Sādhanā (2019) 44:62
https://doi.org/10.1007/s12046-018-1022-8

A framework for intelligent question answering system using semantic context-specific document clustering and Wordnet

K KARPAGAM[1,*] and A SARADHA[2]

[1]Department of Computer Applications, Dr Mahalingam College of Engineering and Technology,
Pollachi 642003, India
[2]Department of Computer Science and Engineering, Institute of Road and Transport Technology, Erode, India
e-mail: karpagam80@gmail.com; saradha.irtt@gmail.com

MS received 12 January 2017; revised 11 June 2018; accepted 4 September 2018; published online 18 February 2019

Abstract. The question answering system plays an important role in information retrieval field, where the user is in need of getting a precise answer instead of large collections of documents. The aim of this paper is to investigate techniques for improving sentence-based question answering system. To achieve this, a POS-Tagger-based question pattern analysis model is proposed to identify question type based on pattern template for the user-submitted query. Next, the knowledge base is created from a large corpus by clustering the documents by grouping on domain context. The proposed semantic-word-based answer generator model deals with the user query mapping with an appropriate sentence in the knowledge base. By the proposed models, the system reduces the search gap among user queries and answer sentences using Wordnet. It considers word order, overlap, sentence similarity, string distance, unambiguous words and semantic similarity of words. The proposed algorithm evaluates with benchmark datasets such as 20Newsgroup and TREC-9 QA, and proves its efficiency by statistical test for significance.

Keywords. Document clustering; POS-Tagger-based question analysis; mean average precision; semantic similarity; Wordnet.

1. Introduction

The amount of data in web resources and their needs are growing enormously day by day. In the real scenario, the information searched by the user is lost by its way due to large collection of documents. To overcome this issue, the role of intelligent question answering (QA) system is evolved. QA system is one of the major application areas of information retrieval techniques. The QA system is composed of three main modules: they are question processing, information retrieval and information extraction [1]. These techniques aim at producing short, precise answers based on the semantic and syntactic relations among documents and also similar document grouping and co-occurrence of keywords. The QA systems are categorized into open-domain QA and closed-domain QA. The open-domain QA system deals with the queries and answers that are independent in nature of any domain. Closed-domain QA systems are able to deal with questions and answers of specific domain like commercial, education, music, weather forecasting, tourism, medical health, etc. [2]. Document clustering is a technique that organizes text documents into meaningful clusters or groups. It has two approaches, namely traditional approach and semantic approach. Traditional document clustering approach uses Bag of Words model to generate the clusters by finding the frequency of keywords occurring in each document. *K*-Means is the most popular clustering algorithm that groups the given data objects into *K* number of clusters depending on similarity/dissimilarity between the data. As a result, similar documents are placed in the same cluster and dissimilar documents are placed in different clusters. The major disadvantage is ignoring the semantic relationship among the words that leads to insignificant documents clusters and also it is not able to discriminate between two different clusters. The semantic document clustering is a technique used to group the documents into meaningful clusters that are semantically related to each other, which helps easily to map with user query. The proposed method enhances the grouping of clusters by adding semantic and syntactic similarity for grouping. This paper is

*For correspondence

organized as follows. Section 2 discusses the related works; section 3 deals with system architecture, question pattern analysis, knowledge base building and semantic-relation-based document clustering; section 4 deals with the experimental results compared to the existing models. Section 5 gives an evaluation of the system for information retrieval. Conclusion and future works are provided in section 6.

2. Related works

The major challenges of information retrieval system are about search space, response time, sentence length, word mismatch, overlap, order and word ambiguous among the user queries and the candidates answers. To overcome this dispute of information retrieval system, the following techniques are considered such as semantic similarity technique using Wordnet, translation language model, query like-hood model, machine learning, artificial intelligence, supervision/non-supervision-based learning models, ranking model, etc. by various subject experts. The learning model is trained and tested with social-medium-based QA pairs such as Quora, Stack Overflow and Yahoo! Answers. In paper [3], authors discussed the learning on question classifiers for factoid QA, which is able to provide the answers for Wh-type questions like What, Where, Which and When from various knowledge sources. Paper [4] converses about the system analyses on user question received in natural language. A Stanford POS-Tagger, parser for Arabic language, employs numerous detection rules and a trained classifier for answering the question. Paper [5] discusses the simple language modelling technique called query likelihood retrieval, which is considered for sentence retrieval, and proves that it outperforms TF-IDF for ranking sentences. Comparisons of sentence retrieval techniques such as topic-based smoothing, dependence models, relevance models and translation-based models are discussed. In paper [6], a model for answer representation for long answer sequence and passage answer selection with deep learning models is proposed. The results are evaluated with TREC-QA and Insurance QA datasets. The passage-level QA system produces answers by text summarization for the complex questions from different documents. The author proposes a deep learning hybrid model with convolution and recurrent neural networks for passage-level question and answer matching with semantic relations. In paper [7], the procedure is incorporated to transform the Wordnet glosses with logical forms in first order and position of words as arguments in syntactical information. It inculcates the knowledge about the role of Wordnet glosses in performing better QA systems. The contribution of this paper is to (i) propose a POS-Tagger-based question pattern analysis (T-QPA) model for question-type identification, (ii) create a domain-based knowledge base, (iii) develop a semantic-word-based answer generation model, (iv) achieve state-of-art results on both TREC-9 QA and 20Newsgroup dataset and (v) statistical test for significance.

