2015 Seventh International Conference on Computational Intelligence, Modelling and Simulation

NLP Algorithm Based Question and Answering System

Sanglap Sarkar
Global Technology Office.
Cognizant Technology Solutions
Chennai, India
sanglap.sarkar@cognizant.com

Venkateshwar Rao Madasu
Global Technology Office.
Cognizant Technology Solutions
Chennai, India
Venkateshwar.madasu@cognizant.com

Baala Mithra SM
Global Technology Office
Cognizant Technology Solutions
Chennai, India
baalamithra.sm@cognizant.com

Subrahmanya VRK Rao
Global Technology Office
Cognizant Technology Solutions
Chennai, India
Subrahmanyavrk.rao@cognizant.com

***Abstract*— Question and Answering (QA) systems are referred as virtual assistants and are envisioned to be the next generation call center. However the accuracy of such QA systems is not as desirable and needs significant enhancement. Understanding the intent of the query is a significant contributor to an efficient system which has not been often analyzed. The current study intends to develop a QA system which can understand the query intent by using NLP based classification along with a novel scoring mechanism to extract the related information. Initially an insurance domain based simple frequently asked QA system was developed, which was then operationally enhanced into a system that can answer any type of insurance related query. The system was tested with 200 different queries and the output was cross validated by 3 experts from insurance field.**

Keywords-QA system; NLP algorithms; semantic similarity;user intent

I. INTRODUCTION

Evolution of web from a read only to read write mode has made way for a huge load of information in the form of knowledge bases. Wikipedia, Freebase, YAGO, Microsoft Satori and Google Knowledge Graph are some of the well-known knowledge bases [1]. Information present in them could be used to build specific decision making /advisory systems. QA systems, which are a part of advisory systems are viewed as futuristic replacement of call centers and are called as virtual assistants. QA systems generally are classified based on the type of queries asked by the user and by the way system retrieves information while responding to the queries. While the former could be again classified as supervised (frequently asked questions (FAQ)) and unsupervised (generic questions) the latter could be classified based on logical reasoning, semantic understanding or plain key word matching. Most of the systems reported in literature are FAQ/key word matching type while semantic /logical reasoning systems have been rare [2-11]. Common objective of all QA systems has been to find a relevant response to a precise natural language query. In spite of the recent advances, accuracy and performance are the two cardinal areas in search query processing where there is still a huge scope for enhancement.

Accuracy issues could be attributed to the fact that the typical queries are generally insufficient and don't completely describe the user's need. Hence to classify the huge content into predefined categories presents a huge challenge. Literature reports use of machine learning algorithms to train a classifier and predict the category of an input query [12-15]. However accuracy of such systems could be enhanced only when both discriminative features as well as sufficient sample size co exists, which is a rarity in a real world scenario. It must be noted that an ideal system should be context aware and be able to respond to the queries with high accuracy. Hence understanding the intent of the user is important for providing relevant responses to the user queries. Another significant factor that has to be taken care of is the ever growing size of the content. Optimal method of indexing the content and scaling the solution is also as important as the response of these systems. However with recent advances in cloud and distributed computing the scalability part could be solved.

QA systems have evolved from a very generic solution provider to be more specific to a particular domain. Healthcare and retail are the domains that have started to deploy these systems [16-18]. Primary objective of the current study is to develop a context aware QA system using an improved approach that would be able to provide relevant responses using algorithms in supervised and unsupervised model followed by a novel scoring mechanism.

II. METHODOLOGY

QA system that has been developed is an intranet solution. The user can ask queries either by typing using the search box in the user interface or through a voice input. Google API was used to convert the voice input into text and perform the necessary operations on the query. The system had two modes namely supervised and unsupervised, both of which have been explained.

Knowledge base was constructed by crawling FAQ public websites of insurance companies and stored in a number of flat files. All the possible queries were labelled with help of an expert. Responses to the user queries were based on key word matching.

2166-8523/15 $31.00 © 2015 IEEE
DOI 10.1109/CIMSim.2015.29

System would be able to provide response to a predefined structure/query only. Handling question ambiguity which is heart of natural language question was a major drawback of this supervisory mode of the system and hence context-centric natural language processing (NLP) algorithms were designed and developed to make it viable in real world scenario.

