International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-2, December, 2019

ACTSMLT: Automatic Classification of Text Summarization using Machine Learning Technique

Ramya R S, Darshan M, Sejal D, Venugopal K R, Iyengar S S, Patnaik L M

Abstract— ***In today's world, due to the steep rise in internet users, Community Question Answering (CQA) has attracted many research communities. In order to provide the correct and perfect answer to the user asked question from a given large collection of text data, understanding the question properly to suggest a precise answer is a challenging task. Therefore, Question Answering (QA) system is a challenging task than a common information retrieval task done by many search engines. In this paper, an automatic prediction of the quality of CQA answers is proposed. This is accomplished by using five well known machine learning algorithms. Usually, questions asked by the user are based on a topic or theme. We try to exploit this feature in our work by identifying the category of the question posted and further map with the corresponding question. Similarly, for the answers posted by the multiple user's are processed as answer for category mapping. Here, the results show that for Question Classification (QA), Linear Support Vector Classification (LSVC) is found to be the best classifier and Multinomial Logistic Regression (MLR) is the most suitable for Answer Classification (AC). The MS Macro dataset is used as the underlying dataset for retrieving and testing the question and answer classifiers. The Yahoo Answers are used as a golden reference during the testing throughout our experiments. Experiments results show that the proposed technique is efficient and outperforms Metzler and Kanungo's (MK++) [1] while providing the best answer summary satisfying the user's queries.***

Keywords: Question answering, Answer biased summaries, Information Retrieval, Classification, Document summarization.

I.INTRODUCTION

Social Question Answering (SQA) sites such as quora, yahoo, stack overflow have become a significant and huge database in to extract knowledge. Millions of internet users depend on SQA websites to obtain the best answers to their asked queries (question). Question Answering (QA) task has recieved a great attention in the last few decades. It is a sophisticated form of Information Retrieval (IR) and also one of the significant natural form of user computer system interaction.

Revised Manuscript Received on December 30, 2019.

* Correspondence Author

Ramya R S*, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India.

Darshan M, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India.

Sejal D, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India.

Venugopal K R, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India. E.mail: rs.ramya.reddy@gmail.com.

Iyengar S S, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India.

Patnaik L M, Department of Computer Science and Engineering, UVCE, Bengaluru-560001, India.

In the traditional information retrieval techniques, the whole documents were considered to be relevant to the user request. However, in question answering task, only a relevant piece of information are extracted to the user as an answer. The answer may be in the form of a sentence, a phrase, a term or even a paragraph. QA task have become a significant directions in natural language processing. It explores a correct and concise answer to a question from a large set of textual data. Document summarization are of two categories i.e., abstrac-tive and extractive summarization. Abstractive methods deals with compression and reformulation of text sentences. The main drawback of this method is that implementation is a challenging task. In extractive method, the important sentences are extracted from the collection of document and provides only a short sentence summary to the users. Extractive text summarization are classified into three types. 1) Extracting position of the sentence in the documents, 2) Unsupervised method and 3) Supervised method. In the first method, only significant sentences in the documents containing introduction and conclusion parts are extracted and given as a final summary. However, this method can be applied on only a few documents. In unsupervised methods, the sentences are ranked by the scores obtained based on the statistical feature and only top sentences are mined to generate the summary. Some of the models that are associated with unsupervised learning method are Integer Linear Programming (ILP). Markov Random Walk (MRW), Vector Space Model (VSM), Language Modelling (LM) for extracting relevant sentences from whole documents. On the other hand, in supervised learning method a collection of training documents are considered along with their manually created summaries to train the classifiers in order to predict whether the predicted sentences should be text summary to provide to users. Some of the models that are associated with supervised method are Bayessian Classifier (BC), Gaussian Mixture Model (GMM), Conditional Random Field (CRF), to capture the feature of sentences and rank these sentences. From the aforementioned literature survey, it is found that any text mining classification problem has been handled by machine learning and neural network algorithms and it is proved in solving any text classification problem. Our work mainly focus on supervised learning method. Five different classifiers are utilized namely Naive Bayes Classifier, Multinomial Naive Bayes, Bernoulli Naive Bayes Classifier, Logistic Regression and Linear Support Vector Classifiers to solve the drawback of the existing works that depend on human workers to rate the best answer out of four to five multiple answers for the given question.

It reduces the intensive human workers by using Machine Learning (ML) algorithms to learn and extract features from collection of documents automatically to provide the best answer. Machine learning (ML) algorithms have shown significant performance on text summarization speech and visual object recognition with increased amount availability of computation and documents. ML can be very effective for various Natural Language Processing (NLP) tasks such as sentence classification semantic embedding word embedding to name a few.

A. Motivation:

The existing text summarization methods [1], [2] focus on exploiting summaries from Community Question Answer (CQA). The CQA quality is analysed on different levels by Crowd Flowering (CF) approach. The quality of answer is col-lected manually by submitting a pair of (question, best answer) to CF workers and were asked to assess both question and answer relevance for the given queries. Manually rating the answer is time consuming and very costly to implement when the size of the questions increase. Motivated by this problem our work focuses on automatically predicting the quality of Question and answers using machine learning techniques for the user query.

B. Contributions:

The main contributions of this work are as follows.

1) We utilize machine learning methods namely Naive Bayes Classifier, Multinomial Naive Bayes, Logistic Re-gression and Linear Support Vector classifiers to auto-matically classify the Question and answer.
2) Selecting the best question and answer classifiers. Select-ing the optimal answers from a set of feasible answers that are already summarized.

C .Organisation:

The rest of the paper is organised as follows: Section II introduces a detailed overview of related works. Section III defines the background of Text summarizatioin. The proposed ACTSMLT Framework is presented in Section IV. Section V discusses the performance evaluation. Finally Section VI contains the conclusions and future work.

II RELATED WORK

Kolomiyets *et al.,* [3] designed a general question answer-ing architecture. The architecture defines question answering process as an information retrieval task. Question is given in natural language and the data queried can be of any type consisting of text, images, video and audio. Searches are reduced to term-based, and knowledge based term is queried with structured or logical queries and answers are obtained through reasoning. The question answering system may fail to return a suitable set of answers at an acceptable level of accuracy.

Chiang *et al.,* [4] aims to generate snippets as a detailed overview for the given query on the search-result page. The approach makes use of community based question-answering website for the initial extraction of the attributes from the categories, later generates the snippets based on how salient a sentence is, to the meaning of the query and its category. Experimental results demonstrate that the combining commer-cial search engine and CQA website approach is more effective when compared to other existing methods. The method does not include the extraction of attributes from search engine logs and Wikipedia.