3. System architecture

Communication among the system and users is through user-initiated interface by providing the question in natural language. Normally, search engines use the keywords search for retrieving the relevant documents from the knowledge base. Likewise, QA system acquires input as user query in the form of natural language, then identifies question types and extracts keywords from the question. Next, it matches with the relevant documents, paragraphs and sentences for query and extracts the most appropriate candidates answers. It also ranks the retrieved sentences and displays the top ranked sentences as candidates answers.

The proposed system architecture is shown in figure 1.

3.1 *Question pattern analysis model*

The question classification phase of the proposed framework is to develop the learning model for question-type identification. The POS-Tagger of Stanford University is considered for pattern formation because it is found to be the best in identifying the grammatical structure of the sentence such as nouns, verbs, adverbs and adjectives [8]. Using POS-Tagger, pattern for the

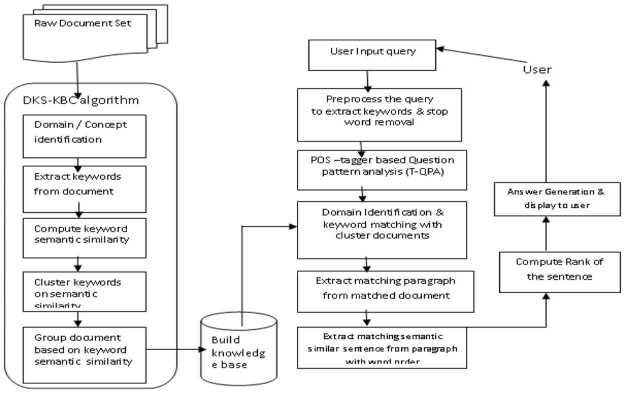

Raw Document Set
DKS-KBC algorithm
Domain / Concept identification
Extract keywords from document
Compute keyword semantic similarity
Cluster keywords on semantic similarity
Group document based on keyword semantic similarity
Build knowledge base
User Input query
User
Preprocess the query to extract keywords & stop word removal
POS –tagger based Question pattern analysis (T-QPA)
Domain Identification & keyword matching with cluster documents
Extract matching paragraph from matched document
Extract matching semantic similar sentence from paragraph with word order
Answer Generation & display to user
Compute Rank of the sentence

Figure 1. System architecture.

question is formulated and the learning model with structured question patterns is trained. The knowledge of the intelligent QA system is based on the learning model that uses the supervised approach to roll out the exact answer. A set of 1000 questions with positive and negative tagging, for example, do and dont, are given as inputs, which in turn identifies question type of user query input.

The question pattern is formed using the grammatical structure for each type of questions. The question types include Evaluative Question (Q_{EV}) (what, why, when, where, which), Choice Question (Qch), Hypothetical Question (Qhp), Confirmative/Rhetorical question (Q_{RC}) and non-Factoid Question (Q_F), which return qualitative/quantitative information based on their question pattern.

The POS-Tagger algorithm of Stanford University identifies the question type using pattern template and it is shown in Algorithm 1.

For example, User Question: What country was Mahatma Gandhi born? POS-Tagger result: In —IN what —WP country — NN was — VBD Mahatma — NNP Gandhi — NNP born — NN, question type: Q_{EV}, answer type: country, domain: politics. By incorporating the T-QPA model, the proposed system outperforms in identifying question patterns along with positive and negative question tags in producing efficient results. The user input question Q analysed by the learning model of T-QPA and the question type identified along with positive and negative tagging are as shown in table 1.

3.2 *Knowledge base building model*

The knowledge base is built using the 20Newsgroup dataset, which acts as a source of documents consisting of different domain information types such as politics, entertainment, sports, etc. This dataset is pre-processed with Apache OpenNLP library, which supports the NLP tasks such as tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing and co-reference resolution [8].

Table 1. Questions with positive and negative tagging.

Text	Question type	Description
Do you know the correct distance between Coimbatore and Chennai?	Confirmative question	Positive tag
How much mark is required to pass in an internal exam ?	Evaluative question	Positive tag
Isn't cold outside due to winter?	Choice question	Negative tag

Algorithm 1: Algorithm for POS-Tagger-based Question Pattern Analysis (T-QPA) Model

```
Let Q be the User Question
Split Q using POS-Tagger;
Analyse tags based on the patterns below
Let $Q_p$ be the question pattern identified
Evaluative Question ($Q_{EV}$)

    i   What returns a statement or definition or an explanation.
    ii  Why returns a reason/cause of an action.
    iii When returns time and date information
    iv  Where returns location information item Who, whos, whose, whom, represents a person
    v   Which returns information about a thing?

Choice Question ($Q_{ch}$)-Or returns one of answers from the choice/options
Hypothetical Question (Q-hp) – What and if, if and what Pattern refers to a hypothetical
question and returns an imaginary/probabilistic answer.
Confirmative/Rhetorical question ($Q_{RC}$)

    i   Any user question ending/starting with any of the following tags are found to be Confirmative/Rhetorical question
    ii  /MD (or) /VB (or) / VBZ (or) /VBN (or) / VBG (or) /VBD (or) / VBP + /PRP
    iii /MD (or) /VB (or) / VBZ (or) /VBN (or) / VBG (or) /VBD (or) / VBP+ /RB +/PRP

Non-Factoid Question ($Q_F$)
How refers to a factoid question, which returns a Qualitative/Quantitative information.
The questions with any of the following pattern are factoid questions
/IN (or) /RP (or) /TO + /WDT
/IN (or) /RP (or) /TO + /WP
/IN (or) /RP (or) /TO + /WRB
```