A. *Classifying the query*

Intention of query needs to be understood to have an accurate QA system. Hence query provided by the user was replaced with equivalent synonyms by comparing each important key word present in the query with WordNet library .This was done to produce as many variations for a single query using natural language. Subsequently the query was tokenized using Stanford Open NLP tool [19]. POS tagging was performed using POS tagging Stanford parser [20]. Important words of the query were extracted using a stop word database. Stop word database contains a list of high-frequency irrelevant words which were required to make the sentence grammatically correct.

Standard phrases, which represent a specific type of response (numerical or logical), were stored in a database file. Intent of the query could be understood by identifying the starting phrase or words of the question. The current study analyzed the starting phrase of the query to verify whether there were standard phrases such as “what” or “how much”, by matching the predefined standard phrases. A typical example would be the query “What is health insurance in brief?” where the intension of the user would be to understand the general information about health insurance. Whereas, query similar to “How much is the sum assured for XYZ policy?” would be expected to get a user specific numerical response as answer. If similar type of predefined standard pattern exists, the query/sentence is given a score $s_1 > k$, otherwise it is given a score $s_1 < k$ where constant k is determined by trial and error process by training the QA system with FAQs). Depending on the scores the user intention can be broadly classified as generic (queries with implicit intent related to an insurance type) and specific type (queries that require information rooted from multiple issues that are user-specific as well as insurance-type specific).

Challenges of analyzing a question include the following:

a. Constraints imposed on the vocabulary and syntax thereby stemming into databases of knowledge used by the QA system. The constraints might be in terms of form-filling. This limits the expressivity of the user.

b. Presence of ellipsis or anaphora requires understanding the context and the intent of the query. This enables the system to use the knowledge base of the QA system for interpreting the user’s query and his/her goals of getting relevant information.

Few techniques reported in the literature are:

a. Moldovan et al. (2000) manually constructed a question type hierarchy of about 25 types from the analysis of the TREC-8 training data.

b. Srihari and Li (2000) base their question type hierarchy on an extension of the MUC named entity classes and use a shallow parser to identify the question type, or what they call the asking point.

c. Hovy, Gerber, Hermjacob, Junk and Lin (2001) constructed a QA typology of 47 categories based on an analysis of some 17,000 `real' questions.

Processing of generic query ($s_1 > k$):

This involves the following steps:

a. Parse question

b. Determine repeated query point

c. Elaborating questions

Processing of specific query ($s_1 < k$):

This requires

a. Background knowledge indexing

b. Mapping the constraint and tag-set of a simplified ones used by the system

B. *Extraction of relevant answer to the query*

Usually content is voluminous and hence identification of relevant information for the given query is a challenge. Majority of the reported QA systems have focused on structural data and the queries were mainly straight forward, which might not be viable in real time enterprise scenario. Knowledge base for the current study was an unstructured and a two stage methodology has been adopted to extract relevant answer to the query. The study also considered how informative, relevant and dissimilar the responses were in comparison to other probable answers. The two-stage process which reduces the search space significantly, making it easily scalable in big data landscape, has been described below.

a) First Stage

Identifying areas of interest from a voluminous content for a particular query is objective of this stage. First step would be to identify the subject –verb-predicate (SVP) triplets of the query. This was determined using Apache OpenNLP tool. Subsequently each word of the query was stemmed into a basic or primitive word using WordNet. Nouns, verbs, adjectives and adverbs which were extracted from the query phrase were grouped into sets of context-centric synonyms, each expressing a distinct concept. Content which had the similar set of context-centric synonyms were then extracted. Therefore for each query depending on the context, the extracted contents/areas of interest from different documents might be relevant or irrelevant. These extracted contents were then ranked according to the following steps.

Extracted contents could be in paragraphs or sentences. Context-centric synonym matching was performed between the query words and the sentences that are present in the extracted paragraphs. Each such sentence was given a score s_2 based on SVP matching. Some of these sentences would be directly related to the given query (which could be either generic or specific) where as some of them might be related to the sentence rather than the query. Hence these sentences that are not directly related to the query were assigned a relatively lower score.

By adopting a simple mathematical formula explained below all the extracted paragraphs were given a score. If a paragraph score was more than the threshold, then that paragraph was retrieved as areas of interest to the given query.

