Search relevant retrieval on indonesian translation hadith document using query expansion and smoothing probabilistic model

IOP Conf. Series: Journal of Physics: Conf. Series **1192** (2019) 012032 doi:10.1088/1742-6596/1192/1/012032

Search relevant retrieval on indonesian translation hadith document using query expansion and smoothing probabilistic model

Ika Rahayu Ponilan[1], Adiwijaya[2], Moch. Arif Bijaksana[3], Agus Suyadi Raharusun[4]

[1,2,3]School of Computing, Telkom University, Bandung, Indonesia,
[4]Universitas Islam Negeri Sunan Gunung Djati Bandung, Bandung, Indonesia

E-mail: [1] kaponilan@student.telkomuniversity.ac.id,
[2] adiwijaya@telkomuniversity.ac.id,
[3] arifbijaksana@telkomuniversity.ac.id,
[4] agussuyadi@uinsgd.ac.id

Abstract. Hadith are words, deeds, decrees and approvals of the Prophet Muhammad SAW which are used as the basis for Islamic Shari'a law after the Qur'an. Currently there are many websites that provide information about hadith to facilitate users in the process of hadith learning or what we usually know as Information Retrievel (IR), such as the Lidwa Pustaka website. Basically, IR provides a search box for users to enter queries that reflect the user's information needs. The hadith search process on Lidwa Pustaka uses exact string matching method, which in the process of searching the hadith to the user's query must be the same as the hadith document in order per word (term), so that for partial matching search (matching the query in each word without sequential) can't be done yet. In addition, the writing of synonyms or variant strings that differ in Indonesian hadith translations, such as **"الخمر" (al-khomru)** are written as "khamar", "khamer", "khamr" or "minuman keras", making the process of the hadith search less precise. Therefore, this study aims to improve the search system for the hadith, using the approach of query expansion and smoothing probability models, namely Jelinerk-Mercer Smoothing, Dirichlet Smoothing and Absolute Discounting Smoothing. The use of query expansion and smoothing probability models in this study resulted in Mean Average Precision ALL (MAP ALL) values in all hadith documents, Mean Average Precision@30 and the highest recall value of 30 compared to other methods, such as exact matching method for Lidwa Pustaka, Latent Semantic Indexing, Probability Model and Cosine Similarity.

1. Introduction

Recently, many researcher was doing research in the text classification field [1, 11, 14, 15], especially for islam content (Quran and Hadith) such as in [2, 12,13]. Hadith are words (words), deeds, decrees and approvals of the Prophet Muhammad SAW which are used as the basis for Islamic Shari'a law other than the Qur'an, Ijma '(the agreement of the ulamas) and Qiyas (establishing a law on new cases that have never existed before In the time of the Prophet, in terms of facilitating the user for the process of hadith learning, there are now many websites that provide information about hadith,

Information Retrieval (IR) like search engines has become an important tool for users to retrieve information on the website [3]. the user matches the hadith document collection index to find the hadith containing the query, which is then sorted according to various methods or models.

The current Indonesian search for the hadith system is the Lidwa Pustaka website. Search in Lidwa Pustaka cannot complete partial matching, meaning that it cannot handle queries that only have the same partial term with documents. For example, searches using "dilarang kikir" queries generate zero hadith documents retired by the system, whereas if the query is changed to "kikir" only, it produces 52 hadith documents. That is, queries "dilarang kikir" only have a partial similarity to the "kikir" term, so the system cannot handle partial matching searches.

Another problem in the search for the Indonesian translation of the hadith is that there are several different types of writing, such as the word "الخمر" (al-khomru) written in Indonesian to "khamar", "khamer", "khamr" or "minuman keras". This difference in writing is often referred to as synonym (the writing of words is different but has the same meaning / meaning) and string variants (writing words that are almost similar and meaning the same). In addition, user queries are still short so the system still cannot display documents that reflect user needs, such as queries: "jangan kikir," documents that contain the words "dilarang pelit" and "tidak diperbolehkan bakhil" will not be displayed by the system. Other problems found in users often experience difficulties in forming queries intended to retrieve information. This is because, they do not know the details of the construction of hadith and environmental collections from IR, even though the number of relevant documents obtained is influenced by the number of keywords in the query [4].