The role of WordNet is used to find the semantic similarity among the sentences and query relationships. Rita Wordnet is used to provide various utility functions for annotating the corpus and positive and negative word separation and word preposition. It is also used for analysing different words with the same meaning like good, better, nice and best and for finding similarity between grammar of same words like sing and sang [9, 10]. Based on the grammar, context and semantic similarity, the keywords in the document are grouped together as clusters to form the knowledge base. The POS-Tagger splits each sentence into nouns, verbs, adjectives and adverbs in the document. From the extracted split words, the stem words and noun words are taken into account for indexing, domain grouping and categorization for faster cluster formation [11].

The keyword can be extracted using the empirical formula given below (1):

$$KW_{\mathrm{N}} = \sum_{i=1}^{N} Ext(Pos\_noun(d_i)) \qquad (1)$$

where $\mathrm{KW_N}$ is the number of keywords extracted from the document d_i of dataset.

These extracted words are stored in the database as keyword id, keyword along with its document index to map the query keyword with the document. Then the keywords semantic similarity computed with Wordnet is used to identify how the keywords are similar to each other and occurrence of the related keywords available in all documents [12].

The similarity computation can be performed using (2):

$$sim(x,y) = \frac{1}{3}\left(\frac{m}{l1} + \frac{m}{l2} + \frac{m-n}{m}\right) \qquad (2)$$

where m is the matching characters, n is the misplaced characters and $l1$, $l2$ are the length of the two words.

The group index G_{N} and keyword index k_{i} can be computed using the following formula (3):

$$[G_{\mathrm{N}}, k_{\mathrm{i}}] = max_{\mathrm{j=1:N}}\left(sim\left(KW_{\mathrm{j}}, KW_{\mathrm{j+1}}\right)\right) \qquad (3)$$

where G_{N} is group index and k_{i} is keyword index.

The algorithm proposed for Knowledge base building model is shown in Algorithm 2. After execution of algorithm 2, keywords are extracted from documents and are grouped together based on the context of documents. The group count, which is dynamically changed according to query keyword, leads to dynamic clustering of documents, which improves the efficiency. This dynamic clustering also supports in adding new keyword documents and forming a tree structure for easy retrieval. Recently updated documents are inserted at appropriate places in the tree structures. Based on the keyword similarity value, the documents are grouped into sizes of clusters. For example, the given $k = 4$ and denoted as G1–G4 is shown in table 2.

Table 2. Documents grouping based on semantic similarity of keywords.

ID	Keyword	Group
1	Gallery	G1
2	Tree	G1
1773	Art	G2
2134	Ballet	G3
3877	Role	G4

The similarity table is generated from the set of given documents, domain grouping, group id and keyword id. Information is extracted from unstructured text from the Internet based on the grammar, context and semantic similarity; the keywords in the document are grouped together as clusters to form the structured knowledge base. Hence, these groups are loaded into the knowledge base.

Algorithm 2: Domain-specific keyword-similarity-based knowledge-based creation (DKS-KBC) algorithm

```
(Domain-specific Keyword-similarity-based knowledge-based creation (DKS-KBC) algorithm)
Let $R_x$ be the Set of Concepts/Domains based on which the Question/ Answer model is designed, where x denotes the set of Domains/Concepts.
Let D be the Database to be loaded as resource of the Answering model.
Let Dc Donate the Documents in the Database.
Let $N$ be the No. of Documents in the D.
For $i$ = 1 to $n$
Extract keywords from the document.
Let $Kw_i$ be the set of keywords extracted from $Dc_i$
Analyse $Kw_i$
Compute Similarity $Sim_K$ ($Dc_i$) between the $Kw_i$ and keywords in $Dc_i$
Where $i$ = 1 to $n$
Split documents based on the $Sim_K(Dc_i)$
Load $Kt_n \leftarrow (Kw_i, ind_{Kw})$
Where $Kt_n$ is the table that holds the keywords with index
Group keywords based on keyword Similarity.
Load $R_x$ ($Dc_i$, $Kw_i$, $G_n$)
Continue until $i$ = $n$
End for
Load set of $R_x$ to the knowledge base.
End while
```

3.3 *Information extraction*

The final phase of proposed framework is semantic-word-based answer generator (SWAG) model. The conventional methods have the limitation of finding answer boundaries and recognizing the desired type of information and answer size. This can be overcome by the semantic and syntactic QA analysis with pattern matching. The system takes the user input in the form of natural language, pre-processing using NLP techniques such as tokenization, stop words removal, stemming, noun-phrase identification, parsing, etc. It determines the question type from the T-QPA model by applying POS-Tagger on the input query. The system maps the user's query with the answer sentences using machine learning techniques. The question type focuses on text chunks to retrieve matching query keywords for providing the answers to user questions. Keywords extraction from the user query is performed using (4):

$$Q_K = Ext(Keywords(Q)) \quad (4)$$

where Q_K is the keyword extracted from user query.

