SEMANTIC INDEXING FOR QUESTION ANSWERING SYSTEM

Kasturi Dewi Varathan[1]*, Tengku Mohd Tengku Sembok*[2]*, Rabiah Abdul Kadir*[3]*, Nazlia Omar*[4]

[1]Department of Information Systems, Faculty of Computer Science & Information Technology, University of Malaya
Kuala Lumpur, Malaysia

[2]Cyber Security Centre, National Defence University Malaysia
Kuala Lumpur, Malaysia

[3]Faculty of Information Science & Technology, UKM,
43600 Bangi, Malaysia

[4]Center for AI Technology, Faculty of Information Science and Technology, UKM,
43600 Bangi, Malaysia

Email: [1]kasturi@um.edu.my, [2]tmts@iium.edu.my, [3]rabiah@fsktm.upm.edu.my, [4]no@ftsm.ukm.my

Abstract

With the vast growth of various forms of digital data, automated indexing has become very important so that it enables the needs of the current users to be fulfilled. Keywords based indexing has failed to accommodate to the needs of the present demands. The representation of the document content as well as the indexing process is a crucial factor that ensures the success of retrieval process. Therefore, this research introduces a new approach in creating semantic indexing that uses Skolem representation which automatically indexes multiple documents into a single knowledge representation. This knowledge representation will then be used by the proposed question answering system in retrieving the answers as well as pointing to the documents the answer contains based on the user's query. The system managed to achieve 93.84% of recall and 82.92% of precision.

***Keywords*:Skolem clauses; Skolem indexing; semantic indexing; question answering**

1.0 INTRODUCTION

We are in the information intensive environment in which there is a rapid growth in the digital contents. Users are no longer interested in retrieving a set of documents for their query. Eventhough the relevant information that they required lies in the first hit returned but finding the result of their query within the document can be time consuming. A question answering (QA) system managed to solve this hassle of retrieving answers based on users' questions. This system managed to come up with answers exemplified by “What is the capital of Malaysia?” or “Who is van Rijsbergen?” or “When the tsunami hits Japan?” It managed to eliminate the burden of query formulation and the tediousness of reading a lot of unrelated documents in order to retrieve the required answer [1]. In fulfilling this kind of goal, knowledge management and information retrieval has become the most important issues that need to be tackled. Past research has proven that there exists a consistent relationship between knowledge representation and the performance of retrieval results [2]. The demand of automated indexing has increased tremendously with the incredible rate of growing corpus and digital data. In dealing with such a huge amount of data, many questions arise on how to handle with the inconsistencies of the knowledge representation and how to normalize these documents into a single standard representation that can be used for retrieval purposes.

In order to face the challenges of digital contents, knowledge management and related theories as well as technologies for managing the digital contents have risen to be the most important issues to be tackled [3]. Knowledge representationis a crucial component of any information retrieval system [4], [5], [6]. The representation itself is considered as the major problem especially in representing the content of unstructured text in an effective way that leads to better system performance [7]. Most question answering systems that are built nowadays are not strong in knowledge representation even though there has been some recent progress towards that direction [8]. As a consequence, the output obtained from the retrieval is not accurate no matter how good the retrieval engine is [9]. The authors in [10] also agreed that if the document is incorrectly represented, then definitely the answer that is going to be retrieved from the erroneous document representation will also be incorrect. The knowledge representation of each and every single document that lies in the document repository has to be represented in a single uniform representation. In order to achieve this, each of the documents has to be indexed in a proper manner. To achieve this objective, a proper plan needs to be outlined and executed carefully.Thus, this research has focused on a new approach in creating single knowledge representation which is known as semantic index that represents multiple documents. This indexing will then be used in retrieving result(s) based on the queries posted by the users and also the document source which contains the answer(s).The following sections describe the review of related works based on question answering and semantic indexing followed by design of the framework for semantic indexing for question answering and the semantic Skolem index creation. An illustrated example is also shown together with experimental results and discussion. Performance of the system is evaluated and discussed followed by a conclusion which summarizes the findings and further works.

2.0 RELATED WORK

Past research in IR revealed that each document is characterized as a set of index terms that exist in the document [11]. These index terms represent the keywords of the documents. Bag of Wordsmodel requires documents to be indexed by considering all terms in them as independent keywords [12]. Besides that, vector space model represents a document with an unstructured collection of keywords or terms which in general have been assumed to be statistically independent. It has been proven that no matter how much of statistical approach applied to bare keywords, it will not be able to recover the information that was cast away during the keyword extraction process [13]. On top of that, this model is the presumption that the dimensions are orthogonal in which the words are represented independently [14]. This model along with the document representation, the user’s question and ranking function allows for retrieving relevant documents which suit the user’s query.

However, it does not go beyond the idea of counting word occurrences [15]. The semantic relation is neglected in the document representation. As a consequence, a lot of semantics in a document are lost when the texts are replaced with sets of words. Detailed analysis showed that this is mainly due to the fact that emphasis was only given to bare lexical skeleton and linguistically relevant information was neglected [16,17]. On top of that, inconsistencies issues between document contents could not be overcome by just using keyword based indexing method [17, 18, 19]. These lead to low precisions. An alternative way is to capture the semantic information to enhance performance automatically in which higher precision can be achieved by indexing semantic representation rather than keywords [19].

Many researchers nowadays are working on semantic representation of documents [20]. The past research had indexed concepts [21], word senses [22], topics [23], integrating word relationships into a language model [24] and thematic relationships between parts of texts [19]. Besides these studies, there are also studies which identify patterns in the relationships between terms and concepts contained in unstructured collections of text [25]. Meanwhile OpenEphyra QA system [26] used verb predicateargument for its knowledge representation. As for [27], the authors have used the logical structure of a document by utilizing the hierarchy of titles and paragraphs to extract the semantic relations between terms. Meanwhile, the authors in [28] have used first order logic representation in performing document indexing for its logical linguistic document retrieval system. Besides [28], the authors in [29] have used predicate in indexing for incremental multi-query optimization. On the other hand, the authors in [30] translated the first order logic representation to Skolem representation and used Skolem representation in indexing single documents. We have extended the work of [30] and introduced a new approach in indexing multiple documents in an effective way by using semantic Skolem indexing approach.