Based on the explanation above, it can be concluded that the search for the Indonesian translation hadith has not been able to handle the partial matching search and identify synonyms or variant strings, so that it is possible in the search process to produce hadiths that are less relevant to the user's needs. Therefore, another approach is needed such as rearranging queries (query reformulation) that are entered by users using expansion queries and smoothing probability models. The Expansion Query (QE) functions to extend the query entered by the user by adding several terms to it, such as synonyms and variant strings utilizing the Indonesian language thesaurus, while the smoothing probability model to handle partial matching problems.

2. Related work

One of IR model from the user side (user task) is a classic model. The concept of the classic model is that the document is represented by using the index term and the weight of the index term indicating the specifications for a particular document. One example of a classic model is the Vector model (Vector Space Model) and Probabilistic model [5]. While query expansion is an IR model from the query side. The following is an explanation for each method.

2.1. Vector Space Model (VSM)

Research using VSM is a research conducted by [6] using the LSI method. This LSI method will index (calculate the number of occurrences of words in documents and queries), then calculate simmilarity computation between queries and all hadith documents, then sort them according to what is most relevant to the user's input query. However, this LSI method has its drawbacks, that is, it must determine the dimension k value for cutting VSM between queries with hadith documents, so that if the determination of the k value is incorrect, the relevant hadith documents are not retrieved by the system. Another disadvantage to the VSM model is that it requires calculating the distance between the document vector and the query term, so that in this model there is a possibility that there is no document that can be displayed in one search [7].

2.2. Probabilistic model

The probabilistic model assumes that each term in the query is assumed to have the same term in the document. Each term in the initialized query is likely to have an appearance in the document called the

term frequency (tf) according to the number of terms in a document known as inverted document frequency (idf) as a reference for ranking documents to be displayed. This explains that, in addition to being able to rank documents, other capabilities of this model are also able to perform partial matching queries with documents that are considered appropriate [8]. In the study [9] using three probabilistic smoothing models, namely Jelinerk-Mercer Smoothing, Dirichlet Smoothing and Absolute Discounting Smoothing.

2.3. Query expansion

Query Expansion (QE) is the process of re-forming (reformulating) querying users by adding several terms to it [4]. QE can act as a connector because of the vocabulary gaps between queries and documents. Queries entered by users are generally short and QE can complete the information that users want to find. The purpose of query expansion is to improve system performance. Research [10] uses QE by utilizing thesaurus.

3. Propose

The general description of the system built in this study includes preprocessing the hadith and queries documents, document indexing, expansion queries on user input queries, calculating the level of relevance of the hadith documents to expanded queries and retrieving hadith documents deemed relevant by the system based on their relevance level. System design can be seen in Figure 1 and the description can be seen below.

