Geographic Information Retrieval using Query Aware Document Ranking Method Case Study for Surakarta

Viny Christanti M, Steven Tionardi, Ery Dewayani
Faculty of Information Technology, Tarumanaga University, Jl. Letjen. S. Parman no. 1, Jakarta, Indonesia
viny@untar.ac.id

Abstract—**This paper discusses the development of a Geographic Information Retrieval for Surakarta City in Indonesia. Surakarta City was chosen as the location of the place in the system because Surakarta got an award for the best tourist spot in Indonesia. In this case, Geographic Information Retrieval is a system that can handle geographic data by analyzing existing text data and generate output that can be used as decision-making on problems related to geographical. The method used in processing the information is Query Aware Document Ranking. The purpose of using this method is to provide relevant results such as output answer, answer's images and coordinates of the answer.**

Index Terms—**Geographic Information Retrieval; Language Model; Query Aware Document Ranking; Question Answering.**

I. INTRODUCTION

In daily life consciously or not, location has a very important role such as where we are, where the person is, how we achieve our destination, what can be found around, and so forth. Location also shows the address of the place where we live which each address is different from the others address and that is an important factor in human life.

People also use locations to search a person's address because the location is related to the address of a place or something or someone. Location is important to the demands of the geographic information because without a location, it is easy to confuse with the destination address. It will spend a lot of time just to find where someone or something is.

Geographic Information Retrieval (GIR) is a system that can handle geographic data by analyzing existing text data and producing outputs that can be used as decision-making on geography-related problems [1]. Geographic Information Retrieval (GIR) is different from GIS (Geographic Information System). Geographic Information System is an information system that can process data that has geographical information with the ability of computers to build, store, manage, and display geographical information that has been stored in database [2].

The method used in this Geographic Information Retrieval system is Query Aware Document Ranking. Query Aware Document Ranking works to match between documents and query. Query Aware Document Ranking method is also used because it can produce the most relevant output answers, supporting documents, and visualization on the map.

II. METHODOLOGY

This section discusses some of the methods used in designing the Geographic Information Retrieval, its function in this project, and the order of use of those methods.

A. Pre-processing

Pre-processing is the earliest step before processing text data. The main purpose of pre-processing is to facilitate the calculation of the indexing process on systems designed in the Information Retrieval so that the retrieval process is faster and more accurate. In this case, the pre-processing includes [3]:

1. Tokenization
   Tokenization function is to split sentences in the document text by removing any punctuation in the document into a word or token. Examples of tokenization are "Serabi yang terkenal ada di Solo, Jawa Tengah" to "Serabi", "yang", "terkenal", "ada", "di", "Solo", "Jawa", "Tengah".
2. Stopword removal
   Stopword removal is to remove stopwords on a document. Stopwords are words that have the most frequencies on a document or corpus so that words are considered less important.

B. N-gram

N-gram is a substring along the n character of a string. In another definition, N-gram is a chunk of the number of n characters of a string.

For example the word "TEXT" can be divided into the following N-grams ("_" represents blank) [3]:

Uni-grams: T, E, X, T
Bi-grams: _T, TE, EX, XT, T_
Tri-grams: _TE, TEX, EXT, XT_
Bi-grams skip one: TX, ET
Tri-grams two skip one: TET
Tri-grams one skip two: TXT

One of the advantages of using N-gram and not a whole word intact is that N-gram will not be too sensitive to errors written in a document [3].

C. Language Model

Language Model is a function or learning algorithm for a function that captures the prominent characteristics of the sequence distribution of words in a natural language which makes it possible to make the prediction probability of the next word given by the previous word [4]. The system designed using Language Model and ranking method used is Query-likelihood.