Wu *et al.,* [2] have proposed Question-Type-Specific Method (QTSM). It extracts knowledge from the repositories to enhance sentence-based complex QA systems and uses social question-and-answer (QA) websites to store knowledge. Social QA collections are explored automatically and for every question type a question-type-specific classifier is created to filter out noisy sentences before the selection of the answer. The approach fails to handle complex questions adequately. Yulianti *et al.,* [5] introduced Tweet-Biased Summary (TBS). The technique uses the concept of query-biased sum-marization and extractive summarization that can be applied to the Twitter microblog domain. Different summaries that are generated without using tweets called Generic Summary (GS) are compared with TBS. The results of the experiments show that the quality of the summary of a web page can be improved by even a single tweet. Web pages that are pointed to the tweets can only use TBS technique. Asur *et al.,* [6] constructed a linear regression model to fore-cast box-office revenues for movies to be released. The model makes effective use of social media feeds like tweets from the twitter site to predict box-office outcome. The methodology can also be applied to a large collection of topics, ranging from the E-commerce product rating to election consequence prediction. Since there are many users sharing their opinions and experiences *via* social media, there are limitations as viewpoints subject to change with time. Losada *et al.,* [14] aim to make a detailed work on the outcomes of the strategies used for query expansion for retrieving sentences and also present's an excellent statistical query expansion method. The experiment assesses different term selection methods and provide a fact-based documentation to show that performance that can be improved if expansion is done before sentence retrieval. Query expansion before the sentence retrieval is more effective when handling poor queries and efficient as well. The proposed method is advantageous in the sentence retrieval only if the amount of relevant sentences in the set of documents that are ranked is not extremely low. Leal Bando *et al.,* [12] [15] have investigated the techniques for the generation of query-biased summaries including significant terms, sentence position, sentence length and query expansion techniques. The method concludes that the query expansion techniques can be effective in choosing the sentences for generating query-biased summaries preferably at the summary level than at the sentence ranking level. The study does not evaluate the summaries composed of only two

TABLE I
COMPARISON OF DOCUMENT SUMMARIZATION TECHNIQUES

Sl.no.	Authors	Model	Concept	Advantage	Disadvantage
1.	Yulianti *et al.*, [1] 2018	Optimization methods and a learning-to-rank method	Summarizes the document for Answering Queries	Accuracy of summaries is better than CQA answers	Approach is effective only for CQA answers and are focused in answering the user queries
2.	Ren *Et al.*, [7] 2018	Sentence relation-based summarization (srsum) model	Automatically learn features contained in sentences	Modelling csr, tsr, and qsr relations is useful for query-focused summarization	Complex neural network architecture
3.	Liu *et al.*, [8] 2018	Palp (personalized prediction)	Predict the activity Levels of users of CQA websites	Improved learning efficiency and prediction accuracy	Information channels incorporated with the study are less
4.	Chen *et al.*, [9] 2018	Essence Vector (Ev) model Denoising essence vector (D-Ev) model	Paragraph Embedding framework	Framework is robust	Cannot be applied for information retrieval and language Modelling
5.	Yulianti *et al.*, [5] 2018	Tweet-biased summarization	Query-biased summarization	Improves the quality of the summary	It is used for web pages pointed by tweets.
6.	Williams *et al.*, [10] 2016	Gesture model + query-session model	Use user's gestures that provide signals to differentiate between good and bad abandonment	Models canautomatically identify good abandonment in mobile search	Additional information on the search engine result page are not shown in the screenshot, that may have satisfied the user
7.	Baralis *et al.*, [11] 2016	Itemset-based summarizer	Presented the results of A summary Evaluation experience in an e-learning context	Supports individual and collective learning activities in a real context	Summarizing learning documents ranging over different subject are not supported
8.	Y Wu *et al.*, [2] 2015	Question-type-specific method (qtsm)	Techniques for mining knowledge from Social question answer websites	Filters out noisy sentences before the selection of the answer.	Complex questions cannot be adequately handled
9.	Leal Bando *et al.*, [12] 2015	Rocchio-s And Local Context Analysis (LCA)	creating subcollections by bucketing sentences of similar length	Effective in the selection of sentences at the summary level	Do not include evaluation for summaries composed of only two sentences.
10.	Chiang *et al.*, [4] 2014	Integer Linear Programming (ILP)	Generates Snippets for the given query	Combining commercial search engine and CQA website approach is more effective	Do not extract the attributes from search engine logs and wikipedia.
11.	Soricut *et al.*, [13] 2006	Answer/question translation model	Evaluates Question Answering (QA) system beyond factoid questions	Domain or type of the questions to be handled are not restricted	Robust Non-factoid QA is not achieved

sentences. The summary of recent research works along with their advantages and disadvantages is given in TableI.

Shrout *et al.*, [16] illustrated six forms of the intraclass correlation for reliability studies and also guidelines for choos-ing among six different forms. The attempt is to give a set of guidelines for researchers who use intraclass correlations. The six forms guides many researchers who are not aware of the differences between the various versions of the intraclass correlation coefficient. The discussion has been limited to a relatively pure data analysis case.

Williams *et al.*, [10] proposed a solution to the challenge faced by the search providers i.e., user satisfaction especially for the users using mobile devices. The solution is based on the signals provided by the gesture interactions that dif-ferentiate between positive and negative abandonment. The work suggests an analysis of the relationship between user gesture characteristics and fulfilment of user needs. A model is developed through this analysis to automatically identify good abandonment in mobile search . The conclusions of the observations are presented only with screenshots of the mobile Search Engine Result Page (SERP) and hence are not able to swipe, if there is an additional information on the SERP (that are not shown in the screenshot). The summary of different summarizations techniques, context and types based on multiple documents are summarised in Table II.

III. BACKGROUND WORK

Yulianti *et al.*, [1] developed three optimization methods namely question biased, answer biased and expanded question biased to extract the answer biased summaries from

a collection of text documents. An external data from CQA content is retrieved to the direct the answer summaries that are retrieved from text corpus. Inorder to interpret the extracted CQA information, a crowdsourcing service is incorporated to gather the quality of the answers. This sort of judging the quality of CQA answers manually is a time and cost consuming. Our work focuses on utilizing the machine learning techniques to predict the best answers.