Total score of a sentence

The total score S_j^k of j^{th} sentence of k^{th} paragraph was calculated by summing up the weighted scores as follows

$$S_j^k = \sum_{i=1}^{2} w_i s_i \quad (0 \leq w_i \leq 1) \qquad (1)$$

Where w_1 is the weightage given to the score obtained by analyzing the user intention and w_2 is the weightage given to the score obtained through SVP matching. In case, the query was of generic type, the SVP matching was given more priority ($w_1 < w_2$). On the other hand, if the user query was of specific type, the score obtained based on user intention i.e. s_1 was given more priority and hence, in this case $w_1 > w_2$. The exact values of the weightage $\{w_1, w_2\}$ was obtained through a trial and error process by training the QA system with FAQs.

Total score of a paragraph

Total score of a paragraph S_k was obtained by adding up individual sentence scores as follows,

$$S_k = \sum_{j=1}^{N} S_j^k \tag{2}$$

Where N is the total number of sentences in the paragraph.

Paragraph Extraction:

Paragraphs for which total score were above a given threshold were extracted. Average paragraph score was set up as the threshold. Therefore, the l^{th} paragraph is extracted if

$$S_l \geq \frac{1}{p} \sum_{k=1}^{p} S_k \tag{3}$$

b) Second Stage

After identifying the related paragraphs, specific sentences which are relevant to the given query were extracted using a scoring mechanism. Features such as query sentence similarity, sentence length, word relevance, sentence position, cosine similarity between sentences, existence of numerical data were used to determine relevance score. The scoring mechanism which was developed for the study has been detailed below.

Question-sentence similarity score:

The user query and a particular sentence of an extracted paragraph are stemmed and the stop words are removed. Now, the numbers of word matches are calculated.
The score is calculated as

$$s_1' = \frac{\text{the numbers of important word matches between sentence and query}}{\textit{the number of important words in query}}$$

Sentence Length:

Normalized sentence length was calculated as follows

$$s_2' = \frac{\text{the numbers words in sentence}}{\text{the numbers words in longest sentence}}$$

Short sentences (e.g. author names, subtitle, date etc) may not contain useful information in case of a generic query. However, they may contain useful information in case of specific query. Accordingly, weightage w_2' was assigned depending on the query type.

Word relevance:

A sentence can be considered to be important if it contains words that occur frequently in the document. Term frequency and inverse sentence frequency (*tf.idf*) were computed as described in an earlier study [21]. Subsequently score s_3' was given to each sentence.

Sentence position:

In case of a generic query, the first few sentences of a paragraph may contain more information. However in case of a specific query, the sentences adjacent to the sentence containing specific information may contain more information. Accordingly score s_4' was assigned and the weightage depended on the type of query (i.e. generic or specific).

Cosine-similarity between the sentences:

The word relevance was calculated for all the words of two set of sentences. Cosine similarity between the two sentences was computed by appropriate zero-padding and using dot product.

Proper noun extraction:

A sentence with more proper nouns may be important and hence was considered to be a part of answer segment. In this context, the score was calculated as

$$s_5' = \frac{\text{the numbers proper nouns in sentence}}{\text{the numbers words in sentence}}$$

Existence of numeric data:

The score was calculated as the ratio of the number of numerical data that occurs in sentence over the total number of words in the sentence.

$$s_6' = \frac{\text{the numbers of numerical data in sentence}}{\text{the numbers words in sentence}}$$

In case of specific numerical query, this feature was given more weightage. Accordingly weightage w_6' was assigned.

Evaluate relevance score:

The total score S'^k_j of j^{th} sentence of k^{th} paragraph was calculated by summing up the weighted scores as follows

$$S'^k_j = \sum_{i=1}^{6} w_i' s_i' \tag{4}$$

Where, w_i' is the weightage given to the score s_i' and $0 \leq w_i' \leq 1$.

Sentence extraction:

A sentence was considered to be a part of answer segment if its score was above a particular threshold. Average sentence score across all the sentences of extracted paragraphs was considered as the threshold for the study. Hence, the j^{th} sentence of k^{th} paragraph would be a candidate for the probable answer segment if

$$S'^k_j > \frac{1}{M} \sum_k \sum_j^{N_k} S'^k_j \tag{5}$$

where N_k is the number of sentences in k^{th} paragraph and *M* is the total number of sentences in the extracted paragraphs. The sentences which are above the threshold would be provided as response to the given query.

