	Others

Table 4: Clue Words in Answer Sentences for Non-Factoid Question

Category	Before Keywords	After Keywords
Definition	disebut, dikenal, dinamakan, diistilahkan	adalah, bermakna, ialah, diartikan, berarti, memiliki arti, merupakan
Reason	karena itu, oleh karena itu, oleh sebab itu, maka, itulah sebabnya, mengapa, sehingga, memungkinkan, menyebabkan, dengan demikian, mengakibatkan, penyebab,	karena, bertujuan, dikarenakan, agar disebabkan, sebab, akibat, kemudian
Method	cara, langkah, proses, untuk, prosedur, tahapan, tahap	dengan, melalui, pertama, dimulai, diawali, sebelum, setelah, kemudian

Each candidate answer for non-factoid questions is a complete paragraph. Steps performed on the answer finder to find candidate answers of non-factoid questions are:
1. Each paragraph from passage retriever component is separated into sentences.
2. Check the sentence pattern that is used in each sentence and count the occurrence of the keywords in that sentence.
3. For each paragraph, find the sentence with the highest priority and largest occurrence of keywords. This sentence is chosen as candidate answer for that paragraph.
4. Sort the candidate answers based on the priority value and the number of keyword occurrences in the sentence.

5   Experiments

5.1   Experimental Data

In this experiment, we used 169 questions obtained from respondents. Testing for this system is divided into four main scenarios with black box method: (1) *without additional searching methods (Baseline)*; (2) *with **Query Expansion** method*; (3) *with **Boost EAT** method*; and (4) *with **Boost Term** method.*

5.2   Experiment Result

Question Analyzer

Testing on question analyzer component is done by looking at accuracy in classifying the question according to its EAT and extracting keywords from the question. Accuracy of question analyzer component in EAT classification can be seen in Table 5. Based on the test results, EAT classification by identifying question words and clue words was considered quite good.

This component also succeeds in extracting keywords from questions. It can be seen from the accuracy of passage retriever component. Keyword extraction process through the stop-words elimination, stemming, and POS tagging can produce keywords that can describe the content of the question.

Table 5: EAT Classification Accuracy

EAT	Total Questions	Accuracy
Person	21	100 %
Organization	21	100 %
Location	21	100 %
Datetime	25	100 %
Quantity	20	100 %
Definition	23	100 %
Reason	18	100 %
Method	20	100 %

Passage Retriever

Testing on passage retriever component is done by seeing whether the paragraph that contains the answers is retrieved or not. Tests conducted on all four test scenarios. The test results for each scenario can be seen in Table 6.

Table 6: Passage Retriever Accuracy

Scenario	Total Questions	Accuracy
Baseline	169	85,8%
Query expansion	169	88,76%
Boost EAT	169	88,17%
Boost term	169	85, 8%
Average	169	87,13%

Errors in this component occurred when the document/paragraph that contains the answers did not retrieved by the component. This problem happened because keywords that are used as query appear less in relevant document/paragraph or appear more in irrelevant documents / paragraphs. As a result, the answer was not in the top 30 documents or top 20 documents.

To help overcome these problems, we also modified the passage retriever components using three alternative additional processes: query expansion, boost EAT, and boost term. From the test results, boost term method has no effect to the accuracy of passage retriever component because most of keywords obtained from question are noun or verb. Query expansion and boost EAT methods are able to increase the number of relevant paragraphs that are retrieved by the component, because:

- Boost EAT method can improve the accuracy of passage retriever components because this method can promote the ranking of paragraphs that are predicted containing candidate answers (EAT of question). This helps paragraphs that have a low relevance score in baseline method, to be taken as a candidate paragraph if it contains the EAT.
- Query expansion method can add the expansion of a word in the query if the word is an acronym/ abbreviation. Words that are added can help the QA system to find relevant documents and paragraphs which do not contain the words in the query, but rather the expansion of the word.

Answer Finder

Test results for factoid questions can be seen in Table 7. The highest average MRR value was in Organization category (MRR = 0.7507) and the lowest was in Location category (MRR = 0.4951). Based on the MRR value, the performance of the component in answering factoid questions was quite good. NER method and scoring by calculate distance and keyword occurrences can be used as a good alternative method for answering factoid questions. Example of successfully answered factoid question can be seen in Figure 1.