TABLE II SUMMARY OF THE TEXT SUMMARIZATION CONTEXTS AND MODELS

Sl.no	Authors	Year	Model	Dataset used	Summarization technique	Based on no of document	Summarization contexts	Summary types	Evaluation methods	O/p type	Approaches Used
1	*Ren et al.*, [7]	2018	Sentence Relation based summarization(srsum)	Duc 2001,	Extractive summarization	Multiple document	-	Generic summaries	Rouge	-	Deep neural networks
2	*Liu et al.*,[8]	2018	Personalized prediction model	Stack overflow	-	-	personalised	-	-	Indicative	Logistic regression
3	*Chen et al.*,[9]	2018	Denoising essence vector model	Duc 2002	Extractive	Multiple document	-	-	Rouge	Inform ative	Clustering
4	Yulianti et al.,[1]	2018	Query CQA and Expanded query biased	webap	Extractive	Multiple documents	Web summar Ization	Query Focused	Rouge	Inform ative	Frequency based Approach
5	Yulianti et al.,[5]	2016	Twitter based summary	Twitter dataset from tree 2011	Extractive	Multiple document	Web summar Ization	Generic summar ization	Rouge-n, Rouge-1 Rouge-su	Inform ative	Graph Methods For Summary zation
6	L. Yang et al.,[17]	2016	Learning to rank model	Wedap	Extractive	Multiple document	Web summar Ization	Query Focused	NDCG Mrr	Inform ative	Machine learning
7	Baralis et al.,[11]	2016	Temset based model	Duc 05 suc	Extractive	Multiple document	E-learning Context	Topic based	Rouge-2 and Rouges-u4	Inform Ative	Sentence Based
8	Wu et al.,[2]	2015	Question type specific method	Ntcir 2008	Extractive	Multiple document	Web summar ization	Query Focused	N-gram overlap	Inform Ative	Machine learning
9	Leal Bando et al.,[12]	2015	Sentence ranking approach	Tree novelty	Extractive	Multiple document	Web summar ization	Generic Summaries	Manual	Inform ative	Frequency based Approach
10	Fei wu et al.,[18]	2015	Non-parametric Bayesian model	Duc 2003and Duc2004	Extractive	Multiple documents	Multimedia su mmarization	Generic Extraction	Rouge-1 and rouge-1	Inform ative	Frequency based Approach
11	Jae hyun park et al.,[19]	2015	Statistical translation model	Yahoo 148.102 Qa pairs	Extractive	Multiple document	-	Query Focused	prec@1 recall@5	Inform ative	Machine Learning
12	Chengying liu et al.,[20]	2015	Incrests algorithm	Facebook	Extractive	Multiple documents	Social network Services	Topic Based	F measure, N-gram	Inform ative	Cluster based
13	Chiang et al.,[4]	2014	Integer linear programming model	Pubmed	Extractive	Single document summariz ation	Web summar ization	Query Focused	Rouge-n	Inform Ative	Graph Method For Summar ization
14	Losada,et al.,	2010	Pseudo relevance feedback	The tree novelty track	Extractive	Single document summariz ation	Web summary zation	Query Focused	F measure, precision at ten sentences retrieved	-	Frequency Based (tf/idf)

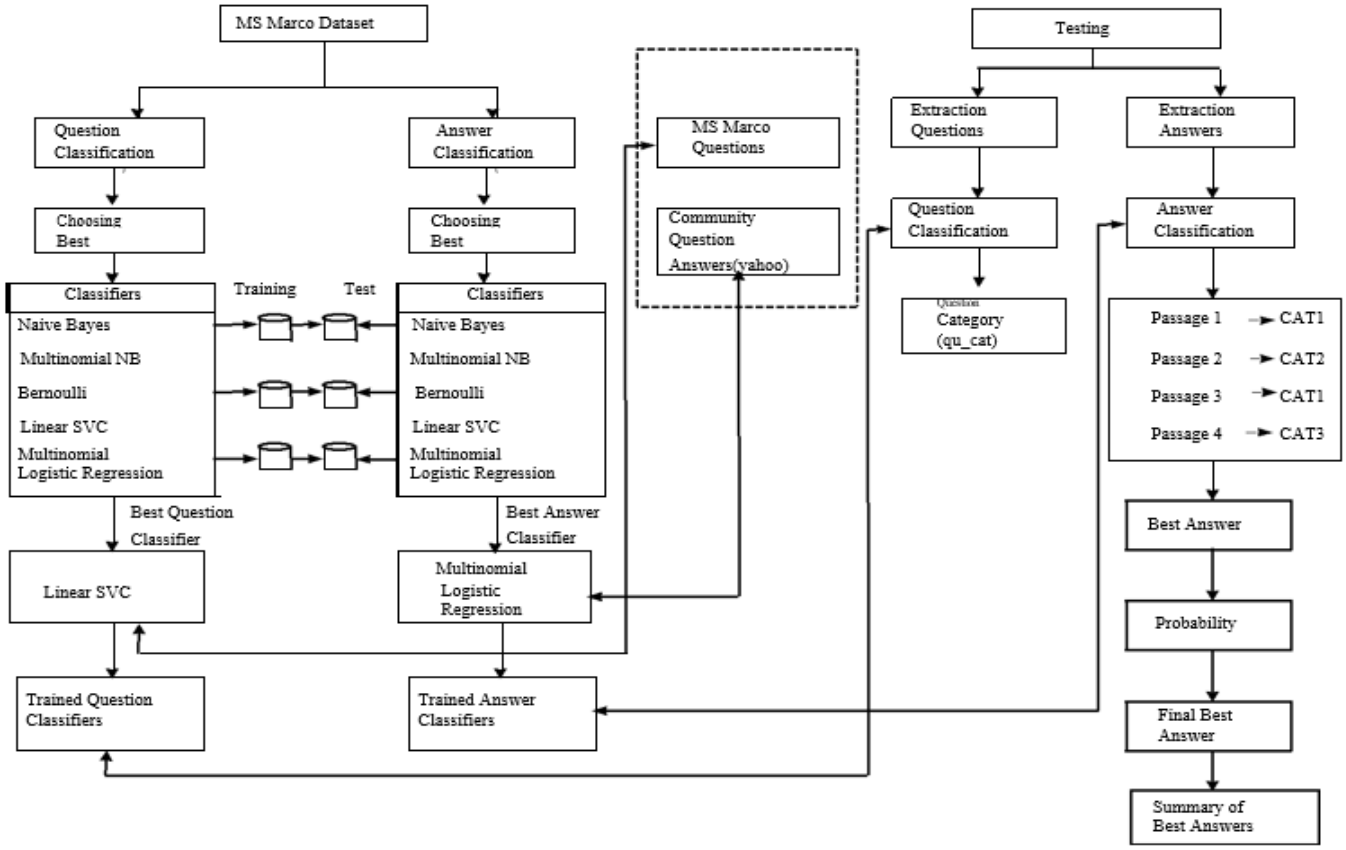

Fig. 1. Automatic Classification of Text Summarization using Machine Learning Techniques Framework ACTSMLT

IV. FRAMEWORK: ACTSMLT: AUTOMATIC CLASSIFICATION OF TEXT SUMMARIZATION USING MACHINE LEARNING TECHNIQUES

A. Problem Definition

On a website, for a given user input question *qu*, the problem is to automatically predict the best answer summaries satisfying the following objectives:

1) To select the best machine learning classifiers for both question and answers.
2) To improve the accuracy of answer summaries.

B. Assumptions

1) It is assumed that the machine learning model is trained and the user is online while entering the input question.

C. ACTSMLT Framework

The ACTSMLT framework retrieves answer text summaries from a large collection of text documents that are relevant for a given user input text question based on machine learned classifiers as shown in Fig. 1. The framework has four different modules:

(i) Text Extraction Module (ii) Question Classification Module (QC) (iii) Answer Classification Module (AC) (iv) Testing MS MARCO Dataset.

The modules are explained in the following sections.

1) Text Extraction Module:

In this work, MS MARCO dataset is used to extract the text

data. Ngdugen *et al.*, [21] have built this dataset by filtering out the question from search engine bing logs,where top *K* relevant answers are extracted and asked to judge manually to select the best answers for the questions. The dataset consists of 4000 question with each question having multiple answers, the URL's for which each answer is retrieved from a question *category*. It consists of five different question categories qu_{cat} = {Person, Location, Description, Numeric and Entity}. The subset questions are selected from the dataset based on the questions that have three to five answers. Further, each question is categorized as one among the five categories.