3.0 DESIGN OF FRAMEWORK FOR SEMANTIC INDEXING FOR QA

The creation of semantic index begins with the creation of Skolem representation. In our research, we are making use of Skolem clauses to create the semantic Skolem index. There are certain processes involved in order to create the Skolem representation. Skolem normal form is named after Thoraf Skolem who is a famous Norwegian mathematician known for his work on mathematic logic and set theory. The authors in [30] employ some similar techniques with our work. One of the major differences is that we use Skolem representation in a much broader context in which we have combined the Skolem representation that have been represented by each document into a single knowledge representation. In this section, an overall design of semantic indexing for QA framework is proposed as shown in Fig. 1. This framework comprises 3 main modules which are automatic lexicon generator, Skolem unification, and answer and document retrieval engine.

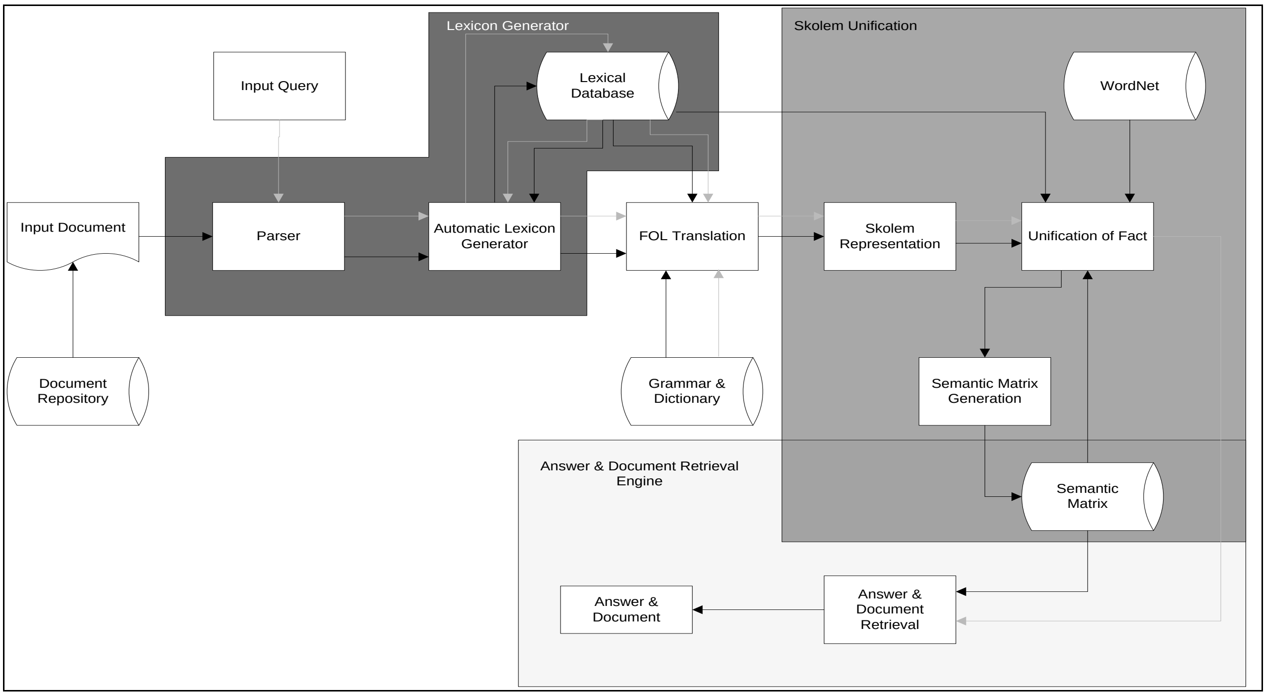


Fig. 1. Design of the semantic indexing for QA framework

Each input document obtained from the document repository will go through our lexicon generator module in generating the lexicon [31]. Once the lexicon is created, the natural language documents will be translated to first order logic(FOL). They will then be translated to Skolem representation. The authors in [30] have created Skolem representation for each of the documents and we have expanded this research to accommodate multiple documents. Once the Skolem are generated for each of the documents, the knowledge that exists in the form of Skolem constant needs to be integrated carefully. In order to perform this, Skolem unification module needs to be executed.

In this module, each of the Skolem representations for each of the documents will go through unification of fact process. A fundamental problem in mapping the input Skolem representation document to a semantic matrix is overcoming the high degree of ambiguity that exists between the sentences and the documents. To overcome this problem, unification of fact process incorporates our previous work [32] on Skolem preprocessing in which WordNet and lexical database had been used. This will ensure strong and elastic knowledge representation be built by overcoming redundancy and inconsistency issues. These issues aroused since we have to deal with similar contents which are represented with different sets of Skolem. In this kind of circumstances, information needs to be filtered before it gets loaded into the semantic matrix database. The unified Skolems will then be fed into the semantic matrix through semantic matrix generation process.

The answering engine will then use the semantic matrix database in retrieving the answer as well as the document source.

4.0 SEMANTIC SKOLEM INDEX CREATION

The Skolem representation for each of the documents has to be unified and integrated into a single knowledge representation in order to be used in the semantic retrieval process. We assume that S={$S_{1,}$ $S_{2,\dots}$ S_n} is the set of all unified Skolems used in indexing the documents from D={$d_{1,}$ $d_{2\dots..}$ d_n}. A document can be seen as a set of Skolem representations, i.e. $d_i=\{s_{1,} s_{2....,} s_k\}$ where s_j denotes the Skolem j in document d_i. Fig 2. shows the detailed steps that had been executed in unifying the Skolems by incorporating Skolem preprocessing and semantic matrix generation.