Flow diagram of the whole process of understanding the user intension and extracting the relevant information is shown (see Figure 1).

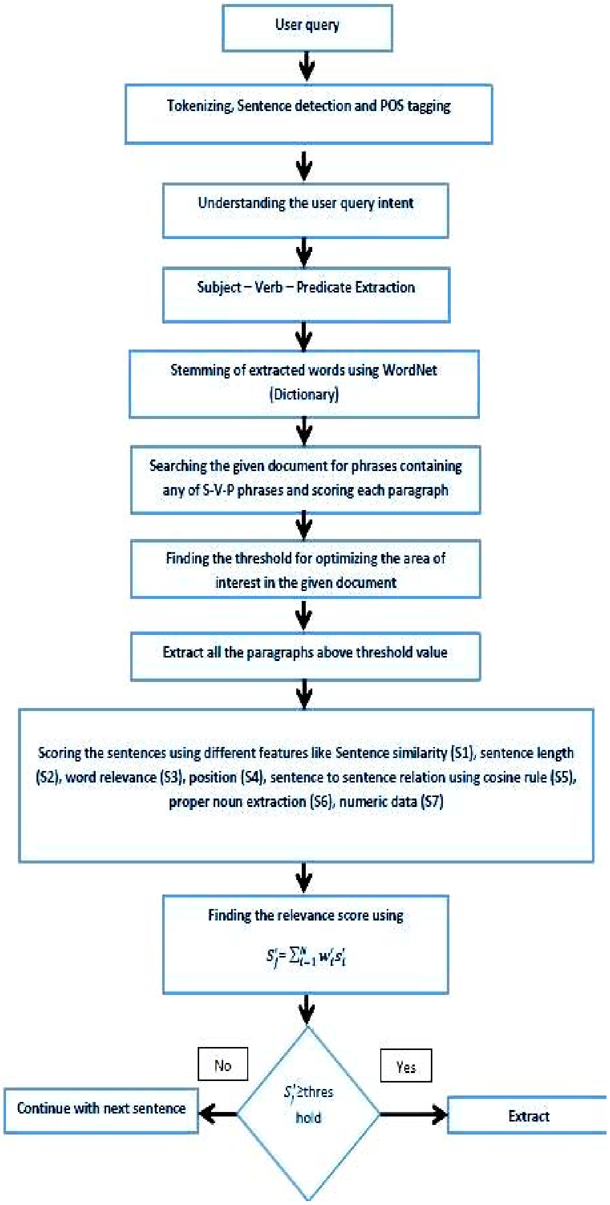

User query
Tokenizing, Sentence detection and POS tagging
Understanding the user query intent
Subject – Verb – Predicate Extraction
Stemming of extracted words using WordNet (Dictionary)
Searching the given document for phrases containing any of S-V-P phrases and scoring each paragraph
Finding the threshold for optimizing the area of interest in the given document
Extract all the paragraphs above threshold value
Scoring the sentences using different features like Sentence similarity (S1), sentence length (S2), word relevance (S3), position (S4), sentence to sentence relation using cosine rule (S5), proper noun extraction (S6), numeric data (S7)
Finding the relevance score using
$S'_j = \sum_{i=1}^{N} w'_i s'_i$
No
Yes
$S'_j \geq$thres hold
Continue with next sentence
Extract

Figure 1. Process flow chart of unsupervised mode of QA System

III. EVALUATION OF QA SYSTEM

A set of 200 queries (100 generic and 100 specific) and answers for them were framed from a randomly selected document. For each topic one query was selected and the corresponding answer were either extracted from the document directly or manually written by understanding the concept. This manually documented set was taken as reference for evaluating our algorithm along with cross validation performed by 3 insurance domain experts. Every query from the set of 200 was provided as input to the system. Metrics such as precision, recall, F-measure, accuracy and rejection were computed to evaluate the system efficiency. F-measure was further grouped into two namely F1 and f1 as was done by a previous study on word pair similarity [22]. F1 is defined as uniform harmonic mean of precision and recall, whereas f1 is defined as uniform harmonic mean of rejection and recall. System was expected to analyze and apply aforementioned logic and extract the possible short answer that is related to the user query. The responses were provided to 3 experts in insurance domain for validating the system accuracy.