Query-likelihood is a ranking method in Information Retrieval where the system can search for documents with queries that are provided with effective and successful results compared to other methods. The steps to calculate Query-likelihood as follows: [5]

1. Calculate the Maximum Likelihood probability of a word (term) against all documents using the following formula:

$$p_{ml}(t|M_d) = \frac{tf_{t,d}}{dl_d} \quad (1)$$

   where:
   $tf_{t,d}$ = the frequency of words in each document
   dl_d = number of words in each document
2. Calculate the average frequency of the word t on each document using the following formula:

$$\overline{ft} = p_{avg}(t)\ x\ dl_d$$
$$p_{avg}(t) = \frac{\sum_{d(t\in d)} p_{ml}(t|M_d)}{df_t} \quad (2)$$

   where:
   df_t = number of documents containing the word t
3. Calculate the Risk Value of each word in each document using the following formula:

$$R_{t,d} = \left(\frac{1}{1+\overline{ft}}\right) X \left(\frac{\overline{ft}}{1+\overline{ft}}\right)^{tf_{t,d}} \quad (3)$$

4. Calculate the probability of word t on the document model by using the following formula:

$$p(t|M_d) = p_{ml}(t|M_d)^{(1-R_{t,d})} x\ p_{avg}(t)^{R_{t,d}} \quad (4)$$

   If there is a probability of 0, then smoothing to the value 0 is replaced by the following formula:

$$Smoot\hbar ing = \frac{cf_t}{c_s} \quad (5)$$

   where:
   c_s = total number of words/tokens
   cf_t = the number of times the word appears across the document
5. Calculate the similarity between queries and documents using the following formula:

$$p(Q|M_d) = \prod_{t\in d} p(t|M_d)\ X \prod_{t!\in d}(1 - p(t|M_d)) \quad (6)$$

The document that has the greatest similarity is the highest candidate answer because it has a high similarity so that it comes out as the output of answers.

D. Vector Space Model

Like other information retrieval system models, the vector space model uses an inverted data structure to simplify the document retrieval process. Inverted lists allow quick access to lists of documents containing a certain term along with other information, such as the weight of each term contained in each document, the position of the term within each document, and so on.

The process of measuring the similarity between the query vector with each vector document on space vector method is generally done by several stages specifically [6]:

1. Create an inverted list of each unique term in the document or query.
2. Calculate the inverse document frequency or idf value of each term using the equation:

$$idf_j = \log \frac{d}{df_j} \quad (7)$$

   where:
   d = number of all documents in the collection
   df_j = the number of documents containing term-j dfj can be called document frequency
3. Calculates the weight value of each term owned by each document by multiplying between idf and tf (term frequency) to represent the document vector. This weighting technique is called tf-idf. The value of tf-idf is obtained by using the equation:

$$w_{i,j} = tf_{ij}\ x\ idf_j \quad (8)$$

   where:
   $w_{i,j}$ = Weight of term j in the i-document
   tf_{ij} = the frequency of occurrence of term j in the i-document
4. Calculating the similarity coefficient with the following equation:

$$Sim\ (d_j, q) = \frac{d_j.q}{\|d_j\|.\|q\|} \quad (9)$$
$$d_j.q = \sum_{j=1}^{t} w_{i,j}.w_{q,j} \quad (10)$$
$$\|q\| = \sqrt{\sum_{j=1}^{t} w_{q,j}^2} \quad (11)$$

   where:
   $w_{i,j}$ = Weight of term j in i-document
   $w_{q,j}$ = The weight of the term on the query
   $sim(d_j,d_k)$ = The cosine angle or coefficient of similarity between the document vector and the query vector
   d_j = The entire term weight contained in the j-document vector
   q = The entire term weight contained in the query vector
   $\|d_j\|$ = Document vector length j
   $\|q\|$ = Query vector length

E. Query Aware Document Ranking

Query Aware Document Ranking is a method in the field of Information Retrieval. Query Aware Document Ranking is basically a development of Geographical Vector Space Model (GeoVSM) method. The Geographical Vector Space Model aims to integrate geographic models with space models based on keywords. Query Aware Document Ranking focuses on search queries against documents. Queries may appear in documents separately or queries may appear together in a document.

The Query Aware Document Ranking provides higher values for queries that appear simultaneously and Query Aware Document Ranking also assigns values to queries that appear separately using N-grams. The way to determine whether the query is specific or not, is to find the specific value of the thematic document by using Inverse Document Frequency (idf). The inverse document value of the document used in the Vector Space Model is replaced by using the inverse document Query Aware Document Ranking which is [7]:

$$Spec_t = -\log(\frac{N_t+1}{N}) \tag{12}$$

where:
N_t = number of documents containing term t
N = total number of document collections