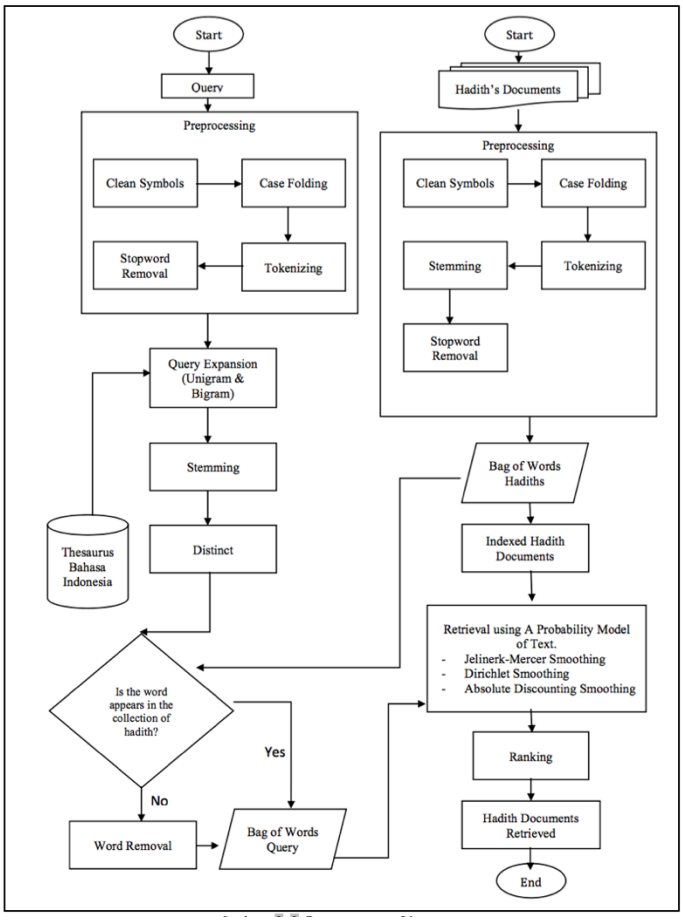

Figure 1. System implementation.

3.1. Hadith's documents

In this study, we collected 2030 data on Hadith Sahih Al-Bukhari, an Indonesian translation, taken from Lidwa Pustaka. In this hadith document is carried out the process of separating sanad and matan, because this research only searches the hadith of the matan (content) of the hadith.

3.2. Preprocessing

After the hadith is separated from the sanad, then preprocessing will be done by including clean symbols, case folding, tokenizing, stemming and stopword removal. User queries will be preprocessed like hadith documents, but not through the process of separating sanad and matan, and the stemming process. The stemming process is not carried out, because this query will be expanded by utilizing the Indonesian thesaurus.

3.3. Query expansion

Query Expansion is a process for extending user queries that aim to find synonyms and variant strings of words in the initial query, given that queries from users are generally short, so that documents retrieved by the system are more accurate. This expansion query utilizes the Indonesian thesaurus from the Department of National Education Language Centre by taking synonyms of nouns (n) and verbs (v).

3.4. Distinct

Distinct is a process to prevent duplication of terms in query expansion after being stemming.

3.5. Checking the appearance of words

The process of checking whether the stem expansion query that has been carried out appears in the bag of words hadith or not. This is to avoid zero probability of the next probabilistic smoothing model process.

3.6. Word removal

Word Removal is the process of deleting each term from the query expansion, if the term does not appear in all the hadith documents.

3.7. Retrieval using a probability model of text

The process of calculating query relevance probability (bag of words query) against hadith documents (bag of words hadith). In this study, using three methods of probability smoothing, namely Jeninerk-Mercer, Dirichlet and Absolute Discounting. The explanation for each smoothing method is as follows:

3.7.1. Jelinerk-Mercer smoothing

The Jelinerk-Mercer Smoothing method uses as a parameter, where the limit value is [0,1]. The smaller means the smaller the term to be smoothed and the probability value is dominated by terms that most often appear in a document. Value = 0 is the same as Boolean AND, while 0 is the same as Boolean OR. The formula for the Jelinerk-Mercer smoothing method can be seen in the following equation 1 [9].

$$P(q_i|D) = (1-\lambda)\frac{(tf\,q_i, D)}{|D|} + \lambda\,\frac{(tf\,q_i, C)}{|C|} \qquad (1)$$

Annotation:

$P(q_i|d) =$ Term probability (query) of a document *d*

λ = parameter

$tf_{qi,D}$ = Number of term *i* (query) occurrences of a document *d*

|D| = Number of terms in a document *d*

$tf_{qi,C}$ = The number of occurrences of term *i* (query) on the entire collection of documents
|C| = The number of terms in the entire document collection