IV. RESULTS

An example computation of the metrics for a query, which was expected to retrieve 40 sentences as per the three experts, is explained below.

Out of these 40 sentences 15 were bound to be related and the remaining unrelated. However the current QA system extracted only 20 sentences for that query. Subsequently all the extracted sentences were subjected to expert verification. It was verified that 9 sentences were related to the query and remaining were not related.

Based on the standard definitions, all the related metric were found to be: Recall=0.6, Precision=0.45, Rejection=0.44, Accuracy=0.225, F1=0.514 and f1=0.507. Mean metric scores for all the queries that were put forth to evaluate the system is shown (see Table 1).

TABLE I. TABLE TYPE STYLES

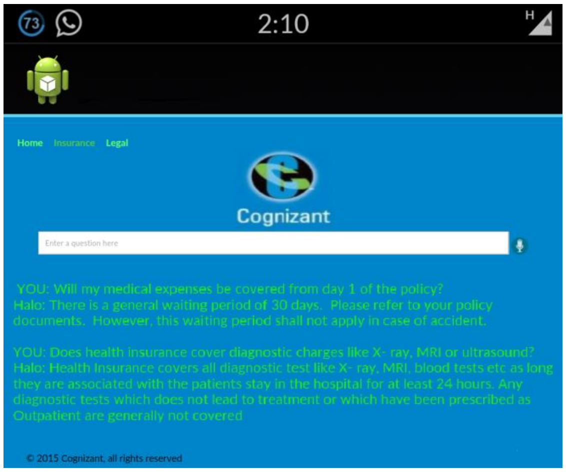
Metrics	**Query Type**	
	Generic	***Specific***
Recall	0.6±0.025	0.8±0.02
Precision	0.45±0.04	0.667±0.08
F1	0.514±0.03	0.727±0.023
Accuracy	0.225±0.025	0.5±0.06
Rejection	0.44±0.08	0.667±0.02
f1	0.507±0.06	0.727±0.02

V. CONCLUSION

Accuracy is a major limitation in most of the QA systems. Understanding the intent of user could determine the accuracy of the QA system response. The study provides a novel way of understanding the intent of the query and provides a scoring mechanism to identify related contents and extract relevant information from then for a given query.

This QA system has been integrated with a Mobile Chat Application, where the users authenticate and queries (See Figure.2). FAQ types of questions are answered directly and user specific questions are answered after getting policy numbers and authentication.

This application can be used by customer care agents to handle telephonic calls whereas they don't have to check database manually for answers and are handled by this application with quick response. In this scenario, the relevance feedback from the agent can act as an input to improve the accuracy of the system. In-case of a new modified answer, that corresponding question – answer pair can be fed to the system for training the FAQ model.



YOU: Will my medical expenses be covered from day 1 of the policy?
Halo: There is a general waiting period of 30 days. Please refer to your policy documents. However, this waiting period shall not apply in case of accident.

YOU: Does health insurance cover diagnostic charges like X- ray, MRI or ultrasound?
Halo: Health Insurance covers all diagnostic test like X- ray, MRI, blood tests etc as long they are associated with the patients stay in the hospital for at least 24 hours. Any diagnostic tests which does not lead to treatment or which have been prescribed as Outpatient are generally not covered

© 2015 Cognizant, all rights reserved

Figure 2. Screen-shot of QA System as a mobile App

The accuracy of the current system suffers due to the ambiguities involved in the user queries and the data available for answering. As a future work, to improve the accuracy of the system, the automatic tagging of the data need to be improved with more accurate anaphora resolution and domain specific Sematic Role Labelling.

REFERENCES

[1] Dong, X. L., Murphy, K., Gabrilovich, E., Heitz, G., Horn, W., Lao, N. & Zhang, W. (2014). *Knowledge Vault: A Web-scale approach to probabilistic knowledge fusion*.

[2] Pal, Saurabh, et al. "A Framework For Automatic Generation Of Answers To Conceptual Questions In Frequently Asked Question (FAQ) Based Question Answering System." *International Journal of Advanced Research in Artificial Intelligence* 1.2 (2012).