Table 7: MRR Value for Factoid Question

EAT	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Person	0,6310	0,6548	0,6230	0,6310	0,63495
Organization	0,7548	0,7423	0,7548	0,7509	0,7507
Location	0,4656	0,5608	0,4884	0,4656	0,4951
Datetime	0,6244	0,7244	0,6044	0,6244	0,6444
Quantity	0,5801	0,5717	0,5496	0,5801	0,570375
Average	0,61118	0,6508	0,60404	0,6104	0,619105

```
======QUESTION ANALYZER======

Question : Dimana Alexander Graham Bell dilahirkan ? (Where Alexander Graham Bell was born ?)

EAT : LOCATION

Keywords : Alexander, Graham, Bell, dilahirkan (born)


======PASSAGE RETRIEVER======

1. Document : Alexander_Graham_Bell

Paragraph : Alexander Graham Bell dilahirkan di Edinburgh, Skotlandia, Britania Raya, pada 3
Maret 1847 dan meninggal di Beinn Bhreagh, Nova Scotia, Kanada, pada 2 Agustus 1922...
(Alexander Graham Bell was born in Edinburgh, Scotland, Great Britain, on March 3, 1847 and died in
Beinn Bhreagh, Nova Scotia, Canada, on August 2, 1922...)


======ANSWER FINDER======

1. Edinburgh
```

Figure 1: Example of Successfully Answered Factoid Question

Test results for non-factoid questions can be seen in Table 8. The highest average MRR value was in Definition category (MRR = 0.902175) and the lowest was in Method category (MRR = 0.74375). Based on the MRR value, the performance of the component in answering non-factoid questions was quite good. Pattern matching method and scoring by calculate rule priority and keyword occurrences can be used as a good alternative method for answering non-factoid questions. We can see that answers for all non-factoid question categories have similar patterns and contain clue word in the sentences. Example of successfully answered non-factoid question can be seen in Figure 2.

Table 8: MRR Value for Non-Factoid Question

EAT	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Definition	0,8913	0,8913	0,9348	0,8913	0,902175

EAT	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Average
Reason	0,75	0,75	0,8611	0,75	0,777775
Method	0,75	0,725	0,75	0,75	0,74375
Average	0,7971	0,7887667	0,8486333	0,7971	0,8079

```
======QUESTION ANALYZER======
```
Question : Apa yang dimaksud dengan hepatitis akut ? (*What is meant by acute hepatitis ?*)

EAT : DEFINITION

Keywords : hepatitis (*hepatitis*), akut (*acute*)

```
======PASSAGE RETRIEVER======
```
1. Document : Hepatitis

Paragraph : Hepatitis adalah peradangan pada hati karena toxin, seperti kimia atau obat ataupun agen penyebab infeksi. Hepatitis yang berlangsung kurang dari 6 bulan disebut "hepatitis akut"... (*Hepatitis is an inflammation of the liver due to toxins, such as chemicals or drugs or agents that cause infection. Hepatitis that lasts less than 6 months is called "acute hepatitis"...*)

```
======ANSWER FINDER======
```
1. Hepatitis yang berlangsung kurang dari 6 bulan disebut "hepatitis akut". (*Hepatitis that lasts less than 6 months is called "acute hepatitis"*)

Figure 2: Example of Successfully Answered Non-Factoid Question

There are still some questions (factoid and non-factoid) that can't be answered by this QA system, either the answers don't exist in the list of answers or the answers are not on 1st rank :

- Problems on the NE tagger: error in the tagging of a word/word group (Figure 3). This is due to the incompleteness of rules and attributes that are made to do the tagging. As the consequences, the answer can't be obtained, the answer is not extracted properly or appear other candidate answers that are not appropriate with the question EAT.
- The effect of the "distances" calculation in the scoring of candidate answers (Figure 4). Problems occur if the distance between candidate answer and the keywords is greater than distance between other candidates and the keywords. The ranking of the correct answer will be lower.
- The effect of the calculation of the number of keywords. Problems occur if the number of keywords in a sentence that contains the correct answer is less than in the sentence that not contains the answer. The ranking of the correct answer will be lower.
- The effect of query expansion when it is not needed can cause in retrieving other documents/paragraphs (not relevant) and the correct answer can be in lower rank.
- Documents or paragraphs are not successfully retrieved, so the correct answer can't be found (problem in passage retriever component).

```
========QUESTION ANALYZER========
```
Question : Siapa nama penemu telepon ? (*Who is the inventor of telephone ?*)