3.7.2. Dirichlet smoothing

Dirichlet Smoothing uses as a parameter to set the boot to the term. The smaller the value, the smaller the smoothing term. The limit value is [500,10.000] or can be adjusted based on the average document length. Precision values are more sensitive to, if the query is long compared to short queries, especially when small. When μ is large enough, all queries are long better than short queries and vice versa. Formula Dirichlet smoothing can be seen in 2 [9].

$$P(q_i|D) = \frac{tf_{qi}, D + \mu \frac{(tf_{qi}, C)}{|C|}}{|D| + \mu} \quad (2)$$

Annotation:
$P(q_i|d) =$ Term probability *i* (query) of a document d

μ = parameter

$tf_{qi,D}$ = Number of term *i* (query) occurrences of a document d
|D| = Number of terms in a document *d*
$tf_{qi,C}$ = The number of occurrences of term *i* (query) on the entire collection of documents
|C| = The number of terms in the entire document collection

3.7.3. Absolute discounting smoothing

Absolute discounting uses as a smoothing parameter, which affects the term depending on the number of unique terms and the number of occurrences of the term in the entire collection of documents. The value limit δ is [0,1], which if $\frac{\delta|d|_u(tf_{qi},C)}{|C|} > 1$, large values of δ will make the term weight flat, but on the contrary, will make the term weight more skewed by the number of terms that appear in a document. The formula for Absolute Discounting Smoothing can be seen in the following 3 [9].

$$P(q_i|D) = \frac{\max(tf_{qi}, D - \delta, 0)}{|D|} + \frac{\delta|d|_u}{|D|}\frac{(tf_{qi}, C)}{|C|} \quad (3)$$

Annotation:
$P(q_i|d) =$ Term *i* probability (query) of a document d

δ = parameter
$tf_{qi,D}$ = Number of term *i* (query) occurrences of a document d
|D| = Number of terms in a document *d*
$tf_{qi,C}$ = The number of occurrences of term *i* (query) on the entire collection of documents
|C| = The number of terms in the entire document collection
$|d|_u$ = Number of unique terms that exist in a document *d*

3.8. Ranking

The ranking process is sorted by the greatest probability value.

3.9. Hadith documents retrieved

The process of displaying relevant hadith documents according to the system based on queries from users. The test in this study uses the mean average precision all (of all the hadith documents), mean average precision@30 and recall@30 (for 30 hadith documents), which is considered by the system to have the highest relevance value to the user's query. Use 30 hadith documents on the mean average precision@30 and recall@30, because remembering the user usually only sees the top documents displayed by the system, so there is no need for energy and long time for the search process. Average Precision (AP) is usually used to calculate system relevance performance for each query, while Mean Average Precision (MAP) is used to calculate the system relevance performance of a query set (many queries). That is, if we have a query of 5 pieces, then each AP value of the query will be summed, then divided by 5. AP formula can be seen in formula 4, while the MAP formula can be seen in formula 5.

$$AP = \frac{1}{|R|}\sum_{i=1}^{n} Prec(i).relevance(i) \quad (4)$$

Annotation:
|R| = the number of documents that are actually relevant
Prec(i) = precision on top *i* of documents
relevance(i) = 1 if relevant, others 0
n = number of relevant documents recommended (system retrieved)

$$MAP = \frac{1}{|Q|}\sum_{Qi \in Q} AP(Qi) \quad (5)$$

Annotation:
|Q| = number of queries
AP = Average Precision value per each query

4. Result and discussion

The test results in this study were carried out by comparing several other methods as baseline research, such as exact matching method on Lidwa Library, LSI, Cosine Similarity and probability models without smoothing. Table 1 is the result of testing that has been done using several methods.

In Table 1 it can be seen that the hadith search based on user queries using a probability model smoothing before the query expansion is applied, produces the highest value (bold text) for MAP all and MAP@30 compared to exact matching, LSI and Cosine Similarity methods. This is the basis of why the probability model smoothing method is chosen to apply query expansion. In addition to these reasons, another reason is the probability model smoothing method when compared to other methods such as, Cosine Similarity and Probability Models produce higher MAP all, MAP@30 and Recall@30 values. The probability model smoothing with Query Expansion model which is gray blocked in Table 1, is the method used in this study. The method has the highest value compared to other methods, this is shown in bold in the table. This is because, query expansion can perform a hadith search semantically.