For example, for the given input query What is da vinci code?, the extracted keywords are what, da, vinci and code. The keywords are matched with the domain-specific groups of clusters to identify the possibility in which the group candidate answer resides. If the keywords are not available in the grouped clusters, an extensive search is made and semantic similar keyword extraction is performed [9]. Semantic keywords extraction is performed using Eq. (5):

$$Q_K = \sum_{i=1}^{N} Q_i + sem(Q_i) \quad (5)$$

where $sem(Q_i)$ is the semantic similar keyword from the documents using Wordnet.

The role of WordNet is used to find the semantic similarity among the sentences and query relationships. It is widely used as an online dictionary and thesaurus for English words for improving text quality by analysing semantic relation among the terms. It is an online lexical database designed for finding English nouns, verbs, adjectives and adverbs organized into sets of synonyms. Semantic relations link the synonym sets for the related words. For example, code is checked with other terms such as code, codification and computer code for the semantic word. This brief analysis on terms facilitates in empirical comparison of terms for efficient results. The extracted keywords are compared to terms in grouped clusters and related list of documents is generated. The domain has been identified by comparing the query keyword and group id using the following formula (6):

$$D_m = \begin{cases} 1 & \text{if}(\max_{i=1:N})\text{Comp}(Q_K, G_N), \\ 0 & \text{elsematchnotfound.} \end{cases} \quad (6)$$

After finding the query keyword and matching domain, calculate the number of occurrences of query keyword in the related domain. The domains with maximum occurrence of context, semantic similar keywords are calculated. The number of query keyword occurrences within the domain documents cluster is calculated. Maximum counts of keyword occurrences and related domains are matched and recognized. For the query example What is da vinci code?, the query keyword related matched group cluster G1 and entertainment domain have been identified for the retrieving answer candidates with keywords with 9 occurrences. Query keyword is matched with the domain documents in the clusters and the retrieved domain based on occurrences is shown in table 3.

The extracted keywords with the domain group and the number of occurrences are checked for semantic and syntactic similarity. It checks using the Wordnet dictionary, when similar words are not found in the group clusters. This process is to identify the most relevant documents with candidate answers, and maintain the list of documents with number of keyword occurrences. The related document is chosen using formula (7):

Table 3. Query-keyword-based domain retrieval with count.

Keywords ID	Matching domain query	Occurrence
da	Entertainment	3
vinci	Entertainment	3
code	Entertainment	3
code	Politics	1
code	Technology	1

$$l_m = D_m(G_N, k_i) \quad (7)$$

where l_m denotes the list of the matched documents for query, D_m is the matched domain, G_N denotes the group number and k_i denotes the query keyword.

The lists rank of matched documents is based on their maximum number of query keywords occurrence in the document. The shilling coefficient is used to compute document similarity based on keywords context. It is used for text analytics of similarity between two documents. The cosine similarity is calculated for finding the semantic relatedness between the words with the summing of the vectors of all words in the text.

The resultant Sim_{Tab} consists of similarity value among query and the matching document for paragraph identification. Average paragraph score is calculated using a threshold of all the paragraphs. The paragraphs are extracted by adding the individual sentence scores. From the paragraphs, answer sentences are retrieved by calculating the matching degree between sentence and question keywords using Eq. (8):

$$keyword_{sim} = \frac{Keyword(Q) \cap Keywords(C)}{Keywords(Q)} \quad (8)$$

where Q is the keywords in the question and C is the number of keywords in the sentence.

From the set of candidates, answers produced are ranked according to likelihood of correctness. The top ranked two sentences are extracted from the documents/paragraph taken for similarity analysis. The machine learning technique defines the features for each n-gram in the sentence and for each n-gram the parts of speech tags are predicted.

Sentences similarities are obtained by analysing the context feature of the keyword in sentence and also by identifying the ambiguous words, i.e., the same word with different meanings based on context. For example, the sentences retrieved as answers for Where was Mahatma Gandhi born? are mapped with the correct answer based on the context such as birthday place, religion, year, etc. Another example is: train, How to train the slow learning student? and train, When does the train arrive at Delhi?.

The proposed algorithm for semantic-word-based answer generator (SWAG) model is shown in Algorithm 3.

