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 (QC), 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. 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...
ACTSMLT: Automatic Classification of Text Summarization using Machine Learning Techniques

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 collected 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 Regression and Linear Support Vector classifiers to automatically 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 summarization. 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 answering 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 summarization 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 forecast 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 sentences. The summary of recent research works along with their advantages and disadvantages is given in Table I.

Shrout et al., [16] illustrated six forms of the intraclass correlation for reliability studies and also guidelines for choosing 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 differentiate 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, that may have satisfied the user).

**TABLE I**

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

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. In order to interpret the

<table>
<thead>
<tr>
<th>Sl.no</th>
<th>Authors</th>
<th>Year</th>
<th>Model</th>
<th>Dataset used</th>
<th>Summarization technique</th>
<th>Based on no of document</th>
<th>Summarization contexts</th>
<th>Summary types</th>
<th>Evaluation methods</th>
<th>O/p type</th>
<th>Approaches Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Liu et al.,[8]</td>
<td>2018</td>
<td>Personalized prediction model</td>
<td>Stack overflow</td>
<td>-</td>
<td>-</td>
<td>personalised</td>
<td>-</td>
<td>-</td>
<td>Indicative</td>
<td>Logistic regression</td>
</tr>
<tr>
<td>3</td>
<td>Chen et al.,[9]</td>
<td>2018</td>
<td>Denoising essence vector model</td>
<td>Duc 2002</td>
<td>Extractive Multiple documents</td>
<td>-</td>
<td>-</td>
<td>Rouge</td>
<td>Informative</td>
<td>Clustering</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Yulianti et al.,[1]</td>
<td>2018</td>
<td>Query CQA and Expanded query biased</td>
<td>webap</td>
<td>Extractive Multiple documents</td>
<td>Web summarization</td>
<td>Query Focused</td>
<td>Rouge</td>
<td>Informative</td>
<td>Frequency based Approach</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Yulianti et al.,[5]</td>
<td>2016</td>
<td>Twitter based summary</td>
<td>Twitter dataset from tree 2011</td>
<td>Extractive Multiple documents</td>
<td>Web summarization</td>
<td>Generic summarization</td>
<td>Rouge-n, Rouge-1 Rouge-su</td>
<td>Informative</td>
<td>Graph Methods For Summarization</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Leal Bando et al.,[12]</td>
<td>2015</td>
<td>Sentence ranking approach</td>
<td>Tree novelty</td>
<td>Extractive Multiple documents</td>
<td>Web summarization</td>
<td>Generic Summaries</td>
<td>Manual</td>
<td>Informative</td>
<td>Frequency based Approach</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Jae hyun park et al., [19]</td>
<td>2015</td>
<td>Statistical translation model</td>
<td>Yahoo 148.102 Qa pairs</td>
<td>Extractive Multiple documents</td>
<td>-</td>
<td>Query Focused</td>
<td>prec@1 recall@5</td>
<td>Informative</td>
<td>Machine Learning</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Chengying Liu et al.,[20]</td>
<td>2015</td>
<td>Incrests algorithm</td>
<td>Facebook</td>
<td>Extractive Multiple documents</td>
<td>Social network Services</td>
<td>Topic Based</td>
<td>F measure, N-gram</td>
<td>Informative</td>
<td>Cluster based</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Losada et al.,</td>
<td>2010</td>
<td>Pseudo relevance feedback</td>
<td>The tree novelty track</td>
<td>Extractive Single document summarization</td>
<td>Web summarization</td>
<td>Query Focused</td>
<td>F measure, precision at ten sentences retrieved</td>
<td>-</td>
<td>Frequency Based (t/t-idf)</td>
<td></td>
</tr>
</tbody>
</table>
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 \( q_u \), 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 \( q_u_{\text{cat}} = \{ \text{Person, Location, Description, Numeric and Entity} \} \). 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

\[ C = \{ \text{cat}_1, \text{cat}_2, \ldots, \text{cat}_n \}, Q = \{ q_{u1}, q_{u2}, q_{u3}, q_{u4}, \ldots, q_{un} \} \].