The exact matching method produces the lowest value compared to other methods, because in the process of searching for hadiths on user queries it must be the same as hadith documents sequentially per word (term), while the queries tested are more than one word, which may have similarities in each term sequentially with hadith documents is very low, so this is what causes the absence of documents that can be displayed in a single search.

LSI produces a performance that is quite low because it uses the matrix dimension cutting k, so it takes precision to determine the k value, so the system can produce more hadiths that are relevant to the query. While the Cosine Similarity method is not quite high in performance, because in this method requires the calculation of the distance between the document vector and the query term, so that in this method there is the possibility of no documents that can be displayed in one search. As with the Cosine Similarity method, ordinary probabilistic methods also have the possibility of missing documents that can be displayed in a search because they produce zero probability.

Table 1. MAP all, MAP@30 and recall value using various methods.

Method	**MAP All**	**MAP@30**	**Recall@30**
Exact Matching (at Lidwa Pustaka)	0.67%	0.67%	0.67%
Latent Semantic Indexing (LSI)	5.56%	4.47%	10.58%
Cosine Similarity	2.73%	2.73%	2.92%
Probability Model	3.69%	3.69%	4.25%
Probability Model Smoothing	**23.75%**	**21.31%**	30.95%
Probability Model Smoothing with Query Expansion	**53.35%**	**48.76%**	65.23%

The system that has been built will be tested against the hadith document, to measure the relevant truth of the hadith search on the user's query. The purpose of this system testing is to determine the effect of the expansion query and the effect of smoothing parameters before and after applying the expansion query to the mean average precision and recall levels. The testing of the effect of query expansion can be seen in Table 2, while the effect on the smoothing parameters can be seen in Table 3 and Table 4.

Tabel 2. The value of MAP all, MAP@30 and recall@30 uses the probability model before and after the implementation of the expansion query.

Method	**MAP ALL**	**MAP@30**	**RECALL@30**
Absolute Discounting Smoothing	26.04%	23.11%	33.25%
Dirichlet Smoothing	26.54%	24.16%	34.71%
Jelinerk Mercer Smoothing	26.97%	23.87%	33.76%
Absolute Discounting Smoothing with Query Expansion	57.60%	52.48%	68.91%
Dirichlet Smoothing with Query Expansion	**62.54%**	**57.39%**	**76.59%**
Jelinerk Mercer Smoothing with Query Expansion	62.02%	56.87%	73.93%

Table 3. MAP all, MAP@30 and recall@30 values using the probability model before implementing an expansion query.

Method	Parameter	MAP All	MAP@30	Recall@30
Absolute Discounting Smoothing	**δ**: 0.0	0.01%	0.00%	0.00%
	δ: 0.1	**26.04%**	**23.11%**	33.25%
	δ: 0.2	25.67%	22.95%	**33.25%**
	δ: 0.3	25.59%	22.90%	33.18%
	δ: 0.4	25.61%	22.97%	33.13%
	δ: 0.5	25.73%	23.06%	32.95%
	δ: 0.6	25.63%	23.00%	32.85%
	δ: 0.7	25.69%	23.08%	32.58%
	δ: 0.8	25.11%	22.71%	32.69%
	δ: 0.9	24.20%	22.04%	32.15%
	δ: 1	9.99%	9.52%	13.81%
Dirichlet Smoothing	***μ***: average	26.10%	23.57%	32.58%
	μ: 500	**26.54%**	**24.16%**	34.67%
	μ: 1000	26.51%	24.12%	34.68%
	μ: 1500	25.58%	23.17%	34.68%
	μ: 2000	25.50%	23.13%	34.68%
	μ: 2500	25.47%	23.05%	34.64%
	μ: 3000	25.47%	23.08%	**34.71%**
Jelinerk Mercer Smoothing	***λ***: 0.0	0.01%	0.00%	0.00%
	λ: 0.1	**26.97%**	**23.87%**	33.70%
	λ: 0.2	26.81%	23.82%	**33.76%**
	λ: 0.3	26.74%	23.75%	33.71%
	λ: 0.4	26.61%	23.63%	33.71%
	λ: 0.5	26.50%	23.49%	33.46%
	λ: 0.6	26.36%	23.36%	33.40%
	λ: 0.7	26.10%	23.12%	33.33%
	λ: 0.8	25.68%	22.75%	33.19%
	λ: 0.9	25.38%	22.27%	32.65%
	λ: 1	22.65%	19.55%	32.47%
Average		**23.46%**	**21.01%**	**30.48%**