Algorithm 3: Proposed algorithm for semantic-word-based answer generator (SWAG) model

```
Let Q be the user Question
Split Question to tags by POS-Tagger
Update $Q_L$ ← Keywords from Q
Remove Stop Words from $Q_L$
Load $R_x$,
Let $D_m$ be the domain/ concept in $R_x$
Let $Kw_m$ be the keyword list in $D_m$
Let $M_{list}$ be the matched keyword list
For $i = 1$ to $n$ where $N$ is the number of keywords in $Q_L$
If ($Kw_m$ contains ($Q_L$(i)))
Update $M_{list}$ ← ($D_m$,$Kw_m$)
Until $i = n$
End if
End For
Sort $M_{list}$ based on maximum keywords matched with domains
Get maximum matching domain, Max($D_m$)
Update User Search Domain $US_D$ = Max($D_m$)
For $i = 1$ to $n$, where $n$ is the number of keywords extracted from the question.
Extract Semantic word ($Sem_{wd}$) for keywords in $Q_L$
Update $Q_L$ ← $Sem_{wd}$($Q_L$(i))
Until $i = n$
End for
Load the groups ($G_n$) in the $US_D$;
Count=0;
For $i = 1$ to $n$ where $n$ is the number of keywords in $G_i$
If ($G_i$contains ($Q_L$(i)))
Count++;
Update $G_{List}$ ← ($G_i$, count);
Where $G_{List}$ is the list-containing group and the number of keywords available.
Until $i = n$
End if
End For
Update document matching group = Max($G_{List}$);
Load List of documents present in the $G_{List}$.
Compute document similarity with the question to the contents in the document.
For each paragraph in document, get similarity value upload to $Sim_{Tab}$
Get result = Max($Sim_{Tab}$)
Extract and compute the matching degree between the sentence and question keywords, number of mutual keywords between each sentence and questions.
Display the answer candidates
```

The answer generation of candidate sentences is restricted to length of 50–250 bytes from the top rated sentences. The answer is displayed to the user through the interface and user task is to rate the correct answers. The answer representation is performed through the imperative modelling and also the proposed system is evaluated. From the top ranked match documents, relevant paragraph and sentences are extracted considering the features such as word order, sentence similarity, string distance and unambiguous words [13]. The framework dealing with unanswered queries, reducing the search space for the complex question and eliminating non-relevant document and sentences enhances the response time and efficiency of the system. Managing unanswered queries and unpredictable queries is handled by accepting the user answer choice and updating the answers in knowledge base for future usage. This increases the productivity of the proposed framework.

4. Experimental results

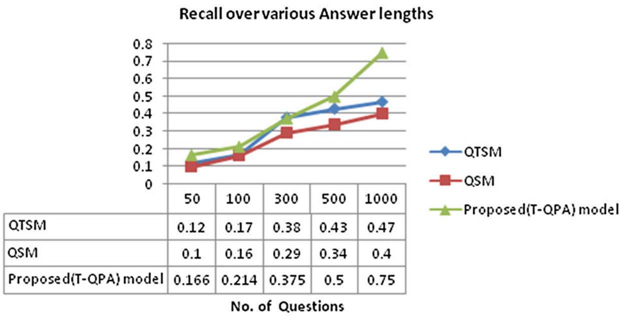
4.1 *Datasets*

The evaluation of the proposed methods is carried out with the benchmark 20Newsgroup dataset with 500 raw data documents collected from UCI machine learning repository [14]. In the 20Newsgroup dataset, raw documents were extracted into five domains like sports, entertainment, politics, etc. for easy retrieval of data. The synthetic question set framed and tested against the learning model with 50 questions of each type is considered for the significance test purpose against 250 documents.

The TREC-9 QA is taken from [15], which was submitted to Microsoft Encarta encyclopedia. TREC -9 QA consists of newspaper and newswire documents collection from various sources such as APnewswire, Financial Times, Los Angeles Times, etc. The TREC-9 QA dataset consists of attributes such as question id, question, document id and judgment answer string. TREC-9 QA deals with semantic similarity for keywords using WordNet and answer tagging with major class labels such as name, time, number, human and earth entities.

The recall or true positive rate is calculated as the ratio of the number of correct positive predictions to the total number of true positive and false negative question predictions. The learning model is tested with 100 questions of each type and the system is evaluated for performance on average retrieves in 20 documents/170 sentences relevant to each topic extract and are processed for precise answer. Labelled question are identified and classified for 320/500 questions. WH questions are identified and classified for 232/500 questions.

The proposed T-QPA method provides the enhanced result in question pattern identification and compares with the existing methods Question-Type-Specific Method



	50	100	300	500	1000
QTSM	0.12	0.17	0.38	0.43	0.47
QSM	0.1	0.16	0.29	0.34	0.4
Proposed(T-QPA) model	0.166	0.214	0.375	0.5	0.75

Figure 2. Recall over various answer lengths.

(QTSM) and Question-Specific Method (QSM) [16]. The result for 500 questions are T-QPA > QTSM, QSM, i.e., $0.5 > 0.43, 0.3$. The T-QPA model result for question pattern prediction is compared to those of the existing models as shown in figure 2.