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 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 \( q_{\text{cat}} \) 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 \( q_{\text{cat}} \) and are segregated using a question list.

\[
Q_1 = (q_{\text{u1}}, q_{\text{cat1}}), (q_{\text{u2}}, q_{\text{cat2}}), \ldots, (q_{\text{un}}, q_{\text{catn}}).
\]

(c) Word Tokenization and Redundancy Removal:
Information available in natural language is a sequence of characters. Word tokenization is a significant phase in NLP. Every sentence must undergo tokenization inorder to precisely represent the word in the form of token as a basic unit. \( Q_1 \) 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.

\[
doc_{\text{list}} = (\text{word}_1, q_{\text{cat}}), (\text{word}_2, q_{\text{cat}}), \ldots, (\text{word}_i, q_{\text{cat}}).
\]

(d) Frequent Word Extraction:
Extracting frequent words from the document is The significant task in text mining. The document list \( doc_{\text{list}} \) 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 \( doc_{\text{list}} \) to obtain frequency list \( freq_{\text{list}} \) that consists of a number of words that frequently occurs in the documents. The frequency list \( freq_{\text{list}} \) is given by

\[
freq_{\text{list}} = (\text{word}_1, x_1), (\text{word}_2, x_2), \ldots, (\text{word}_i, x_i).
\]

where \( x_i \) is term frequency of the \( \text{word}_i \) 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 \( F_{\text{set}} \) 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 \( \text{word}_i \) present in the \( freq_{\text{list}} \).

(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 \( \text{Whom} \) or \( \text{Who} \): It belongs to a category \( \text{Person} \)
When a question starts with \( \text{Where} \): It belongs to a category \( \text{Location} \)
When a question starts with \( \text{What} \) or \( \text{Which} \): It belongs to a category \( \text{Description} \)
When a question starts with \( \text{How} \): It belongs to a category \( \text{Numeric} \)
When a question starts with \( \text{Who} \): It belongs to a category \( \text{Entity} \)

Examples for each category:

Question: Who is the father of history in the world? \( \text{Person} \)
Question: Where is University Visveswaraya College of Engineering located in Bangalore? \( \text{Location} \)
Question: Which is the largest state in India? \( \text{Description} \)
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 $C = \{ cat_1, cat_2, \ldots, cat_n \}$, the collection of all the categories. The classifier finds out a collection of labels, $C_i = \text{Classifier}(C, qu)$, $C_i \subseteq C$ so that $|C_i| \leq 5$ (5 is chosen throughout the experiments). During the training stage, a question is processed and are classified using five different machine learning classifiers.

(b) **Machine Learning Classifiers:**

In the following, we briefly explain five different 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 algorithms. 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).

$$ P(\text{cat} \mid F_{\text{set}}) = \frac{P(\text{cat}) \times P(F_{\text{set}} \mid \text{cat})}{P(F_{\text{set}})} $$ (1)

Where,

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

II. **Multinomial Naive Bayes Classifier (MNB):**

Multinomial Naive Bayes classifiers are 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, $\theta_{\text{cat}} = \{(\theta_{\text{cat}_1}, \theta_{\text{cat}_2}, \ldots, \theta_{\text{cat}_n})\}$ for each category. Relative frequency count is calculated by

$$ \hat{\theta}_{\text{cat}_i} = \frac{N_{\text{cat}_i} + \alpha}{N_{\text{cat}_i} + \alpha n} $$ (2)

where, $n$ is the number of feature set.

$\theta_{\text{cat}_i}$ is the probability $P(\text{cat})$ of feature $i$ appearing in a sample belonging to a category cat.

$N_{\text{cat}_i}$ is the number of times feature set $i$ appears in a sample of category cat in the training set $T$, $n_{\text{cat}_i} = \sum_{i=1}^{n_{\text{cat}_i}}$ 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, Bernoulli classifiers implements Naive Bayes algorithm for the distributed data. The main difference between MNB and Bernoulli is that MNB works with word occurrence 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 applying 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 a multiclass classification for one against all method. It builds a Categorical SVC model where 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 $Q_i = \{(qu_1, cat_1), \ldots, (qu_n, cat_l)\}$ be the set of training questions. We extract every
question from a domain \( q_u, \in R^n \), where, \( i \in \{1, 2, \cdots , Q_i\} \) and \( cat, \{1, 2, \cdots , c_n\} \) is the number of categories of \( q_u \). A multiclass classifier is a function \( S = Q_i \rightarrow cat \) that maps an instance \( q_u \) to an element \( cat \) by a function. and \( C \) as it is the penalty parameter. The \( m^n \) SVC solves the following problem

\[
\min_{w, w^m, \xi = 0, \cdots , c_n} \frac{1}{2} \sum_{i=1}^{Q_i} \xi_i^m + C \sum_{i=1}^{Q_i} \xi_i^m
\]

\[
(w^m)^T \phi(q_u) + b^m \geq 1 - \xi_i^m, \text{ if } cat_i = m,
\]

\[
(w^m)^T \phi(q_u) + b^m \leq -\xi_i^m, \text{ if } cat_i \neq m,
\]