Then, calculate the query-specific value based on the specific value of the existing document by using the following formula:

$$Spec_T = \sum_{t \in Q} w_t Spec_t \tag{13}$$

where:
w_t = weight for each term

After that, calculate the value of the specificity of the geographic document by using the formula:

$$Spec_G = -\log(\frac{Area(G_Q)}{Area(G_d)}) \tag{14}$$

where:
G_Q = the number of geographical words in the query
G_d = number of geographic documents

Then, the two specificity values from thematic documents and geographical documents are normalized using the formula:

$$w_t = 1 - \left(\frac{1}{\ln(e+Spec_T)}\right), w_g = 1 - (\frac{1}{\ln(e+Spec_g)}) \tag{15}$$

where $e = 2.71828$

After obtaining the weight from both documents, then calculate the relevance between the two documents using the following formula:

$$Rel_{(q,d)} = w_t * Rel_{T(q,d)} + w_g * Rel_{G(q,d)} \tag{16}$$

where:
$Rel_{T(q,d)}$ = the similarity value of the thematic document
$Rel_{G(q,d)}$ = the similarity value of the geographic document

F. Passage Searching

To simplify the process of searching for answers, existing documents are split into fewer sentences called passages. In the process of searching for passage, there are several steps that must be passed, that is construction a passage and passage ranking. When constructing a passage, the passage is usually broken down into several sentences per document. Next, the passage ranking is to check whether the passage contains the answer or not. If the passage does not contain an answer, then the passage will be discarded. The remaining passage is ranked based on the weight determined by the user.

In passage ranking, some examples of features that can be used are the same entity name type in the passage, the number of keywords in the passage, The longest number of words of a consecutive keyword that match the query, and the ranking of the document in which the passage will be split [8]. In this study, the construction of passages is done by using features that have been determined by the user. In addition, the answer taken is the highest ranked passage. Weights for each feature can be seen in Table 1.

Table 1
Weight for each feature

Weight	Feature
5.0	Is there any entity searched in the passage
0.5	If the entity searched in passage> 1
words in passage / total	Query keywords in passage / total words

G. Paragraph Passage

Paragraph Passage serves to obtain information that most appropriate to the query by splitting the sentences inside the document into a narrower scope called the passage. After the documents are split into several passages, they are treated as single documents or stand-alone documents. The passage is split by sentence and in this study used as many as 5 sentences per passage.

H. Answer Searching

The search for the answer from the query against the passage is done only to the highest ranked passage, so that the answering entity is most likely to match the query entered by the user. In general, the existing top passage will be split down into word-by-word form (token), except for words containing tags. Therefore, the word splitting process is based on tags and stored in an array

If the number of candidate responses in the candidate list is still the same, then the candidate of the answer to be used is the candidate of the answer with the first position to be processed.

I. Precision and Recall

Precision and Recall is a method used to evaluate the Language Model method used to work well or not. Precision and Recall measures the accuracy of the retrieved documents by using the Language Model. Here is the formula of precision [9] :

$$p = \frac{Number\ of\ documents\ that\ have\ been\ calculated\ relevant}{the\ total\ number\ of\ documents\ all\ relevant} \tag{17}$$

Recall formula:

$$r = \frac{Number\ of\ documents\ that\ have\ been\ calculated\ relevant}{Number\ of\ documents\ that\ have\ been\ count} \tag{18}$$

Both values are used to calculate average precision by taking precision values and summing up all relevant values and dividing them into many relevant documents and multiplied by one hundred percent. The value of the average precision is used to calculate the mean average precision, which is the value that becomes the base for determining the performance of the Language Model in retrieving relevant documents. The formula of the average precision used is [9]:

$$Average\ Precision = \frac{\sum_{i=1}^{n} nilai\ precision\ relevan_i}{n} . 100\% \qquad (19)$$

From the formula above, Mean Average Precision can be searched by using the following formula:

$$Mean\ Average\ Precision = \frac{\sum_{i=1}^{n} average\ precision_i}{n} . 100\% \qquad (20)$$

where n is the number of tests that are top 5 documents, so n = 5.

III. Planning and Implementation

This section discusses the plan that has been made. In this section a flowchart is presented along with an explanation of how the program works.

A. System Plan

The system designed is a web-based program that can display answers in the form of a location or place in accordance with the query that is input by the user and visualized in the form of maps. The input query is a sentence or question about a location or place that users want to search for.