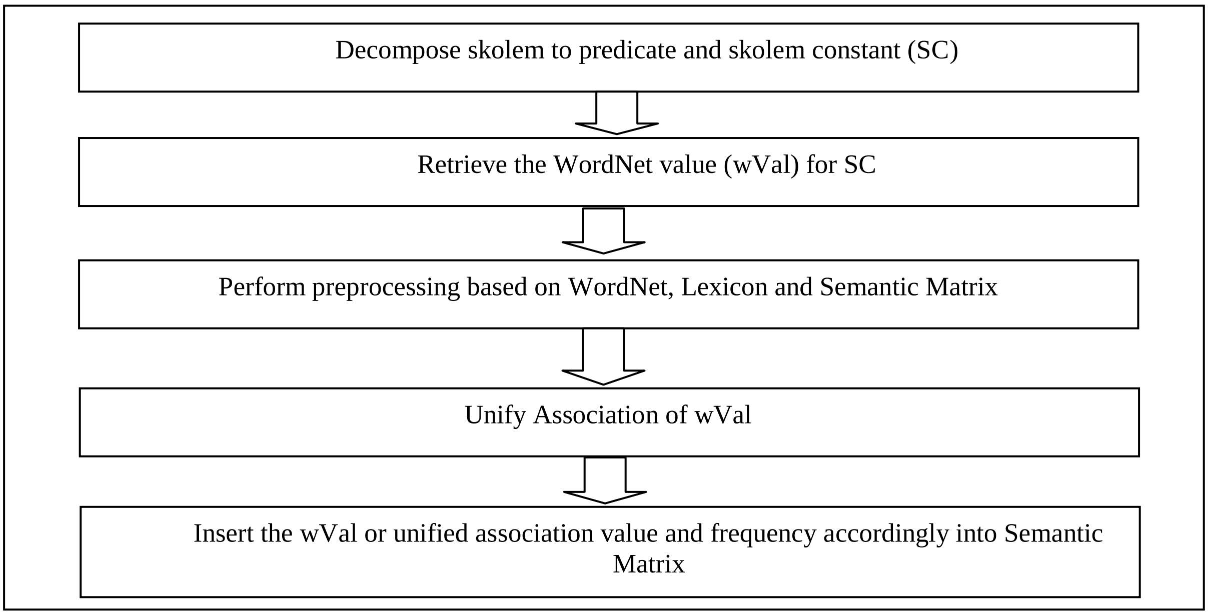

Decompose skolem to predicate and skolem constant (SC)
Retrieve the WordNet value (wVal) for SC
Perform preprocessing based on WordNet, Lexicon and Semantic Matrix
Unify Association of wVal
Insert the wVal or unified association value and frequency accordingly into Semantic Matrix

Fig 2. Steps for Skolem Unification and Matrix Generation

The semantic Skolem indexing that we have created managed to tie the associations that exist between the skolem representations for each document, and these associations become the ultimate information that helps in the retrieval process. The semantic Skolem indexing matrix that has been created is also scalable in parallel with the growth of the documents. This scalability feature of the semantic matrix enables the real-time environment's data in document form to be integrated automatically to it.

5.0 ILLUSTRATED EXAMPLE

Here is an example of small text documents that have been used for illustrative purpose. In this example, each document consists of 1 or 2 sentences. The sentences are represented in natural language.
Document1: The author writes books.
Document 2: Melissa composes books. Melissa composes these books to simplify complex algorithms.
Document 3: The writer composed booklets in English.
Document 4: The teacher books a room.
Document 5: Harry Porter was written in 1995.
Document 6: Writing is a tedious process.
Document 7: The teacher wrote few famous books. The teacher writes these books to show her views on teaching.
Document 8: An educator has composed few known booklets. The educator writes these booklets to present her perspective on pedagogy.

Each of the sentences will go through the parser, automatic lexicon generator and FOL translation process as shown in Fig. 1. Then, each of the FOL will be converted into a Skolem representation. The Skolem representation for each of the documents mentioned earlier (document 1 to document 8) and the unification of fact for each of the documents are shown in Table 1.

Table1. TheSkolemRepresentationforDocument 1 toDocument 8

Document Number	Skolem Representation	Unification of Fact

1	author(g47).
	book(g48).
	writes(g47,g48).
		author(f110090311)
		book(f106013091)
		writes(f110090311, f106013091)

2	book(g26).
	composes(melissa,g26).
	
	book(g26).
	composes(melissa,g26).
	complex(g27).
	algorithm(g27).
	simplifies(writes(melissa,g26),g27).
		book(f106013091)
		writes(melissa, f106013091)
		
		book(f106013091)
		writes(melissa, f106013091)
		complex(f302102223)
		algorithm (f105509072)
		simplifies(writes(mellissa, f106013091), a1)

3	writer(g48)
	booklet(g49).
	composes(g48, g49).
	in(composes(g48, g49),english).
		author(f110090311)
		book(f106013091)
		writes(f110090311, f106013091)
		in(writes(f110090311, f106013091), english)

4	teacher(g27).
	room(g14).
	books(g13, g14).
		teacher(f109997151)
		room(f103951013)
		books(f109997151,f103951013).

5	***writes(r(harry & porter),1995)***	***writes(r(harry&porter),1995).***

6	writing(g27).
	tedious(g3).
	process(g3).
	isa(g27,g3).
		writing(f100874197)
		tedious(f301298622)
		process(f100964359)
		isa(f100874197,a2).