2) Question Classification Module (QC):

(a) Question Classifier:

The process of summarization for the query presented in natural language can be viewed as text classification based on the topic. Question Classification is a multiclass classification task that can be represented as C = $\{cat_1, cat_2, \cdots, cat_n\}$, Q = $\{qu_1, qu_2, qu_3, qu_4, \cdots, qu_n\}$. The main purpose of question classification is to group the questions into various classes and to provide constraints on the type of answers to the user to correctly place and verify the answers in the documents.

For example: Question: Which is the largest state in

is the largest state in India?

From the example above here, every noun phrase in the document is not tested inorder to extract an answer. Instead, the question is classified using machine learning techniques giving an answer category *location* We define five categories namely Person, Location, Description, Numeric, Entity. In MS-MARCO dataset every question is associated with a question category, represented by qu_{cat} that can be used in training the classifier.

(b) Preprocessing:

In this phase, preprocessing of MS-MARCO dataset is done for every question that are associated with a tag of its related category. The questions are extracted within the respective categories qu_{cat} and are segregated using a question list.

$Q_l = (qu_1, qu_{cat_1}), (qu_2, qu_{cat_2}), \cdots, (qu_n, qu_{cat_n}).$

(c) Word Tokenization and Redundancy Removal:

Information available in natural language is a sequence of characters. Word tokenization is an significant phase in NLP. Every sentence must undergo tokenization inorder to precisely represent the word in the form of token as a basic unit. Q_l obtained from the pre-processing consists of collection of questions. Initially, the question are given as the input and are automatically segmented into collection of words or tokens. In the Second stage, the sentence final punctuation characters i.e., sentence boundaries are removed. for example ',', '!', '@', "","' " are pruned from every sentence because the characters may vary the results. The document list is obtained for every word in dataset along with their specific category. doc_{list}=(word$_{i1}$,qu_{cati}), (word$_{i2}$,qu_{cati}),…. (word$_{j3}$, qu_{catj}).

(d) Frequent Word Extraction:

Extracting frequent words from the document is The significant task in text mining. The document list doc_{list} obtained from word tokenization phase contains a large number of words. Processing these words is a time consuming process. Inorder to overcome this issue, a frequent distribution is applied on the list of document

represented by doc_{list} to obtain frequency list $f\,req_{list}$

that consists of a number of words that frequently occurs

in the documents. The frequency list $freq_{list}$ is given by

$f\,req_{list} = (word_1, x_1), (word_2, x_2), \cdots (word_i, x_i).$

where,x_i is term frequency of the $word_i$ in the document. In this work, the first four thousand words are considered from the document in the experiment.

(e) Feature set Representation:

The Machine Learning classifiers basically takes a feature-based representation of the domain element as

input (e.g., a question). In our work, a sentence in the question is taken as a vector of features and this is considered as a training data for learning. Linear function

is defined to this feature vector to map from a question

to a class label. Feature set F_{set} is a boolean array of size equal to the number of words in any document. The

value of the feature set at index i depends on $word_i$ present in the $freq_{list}$.

(f) Learning a Question Classifier:

In the proposed work, a question can be mapped to one of the five possible categories as explained in text extraction module. Some of the rules followed in the question Classification are as follows:

When a question starts with *Whom* or *Who*: It belongs to a category *Person*

When a question starts with *Where*: It belongs to a category *Location*

When a question starts with *What* or *Which*: It belongs to a category *Description*

When a question starts with *How*: It belongs to a category *Numeric*

When a question starts with *Who*: It belongs to a category *Entity*

Examples for each category:

Question: Who is called the father of history in the

world? *Person*

Question: Where is University Visveswaraya College

of Engineering located in Bangalore? *Location*

Question: Which is the largest state of India? *Descrip-*

tion

Question: How many numerical answer questions are there in gate exam? *Numeric*

Question: What is the latin word for tomato?

Entity

(g) Training Question Classifier:

The initial set of any question *qu* is $C = \{ cat_1, cat_2, \cdots, cat_n \}$, the collection of all the categories. The classifier finds out a collection of labels, $C_1 = Classifier(C, qu)$, $C_1 \subseteq C$ so that $|C_1| \leq 5$ (5 is chosen through out the experiments). During the training stage, a question is processed and are classified using five different machine learning classifiers.

(h) Machine Learning Classifiers:

In the following, we briefly explain five different classifiers for the question and answer.

classifiers for the question and answer. 1) Naive Bayes, 2) Multinomial Naive Bayes, 3) Bernoulli Naive Bayes, 4) Multinomial Logistic Regression and 5) Linear SVC.

I. *Naive Bayes (NB) Classifier:*

Naive Bayes classifiers are a collection of supervised learning algorithm. NB is a simplest and efficient classifiers in machine learning as it calculates the probability of a class based on the bag of words present in the documents, However, the position of each word in text document are ignored. NB applies Bayes theorem to predict the probability for a given feature set whether the set belongs to a particular class (category) and is represented by an Equation (1).

$$P(cat \mid F_{set}) = \frac{P(cat) * P(F_{set} \mid cat)}{P(F_{set})} \quad (1)$$

Where,

- $P(cat)$ is the prior probability of a category,
- $P(F_{set} \mid cat)$ is the prior probability that the given feature set is classified as a category and
- $P(F_{set})$ is the prior probability that a feature set is occurred.

II. *Multinomial Naive Bayes Classifier (MNB):* Multinomial Naive Bayes classifiers is the most commonly used classifiers that implements Naive Bayes algorithm on distributed data. The input data in MNB is represented as a word vector counts.

The distribution is parameterized by vectors counts. $\theta_{cat} = \{(\theta_{cat1}, \theta_{cat2}, \cdots, \theta_{catF})\}$ for each category. Relative frequency count is calculated by

$$\hat{\theta}_{cati} = \frac{N_{cati} + \alpha}{N_{cati} + \alpha n} \quad (2)$$

where, n is the number of feature set, ˆ

θ_{cati} is the probability $P(cat)$ of feature i appearing in a sample belonging to a category *cat*.

N_{cati} = is the number of times feature set i appears in a sample of category *cat* in the training set T,

$^{N}cat \sum_{i=1}^{n} {}^{N}cati$ is the total count of all features for category *cat*. Smoothing priors $\alpha \geq 0$ represents features not present in the learning samples.

III. *Bernoulli Naive Bayes Classifier (MNB):*

Like MNB classifiers, Bernolli classifiers implements Naive Bayes algorithm for the distributed data. The main difference between MNB and Bernolli is that MNB works with word occurence counts, whereas BNM is suitable for only binary features.

IV. *Linear Support Vector Classification (SVC).*

Linear SVC is a machine learning algorithm that solves multiclass classification problem efficiently for large dataset. The data is pre-processed before ap-plying SVM package. Initially, the original data i.e., all the questions and answers are transformed to the format of SVM package. The question and answer are represented as a vector of real numbers. Further, it is very significant to apply scaling. The categories of attributes are linearly scaled because if there are enormous numeric values it is very difficult during numeric calculation.

Linear SVC is an multiclass classification for one against all method. It builds a *Categorical* SVC modelswhere *Cat* is the number of categories. For all the questions, Multiclass m^{th} SVC is trained in m^{th} class with proper positive labels.