This system is designed using System Development Life Cycle (SDLC). System Development Life Cycle (SDLC) is a methodology used to design, develop, maintain, and use information systems [10].

System Development Life Cycle (SDLC) has several models: waterfall, fountain, spiral, build and fix, rapid prototyping, incremental, and synchronize and stabilize. From some of the above models, the waterfall is the most famous and often used model [11]. In the design of this system, the SDLC waterfall model is used as the model. The SDLC consists of 6 stages, namely: the stage of system planning, system analysis, system design, system testing, system implementation, and system maintenance. Explanations of the first to fourth stages are described separately into sections.

B. System Planning

The first stage of SDLC is system planning. The system designed is Geographic Information Retrieval in the Indonesian language that can receive input in the form of sentences or questions that are geographical so the answer can be searched, supporting documents, and visualization on maps. Input is limited only to the question of "what", "where", and "how". The relationship between question and answer type can be seen in Table 2.

Table 2
Relation between question and answer type

Question	Answer Type
What	Object
How many / much	Distance
Where	Location

The thematic documents used in the research are articles with a total of 178 documents. The data was obtained from the previous alumnus of Dessy Yanty, a student of Tarumanagara University. Data obtained from Dessy Yanty include the thematic documents that already entities with a total of 178 documents. Other data is geographic information system data used to match answers with existing information systems [12].

C. System Analysis

In this section discusses the software and hardware used to support the process of making this system. Each software and hardware along with its usability can be seen in Table 3 and Table 4.

Table 3
Hardware used

Hardware	Function
Processor Intel(R)Core(TM) i5-2430M CPU @2.40GHz	Controls the process of running a computer system.
RAM 12 GB	For temporary data storage.
Harddisk 1TB	As a data storage and information center.
Mouse	Cursor Mover.

Table 4
Software used

Software	Function
XAMPP 7.1.1 / PHP 7.1.1	As the server of the website to be created.
Sublime Text	Editor for programming PHP, CSS, and HTML.
Google Maps API	Call the functions needed to display a geographic map.
Active Perl	Interpreter used to perform system programming.
Padre	IDE or editor to program or run Perl

D. System Design

System design stage is the third stage of SDLC. This stage explains what designs are required for the designed system. This stage is divided into several parts, namely the design process and data flow, as well as the design of the application program interface.

a. Process Design and Data Flow

The design of the process and the flow of data describe a process of the running application program and its work process. The design of the process and the flow of data on this design are explained by flowchart and State Transition Diagram (STD)

State Transition Diagram (STD) is a diagram illustrating how a process is connected to one another at a time. State Transition Diagram is described as a state in the form of a system component showing how the events are from one state to another. Flowchart is a scheme that shows the sequence of program processes designed.

The following sequence of system processes can be explained as follows:

1. The first stage is the user can choose to try the application by entering a query or directly out of the program.
2. When the user enters the query and presses the "Search" button, the user will switch to the output or results page, while if the user chooses to exit, then the program will stop.
3. After inputting input, the system will process the input. The query in the input will be calculated for its relevance to documents already in the system. The first

process of pre-processing on thematic documents is tokenizing, stopword removal, and weighting.

4. Search for thematic documents totaling 178 pieces of documents relevant to the query using the Language Model method that generates an output of top 90 documents relevant to the query.
5. On thematic documents and geographical documents the calculation process is done using the Vector Space Model. The number of documents calculated is only a portion of the existing 178 documents, which are about 90 documents relevant to the query. The geographical document is also calculated using the Vector Space Model which amounted to 178 documents.
6. From the results of thematic documents and geographical documents, a process of calculation is performed using Query-Aware Document Ranking to see the relationship between thematic documents and geographic documents. The output of Query-Aware Document Ranking is the most relevant value from existing documents.
7. After the calculation is complete, then output is given on the output page. The output of the system is an answer to the query, query support document, and visualization of the query on the map.

b. Dialog Design

The dialog design is represented by the Hierarchical Diagram. A hierarchy diagram is also called a functional diagram. This diagram shows the hierarchical relationship of modules to a system. In the designed program, there are two modules: front page or input page and output page.

c. Interface Design

Interface design is a design module which designed based on the program created. This section will describe the modules and the functions of the components in the module. A complete explanation of the following modules:

1. Front Page Module

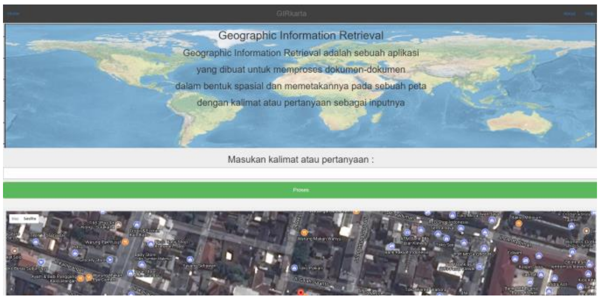
   This module is the front page of the program designed as well as this module is the input module of the program. Users can see brief details about Geographic Information Retrieval. On this page there is a textbox to enter the query you want to search for and comes with a "Process" button to navigate to the next page. There is also a "Help" button on the right of the web that appears pop up if the push and "About" pop-up also appears if pressed. At the bottom of the page, users can see the maps of Surakarta city that have not been marked with a marker or Pin of interest.
2. Output Page Module

   This module is the output page of the designed program. Results from the output are divided into 3, which is the answer to the query, the document that supports query, and visualization on the map. The documents that support query have a list of relevant documents. If it's clicked then the system displays the contents of the document.

   In this module, the process that occurs is the pre-processing input and then the existing thematic documents are indexed using the Language Model and the last is combining thematic documents and geographic documents using the Query-Aware Document Ranking method. After all the process is done, then the system will direct the user to the output module.

E. Implementation System

The next stage after completing the design stage is the implementation phase of the system, all the designs that have been made will be implemented into the system. Display of the system that has been made can be seen in Figure 1, 2 and 3.

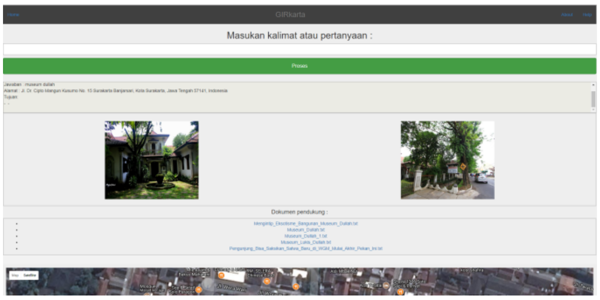


Geographic Information Retrieval

Geographic Information Retrieval adalah sebuah aplikasi yang dibuat untuk memproses dokumen-dokumen dalam bentuk spasial dan memetakannya pada sebuah peta dengan kalimat atau pertanyaan sebagai inputnya

Masukan kalimat atau pertanyaan :

Figure 1: Website homepage



Masukan kalimat atau pertanyaan :

Jawaban : museum dullah
Alamat : Jl. Dr. Cipto Mangun Kusumo No. 15 Surakarta Banjarsari, Kota Surakarta, Jawa Tengah 57141, Indonesia
Tujuan:

Dokumen pendukung :

- Mengintip_Eksotisme_Bangunan_Museum_Dullah.txt
- Museum_Dullah.txt
- Museum_Dullah_1.txt
- Museum_Lukis_Dullah.txt
- Pengunjung_Bisa_Saksikan_Satwa_Baru_di_WGM_Mulai_Akhir_Pekan_Ini.txt

Figure 2: Output page

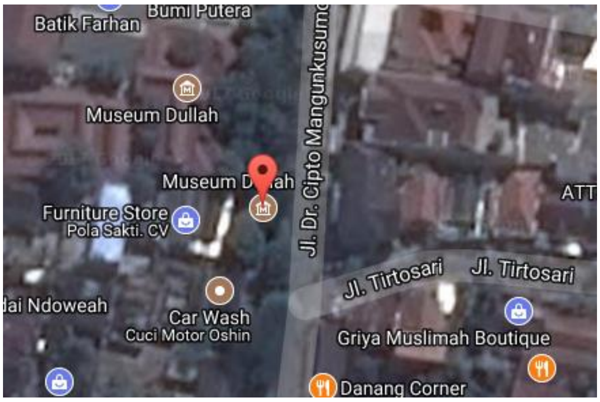


Figure 3: Visualization on maps

IV. Result and Discussion

A. Testing Method

After the process of designing and making the system, there will be the process of testing the system. In this case, the test is divided into 3 tests, namely the website testing section, the accuracy of obtaining the document, the test on the Mean Reciprocal Rank (MRR), and the test of the answer whether the answer is right or wrong.