7	teacher(g15).
	famous(g16).
	book(g16).
	writes(g15,g16).
	
	book(g17)
	her(g18).
	view(g18)
	teaching(g19).
	on (g18,g19).
	shows(writes(g15, g17), g18).
		teacher(f109997151).
		famous(f301328419)
		book(f106013091).
		writes(f109997151,a3)
		
		book(f106013091)
		her(f500000001)
		view(f105831732)
		teaching(f100834401)
		on(a4, f100834401)
		shows(writes(f109997151,f106013091), a4)

8	educator(g50).
	known(g51).
	booklet(g51).
	composes(g50,g51).
	
	booklet(g52)
	her(g53).
	perspective(g53)
	pedagogy(g54).
	on (g53,g54).
	presents(writes(g50, g52), g53).
		teacher(f109997151)
		famous(f301028419)
		book(f106013091)
		writes(f109997151, a3).
		
		book(f106013091).
		her(f500000001)
		view(f105831732)
		teaching(f100834401)
		on(a4, f100834401)
		shows(writes(f109997151,f106013091), a4)

6.0 EXPERIMENTAL SYSTEM DEVELOPMENT

The translation process that translated text documents to first order logic is done by using prolog. We have used pragmatic skolemization technique by using prolog as a tool in translating first order logic to Skolem representation. The process of indexing the Skolem clauses has been done using php and the index has been stored in mysql database. In retrieving the answer for the user's query, resolution theorem proving approach has been used [29]. The question will be used as a theorem to be proven in order to derive the answer which has been stored in the Skolem-document index matrix. Skolem clause binding approach as in [32, 33] has been used to bind all the interrelated Skolems together by the answer key.

7.0 EXPERIMENTAL RESULTS AND DISCUSSION

The matrix employs harvesting tools that access and index Skolem representations automatically. New and old Skolems are merged smoothly and indexed efficiently so that new information is disseminated as soon as it is created. The result in Table 2 shows the Skolem-document matrix where rows represent all the possible Skolem clauses. As for the columns, the first column of Table 2 shows the predicate that has been extracted from the Skolems. Second and the third columns of Table 2 shows the arguments extracted (Skolem constants) from the Skolems. The rest of the columns are specially meant to store the number of occurrences of each of the Skolems in the documents. These Skolem occurrences will be incremented if any of the Skolems co-occur within the same document. As for document 1, the predicate which will be inserted in this Skolem-document indexing matrix is 'writes' and arg1 and arg2 both will be retrieved from Wordnet (wVal) which are 'f110090311' and 'f106013091'. Since it occurs only once in document 1, the frequency of occurrences is stated as 1 as shown in the last column of Table 2. Meanwhile, there is inexistence of other predicates shown in Table 2, thus 0 is assigned to all the other predicates of document 1.

Table 2. Skolem-Document Indexing Matrix

predicate	**arg1**	**arg2**	**doc 8**	**doc 7**	**doc 6**	**doc 5**	**doc 4**	**doc 3**	**doc 2**	**doc 1**
writes	f110090311	f106013091	0	0	0	0	0	1	0	1
writes	melissa	f106013091	0	0	0	0	0	0	2	0
simplifies	writes(mellissa, f106013091)	**a1**	0	0	0	0	0	0	1	0
in	writes(f110090311, f106013091),	english	0	0	0	0	0	1	0	0
books	f109997151	f103951013	0	0	0	0	1	0	0	0
writes	r(harry&porter)	1995	0	0	0	1	0	0	0	0
Isa	f100874197	**a2**	0	0	1	0	0	0	0	0
writes	f109997151	**a3**	1	1	0	0	0	0	0	0
On	**a4**	f100834401	1	1	0	0	0	0	0	0
shows	writes(f109997151,f106013091)	**a4**	1	1	0	0	0	0	0	0

Table 3. Association Matrix

Association Value(AVal)	Skolem Constant(SC)
a1	f302102223, f105509072
a2	f301298622, f100964359
a3	f301328419, f106013091
a4	f500000001, f105831732

We took advantage of the association that lies between each of the Skolem representations from multiple documents in indexing it into a single knowledge representation. "a" value that has been highlighted bold as shown in Table 2 shows the association value that captures the relation of the SC. The association of Skolems is shown in Table 3 in which the SCare the Skolem constant(s) that are mapped to an AassociationValue(AVal). As for doc 2 which is shown in Table 2, it contains an association value of "a1" in its arg2. "a1" basically captures the wVal of SC for'complex(g27)' and 'algorithm (g27)'. Since the SC contains the same value of "g27", these two Skolems have been associated to an association value of "a1". These associations of SC incorporate the Skolem preprocesssing. Normalizing the SC with WordNet gives a standard value for each of the SCs as shown in columns 2 and 3 of Table 2. For an example, "complex" will retrieve the value of "f302102223" and "algorithm" will retrieve the value of "f105509072". The association of these two SCs to "a1" is shown in Table 3.

Besides taking advantage of the relation that exists between the Skolems, we have also tackled problems associated with redundant Skolems. The redundant representation of Skolems occurs when the same Skolem clause is being represented more than one time. Besides that, in some circumstances, redundancy may also occur in which different Skolem clauses that represent the same meaning occurs. For an example, "author writes books", and "writer composed booklets" have the same meaning although they are being represented differently as shown in the example mentioned above(Document 1 and Document 3). If the Skolems that represent "author writes books" has been added in the semantic index matrix, then the second statement's Skolem representation which represents "writer composed booklets" will not be considered as unique representation. As a consequence, the later Skolem which contains the same meaning will not be added as a new Skolem representation in the matrix. Thus, the final representation will only have the Skolem representation value of "author writes books" and this statement has been semantically represented in the matrix. The matrix also shows that document 1 and document 3 co-occur with the same Skolem clauses.

Table 4 shows the Skolem-Document indexing matrix for all the eight documents that does not include Skolem preprocessing during the unification of fact. The generated matrix is represented without the integration of WordNet and lexical database.