Let $Q_l = \{(qu_1, cat_1), \cdots, (qu_l, cat_l)\}$, be the set of training questions. We extract every question from a domain $qu_i \sqsubseteq R^n$, *where,* $i = \{1, 2, \cdots, Q_l\}$ and $cat_i \{1, 2, \cdots, c_n\}$ is the number of categories of qu_i. A multiclass classfier is a function $S = Q_l \rightarrow cat_i$ that maps an instance qu_i to an element cat_i by a function. and *C* as it is the penalty parameter. The m^{th} SVC solves the following problem

$$\min_{w^m, b^m, \xi^m} \frac{1}{2}(w^m)^T w^m + C \sum_{i-1}^{Q_l} \xi_i^m$$
$$(w^m)^T \phi(qu_i) + b^m \geq 1 - \xi_i^m, \text{ if } cat_i = m,$$
$$(w^m)^T \phi(qu_i) + b^m \leq - + \xi_i^m, \text{ if } cat_i \neq m,$$
$$\xi_i^m \geq 0, i = 1, \ldots, Q_l,$$

(3)

where the training questions qu_i are mapped to a higher dimensional space by the function φ and C is the penalty parameter. Minimizing $1/2(w^m)^T w^m$ means that we would like to maximize $2\| w^m \|$, the margin between groups of data. When data is not linearly separable, there is a penalty term $C \sum_{i-1}^{Q_l} \xi_i^m$ that reduces some errors in the training set. The basic concept behind SVM is to search for a balance between the regularization term $1/2 (w^m)^T w^m$ and the training errors. Calculating (3), there may be k decision functions:

$$(w^1)^T \varphi(x) + b^1,$$
$$\vdots$$
$$(w^k)^T \varphi(x) + b^k.$$

We say Q_l is present in class that has the biggest value of the decision function:

$$\text{class of } Q_l \equiv argmax_{m=1,\ldots,k}((w^m)^T \varphi(Q_l) + b^m). \quad (4)$$

V. *Multinomial Logistic Regression (MLR).*

In this section, we show how the multinomial logistic regression is used to predict a given question query belongs to one of the five question categories. The MLR measures the relationship between the categor-ical dependent variable and one or more independent variables by estimating the probabilities using a logistic function. The dependent variable is the target variable that is to be predicted and the five possible outcomes of this variable includes person, location, description, numeric and entity. The independent variables are the features or attributes used to predict the outcome on the *likelihood occurrence* of the match between the question and the question category.[1] used in the logistic regression model for multiclassification

The MLR model passes the likelihood occurrences through the logistic function to predict the corresponding target class. In other words, this approach consist of training a regressor over all the features defined for a given< *question, question category* > tuple. The idea is to rely on the learning to optimally use all available features to predict the final target variable. For some match between the question and the question category with probability *p* of being 1, the *odds* of that match are given by:

$$Odds = \frac{p}{1-p}$$

which means that the odds of the likelihood that the match takes place, while odds against reflect the likelihood that does not occur. Now, we obtain the *logit* transform natural log of the odds i.e.

$$logit(p) = log(odds) = \log \frac{p}{1-p}$$

In the logistic regression, we seek a linear model of the form:

$$logit(p) = \beta_0 + \beta_i x_i$$

That is, the *log odds* (logit) is assumed to be linearly re-lated to the independent variables x_i for $i = 1, 2, \ldots, 5$. One of the popular logistic function, the softmax function[1] to calculate the probabilities for the given *logits* or *scores* is used in this work. The softmax function is defined by:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{k} e^{z_i}}$$

Such a softmax function output values in the range *(0, 1)* and the sum of the output values is equal to 1. This function takes each logit and finds the probability. Once the probabilities are calculated, transfer them into *one hot encoding* and use the *cross entropy* methods in the training process for calculating the optimized weights. Now, to solve to which category the question belongs, we implement the following MLR stages.

i) **Inputs:** The inputs to the MLR are the features in the dataset. Hence, to predict the question category, the features like word count, etc. are considered as features.

ii) **Linear Model:** The linear model equation is the same as the linear equation in the linear regression model of the form $y = \beta_0 + \beta_1 x$ where *x* is a feature vector and β_1 is the weight vector. The weights gets updated in the training phase.

iii) **Logits:** These are the outputs of the linear model. The logits (or score) change with the changes in the calculated weights.

iv) **Softmax Function:** This is a probabilistic function that calculates the probabilities for the given logit. The function returns the high probability value for the high scores and fewer probabilities for the remaining scores. However, the calculated proba-bilities occurs in the range *(0, 1)* and the sum of the probabilities is equal to *1*.

v) **Cross Entropy Function:** This function is used to find the *similarity distance* between the probabili-ties calculated from the softmax function and the target one-hot-encoding matrix.

Parameter optimization: The expected output af-ter training the MLR classifier is the calculated weights. These calculated weights are used for the prediction task. The process stops when the cost function value is less or negligible.

In this way, the given question is classified to which category it belongs. Similarly, the answer classification is accomplished.

3) Answer Classification Module (AC): In this module, the best classifiers is identified for the answer classification. The Multiple answer from Yahoo Answers (YA) are collected for every question in the MS MARCO dataset. In total, multiple answers for a seven hundred questions are collected. Out of which, the best answers for four hundred questions are used for training the classifiers and remaining three hundred answers are used for testing. Similar to the question classification, all the answers are preprocessed with their categories an_{cat} and these answers are segregated with a list called answer list.

$An_{list} = \{(an_1, an_{cat1}), (an_2, an_{cat2}), \cdots, (an_n, an_{catn})$. An_{list} obtained from preprocessing contains collection

of answers. Each answers consists on an average eight to ten sentences. The word tokenization is applied on every sentence in the answer. Further, all the punctual characters like (',', '!', '@', '"',' ') are removed from every sentences to form document answer list represented by $docan_{list} = \{(word_{i1}, an_{cati}), (word_{i2}, an_{cati}), \ldots (word_{j3}, an_{catj})\}$.

Frequent distribution is applied on the list of document represented by $docan_{list}$ to obtain $f\ req_{list}$ that consists of a number of words that frequently occurs in the documents. $f\ req_{list}$ is given by $f\ req_{list} = \{(word_1, x_1), (word_2, x_2), (word_3, x_3), \cdots (word_i, x_i)\}$. where x_i is term frequency of $word_i$ in the document.

For the current task, a answer sentence is represented as a vector of features and treated as a training or test example for learning. The mapping from a answer to a class label is a linear function defined over this feature vector.
The Feature set F_{set} is boolean array of size equal to number of words in any document. Once the feature set is obtained, five different machine learning classifiers are applied to find the classifiers that provide the best results. From the results obtained in table, we observe that multinomial logistic regression is the best classifier to classify the answers.