Table 4. Value of MAP all, MAP@30 and recall@30 using probability model smoothing after implementation of expansion query.

Method	Parameter	MAP All	MAP@30	Recall@30
Absolute Discounting Smoothing with Query Expansion	**δ**: 0.0	0.00%	0.00%	0.00%
	δ: 0.1	**57.60%**	**52.48%**	68.69%
	δ: 0.2	57.32%	52.44%	**68.91%**
	δ: 0.3	57.02%	52.16%	68.63%
	δ: 0.4	56.68%	51.77%	68.23%
	δ: 0.5	56.19%	51.42%	67.67%
	δ: 0.6	55.04%	50.17%	66.90%
	δ: 0.7	55.03%	50.37%	66.58%
	δ: 0.8	52.47%	48.12%	65.93%
	δ: 0.9	50.64%	47.21%	66.24%
	δ: 1	21.44%	20.31%	30.81%
Dirichlet Smoothing with Query Expansion	***μ***: average	55.16%	51.09%	67.32%
	μ: 500	**62.54%**	**57.39%**	75.88%
	μ: 1000	61.82%	56.76%	76.34%
	μ: 1500	60.83%	55.62%	76.35%
	μ: 2000	60.65%	55.52%	**76.59%**
	μ: 2500	60.34%	55.09%	76.39%
	μ: 3000	60.26%	54.92%	76.25%
Jelinerk Mercer Smoothing with Query Expansion	***λ***: 0.0	0.00%	0.00%	0.00%
	λ: 0.1	61.28%	55.94%	72.57%
	λ: 0.2	61.53%	56.09%	72.52%
	λ: 0.3	61.76%	56.39%	72.96%
	λ: 0.4	61.90%	56.58%	73.16%
	λ: 0.5	61.67%	56.29%	73.16%
	λ: 0.6	**62.02%**	**56.87%**	73.93%
	λ: 0.7	61.86%	56.64%	73.81%
	λ: 0.8	61.61%	56.39%	**73.81%**
	λ: 0.9	59.67%	53.97%	72.92%
	λ: 1	52.83%	45.98%	69.14%
Average		**53.35%**	**48.76%**	**65.23%**

Analysis of test results for the effect of smoothing parameters on system performance are as follows:

4.1. Absolute discounting smoothing

In Table 3 and Table 4 the highest performance value of MAP all & MAP@30 for the Absolute Discounting Smoothing method is when the value δ= 0.2, meaning the change in value δ, has no effect on the short query (the pre-expanded query) and the long query (the query after expansion). At δ = 0.2 the weight of the term in the query tends to the weight of the appearance of the term in a document (seen term) rather than the weight of the appearance of the term in the entire document (unseen term). The meaning of change in value δ, is the higher the parameter value δ, the higher the smoothing done and vice versa.

Absolute Discounting Smoothing method uses unique terms in the formula, as a measure of how focused the topic of the document is. This method uses parameters $\alpha_d = \frac{|d|u}{|d|}$,, where the length of the document and the unique term determines the probability of the relevance of the hadith document to the query. It can be concluded that, the greater the value δ, will affect the document that has many unique terms, so that the document has a large probability and is in the top rank. In other words, documents that have many unique terms tend to be retrieved by the system when the value gets bigger.