a. Testing Document Accuracy

Document accuracy testing is done by calculating precision and recall. The first step is to calculate the Average Precision for each tested query. Then, the results of the Average Precision calculation of each query is averaged to obtain the

Mean Precision value. In this case, document accuracy testing is performed on 50 different queries.

b. Mean Reciprocal Rank (MRR) Testing

In testing this system, testing is done to find Top-n answers which are used to guide the system, how the system can produce the most appropriate answer. Testing will be performed on Top-5 answers. In addition, the calculation process is performed using the Mean Reciprocal Rank (MRR). Basically the Reciprocal Rank Mean is a method used to evaluate the accuracy of a system by calculating the correct order of answers first then the value will be averaged for each query.

c. True False Answer Testing

In this part, the system will generate an answer. The “True or False level” of the answer is rated by the user with the answer key. The answer key is obtained from a list, derived from 50 queries that have been searched the correct answer to the question manually. If the system output is true and match with the answer key, the evaluation value will be marked as “True” else if the system output is is false, then the evaluation will be marked as “False”. If the system did not generate any answer, the evaluation also would be marked as “False”.

B. Discussion

Based on the test sub-section, then the discussion will be conducted on the tests that have been described in the sub-section of the test is the accuracy of the acquisition of documents, discussion of Mean Reciprocal Rank (MRR) and discussion of the true or false answer. Each test will be explained in more detail in Section 1 until Section 3.

a. Document Accuracy Discussion

In this research, accuracy test of document acquisition is done to 50 queries by using precision and recall. The testing process is done by calculating Average Precision for each existing query. After that, the obtained results will be calculated average to obtained Mean Precision. In this case, accuracy testing is performed on Top-5 documents. An example of a precision calculation for the query "KRA Sosorodiningrat IV" can be seen in Table 5.

Table 5
Query Precision Result "Museum apa yang didirikan oleh KRA Sosorodiningrat IV?”

Rank	Document name	R(Relevant)_ /TR(Not Relevant	*Top*-5 Document	
			Precision	*Recall*
1	Museum_Radya_Pustaka_ Solo.txt	R	1	0.2
2	Mengunjungi_Museum_ Radya_Pustaka_Solo.txt	R	1	0.4
3	Museum_Radya_Pustaka_S olo_Jawa_Tengah.txt	R	1	0.6
4	Museum_Radya_ Pustaka.txt	R	1	0.8
5	Museum_Radya_Pustaka_ Wisata_Seru_Mengenal_ Kota_Solo.txt	R	1	1

From the results in Table 5, it can be seen that the Average Precision value for Top-5 documents is 100%.

An example of the result of precision and recall that has been done on 50 queries can be seen in Table 6 and its Mean Average Precision results can be seen in Table 7.

Table 6
Sample Results of Average Precision and Mean Average Precision for 5 Queries

No	Query tested	Average *Precision Top*-5 Document	Duration
1	Objek yang dibangun oleh Susuhunan Pakubuwana II	100%	00:18:01
2	Museum apa yang didirikan oleh KRA Sosorodiningrat IV?	100%	00:18:49
3	Kel Kampung Batik Kauman	100%	00:17:46
4	Berapa jarak Candi Plaosan dari Candi Prambanan?	71%	00:19:49
5	Lokasi kecamatan Candi Sambisari?	100%	00:17:41

Table 7
Mean Average Precision Result

Mean Average Precision	73.02%

From the results above, it can be seen that the value of Mean Average Precision for 50 queries is 73.02% and the average time required to process one sentence or question is 18 minutes 34 seconds.

b. Mean Reciprocal Rank (MRR) Discussion

In this study, evaluation for top-n answer is done to some systems can produce the most appropriate answer with the query entered. Testing will be performed on top-5 answers from 50 queries tested. In this case, the calculation of accuracy will be compared by using the Mean Reciprocal Rank or the value contained in the first occurrence rank of the correct answer. The result of Mean Reciprocal Rank value of query tested against Top-5 document can be seen in Table 8.