4) Probabilistic Model for Selecting the Best Answer: In this section, a probabilistic model for selecting the best answer from a set of feasible answers using the Probabilistic Ranking Principle (PRP) is provided. In this model of probabilistic selection, let $P\ (x|y)$ be the proba-bility that a feasible answer y is ranked and categorising to category x is calculated. In other words, for every feasible answer in the answer list y_i is calculated and all the feasible answers are ranked in a non-increasing order according to their probabilities. The larger the value of $P\ (x|y_i)$ a feasible answer y_i has, the more probable it will be termed as the best answer. Let $T = t_i$ be a randomly selected term from a answer is t_i. Hence, $P\ (x|y)$ can be written as:

$$P\ (x|y) = \sum P\ (x|y, T = t_i) P\ (T = t_i|y) \quad (5)$$

If we assume conditional independence between x and y, given that $T = t_i$, i.e. $P\ (x|y, T = t_i) = P\ (x|T = t_i)$, we obtain,

$$P\ (x|y) = \sum P\ (x|T = t_i) P\ (T = t_i|y) \quad (6)$$

Using Bayes' Theorem,

$$P\ (x|y) = P\ (x) \sum_{t_i} \frac{P\ (T = t_i|x) P\ (T = t_i|y)}{P\ (T = t_i)} \quad (7)$$

5)

The probabilities in Equation. 7 can be estimated from the training data based on $P\ (T = t_i|x)$, the probability that a randomly selected term in the set of feasible answers is t_i given that the answer is assigned to x.

Example of Text Summarization

Golden Reference (GR) Summary

A UFO is not known as anything. It's unknown. Since UFO stands for Unidentified Flying Object, it must be unknown. If it were known, it would not be classified as a UFO. The most convincing UFOs are those that COULD NOT POSSIBLY BE any other known object (insects, birds, planes, clouds, satellites, planets, meteors, etc). There are documented instances of true UFOs (something that could not be positively identified as anything) but it should only be classified as such after all other possibilities have been found impossible. That makes it difficult to come up with a solid UFO. The key thing to remember here is a UFO does not necessarily equate to aliens.

ACTSMLT Framework Summary

There are documented instances of true UFOs (something that could not be positively identified as anything) but it should only be classified as such after all

other possibilities have been found impossible. The most convincing UFOs are those that COULD NOT POSSIBLY BE any other known object (insects, birds, planes, clouds, satellites, planets, meteors, etc).

MK++ Summary

The most convincing UFOs are those that COULD NOT POSSIBLY BE any other known object (insects, birds, planes, clouds, satellites, planets, meteors, etc.,). There are documented instances of true UFOs (something that could not be positively identified as anything) but it should only be classified as such after all other possibilities have been found impossible.

From the above example it is noticed that ACTSMLT is a very concise summary about UFO and the second summary, MK++ can be seen that it is neither verbose nor overly concise however, both mean to summarize the description of the word UFO's. The Golden Reference summary, GR is a more verbose summary.

5) Testing MS MARCO Dataset: In this module, a set of experiments is done for MS MARCO and Yahoo answers dataset using automatically extracted features from questions and answers is briefly described. As we discussed in the previous section, there may be a common perception of what constitutes to the quality of an answer whether the answer satisfies the user asked question or not. The MS MARCO dataset contains more than 90K queries with each having a set of relevant paragraphs, the document URLs for which each relevant paragraph is extracted from the annotated answers, and the query category. In this experiment, a subset of queries from MS

MARCO data is selected as discussed in text extraction module. The above steps result in five thousand queries. For the experiments, four thousands questions are given to the trained question classifiers and the related YA answers that are available are extracted as the answers and are given to the trained answer classifiers. The generated answers that are obtained after testing by multinomial logistic regression are then taken as ground truth answers. If the question classification, classifies the given question as qu_{cati} and if the answer classification, produces a same category an_{cati} , then it is said that the prediction made by the question and answer classification is correct. If two answers are obtained with the same category, then the probability score is to be calculated for the two answers. The probabilty score that has the highest score is considered to be the best answer.

v PERFORMANCE ANALYSIS

A. Datasets Used:

To investigate the effectiveness of our proposed ACTSMLT framework, the MS MARCO dataset is used. The dataset consists of 4000 question with each question having multiple answers, the URL's for which each answers are retrieved from a question *category*. It consists of five different question categories qu_{cat} = {Person, Location, Description, Numeric and Entity}. The subset question are selected from the dataset based on the question that have three to five answers. Each answers consists of average ten sentences and every question is categorized as one among the five categories.

B. Experiment Settings

All the algorithms are implemented in Java 1.8.0 and experiments have been processed on a Windows 10 PC with 2.2 GHz Intel i5 processor, 8 GB RAM and 1 TB HDD.Our experimental study focuses on testing the perfor-mance of the learned classifier in classifying factual questions into classes.

a) Choosing the best Question Classifier.
b) Choosing the best Answer Classifier.
c) Extracting Summaries for related CQA answers.

In the following, the aforelisted are briefly expalined.

a) Choosing the best Question Classifier: All the classi-fiers are trained on the 4000 training questions and tested on the 1,000 MS-MARCO dataset in the ex-periments. The most significant criteria for evaluating the performance of the classifiers is the accuracy rate. For question classifier, we train MS-MARCO dataset with Radial Basis Function (RBF) kernel [22]. The size of the training set by training the classifier is varied with different sizes. The size of training and testing dataset is varied by 75% - 25%, 80% - 20% and 85% - 15% using the kernel parameter γ and cost parameter C pair ie., (C, γ). After training the dataset for the aforementioned ratios, the best results are tabulated for 75% - 25% ratio. The results in the Table III shows that, for Question classification (QC), the Linear Support Vector Classification is found to be the best classifier compared to other classifiers.

The performance is evaluated by the global accuracy of the classifiers for all the classes (Accuracy), and the accuracy of the classifiers for a specific class (Precision[c]), defined as follows:

$$Accuracy = \frac{\sum Number\ of\ Correct\ prediction}{Number\ of\ prediction}$$

TABLE III
COMPARISON OF DIFFERENT CLASSIFIERS EFFICIENCY FOR QUESTIONS

Sl. no.	Classifiers	Number Of question	Correctly Classified	Efficiency
1.	Naïve Bayes Classifier	375	281	75%
2.	Multinomial Naïve Bayes	375	263	70%
3.	Bernoulli	375	274	73%
4.	Multinomial Logistic Regression	375	323	86%
5.	Linear SVC	375	330	88%

b) Choosing the best Answer Classifier:

All the classifiers are trained on the 4000 training answers and tested on the 1,000 MS-MARCO dataset in the experiments. The size of the training set is varied with different sizes while training the classifier. The size of training and testing dataset is varied by 75% - 25%, 80% - 20% and 85% - 15%. After training the dataset for the aforementioned ratios, the best results are tabulated in Table IV for 75% - 25%. The results shows that, for Answer classification (AC), Multinomial Logistic Regression (MLR) is found to be the best classifier.

The MLR requires relatively more time during training phase when compared to other machine learning techniques because it uses an iterative algorithm to estimate the parameters of the model. However, MLR is useful when the data set is large. Hence, in this work, the soft max regression function is used that is found to be competitive in CPU and memory consumption. Also, the MLR does not assume a linear relationship between the dependent and independent variables.