[3] Shaikh, Anwar D., et al. "Improving Accuracy of SMS Based FAQ Retrieval System." *Multilingual Information Access in South Asian Languages. Springer Berlin Heidelberg*, 2013. 142-156.

[4] Moreo, A., Navarro, M., Castro, J. L., & Zurita, J. M. (2012). A high-performance FAQ retrieval method using minimal differentiator expressions. *Knowledge-Based Systems*, 36, 9-20.

[5] Chen, L., & Shen, R. (2011, October). Faq system in specific domain based on concept hierarchy and question type. *In Computational and Information Sciences (ICCIS), 2011 International Conference on* (pp. 281-284). *IEEE.*

[6] Kabakus, A. T., & Cetin, A. (2014). A Question and Answer (Q&A) System to Extend Capabilities of Distance Education. *Global Journal of Computer Science*, 4(1).

[7] Shaw, Ruey-Shiang, Chin-Feng Tsao, and Pei-Wen Wu. "A study of the application of ontology to an FAQ automatic classification system." *Expert Systems with Applications* 39.14 (2012): 11593-11606.

[8] Graesser, Arthur C., et al. "Question answering and generation." *Applied NLP*. IGI Global, Hershey (2011).

[9] Unger, Christina, and Philipp Cimiano. "Pythia: Compositional meaning construction for ontology-based question answering on the Semantic Web". *Natural Language Processing and Information Systems. Springer Berlin Heidelberg*, 2011. 153-160.

[10] Moschitti, Alessandro, and Silvia Quarteroni. "Linguistic kernels for answer re-ranking in question answering systems." *Information Processing & Management* 47.6 (2011): 825-842.

[11] Schneider, Michael, and Geoff Sutcliffe. "Reasoning in the OWL 2 full ontology language using first-order automated theorem proving." *Automated Deduction–CADE-23. Springer Berlin Heidelberg*, 2011. 461-475

[12] Beitzel, S. M., Jensen, E. C., Frieder, O., Lewis, D. D., Chowdhury, A., & Kolcz, A. (2005, November). Improving automatic query classification via semi-supervised learning. *In Data Mining, Fifth IEEE international Conference on* (pp. 8-pp). *IEEE.*

[13] Broder, A. Z., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., & Zhang, T. (2007, July). Robust classification of rare queries using web knowledge. *In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 231-238). *ACM.*

[14] Shen, D., Sun, J. T., Yang, Q., & Chen, Z. (2006, August). Building bridges for web query classification. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 131-138). ACM.

[15] Shen, D., Pan, R., Sun, J. T., Pan, J. J., Wu, K., Yin, J., & Yang, Q. (2005). Q 2 C@ *UST: our winning solution to query classification in KDDCUP 2005. ACM SIGKDD Explorations Newsletter*, 7(2), 100-110.

[16] Cao, YongGang, et al. "AskHERMES: An online question answering system for complex clinical questions." *Journal of biomedical informatics* 44.2 (2011): 277-288.

[17] Athenikos, Sofia J., and Hyoil Han. "Biomedical question answering: A survey." *Computer methods and programs in biomedicine* 99.1 (2010): 1-24.

[18] Janzen, Sabine, and Wolfgang Maass. "Ontology-based natural language processing for in-store shopping situations." *Semantic Computing, 2009. ICSC'09. IEEE International Conference on. IEEE, 2009.*

[19] "The Stanford NLP (Natural Language Processing) Group." *The Stanford NLP (Natural Language Processing) Group.* N.p., n.d. Web. 06 Jan. 2015.

[20] "Apache OpenNLP Developer Documentation." *Apache OpenNLP Developer Documentation.* N.p., 06 Jan. 2015. Web. 06 Jan. 2015.

[21] Suanmali, L., Salim, N., & Binwahlan, M. S. (2009). Fuzzy logic based method for improving text summarization. arXiv preprint arXiv:0906.4690.

[22] Manna, S., & Mendis, B. S. U. (2010, July). Fuzzy word similarity: a semantic approach using WordNet. *In Fuzzy Systems (FUZZ), 2010 IEEE International Conference on* (pp. 1-8). *IEEE.*