Table 4. Skolem-Document Indexing Matrix(Without Skolem Preprocessing)

predicate	arg1	arg2	doc8	doc7	doc6	doc5	doc4	doc3	doc2	doc1
writes	f1	f2	0	0	0	0	0	0	0	1
composes	melissa	f2	0	0	0	0	0	0	2	0
simplifies	writes(mellissa, f2)	a1	0	0	0	0	0	0	1	0
composes	f5	f6	0	0	0	0	0	1	0	0
in	composes(f5,f6)	english	0	0	0	0	0	1	0	0
books	f7	f8	0	0	0	0	1	0	0	0
writes	r(harry&porter)	1995	0	0	0	1	0	0	0	0
isa	f9	a2	0	0	1	0	0	0	0	0
writes	f12	a3	0	1	0	0	0	0	0	0
on	a4	f17	0	1	0	0	0	0	0	0
shows	writes(f12,f2)	a4	0	1	0	0	0	0	0	0
composes	f18	a5	1	0	0	0	0	0	0	0
on	a6	f21	1	0	0	0	0	0	0	0
presents	writes(f18,f6)	a6	1	0	0	0	0	0	0	0

Comparison has been made between both the matrixes (Table 2 and Table 4). Similar document content exists between document 1 and document 3, and document 7 and document 8. These documents represent the same meaning although they are represented with different sets of Skolems as shown in Table 1. All these similar documents have been represented in an effective manner as shown in Table 2 in which it caters for the similarity of these different Skolems. Duplication and overlapping Skolems are filtered more efficiently in the Skolem-document matrix with the integration of Skolem preprocessing. Besides that, since all the documents have been unified into a single representation, the matrix enables to show the semantically similar Skolems that co-exist in other documents. This information is very useful in identifying the similarity of Skolems across the documents. On the other hand, the Skolem-document matrix shown in Table 4 is represented independently without any similarities between the documents. This can be seen in the number of occurrences of Skolem that are represented in document 1 does not overlap with document 3, and document 7 does not overlap with Skolems in document 8.

The semantic matrixes generated as in Table 2 and Table 4 had been used as knowledge representation for our question answering system. Table 5 shows the answers, documents and frequencies of the answers for the questions posed based on two different knowledge representations (Skolem-document indexing matrix with Skolem preprocessing and Skolem-document indexing matrix without Skolem preprocessing).

Past research in information retrieval has dealt with matching the query content with the available documents together with the most appropriate fragments of these documents [34]. On the other hand, the authors in [30,35] managed to retrieve the exact answer in logic representation from single documents. As for our QA system, we managed to go a step ahead compared to other research in providing the user not only the answers for his/her queries from multiple documents, but the document in which an answer contains. This feature is provided as a proof for the answers according to the respected queries posed [3]. Here is an example of query posed by the user and the results obtained from both the matrixes are shown in Table 5.

Query: *Who writes book?*

Table 5. Comparison of Answers and Documents Retrieved for Semantic Skolem Indexing

Answer Retrieved from Skolem-document matrix(with SkolemPreprocessing)			**Answer Retrieved from Skolem-document Matrix(without SkolemPreprocessing)**		
Retrieved Skolems	Doc No	Frequency	Retrieved Skolems	Doc No	Frequency
writes(f110090311,f106013091)	1	1	writes(f1,f2)	1	1
	3	1			
			writes(f12,a3)	7	1
writes(melissa, f106013091)	2	2			
writes(f109997151,a3)	7	1			
	8	1			
Set of Skolem Clauses(Answer): *author* *melissa* *teacher*			Set of Skolem Clauses(Answer): *author* *teacher*		

Based on the results obtained, we managed to successfully prove that the new approach of semantic Skolem indexing matrix succeeded in enabling the QA system to retrieve answer(s) for the question(s) posed. However, the precision and recall increased by integrating the Skolem preprocessing into the semantic Skolem-Indexing matrix. Our QA engine managed to retrieve answers together with the source document in which an answer contains, as well as the frequency of occurrences of the Skolems from the matrix generated.

8.0 PERFORMANCE EVALUATION

This section evaluates the results of Skolem-document matrix for all the eight documents mentioned. As a whole, the number of predicates for Skolem-document indexing matrix which incorporates Skolem preprocessing contains

10 predicates compared to 14 without preprocessing as shown in Fig. 3. This gives 28.5% lesser predicates that are being stated in the matrix compared to the one without Skolem preprocessing.

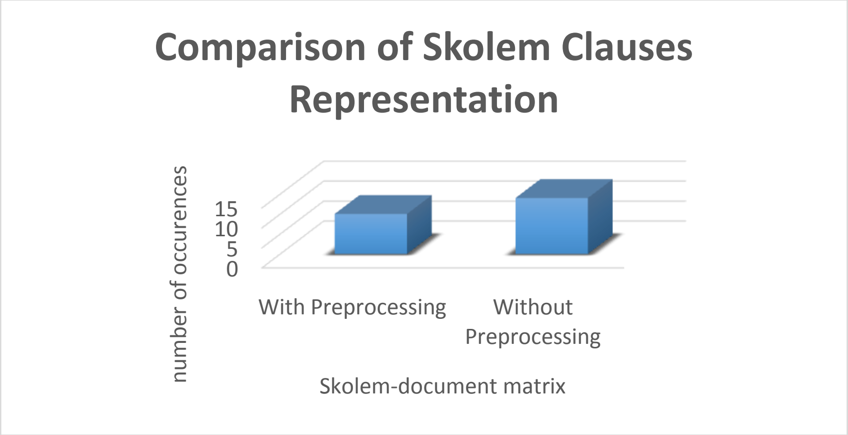


Fig. 3. Comparison of Skolem Clauses Representation

The evaluation of QA system is analyzed with two different knowledge bases: Semantic-document indexing matrix that incorporates Skolem preprocessing and Semantic-document indexing matrix without preprocessing. In order to perform this experiment, a familiar Remedia publications data set is used. Remedia publications data set is a famous data set which is used in QA system [35]. This data set is supplied by MITRE corporation for the purpose of research that contains reading materials for grade 3 to grade 6 [36]. The materials cover a broader range of topics with a length of 150-200 words for each document. A total of 20 documents at random have been used for the test set. Due to unavailability of grammar and inexistence of automatic grammar generation system that suits logical representation, the author has used a test set of 20 documents. A total of 68 queries have been posted into the QA system. Table 6 shows the statistics on the queries that are classified according to the question types. The percentage (%) shows the percentage of occurrences of each type of query that are used in the QA system in order to generate the required answer(s).