Table iv comparison of different class ifiers efficiency for answers

Sl.no.	Classifiers	Number of Answer	Correctly Classified	Efficiency
1.	Naive Bayes Classifier	375	274	73%
2.	Multinomial Naive Bayes	375	244	65%
3.	Bernoulli	375	285	76%
4.	Multinomial Logistic Regression	375	319	85%
5.	Linear SVC	375	300	80%

c) Extracting text Summaries for related CQA answers: In this work, the text summarization[2] is achieved by extracting several portions of the query and answer result such as sentences and phrases. The whole process of text summarization is briefly explained as follows step-wise:

i) Read the text (or query and answer result) and split it into sentences.
ii) The similarity between two sentences are identified by computing the similarity scores that are stored in the square matrix. In other words, the similarities between the sentence vectors are calculated.
iii) This similarity square matrix is converted to a graph with sentences as vertices and scores as edges to calculate the rank of each sentence.
iv) Finally, the top-three ranked sentences are group to form the summary.

The quality of 30 word summaries was evaluated using ROUGE Recall Oriented Understudy of Gisting Evaluation by comparing the produced summaries from the proposed ACTSMLT framework with that of ground truth (golden reference) answers i.e., ROUGE compares the number of overlapping words between system summary and golden reference summary However it do not produce good metric. Inorder to produce a quantitative value in our evaluation recall, precision and F- score is computed using the overlap words. We choose only ROUGE 1 (unigram) and ROUGE 2 (bigram) scores in our evaluation. In the calculation of ROUGE scores, the maximum value of term overlap between the generated summary and ground truth answers are considered. Recall, Precision

$$Recall = \frac{\sum Overlap(GR_{ipos}, ACT\ SM\ LT_{jpos})}{|\ GR_{ipos}\ |}$$

$$Precision = \frac{\sum Overlap(GR_{ipos}, ACT\ SM\ LT_{jpos})}{|\ ACT\ SM\ LT_{jpos}\ |}$$

$$F\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision}$$

After calculating these scores for each document, the average is taken and we consider as its final score. The proposed framework is evaluated and compared with the Metzler and Kunungo's (MK++) [1] as our baseline system.

TABLE V
SUMMARY ACCURACY ROUGE-1 SCORES FOR ACTSMLT AND MK FRAMEWORK

Sl.no.	Method	Recall	Precision	F- Score
1.	ACTSMLT Description	0.6010	0.9978	0.7501
2.	ACTSMLT Entity	0.5132	1	0.6782
3.	ACTSMLT Location	0.5671	0.9695	0.7156
4.	ACTSMLT Numeric	0.5932	0.9997	0.7445
5.	ACTSMLT Person	0.5836	1	0.7370
6.	MK++ Description	0.5936	0.9683	0.7393
7.	MK++ Entity	0.5050	0.9985	0.6707
8.	MK++ Location	0.5560	0.9575	0.7034
9.	MK++ Numeric	0.5826	0.9982	0.7357
10.	MK++ Person	0.5796	0.9991	0.7336

While scrapping the yahoo answers, a very high quality of answers are obtained for descriptive category. Therefore, descriptive category has high weightage. The Table V shows the summary accuracy of ROUGE-1 scores for ACTSMLT and MK++ framework and it is observed that the quality of answers for descriptive category has good quality. The low value indicates that we do not get very good quality answers from yahoo scrapped data. Therefore, we justify that the accuracy varies based on the quality of answer. The Rouge 1 is used more for text summarization evaluation because the summary obtained is more accurate as we follow the word orderings of reference summary.

Accuracy is also affected based on the number of answers to a particular question. Suppose, if a question has only one answer, then considering that has a best answer is not so relevant. Therefore, the knowlegde of the classifier may vary if there is only one answer for a question. On the other hand, if there are multiple answers for a question and one best answer is highly rated by the user. Hence, the quality of the answer is high. The Table VI shows the summary accuracy of ROUGE-2 scores for ACTSMLT and MK++ framework for five different categories. It is noticed that as the summaries in both the system and golden reference get longer and longer,

TABLE VI
SUMMARY ACCURACY ROUGE-2 SCORES FOR ACTSMLT AND MK FRAMEWORK

Sl.no.	Method	Recall	Precision	F- Score
1.	ACTSMLT Description	0.6276	0.9870	0.7672
2.	ACTSMLT Entity	0.5183	0.9719	0.6760
3.	ACTSMLT Location	0.5218	0.9100	0.6632
4.	ACTSMLT Numeric	0.5663	0.9651	0.7173
5.	ACTSMLT Person	0.5723	0.9723	0.7205
6.	MK++ Description	0.6191	0.9775	0.7580
7.	MK++ Entity	0.4995	0.9729	0.6600
8.	MK++ Location	0.5193	0.9090	0.6609
9.	MK++ Numeric	0.5654	0.9641	0.7127
10.	MK++ Person	0.5622	0.9717	0.7122

Their are fewer overlapping bigrams especially in case of abstractive summarization. Therefore, the proposed framework removes related CQA answers that are bad in quality.

VI CONCLUSIONS

Due to the increasing amount of internet user's Com-munity Question Answering (CQA) sites such as quora, yahoo, stackoverflow have become a significant knowl-edge database in enormous domains to extract knowl-edge. In order to provide correct ansers from a large text sources, one need to understand the question to suggest a perfect answer. Therefore, it is necessary to correctly provide the ansers to user asked question. In this work, an ACTSMLT framework has been proposed to automatically predict the quality of CQA answers using five different machine learning classifiers. A series of experiments are conducted to find the best classifiers for question and answer by varying the size of training and testing the dataset. The results shows that, for Question Classification (QA), Linear Support Vector Classification (LSVC) is found to be the best classifier and Multinomial Logistic Regression (MLR) is most suitable for Answer Classification (AC). The MS MARCO dataset is used for testing the proposed framework. While training the classifiers the Yahoo answers are extracted for all the questions. The automatically generated best answers are then considered to be the golden reference. Further, we plan to use semantic similarity between sentences and provide more accurate and relevant summaries by increasing the number of categories and predicting the emotions involved in the documents.

REFERENCES

1. E. Yulianti, R.-C. Chen, F. Scholer, W. B. Croft, and M. Sanderson, "Document Summarization for Answering Non-Factoid Queries," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, pp. 15–28, 2018.
2. Y. Wu, C. Hori, H. Kashioka, and H. Kawai, "Leveraging Social Q Collections for Improving Complex Question Answering," *Computer Speech Language*, vol. 29, no. 1, pp. 1–19, 2015.
3. O. Kolomiyets and M.-F. Moens, "A Survey on Question An-swering Technology from an Information Retrieval Perspective," *Information Sciences*, vol. 181, no. 24, pp. 5412–5434, 2011.
4. C. L. Chiang, S. Y. Chen, and P. J. Cheng, "Summarizing Search Results with Community-Based Question Answering," *In Pro-ceedings of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, vol. 1, no. 2, pp. 254–261, 2014.
5. E. Yulianti, S. Huspi, and M. Sanderson, "Tweet-Biased Sum-marization," *Association for Information Science and Technology*, vol. 67, no. 6, pp. 1289–1300, 2016.
6. S. Asur and B. A. Huberman, "Predicting the Future with Social Media," *In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 01, pp. 492–499, 2010.
7. P. Ren, Z. Chen, Z. Ren, F. Wei, L. Nie, J. Ma, and M. D. Rijke, "Sentence Relations for Extractive Summarization with Deep Neural Networks," *ACM Transactions on Information Systems (TOIS)*, vol. 36, no. 4, p. 39, 2018.
8. Z. Liu, Y. Xia, Q. Liu, Q. He, C. Zhang, and R. Zimmermann, "Toward Personalized Activity Level Prediction in Community Question Answering Websites," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 2s, p. 41, 2018.
9. K.-Y. Chen, S.-H. Liu, B. Chen, and H.-M. Wang, "An Infor-mation Distillation Framework for Extractive Summarization," *IEEE/ACM Transactions on Audio, Speech, and Language Pro-cessing*, vol. 26, no. 1, pp. 161–170, 2018.