\( \xi_i^m \geq 0, i = 1, \ldots , Q_i, \)

(3)

where the training questions \( q_u \), are mapped to a higher dimensional space by the function \( \phi \) and \( C \) is the penalty parameter. Minimizing \( 1/2 ||w^m||^2 \) \( w^m \) 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 \( C \sum_{i=1}^{Q_i} \xi_i^m \) that reduces some errors in the training set. The basic concept behind SVM is to search for a balance between the regularization term \( 1/2 ||w^m||^2 \) \( w^m \) and the training errors. Calculating (3), there may be \( k \) decision functions:

\[
(w^j)^T \phi(x) + b^j,
\]

\[
\cdots
\]

\[
(w^k)^T \phi(x) + b^k.
\]

We say \( Q_i \) is present in class that has the biggest value of the decision function:

\[
\text{class of } Q_i = \arg \max_{m=1, \ldots , Q_i} ((w^m)^T \phi(Q_i) + b^m).
\]

(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 \(< \text{question}, \text{question category}> \) 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:

\[
\text{Odds} = \frac{p}{1-p}
\]

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.

\[
\log(p) = \log(\text{odds}) = \log \left( \frac{p}{1-p} \right)
\]

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

\[
\logit(p) = \beta_0 + \beta x_i
\]

That is, the \( \log \text{ odds} \) (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 \( y = \beta_0 + \beta x \) where \( x \) is a feature vector and \( \beta \) 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-ibilities 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

---

 DOI: 10.35940/ijeat.B2993.0402

\[\text{Published By:}\]

Blue Eyes Intelligence Engineering & Sciences Publication
similarity distance between the probabilities calculated from the softmax function and the target one-hot-encoding matrix.

**Parameter optimization:** The expected output after 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 and these answers are segregated with a list called answer list.

\[ \text{An}_{\text{list}} = \{(a_1, a_{\text{cat}1}), (a_2, a_{\text{cat}2}), \ldots, (a_m, a_{\text{cat}m})\} \]

An 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 \( \text{docan}_{\text{list}} = \{(\text{word}_{i1}, a_{\text{cat}}), (\text{word}_{i2}, a_{\text{cat}}), \ldots, (\text{word}_{ip}, a_{\text{cat}})\} \).

Frequent distribution is applied on the list of document represented by \( \text{docan}_{\text{list}} \) to obtain \( f_{\text{req}_{\text{list}}} \) that consists of a number of words that frequently occurs in the documents. \( f_{\text{req}_{\text{list}}} \) is given by \( f_{\text{req}_{\text{list}}} = \{(\text{word}_{i1}, x_1), (\text{word}_{i2}, x_2), (\text{word}_{ip}, x_i)\} \), where \( x_i \) is term frequency of \( \text{word}_i \) in the document.

For the current task, a answer sentence is represented as a vector of features and trained 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 \( F_{\text{set}} \) 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 probability 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 \( P(x|y) \), a feasible answer \( y_i \), 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:

\[ P(x|y) = \sum P(x|y, T = t_i)P(T = t_i|y) \]  \( (5) \)

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,

\[ P(x|y) = \sum P(x|T = t_i)P(T = t_i|y) \]  \( (6) \)

Using Bayes’ Theorem,

\[ P(x|y) = P(x) \sum \frac{P(T = t_i|x)P(T = t_i|y)}{P(T = t_i)} = \frac{1}{5} \]  \( (7) \)

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 q\(_i\) and if the answer classification, produces a same category a\(_i\) 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 probability 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 q\(_{cat}\) = {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.

\[ \frac{\text{Accuracy}}{\text{Number of prediction}} = \sum \frac{\text{Number of Correct prediction}}{\text{Efficiency}} \]