Table 6.Statistics on the queries

Question Type	**Number of Questions**	**Percentage(%)**
Who	14	20.58
What	13	19.12
When	14	20.58
Where	14	20.58
Why	13	19.12
Total Number of Questions	**68**	**100**

The results that are obtained from the QA system that works on each of the knowledge bases will be in the form of answer(s) and document(s) sources. Thus, the results of the experiments are compared to each knowledge base and evaluated based on the recall and precision metrics[37] of the answer retrieved as shown below.

$$\text{Recall}=|\{\text{relevant documents}\}\cap\{\text{retrieved documents}\}|/|\{\text{relevant documents}\}|$$

$$\text{Precision}=|\{\text{relevant documents}\}\cap\{\text{retrieved documents}\}|/|\{\text{retrieved documents}\}|$$

Two experiments were conducted in order to compare the results of semantic matrix with preprocessing and semantic matrix without preprocessing. The results shown in Table 7 states the recall and precision rates obtained for each of the “wh” question types that are posted to the QA system using Skolem without preprocessing index.

Table 7. Results of experiment 1(without Skolem preprocessing)

Question Type	**Recall(%)**	**Precision(%)**
Who	91.43	85.8
What	83.33	83.33
When	78.57	61.2
Where	75	71.43
Why	65.38	52.54
Overall	**78.74**	**70.86**

The highest recall result was obtained by “who” type of question in which more than 92% of recall has been achieved. This was followed by “what”, “when”, “where” and “why”. “Why” type questions have a reduction of more than 25% from the recall result obtained by “who” type of question. This is due to the higher complexities that are involved in “why” type questions compared to the other types of questions. The overall result achieved for recall is 78.74% compared to 70.86% for the precision.

The second experiment uses Skolem semantic matrix with Skolem preprocessing. Combining Skolem preprocessing in semantic matrix generation has shown promising results in the QA system as shown in Table 8.

Table 8. Results of experiment 2(with Skolem preprocessing)

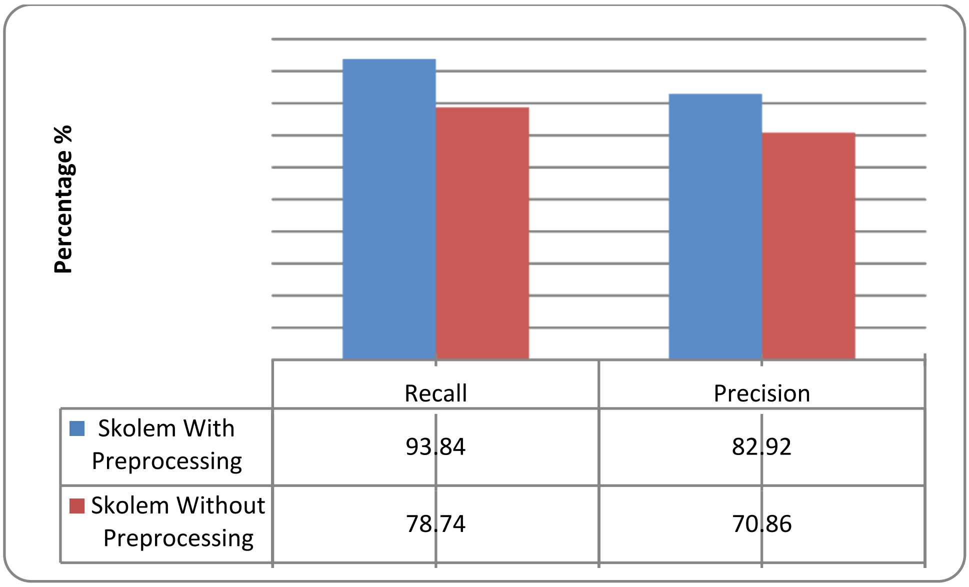
Question Type	**Recall(%)**	**Precision(%)**
Who	100	93.7
What	100	88.5
When	92.3	71.9
Where	100	96.4
Why	76.9	64.1
Overall	**93.84**	**82.92**

This integration enables the QA system to retrieve answer(s) correctly although with the semantically similar document content which is represented with different sets of words exist. There are 3 question types which managed to obtain 100% recall which are “who”, “what” and “where” type of questions. On average, the overall recall and precision for the questions are 93.84% and 82.92%.

Fig. 4 shows the comparison of recall and precision of two different knowledge bases that we have tested. The result shows that the recall of using Skolem with preprocessing knowledge base is higher than the one without the Skolem preprocessing. The recall of Skolem with preprocessing is 93.84% compared to 78.74% of Skolem without preprocessing. The difference obtained is 15.1%. On the other hand, precision also shows a better result compared to Skolem without preprocessing with a diffence of 12.06%.