EAT : PERSON

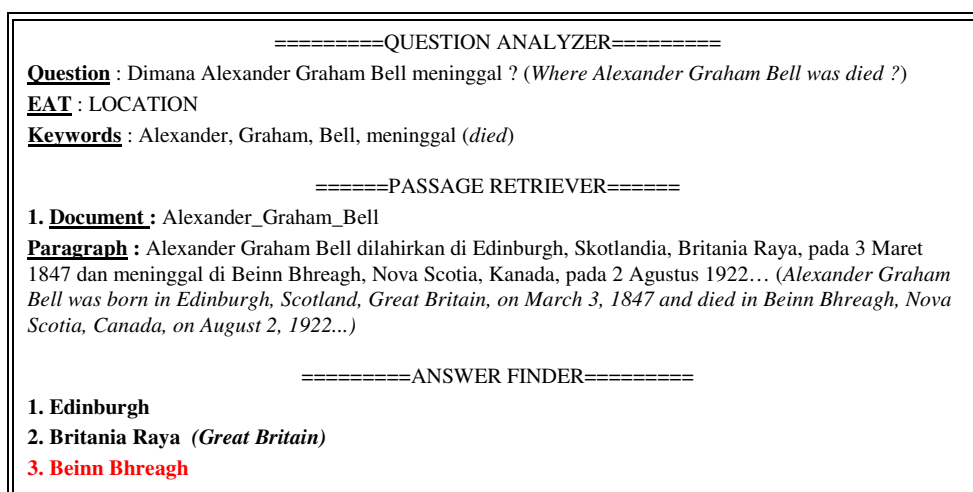**Keywords** : penemu (*inventor*), telepon (*telephone*)

```
========ANSWER FINDER=======
```
1. Umumnya (*Generally*)

Sentence : Umumnya penemu telepon yang lebih dikenal masyarakat adalah Alexander Graham Bell… (*Telephone inventor who is generally known to the public was Alexander Graham Bell*)

2. Alexander Graham Bell

Sentence : Lebih dari seabad dan di seluruh penjuru dunia, Alexander Graham Bell dikenal sebagai penemu telepon. (*More than a century and throughout the world, Alexander Graham Bell is known as the inventor of the telephone.*)

Figure 3: Sample Error Case for NE Problem

```
========QUESTION ANALYZER=========
Question : Dimana Alexander Graham Bell meninggal ? (Where Alexander Graham Bell was died ?)
EAT : LOCATION
Keywords : Alexander, Graham, Bell, meninggal (died)

======PASSAGE RETRIEVER======
1. Document : Alexander_Graham_Bell
Paragraph : Alexander Graham Bell dilahirkan di Edinburgh, Skotlandia, Britania Raya, pada 3 Maret
1847 dan meninggal di Beinn Bhreagh, Nova Scotia, Kanada, pada 2 Agustus 1922… (Alexander Graham
Bell was born in Edinburgh, Scotland, Great Britain, on March 3, 1847 and died in Beinn Bhreagh, Nova
Scotia, Canada, on August 2, 1922...)

========ANSWER FINDER=========
1. Edinburgh
2. Britania Raya  (Great Britain)
3. Beinn Bhreagh
```

Figure 4: Sample Error Case for Distance Calculation Problem

6   Conclusion

Conclusions obtained from this work as follows:

1. To achieve the criteria of open-domain QA system, we used database dump from Indonesian Wikipedia articles as corpus. The database dump needs to be processed before it can be used as a source of searching by the QA system.

2. EAT classification is done by using identification of questions words and clue words. Keyword extraction from the question is done by looking at POS information of each word in question, removing stop words, and stemming. From the test results, this system is able to classify all the questions according to their EAT and extract all of the keywords.

3. QA system is built using Apache Lucene framework as the base of passage retriever component. There are three additional processing on the passage retriever component to help the system in searching relevant documents and paragraphs: query expansion, boost EAT, and boost term. This component is quite good in retrieving relevant documents and paragraphs, with the accuracy obtained were 87.13%. Query expansion and boost EAT methods considered can help component to retrieve relevant documents and paragraphs.

4. Answer finding method for factoid question that is used in this system is NE Recognition. Scoring of candidate answers is done by calculating keyword occurrences in the sentences and distance between candidate answers and keywords. The average MRR value obtained was 0.61910.