Table 8
MRR results for queries "Kecamatan Candi Bubrah"

No.	Answer	Top-5 Answer
1	Prambanan	1/1
2	Prambanan	
3	Prambanan	
4	(no answer)	
5	(no answer)	

From the results of testing of Mean Reciprocal Rank that have been done, obtained a result Average Mean Reciprocal Rank when using Top-5 document is 0,8 or 80% for 50 query.

c. True False Answers Discussion

In this test, the query used is 50 queries. Basically, true or false evaluation is a test done for the purpose of checking the top answers generated by the Q & A system. In this case, true or false answer to a query is judged on the basis of the topmost response generated by the system. The true or false answer in this research was conducted on Top-5 documents used on the question and answer system. The True or false answer test results for 50 queries tested in this study is 78%, which calculated from 39 correct answers and 11 wrong answers.

C. Evaluation

Based on the tests that have been done, then the evaluation of the system has been made. The built-in Geographic

Information Retrieval system still contains shortcomings because it still generates incorrect answers or queries entered by users that do not match the answers. This happens because of several factors:

1. The recognition of a word entity is still less precise because of an error in giving the word for example the word "120 m" should be given a distance entity, it gets O entity (other). In addition, the word "Jl." Should get a street entity, it gets an O entity. This is because the word path in everyday life can also be written with jln. or jl. But the system can not accept different entities that word.
2. The Q & A system still finds an incorrect answer to the sentence or question entered. This is proved by the output that some queries are still wrong from the actual answer and even return an irrelevant answer altogether.

V. Conclusions

In this study, the built system aims to find the answer from sentences or questions that are entered by users, and it displays the answer from the query, supporting documents, and visualization on the geographic map. Basically, the system built is Geographic Information Retrieval for Surakarta city. Based on the results of tests that have been done, the following conclusions are obtained:

1. The system that has been build can generate answers from sentences or questions entered by the user. In this case, the system displays the answers, supporting answer documents, and visualizes them in a geographic map using the Google Maps API.
2. The system shows good results to be applied to the acquisition value of accuracy of the system in obtaining the answer or the Mean Reciprocal Rank value with an accuracy of 80%.
3. The accuracy of obtaining document using Mean Average Precision shows the use of top-5 documents with an accuracy of 73.02%.

References

[1] Andrade, L and Silva, M. J. Relevance Ranking for Geographic IR. Proceedings of the workshop on *Geographic Information Retrieval,* SIGIR 06, Seattle, USA, (2006).

[2] Dr. Ir. Barkey, Roland A. et al., “SISTEM INFORMASI GEOGRAFIS”, Makassar : Faculty of Forestry Hasanuddin University, (2009).

[3] Hanani, Ajib et al, *Pemberian Harakat Bahasa Arab Menggunakan Metode N-gram dan C5.0*, Malang: Brawijaya University, (2015).

[4] Dr. B, Yosua, *Neural net language models*. Canada: Université de Montréal, (2008).

[5] Lv, Yuanhua; Cheng Xiang Zhai, *Query Likelihood with Negative Query Generation*, Urbana: University of Illinois, (2012).

[6] Grossman, David A., dan Fireder, Ophir (2004), *Information Retrieval Algorithms and Heuristics*, New York: Springer,.

[7] Cai, Guoray; Yu Bo, *A Query-Aware Document Ranking Method for Geographic Information Retrieval,* USA: University Park, Pennsylvania, (2007).

[8] Jurafksy, Daniel; H. Martin, James, *Speech And Language Processing: An Introduction To Natural Language Processing. Computational Linguistics And Speech Recognition*, Upper Saddle River: Prentice Hall, (2009).

[9] Christopher D. Manning et al, *Introduction to Information Retrieval,* Cambridge, (2008).

[10] Kay, Russel, *System Development Life Cycle,* http://www.computerworld.com/article/2576450/appdevelopment/app-development-system-development-life-cycle.html, (2002).

[11] Top, Syafrudin M., *System Development Life Cycle* (SDLC). https://www.academia.edu/7478619/System_Development_Live_Cicle, (2009).

[12] Dewayani, Ery and Mawardi, Viny Christanti, *Surakarta Cultural Heritage Management Based on Geographic Information Systems,* Jakarta: Tarumanagara University, (2015).