10. K. Williams, J. Kiseleva, A. C. Crook, I. Zitouni, A. H. Awadal-lah, and M. Khabsa, "Detecting Good Abandonment in Mobile Search," *In Proceedings of the 25th International Conference on World Wide Web*, pp. 495–505,2016
11. E. Baralis and L. Cagliero, "Learning from Summaries: Support-ing Learning Activities by means of Document Summarization," *IEEE Transactions on Emerging Topics in Computing*, vol. 4, no. 3, pp. 416–428, 2016.
12. L. Leal Bando, F. Scholer, and A. Turpin, "Query-Biased Sum-mary Generation Assisted by Query Expansion," *Journal of the Association for Information Science and Technology*, vol. 66, no. 5,pp. 961-979,2015.
13. R. Soricut and E. Brill, "Automatic Question Answering using the Web: Beyond the Factoid," *Information Retrieval*, vol. 9, no. 2,pp. 191-206,2016
14. D. E. Losada, "Statistical Query Expansion for Sentence Retrieval and its Effects on Weak and Strong Queries," *Information Re-trieval*, vol. 13, no. 5, pp. 485–506, 2010.
15. R. S. Ramya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, "Feature Extraction and Duplicate Detection for Text Mining: A Survey," *Global Journal of Computer Science and Technology*, vol. 16, no. 5, pp. 1–21, 2017.
16. P. E. Shrout and J. L. Fleiss, "Intraclass Correlations:Uses in Assessing Rater Reliability," *Psychological bulletin*, vol. 86,no 2,pp.420,1979.
17. L. Yang, Q. Ai, J. Guo, and W. B. Croft, "Ranking Short Answer Texts with Attention-based Neural Matching Model," *In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*,pp. 287-296,2016.
18. F. Wu, X. Duan, J. Xiao, Z. Zhao, S. Tang, Y. Zhang, and Y. Zhuang, "Temporal Interaction and Causal Influence in Community-based Question Answering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 10,pp. 2304-2317,2017.
19. J. H. Park and W. B. Croft, "Using Key Concepts in a Translation Model for Retrieval," *In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Infor-mation Retrieval*,pp. 927–930, 2015.
20. T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng, "Ms marco: A Human Generated Machine Reading Comprehension Dataset," *arXiv preprintarXiv:1611.09268*,2016

AUTHORS PROFILE

Ramya R S is currently research scholar in the department of Computer Science Engineering, University Visweswaraya College of Engineering, Bangalore. She has more than 8 years of teaching experience. she has obtained ME and BE degree in computer science and engineering from Visweswaraya Technological University. Her research interest includes data mining, web mining, and sentiment analysis.

.

Darshan M is a UG student in the department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore. He is currently pursuing B.E.degree in Information Science and Engineering. His areas of interest are Data mining, Big Data.

Dr. Sejal Santosh Nimbhorkar is currently working as an Associate Professor at B N M Institute of Technology. She has more than 15 years of industry, research and teaching experience. She has obtained ME and BE degree in Computer Science and Engineering from University Visvesvaraya College of Engineering and Gujarat University respectively. She has received Project Grant from Karnataka State Council for Science and Technology (KSCST). Her research interest includes Data Mining, interest includes Data Mining Web Mining, Sentiment Analysis and IoT.

Venugopal K R is currently the Principal, University Visvesvaraya College of Engineering, Bangalore University, Bangalore. He obtained his Bachelor of Engineering from University Visvesvaraya College of Engineering. He received his Masters degree in Computer Science and Automation from Indian Institute of Science Bangalore. He was awarded Ph.D in Economics from Bangalore University and Ph.D in Computer Science from Indian Institute of Technology, Madras. He has a distinguished academic career and has degrees in Electronics, Economics, Law, Business Finance, Public Relations, Communications, Industrial Relations, Computer Science and Journalism. He has authored and edited 64 books on Computer Science and Economics, which include Petrodollar and the World Economy, C Aptitude, Mastering C, Microprocessor Programming, Mastering C++ and Digital Circuits and Systems etc., He has filed 101 patents. During his three decades of service at UVCE he has over 640 research papers to his credit. His research interests include Computer Networks, Wireless Sensor Networks, Parallel and Distributed Systems, Digital Signal Processing and Data Mining. He is a Fellow of IEEE and ACM.

S S Iyengar is currently Ryder Professor, Florida International University, USA. He was Roy Paul Daniels Professor and chairman of the Computer Science Department of Louisiana state University. He heads the Wireless Sensor Networks Laboratory and the Robotics Research Laboratory at USA. He has been involved with research in High Performance Algorithms, Data Structures, Sensor Fusion and Intelligent Systems, since receiving hid Ph.D degree in 1974 from MSU, USA. He is Fellow of IEEE and ACM. He has directed over 40 Ph.D students and 100 post graduate students, many of whom are faculty of Major Universities worldwide or Scientists or Engineers at National Labs/Industries around the world. He has published more than 800 research papers and published more than 800 research papers and has authored/co-authored 6 books and edited 7 books. His books are published by John Wiley and Sons, CRC Press, Prentice Hall, Springer Verlang, IEEE Computer Society Press etc.. One of his books titled Introduction to Parallel Algorithms has been translated to Chinese. He is a Fellow of IEEE and a Fellow of ACM.

L M Patnaik is currently Honorary Professor, Indian Institute of Science, Bangalore, India. He was a Vice Chancellor, Defense Institute of Advanced Technology, Pune, India and was a Professor since 1986 with the Department of CSA, Indian Institute of Science, Bangalore. During the past 35 years of his service at the Institute he has over 1150 research publications in refereed International Journals and refereed Inter-national Conference Proceedings. He is a Fellow of all the four leading Science and Engineering Academies in Indian Fellow of the IEEE and the Academy of Science for the Developing World. He has received twenty national and international awards; notable among them is the IEEE Technical Achievement Award for his significant contributions to High Performance Computing and Soft Computing. His areas of research interest have been Parallel and Distributed Computing, Mobile Computing, CAD, Soft Computing and Computational Neuroscience. He is the Fellow of all the four leading Science and Engineering Acadamies in India; Fellow the Academy of Science for the Developing World and a Fellow of IEEE.