5. Answer finding method for non-factoid question that is used in this system is pattern matching based on surface expression (sentence patterns) and linguistic clue (clue words). Scoring of candidate answers is done by considering rule priority and keyword occurrences in sentences. The average MRR value obtained was 0.8079.

6. From the test results, there are several questions that can't be answered by system, either the answer is not in the list of answers or the answer is not on 1st rank. The causes are deficiency in the NE tagger, documents/paragraphs that are not successfully retrieved, and the effect of scoring techniques.

References

Adriani, M., B. A. A. Nazief, J. Asian, H. E. Williams and S. M. M. Tahaghoghi. 2007. Stemming Indonesian: A Confix Stripping Approach. *ACM Transactions on Asian Language Information Processing (TALIP)*.

Budi, I., S. Bressan, G. Wahyudi, Z. A. Hasibuan and B. A. A. Nazief. 2005. Named Entity Recognition for Indonesian Language: Combining Contextual, Morphological, and Part-of-Speech Features into a Knowledge Engineering Approach. *Faculty of Computer Science, Universitas Indonesia.*

Fukumoto, J. 2007. Question Answering System for Non-factoid Type Questions and Automatic Evaluation based on BE Method. *Proceedings of NTCIR-6 Workshop Meeting*, Tokyo, Japan.

Harabagiu, S. M., M. A. Pasca and S. J. Maiorano. 2000. Experiments with Open-Domain Textual Question Answering. *Proceedings of the 18th International Conference on Computational Linguistics (COLING),* Saarbruken, Germany.

Lanin, I. and R. Hardiyanto. 2011. *Kateglo on Bahtera Website.* URL: http://kateglo.bahtera.org/.

Mahendra, R., S. D. Larasati and R. Manurung. 2008. Extending an Indonesian Semantic Analysis-based Question Answering System with Linguistic and World Knowledge Axioms. *Proceedings of the 22nd Pacific Asia Conference on Language, Information, and Computation*, Cebu, Philippines.

Niu, Y. 2007. *Analysis of Semantic Classes Toward Non-Factoid Question Answering*. Ph.D. thesis, Department of Computer Science, University of Toronto, Canada.

Purwarianti, A., M. Tsuchiya and S. Nakagawa. 2007. A Machine Learning Approach for Indonesian Question Answering System. *Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications*, Innsbruck, Austria.

Ren, H., D. Ji, Y. He, C. Teng and J. Wan. 2008. Multi-Strategy Question Answering System for NTCIR-7 C-C Task. *Proceedings of NTCIR-7 Workshop Meeting*, Tokyo, Japan.

Wibisono, Y. 2008. *Indonesian Stop Word List*. Universitas Pendidikan Indonesia. URL: http://fpmipa.upi.edu/staff/yudi/stop_words_list.txt.

Wicaksono, A. F. and Purwarianti, A. 2010. HMM Based Part-of-Speech Tagger for Bahasa Indonesia. *Proceedings of the 4th International MALINDO Workshop (MALINDO2010)*, Jakarta, Indonesia.

Wijono, S. H., I. Budi, L. Fitria and M. Adriani. 2006. Finding Answers to Indonesian Questions from English Documents. *CLEF 2006 Workshop*, Alicante, Spain.

Wikimedia. 2011. *Indonesian Wikipedia Dump*. URL: http://dumps.wikimedia.org/idwiki/,

Wisudawan, W. F. 2010. *Analisis Pendekatan Monolingual dan Cross Language Untuk Aplikasi Tanya Jawab Otomatis Bahasa Indonesia*. Bachelor of Engineering thesis, Informatics Engineering, School of Electrical Engineering and Informatics, ITB.

Yusliani, N. 2010. *Sistem Tanya Jawab Bahasa Indonesia untuk Non Factoid Question*. Master of Engineering thesis, Informatics Engineering, School of Electrical Engineering and Informatics, ITB.

Zesch, T, R. E. D. Castilho, E. Wolf and O. Ferschke. 2011. *Java Wikipedia Library on Google Code.* Ubiquitous Knowledge Processing (UKP) Lab., Department of Computer Science, Technische Universität Darmstadt, URL: http://code.google.com/p/jwpl/.

Zhang, G., W. Zhang, Y. Bai, S. Kang and P. Wang. 2010. An Open-domain Question Answering System for NTCIR-8 C-C Task. *Proceedings of NTCIR-8 Workshop Meeting,* Tokyo, Japan.