The experiment results were evaluated with the standard internal measures as mean average precision, accuracy, $F1$, missrate and fallout. The mean average precision is one of the popular performance measures in the field of information retrieval. It is used to evaluate the rank of retrieved relevant documents with the average precision values. It is calculated using Eq. (9):

$$MAP = \frac{1}{n} + \sum Q_{\mathrm{i}} \frac{1}{R_{\mathrm{i}}} \sum_{D_{\mathrm{j}} \subseteq R_{\mathrm{i}}} \frac{j}{r_{\mathrm{ij}}} \tag{9}$$

where n is the number of test questions, r is the rank of the jth relevant document Dj in Qi and Ri is the relevant document for Qi. The mean average precision value is increased by the proposed SWAG algorithm for sentence retrieval from the word clusters formed with 20Newsgroup and TREC-9 QA dataset. It enhances the retrieval rate, which ranges from 0.39 to 0.42, using baseline, AQUAINT-bigram and Google-bigram as seen in figure 3.

The mean average precision resultant value on word occurrences in the sentence is increased from 0.35 to 0.40 in case of applying DKS-KBC algorithm for 20Newsgroup dataset. The result is shown in figure 4.

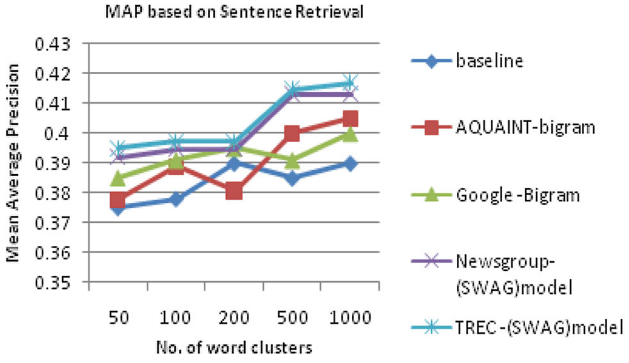


Figure 3. Mean average precision based on sentence retrieval with number of word clusters.

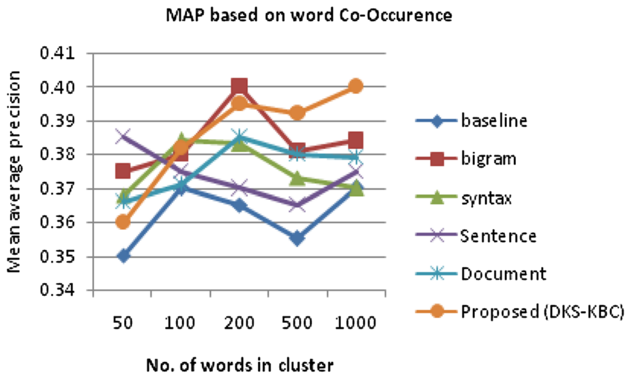


Figure 4. Mean average precision based on word co-occurrence with number of word clusters.

4.2 *ANOVA test*

The ANOVA (ANalysis Of VAriance) is a statistical test for significance used to compare two or more groups to find significant differences between them. ANOVA assumes the following information about the data: all observations are mutually independent and sample populations have equal or unequal variance [17]. The mean values of the groups are significantly same or different from another. The one-way ANOVA form of model is calculated using Eq. (10):

$$y_{\mathrm{ij}} = \alpha_{\mathrm{j}} + e_{\mathrm{ij}} \tag{10}$$

where y_{ij}, the score of observation matrix in each column, represents different domain group clusters; α_{j} is a matrix with domain, which means that α_{j} applies to all rows of the jth column; e_{ij} is a matrix with random disturbance.

The parameters in ANOVA tables considered for significance analysis are (i) sum of square (ss) of each source, (ii) mean square (ms) of source, (iii) F-value for mean squares, (iv) P-value and (v) degree of freedom (df) associated with sources.

The reason behind performing ANOVA test is to test whether there is any significant difference in retrieval of domain-based answers for given user query by the proposed algorithm. The null hypothesis for ANOVA test is stated as absence of significant difference in retrieval of domain answers and alternate hypothesis is there is a significant difference in retrieval of domain answers. The results of ANOVA test are shown in tables 4 and 5.

In the case of 20Newsgroup dataset, Prob $> F$ is not satisfied; hence the null hypothesis is accepted and alternate hypothesis is rejected. However, in the case of TREC-9 QA dataset, Prob $> F$ is obtained; hence the null hypothesis is rejected and alternate hypothesis is accepted. From the observed results, empirical comparison in performance

Table 4. ANOVA table for SWAG algorithm on 20Newsgroup dataset.

Source	SS	df	MS	F	Prob>F
Columns	53.8	4	13.45	0.65738	0.62479
Error	920.7	45	20.46		
Total	974.5	49			

Table 5. ANOVA table for SWAG algorithm on TREC-9 QA dataset.

Source	SS	df	MS	F	Prob>F
Columns	44.0833	1	44.0833	0.33993	0.57278
Error	1296.8333	10	129.6833		
Total	1340.9167	11			

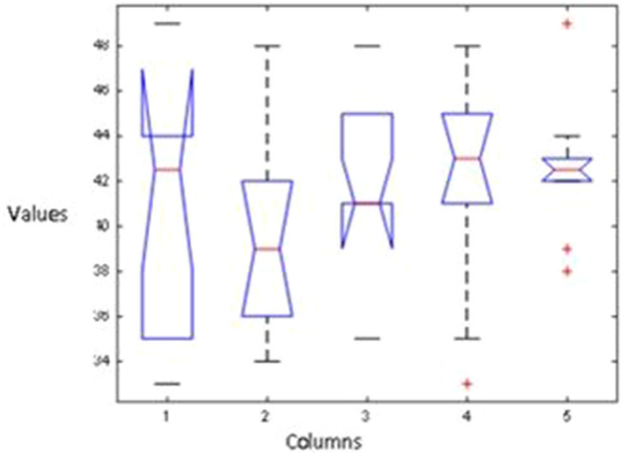


Figure 5. Box plot view of ANOVA test for 20Newsgroup dataset.

accuracy of algorithm is achieved, which varies for different datasets due to its nature.