<table>
<thead>
<tr>
<th>Sl. no.</th>
<th>Classifiers</th>
<th>Number Of question</th>
<th>Correctly Classified</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Naive Bayes Classifier</td>
<td>375</td>
<td>281</td>
<td>75%</td>
</tr>
<tr>
<td>2.</td>
<td>Multinomial Naive Bayes</td>
<td>375</td>
<td>263</td>
<td>70%</td>
</tr>
<tr>
<td>3.</td>
<td>Bernouli</td>
<td>375</td>
<td>274</td>
<td>73%</td>
</tr>
<tr>
<td>4.</td>
<td>Multinomial Logistic Regression</td>
<td>375</td>
<td>323</td>
<td>86%</td>
</tr>
<tr>
<td>5.</td>
<td>Linear SVC</td>
<td>375</td>
<td>330</td>
<td>88%</td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>Sl.No.</th>
<th>Classifiers</th>
<th>Number of Answer</th>
<th>Correctly Classified</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Naive Bayes Classifier</td>
<td>375</td>
<td>274</td>
<td>73%</td>
</tr>
<tr>
<td>2.</td>
<td>Multinomial Naive Bayes</td>
<td>375</td>
<td>244</td>
<td>65%</td>
</tr>
<tr>
<td>3.</td>
<td>Bernoulli</td>
<td>375</td>
<td>285</td>
<td>76%</td>
</tr>
<tr>
<td>4.</td>
<td>Multinomial Logistic Regression</td>
<td>375</td>
<td>319</td>
<td>85%</td>
</tr>
<tr>
<td>5.</td>
<td>Linear SVC</td>
<td>375</td>
<td>300</td>
<td>80%</td>
</tr>
</tbody>
</table>

c) Extracting text Summaries for related CQA answers: In this work, the text summarization 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 grouped 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. In order 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

\[
Recall = \frac{\sum \text{Overlap}(GR_{pos}, ACT\text{SMLT}_{jpos})}{|GR_{pos}|} \\
Precision = \frac{\sum \text{Overlap}(GR_{pos}, ACT\text{SMLT}_{jpos})}{|ACT\text{SMLT}_{pos}|} \\
F\text{ Score} = \frac{2 \times Recall \times Precision}{Recall + Precision}
\]

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

<table>
<thead>
<tr>
<th>Sl.No</th>
<th>Method</th>
<th>Recall</th>
<th>Precision</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>ACTSMLT Description</td>
<td>0.6010</td>
<td>0.9978</td>
<td>0.7501</td>
</tr>
<tr>
<td>2.</td>
<td>ACTSMLT Entity</td>
<td>0.5132</td>
<td>0.6782</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>ACTSMLT Location</td>
<td>0.5671</td>
<td>0.9695</td>
<td>0.7156</td>
</tr>
<tr>
<td>4.</td>
<td>ACTSMLT Numeric</td>
<td>0.5932</td>
<td>0.9997</td>
<td>0.7445</td>
</tr>
<tr>
<td>5.</td>
<td>ACTSMLT Person</td>
<td>0.5836</td>
<td>1</td>
<td>0.7370</td>
</tr>
<tr>
<td>6.</td>
<td>MK++ Description</td>
<td>0.5936</td>
<td>0.9683</td>
<td>0.7593</td>
</tr>
<tr>
<td>7.</td>
<td>MK++ Entity</td>
<td>0.5600</td>
<td>0.9885</td>
<td>0.7067</td>
</tr>
<tr>
<td>8.</td>
<td>MK++ Location</td>
<td>0.5560</td>
<td>0.9857</td>
<td>0.7034</td>
</tr>
<tr>
<td>9.</td>
<td>MK++ Numeric</td>
<td>0.5826</td>
<td>0.9982</td>
<td>0.7557</td>
</tr>
<tr>
<td>10.</td>
<td>MK++ Person</td>
<td>0.5796</td>
<td>0.9991</td>
<td>0.7336</td>
</tr>
</tbody>
</table>

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
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 knowledge extract knowl-edge. In order to provide correct answers from a large text sources, one need to understand the question to suggest a perfect answer. Therefore, it is necessary to correctly provide the answers 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
AUTHORS PROFILE

Ramya R S is currently research scholar in the department of Computer Science Engineering, University Visvesvaraya 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 Visveswaraya 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 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, IEE 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 India; Fellow the Academy of Science for the Developing World and a Fellow of IEEE.