4.2. Dirichlet smoothing

The Dirichlet Smoothing method in Table 3 and Table 4 produces the highest performance values for MAP all & MAP@30 on parameters μ: 500, meaning that the same as the Absolute Discounting Smoothing method, that changes the value in the Dirichlet Smoothing method, does not affect short queries (before expanding)) and long queries (queries after expansion). At μ: 500 the probability calculation is inclined to the calculation of the term weight in a document, compared to the term weights throughout the document. Dirichlet Smoothing method uses parameters $\alpha_d = \frac{\mu}{\mu+|d|}$,, where the length of the document is very influential on the probability value of a document. That is, long documents are more influential when small and vice versa. In other words, small tend to retrieve long documents compared to short documents.

4.3. Jelinerk-Mercer smoothing

Unlike the two previous methods, namely Absolute Discounting Smoothing and Dirichlet Smoothing which involves the length of the document in its parameters, the Jelinerk-Mercer Smoothing method only uses parameters λ. Lamda (λ) serves to balance the effect of the occurrence of terms in the entire collection of hadith documents. Change in value λ, means that the greater the estimated λ value of document probability is skewed towards the term weight in the entire collection of documents, compared to the term weight in a document and vice versa. The Jelinerk-Mercer Smoothing method in Table 3 produces the highest performance value for MAP all & MAP@30 on the parameter λ: 0.1, while in Table 4 produces the highest performance value for MAP all & MAP@30 in the parameter λ: 0.6, meaning that the value λ changes in the Jelinerk method -Mercer Smoothing, is very influential on short queries (pre-expanded queries) and long queries (queries after expansion). That is, the Jelinerk-Mercer Smoothing method acts as a modeling query that explains general and less informative terms in the query.

5. Conclusion

The probability model smoothing with query expansion method produces the optimal performance value of MAP All, MAP@30 and Recall@30 compared to other methods without an expansion query, such as exact matching, LSI, Cosine Similarity and probability models.

Expansion queries improve system performance, for the average MAP all, MAP@30 and Recall@30 by 29.99%, 27.75% and 34.76%. This is because the use of expansion queries can identify synonyms or variant strings per each term in the query, so that the term that does not exist when the initial query becomes available when it is expanded. In other words, the use of query expansion is able to search for the relevance of hadiths to semantic user queries.

Setting parameters in all three smoothing methods after and before the query expansion affects the value of MAP. At Absolute Discounting Smoothing, the parameter δ: 0.2 gives the most optimal MAP All and MAP@30 values, this means that probability calculations tend to lean towards seen terms in a document, so that only requires a little smoothing. Absolute Discounting Smoothing method also affects the number of unique terms in a document, where the more unique terms, the greater the probability value of a document against the query. In Dirichlet Smoothing, the parameter μ: 500 produces the most optimal MAP all and MAP@30, compared to other μ values. This means that small μ tends to be seen term in a document and does not require too much smoothing. The value of μ also affects the retrieval process, because small μ tends to retrieve long documents compared to short documents. This is related to the role of the Dirichlet Smoothing Method as an role estimation. Whereas in Jelinerk-Mercer Smoothing, λ: 0.1 produces the most optimal MAP all & MAP@30 before query expansion, while after the expansion query λ: 0.6 produces the most optimal MAP all & MAP@30. This means that more smoothing is needed for longer queries (after the expansion query), because in long queries there are some less informative terms.

As for the suggestion for the development of the system in a similar study in order to improve the efficiency of the system, a method is needed to identify the hadith documents semantically (similarity of writing / synonyms and variations of writing), because the system that is built now has not been able to identify the meaning of the hadith. In addition, a list of words / dictionaries is needed for translation synonyms or variations in Arabic writing, because the majority of the words in the hadith documents are Arabic translations, so the system still retrieves documents that are relevant to the query and those that are not relevant if the term Expansion is general or often appears in each document.