The box plot view of answer convergence for user queries on 20Newsgroup and TREC- 9 QA datasets is shown in figures 5 and 6, respectively.

If the null hypothesis is accepted, there is no significant difference in the information retrieval with reference to domains or there is no significant influence of domains on the information retrieval. Here, in all constraints where Prob $> F$ and the the null hypothesis is rejected, the accuracies of proposed algorithms are comparatively good.

5. System evaluation

In QA systems, the question identification, candidates answer ranking and appropriate answer validation for given user query are evaluated with standard metrics. The algorithms are implemented using Java on an Intel 2.30-GHz i5 with 4-GB RAM. The system is evaluated with 1000 questions consisting of all question types; it is capable of retrieving candidate answers from 500 raw documents. The results of proposed system are calculated using the standard measurements precision, recall and F-measure for accuracy of the defined inference answers. The accuracy values are calculated with TP – true positive, TN – true negative, FN – false negative and FP – false positive. The true positive rate (TPR) is a measure of the proportion of positive documents correctly identified from the group of raw documents. It is calculated using (11):

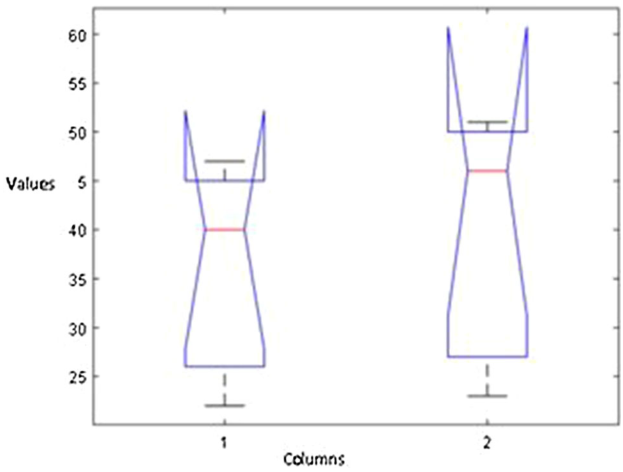


Figure 6. Box plot view of ANOVA test for TREC-9 dataset.

$$TPR = TP/TP + FN. \tag{11}$$

The false positive rate (FPR) is the proportion of all negative values obtained in positive test outcomes, i.e., the conditional probability of positive test results given. It is calculated using (12):

$$FPR = FP/FP + TN \tag{12}$$

where TP means true positive, FN means false positive and FP means false positive.

Precision is calculated as the number of correct positive predictions divided by the total number of positive predictions [18]. Precision for QA system is calculated as relevant document intersection with retrieved document divided by retrieved document. The best precision is 1, whereas the worst is 0. Precision is calculated as the true positives divided by the sum of false positives and true positives. It evaluates the retrieved answer on how it is relevant to the input query. It is calculated using Eq. (13):

$$precision = relevant\ docs \cap retrieved\ docs/retrieved\ docs. \tag{13}$$

Recall is calculated as the number of correct positive predictions divided by the total number of true positives and false negatives. Recall for QA system is calculated as relevant document intersection with retrieved document divided by relevant document. It evaluates on the answers, whether the system retrieved many of the truly relevant documents? The best sensitivity is 1.0, whereas the worst is 0.0. It is calculated using Eq. (14):

$$recall = relevant\, docs \cap retrieved\, docs / relevant\, docs. \tag{14}$$

Accuracy refers to the closeness of a measured value to a standard or known value with the weighted arithmetic mean of precision. It is calculated using Eq. (15):

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{15}$$

The performance analysis is measured for each query run in TREC-9 QA dataset. It is carried out for 50 queries in iterations and the average precision values are analysed. The test is also performed for the same number of queries with different runs. The precision is calculated against the number of documents retrieved from the set of 5, 10, 15, 20, 30, 100, 200, 500 raw documents and the result is shown in tables 6 and 7. The result statistics for the range of questions with possible retrieved answers for both datasets are discussed and shown in tables 6 and 7.

Experimental results show that using information from the external corpora, framework produces imperative improvements on question pattern identification, dynamic document clustering based on domain context, especially on datasets with short documents [19]. The result analysis for the query input “what is da vinci code? ” is validated and verified for correctness of answers and the result is shown in table 8.

The randomly selected questions from the TREC-9 QA extract answers based on keyword matching by also considering semantic and syntactic similarity of terms using Wordnet. Initially five selected answers are ranked; after eradicating incorrect answers, the algorithm filters pinnacle (top) two answers for displaying results to the users. The result of proposed algorithm is compared to the reference paper results [7] and shown in tables 9 and 10.