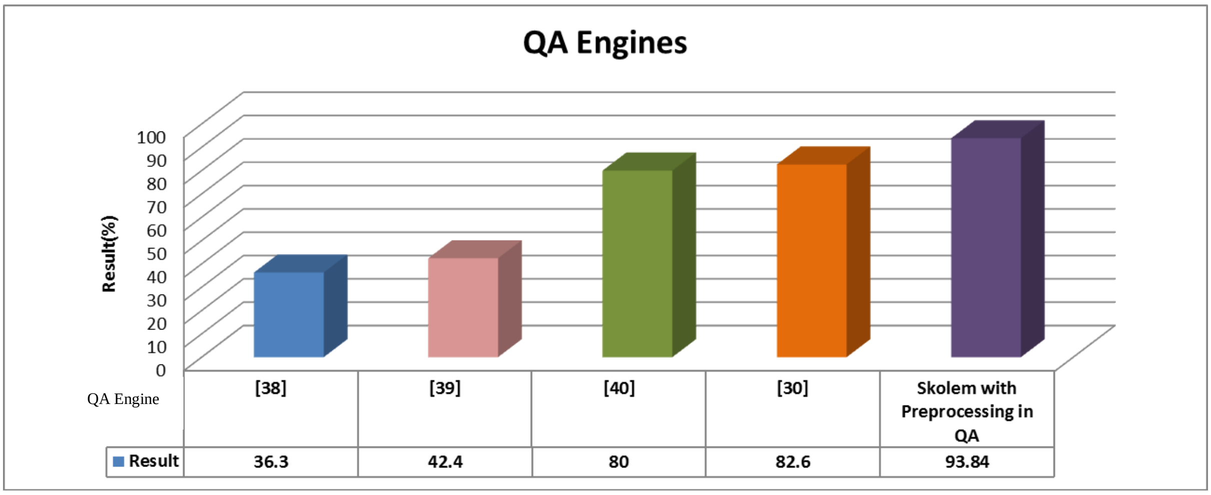
This indicates that by incorporating Skolem preprocessing in semantic matrix generation, the recall result increases to almost 20% and the precision increases by 12.06% compared to that without the integration. Similar answer with different sets of words representation managed to be retrieved by the integration of Skolem preprocessing. Thus, the results obtained are a good indication to reveal that the integration has produced fruitful outcome to the QA system.

The overall result obtained by using skolem with preprocessing in the QA system is compared with other QA engines. The summary of comparison of the QA engines is shown in Fig. 5.



	Recall	Precision
🟦 Skolem With Preprocessing	93.84	82.92
🟥 Skolem Without Preprocessing	78.74	70.86

Fig. 4. Comparison of Recall and Precision



QA Engine	[38]	[39]	[40]	[30]	Skolem with Preprocessing in QA
🟦 Result	36.3	42.4	80	82.6	93.84

Fig. 5. Comparison of QA Engines

MITRE corporation has reported to have attained 36.3% of accuracy in answering the questions from stories[38]. Meanwhile, The authors in [39] have used advance bag of word and bag of verb technique in implementing their QA system. Their system returns an accuracy of 42.4%. On the other hand, The QA system in [30,35] has successfully accomplished a significant result of 82.6% which is an increase of 40.2% compared to that proposed in [39]. Research work on QA engine in [40] managed to achieve 80% accuracy by using CLEF dataset. With the use of semantic indexing using Skolem with preprocessing, we managed to obtain 82.92% of precision and 93.84% of recall in extracting answers from the QA engine.

9.0 CONCLUSIONS

The authors have introduced a new approach in indexing multiple documents in an effective way by using Skolem representation. The knowledge representation which is in the form of semantic representation has been utilized by the QA engine. This research managed to prove that the semantic index is successfully used by the QA engine in order to retrieve answer and the source documents in which the answer contains. This capability managed to boost the users' confidence in using the QA system since the retrieval process managed to also show proof for the retrieved answer. On the other hand, the integration of Skolem pre-processing managed to deliver a better result in terms of indexing and answer retrieval. Recall of 93.84% and precision of 82.92% had been obtained with the integration preprocessing. Thus, it becomes a conclusive evidence to show that the knowledge representation which is in the form of semantic Skolem matrix succeeded as a reliable information provider. Although the information capture, translation and semantic matrix generation could involve some time during indexing, this tediousness had been compensated with higher recall and precision in terms of retrieval. The semantic matrix that has been created is also scalable in parallel with the growth of the documents. This scalability feature is very useful in accommodating the real-time documents.

Future research may enhance the capability of retrieving the most appropriate fragments of the retrieved documents. Besides that, future researchers may also consider in converting the answer that is retrieved by QA system. Since the authors had produced the answers in Pragmatic Skolem form, future researchers may work on transforming this representation into natural language sentences.

10. ACKNOWLEDGEMENT

The authors wish to thank University of Malaya Research Grant (RP04B-13ICT) for funding this research and the Knowledge Technology group (Center for AI Technology, FTSM, UKM) for assisting and giving the opportunity for the authors to conduct this research.

REFERENCES

[1] P. Clark&J.Thompson, "A Study of Machine Reading from Multiple Texts," in *AAAISpring Symposium on Learning by Reading and Learning to Read*, 2009.

[2] Hess, M., "Deduction over Mixed-Level Logic Representations for Text Passage Retrieval," in *International Conference on Tools with Artificial Intelligence (TAI'96)*, , Toulouse, France, 1999, pp. 383-390.

[3] Prince, V.& Labadi´, A., "Text segmentation based on document understanding for information retrieval," in *Applications of Natural Language to Data Bases*, 2007, pp. 295-30.

[4] Clark P. & Thompson J. A study of machine reading from multiple texts. In *Proceedings of AAAI Spring Symposium on Learning by Reading and Learning to Read*, 2009.

[5] Kang B. Y., Kim D. W & Lee S. J. Semantic indexing and fuzzy relevance model in information retrieval. *Computational Intelligence for Modelling and Prediction 2*, 2005, pp. 49-60.

[6] Al-Shalabi R. & Kanaan G. Constructing an automatic lexicon for Arabic language. *International Journal of Computing & Information Sciences 2(2)*, 2004, pp. 114-128.

[7] Liu G. Z. (1997). Semantic vector space model: Implementation and evaluation," *Journal of the American Society for Information Science 48(5)*, pp. 395-417.

[8] Balduccini M., Baral C. & Lierler Y.Handbook of Knowledge Representation. Knowledge Representation and Question Answering, Amsterdam: Elsevier, 2007.

[9] Sammut, C. Knowledge Representation Formalisms in Machine Learning, 2008.

[10] Stoyanchev S., Song Y. C. & Lahti W. Exact phrases in information retrieval for question answering," Coling 2008: *Proceedings of the 2nd workshop on Information Retrieval for Question Answering (IR4QA)*, 2008, pp. 9-16.

[11] Cai D. & Rijsbergen C. J. V. Semantic Relations and Information Discovery," in *Studies in Computational Intelligence*, vol. 5, 2005, pp. 79-102.