References

[1] Asriyanti Indah Pratiwi, Adiwijaya. 2018. On the Feature Selection and Classification Based on Information Gain for Document Sentiment Analysis. *Applied Computational Intelligence and Soft Computing*, 2018.

[2] Al Faraby, S., Jasin, E.R.R. and Kusumaningrum, A., 2018, March. Classification of hadith into positive suggestion, negative suggestion, and information. *In Journal of Physics: Conference Series* (Vol. 971, No. 1, p. 012046). IOP Publishing.

[3] Syazhween Zulkefli, N S, Rahman, N A, & Bakar, Z A 2016 Analyzing Search Retrieval Results on Malay Translated Hadith Text Documents *International Conference on Applied Computing, Mathematical Sciences and Engineering (ACME)*

[4] TP, B P, & Gunawan, I 2015 Sistem Information Retrieval Pencarian Kesamaan Ayat Terjemahan Al Quran Berbahasa Indonesia dengan Query Expansion dari Tafsirnya *Seminar Nasional "Inovasi dalam Desain dan Teknologi"*

[5] Song, F, & Croft, W B 1999 A general language model for information retrieval *Proceedings of the eighth international conference on Information and knowledge management* hal 316-321 ACM

[6] Amirah Nur, N Rahim, T M Mabni, Z Hanum, H M, & Rahman, N A 2016 A Malay Hadith translated document retrieval using parallel Latent Semantic Indexing LSI *In Information Retrieval and Knowledge Management (CAMP), 2016 Third International Conference on*, pp 118-123

[7] Manning, Christopher, D, Raghavan, P, & Schutze, H 2009 Introduction to Information Retrieval ‖ Cambridge University Press

[8] Fuhr, N 1992 Probabilistic Models in Information Retrieval The computer journal, 243-255e Discounting Smoothing Ketiga metode tersebut digunakan karena metode probabilistik biasa m

[9] Zhai, C, & Lafferty, J 2017 A study of smoothing methods for language models applied to ad hoc information retrieval *ACM , 51*, 268-276

[10] Rahman, N A, Bakar, Z A, & Sembok, T T 2010 Query Expansion using Thesaurus in Improving Malay Hadith Retrieval System *Information Technology (ITSim), 2010 International Symposium in* , 1404-1409

[11] Mubarok, M.S., Adiwijaya and Aldhi, M.D., 2017. Aspect-based sentiment analysis to review products using Naïve Bayes. *In AIP Conference Proceedings* (Vol. 1867, No. 1, p. 020060). AIP Publishing.

[12] Reynaldi Ananda Pane, Mohamad Syahrul Mubarok, Nanang Saiful Huda, Adiwijaya, 2018. A Multi-lable Classification on Topics of Quranic Verses in English Translation using Multinomial Naive Bayes. *In 2018 6th International Conference on Information and Communication Technology* (ICoICT). IEEE.

[13] Al Mira Khonsa Izzaty, Mohamad Syahrul Mubarok, Nanang Saiful Huda, Adiwijaya, 2018 A Multi-label Classification on Topics of Quranic Verses in English Translation Using Tree Augmented Naïve Bayes. *In 2018 6th International Conference on Information and Communication Technology* (ICoICT). IEEE.

[14] Fahmi Salman Nurfikri, Adiwijaya, Mohamad Syahrul Mubarok, 2018 News Topic Classification Using Mutual Information and Bayesian Network, *In 2018 6th International Conference on Information and Communication Technology* (ICoICT). IEEE.

[15] Afianto, M.F. Adiwijaya, and Al-Faraby, S., 2018, March. Text Categorization on Hadith Sahih Al-Bukhari using Random Forest. In *Journal of Physics: Conference Series* (Vol. 971, No. 1, p. 012037). IOP Publishing.