Table 6. Top *N* list of possible answers for TREC-9 QA dataset.

No. of queries (*Q*)	Precision	Recall	Accuracy
5	0.9230	0.6410	0.7210
10	0.8510	0.5330	0.6540
100	0.6260	0.4210	0.5340
200	0.6510	0.5300	0.5640
500	0.4540	0.3160	0.3670
1000	0.5230	0.3810	0.4370

Table 7. Top *N* list of possible answers for 20Newsgroup dataset.

No. of queries (*Q*)	Precision	Recall	Accuracy
5	0.7330	0.5230	0.5940
10	0.8660	0.5530	0.6750
100	0.7120	0.4280	0.5310
200	0.6330	0.3870	0.4810
500	0.8330	0.4310	0.5680
1000	0.7660	0.5750	0.6570

Table 8. Result of query sample: what is da vinci code?

Parameters	Values	Parameters	Values
True positive	4	True positive rate	0.6666
True negative	23	False positive rate	0.0415
False positive	1	True negative rate	0.9583
False negative	2	False negative rate	0.3333
Precision	0.8		
Recall	0.6666		
Accuracy	0.9		

Table 9. TREC-9 QA results obtained by the Moldovan systems.

Questions	Dan I Moldovan Initial rank	(Existing) Final rank
Q074	2	1
Q331	2	1
Q381	5	1
Q481	3	1
Q640	2	1

Table 10. TREC-9 QA results obtained by the SWAG proposed algorithm.

	SWAG model (proposed)	
Questions	Initial rank	Final rank
Q074	2	1
Q331	4	2
Q381	5	1
Q481	2	1
Q640	3	2

6. Conclusion and future work

This paper has investigated the techniques on mining candidate answers with less response time to improve a sentence-based QA system. An intelligent question–answer system is proposed with T-QPA model, domain-context-based knowledge base creation and semantic-word-based answer generator model for retrieving the answers. Further, the proposed SWAG model performance is trained and tested with TREC-9 QA and 20Newsgroup datasets. The resultant top ranked sentence is displayed as the answer. In the evaluation with standard metrics and significance test, the proposed SWAG model provides most desirable results and is found to outperform in a variety of strong baselines.

Moreover, a further enhancement is to optimize the results to increase better response with deep analysis of single-word, long-sentence questions and comparative questions in a profound manner.

References

[1] Lin J 2007 Is question answering better than information retrieval? Towards a task-based evaluation framework for question series. In: *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 212–219

[2] Yang H, Chua T S, Wang S and Koh C K 2003 Structured use of external knowledge for event-based open domain question answering. In: *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, pp. 33–40

[3] Li X and Roth D 2002 Learning question classifiers. In: *Proceedings of the 19th International Conference on Computational Linguistics, Association for Computational Linguistics*, vol. 1, pp. 1–7

[4] Ahmed W and Babu A P 2016 Question analysis for Arabic question answering systems. *Int. J. Natl. Language Computing* 5(6): 21–30

[5] Balasubramanian N, Allan J and Croft W B 2007 A comparison of sentence retrieval techniques. In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, pp. 813–814

[6] Tan M, dos Santos C, Xiang B and Zhou B 2016 Improved representation learning for question answer matching. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 464–473

[7] Moldovan D I and Rus V 2001 Logic form transformation of wordnet and its applicability to question answering. In: *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pp. 402–409

[8] https://nlp.stanford.edu/. Accessed date Feb 23, 2003

[9] Christos B and Vassilis T 2012 A clustering technique for news articles using WordNet. *Knowl. Based Syst.* 36: 115–128

[10] Liu S, Liu F, Yu C and Meng W 2004 An effective approach to document retrieval via utilizing WordNet and recognizing phrases. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, pp. 266–272

[11] Momtazi S and Klakow D 2015 Bridging the vocabulary gap between questions and answer sentences. *Inf. Process. Manag.* 51(5): 595–615

[12] Jeon J, Croft W B and Lee J H 2005 Finding semantically similar questions based on their answers. In: *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, pp. 617–618

[13] Severyn A, Nicosia M and Moschitti A 2013 Building structures from classifiers for passage reranking. In: *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, ACM, pp. 969–978

[14] http://qwone.com/∼jason/20Newsgroups/. Accessed date Jan 14, 2008

[15] http://trec.nist.gov/data/qamain.html. Accessed date Nov 12, 2000

[16] Wu Y, Hori C, Kashioka H and Kawai H 2015 Leveraging social QA collections for improving complex question answering. *Comput. Speech Lang.* 29(1): pp. 1–19

[17] Smucker M D, Allan J and Carterette B 2007 A comparison of statistical significance tests for information retrieval evaluation. In: *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*, ACM, pp. 623–632

[18] Kolomiyets O and Moens M F 2011 A survey on question answering technology from an information retrieval perspective. *Inf. Sci.* 181(24): pp. 5412–5434

[19] Pavli M, Han Z D and Jakupovi A 2015 Question answering with a conceptual framework for knowledge-based system development node of knowledge. *Expert Syst. Appl.* 42(12): pp. 5264–5286