[12] Huang L. B.,Balakrishnan V. & Raj R. G. "Improving the relevancy of document search using the multi-term adjacency keyword-order model." *Malaysian Journal of Computer Science* 25.1, 2012, pp. 1-10.

[13] Egozi, O., "Concept-Based Information Retrieval using Explicit Semantic Analysis." vol. *Master of Science in Computer Science*: Israel Institute of Technology, 2009, pp. 1-80.

[14] Tengku Sembok, T.M, "SILOL: A simple logical-linguistic document retrieval system",*Information Processing and Management*vol. 26(1), 1990, pp. 111-134.

[15] Abramowicz W., Kaczmarek T. & W˛ecel K. How much intelligence in the semantic web?," *Advances in Web Intelligence* 3528, 2005, pp. 1-6.

[16] Jamal, N., et al. Poetry Classification Using Support Vector Machines.*Journal of Computer Science***8**(9), 2012, pp.1441-1446.

[17] Hoenkamp, E.& Dijk, S.V., "A Fingerprinting Technique for Evaluating Semantics Based Indexing," *Lecture notes in computer science*, vol. Volume 3936/2006, 2006, pp. 397-406.

[18] Zambach, S., "A Formal Framework on the Semantics of Regulatory Relations and Their Presence as Verbs in Biomedical Texts," 2009.

[19] Ceglarek, D. & Rutkowski W., "19 Automated Acquisition of Semantic Relations for Information Retrieval Systems," *Technologies for Business Information Systems*, 2007, pp. 217-228.

[20] Bounhas, I. & Slimani, Y., "A hierarchical Approach for Semi-Structured Document Indexing and Terminology Extraction," in *International Conference on Information Retrieval and Knowledge Management*, 2010.

[21] Woods W. A. Lunar rocks in natural English: Explorations in natural language question answering." *Linguistic structures processing* 5, 1977, pp. 521–569.

[22] Mihalcea R. & Moldovan D. Semantic Indexing using WordNet senses. In *Proceedings of ACL Workshop on IR & NLP*, 2000, pp. 35-45.

[23] Dou, W., Yu, L., Wang, X., Ma, Z., & Ribarsky, W. Hierarchical topics: Visually exploring large text collections using topic hierarchies. *Preliminary accepted to IEEE VAST*, *2*, 2013.

[24] Cao G., Nie J. Y. & Bai J. Integrating word relationships into language models. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR '05)*, 2005, pp. 298-305.

[25] Park L. A. F., Leckie C. A., Ramamohanarao K. & Bezdek J. C. Adapting spectral co-clustering to documents and terms using Latent semantic analysis, *Lecture Notes in Computer Science* 5866, 2009, pp. 301-311.

[26] Schlaefer N., Ko J., Betteridge J., Sautter G., Pathak M. & Nyberg E. Semantic extensions of the Ephyra QA system for TREC 2007, *The Sixteenth Text REtrieval Conference (TREC)*, 2007.

[27] Abdul Kadir, R., Tengku Sembok, T.M. & Halimah, B.Z., "Towards Skolemize Clauses Binding for Reasoning in Inference Engine," in *Fifth International Conference on Computational Science and Applications*, 2007.

[28] D. Cai & C. J. v. Rijsbergen, "Semantic Relations and Information Discovery," in *Studies in Computational Intelligence*. vol. 5, 2005, pp. 79-102.

[29] Jin, C. and Carbonell, J. Predicate indexing for incremental multi-query optimization. *Foundations of Intelligent Systems,*2008, pp. 339-350

[30] Abdul Kadir, R., Tengku Sembok, T.M. & Halimah, B.Z.,: "Improvement of document understanding ability through the notion of answer literal expansion in logical-linguistic approach",*WSEAS Transactions on Information Science and Applications***6(6) ,** 2009, pp. 966-975.

[31] Varathan, K. D., Sembok, T. M. T., & Kadir, R. A. Automatic Lexicon Generator for Logic Based Question Answering System. In *Computer Engineering and Applications (ICCEA), 2010 Second International Conference on* Vol. 2, 2010, pp. 349-353.

[32] Varathan, K. D., Sembok, T. M., Kadir, R. A., & Omar, N. Skolem preprocessing using WordNet and lexicon in building effective knowledge representation. In *Electrical Engineering and Informatics (ICEEI), 2011,* pp. 1-5.

[33] Varathan, K. D., Tengku Sembok, T.M., Abdul Kadir, R. & Omar, N., "Retrieving Answer from Multiple Documents Using Skolem Indexing," *International Conference on Semantic Technology and Information Retrieval*, 2011.

[34] Prince, V. & Labadi´, A., "Text segmentation based on document understanding for information retrieval," in *Applications of Natural Language to Data Bases*, 2007, pp. 295-30.

[35] Abdul Kadir. Question answering for reading comprehension using logical inference model. Phd thesis, National University of Malaysia, 2008.

[36] Kara, S., Alan, Ö., Sabuncu, O., Akpınar, S., Cicekli, N. K., & Alpaslan, F. N. An ontology-based retrieval system using semantic indexing. *Information Systems, 37*(4), 2012, pp. 294-305

[37] Moghadasi, S. I., et al. Low-cost evaluation techniques for information retrieval systems: A review.*Journal of Informetrics,***7**(2), 2013, pp. 301-312.

[38] Hirschman, L., & Gaizauskas, R. Natural language question answering: The view from here. *Natural Language Engineering, 7*(4), 2001, pp. 275-300

[39] Bashir, A., Kantor, A., Ovesdotter C. A., Ripoche, G., Le, Q., & Atwell, S. Story Comprehension. Retrieved April 11, 2005 from http://12r.cs.uiuc.edu/~danr/Teaching/CS598-04/Projects/First/Term2.pdf , 2004.

[40] Waltinger, U., Breuing, A., & Wachsmuth, I. Interfacing virtual agents with collaborative knowledge: Open domain question answering using wikipedia-based topic models. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, 2011, pp. 1896-1902.