

# DEEP LEARNING APPROACHES FOR ANSWER SELECTION IN QUESTION ANSWERING SYSTEM FOR CONVERSATION AGENTS

K. Karpagam<sup>1</sup>, K. Madusudanan<sup>2</sup> and A. Saradha<sup>3</sup>

<sup>1,2</sup>Department of Computer Applications, Dr. Mahalingam College of Engineering and Technology, India

<sup>3</sup>Department of Computer Science and Engineering, Institute of Road and Transport Technology, India

## Abstract

*The conversation agent acts as core interfaces between a system and user in answering users queries with proper responses. Question answering system acquires an important role in the information retrieval field. The deep learning approach enhances the accuracy in answering complex questions. As outcome, the user is receiving the precise answer instead of large document collections. The aim of this paper is to develop a model with deep learning approach for improving answer selection process which supports more relevant answer displaying by conversation agents. To achieve this, word2vector used for word representation and biLSTM attentive model is used for training, testing and disclosure play precise answer. Question type is identified using POS-tagger based Question Pattern analysis (T-QPA) model. The knowledgebase is created from the bench mark datasets bAbI Facebook (simple QA tasks), TREC QA, Yahoo! Answer, Insurance QA dataset. The proposed framework is built by embedding of questions and answers based on bidirectional long short-term memory (biLSTM) attentive models. The similarity between questions and answers has been measured by semantic and cosine similarity. The proposed model reduces the search gap in extracting among user queries and answer sentences in the education domain. The system results are evaluated with the standard metrics MAP, Top 1 accuracy, F1- Score for the answer selection.*

## Keywords:

*Deep Learning, Question Answering System, Attentive Model, Conversation Agents, Cosine Similarity*

## 1. INTRODUCTION

The growing interest and need for user interface generation between system and user lead to the development of conversation agents. Conversation agent is used as an interface between the system and user for information transformation. The system is user- initiated communication with the Conversation agent by the natural language, gestures, speech with free of cost, access at anytime, anywhere. The system handles the Natural language question using NLP steps such as parsing, tokenization, stemming, keyword extraction etc [2] [16]. Some of the NLP Question answering systems are ASK JEEVES, START, and ANSWER BUS etc are examples of conversation agent. Conversation agent act as chatter bots for providing a precise answer using pattern matching, simple predefined rules, static databases, knowledge base etc.,

In traditional Question Answering System (QAS), steps such as question classification, Information retrieval and information extraction are used for precise answer extraction. QAS systems have required supporting various areas from academics to corporate for exacting precise information. The process is quite comfortable but response time and accuracy should be compensated. Recent days, deep learning is one of the most needed research fields in computer science. To overcome this deep learning facilitate in answering complex questions using

external knowledge, neural network for training and use classifier to seal the answers. The answer generation, deep learning is used for easy retrieval with a proper training and validation set.

## 2. RELATED WORKS

This paper [1] proceeds with the QAS development using deep learning in discussion about various approaches with basics of NLP and algorithmic techniques. Proposed model is implemented and evaluation done with twenty tasks of babI dataset of Facebook.

Turning test is the basic concept used for QAS that provide the answers based on the questions patterns identified using Artificial Intelligence and Machine Learning approaches. Search engines, Natural Language Question-Answering systems, Chatter bots and ECA follow this technology and the comparative studies are carried out for improvement [3].

The ALICE foundation is a widely used standard for creating chat bots with the supporting of artificial intelligence field. ALICE created the Artificial Intelligence Markup Language (AIML) software with scripts markup language which is used to develop chatter bot. The basic tags of AIML are <category> used for storing knowledge bases, <pattern> used to deal with user request; <template> used to deals with the system responses to the user [4].

In paper [5], proposed a deep learning model for answering complex questions based on answer sequence representation and passage answer selection. The author proposes a deep learning hybrid model with convolution & recurrent neural networks for passage-level question and answer matching with semantic relations. The results are trained, tested and evaluated with TREC-QA and Insurance QA datasets. In paper [6], authors discussed about the question classifiers for factoid QA which classifies the question type and able to provide the answers for 'Wh' type questions from various knowledge sources.

In paper [10] author evaluates the performance of proposed question answering system model with a database which consists of pair of questions and answers. AT&T chat data does not have label for question, answers pairs. The proposed model is trained and tested with Insurance QA Corpus.

In paper [13], authors significantly discuss on deep learning related models and methods used for various NLP tasks for answer selection. This paper also summarizes, compare and contrast the various existing models for detailed understanding on the various paradigms of deep learning in NLP.

In paper [14], authors propose a novel approach for distributed representations to match questions with answers by considering their semantic encoding. It has syntactic and semantic features of contexts which suits wide range of domains and languages. TREC

data set is used for experimenting the answer selection process training and evaluation.

### 3. SYSTEM ARCHITECTURE

The proposed system architecture describes various phases involved in complex QA system development such as question processing, question type identification such as who, when, what, which, how, why etc. The identified question type as complex the answer generation is done with the help of deep learning concepts as shown in Fig.1.

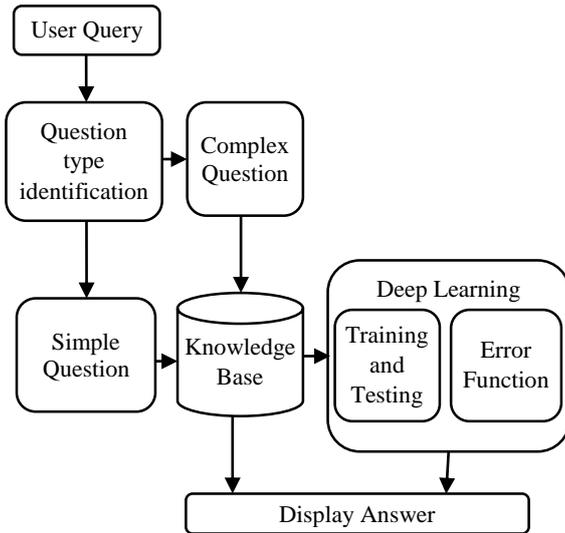


Fig.1. Architecture of QAS with Deep Learning

The question processing phase obtains the query input in natural language through an interface. To reduce the response time of candidate answers generation, the user query compares with the past historical QA pairs maintained in database. The query is pre-processed using tokenization, stop words removal and stemming to extract keywords. Question classifier is trained with pattern template formed by POS-tagger to identify the question types [15]. The question classifier of the system identifies the question types such as WH question (Factoid) and complex question using question pattern. Answers selection is done by keywords matching with user query, if query matching found in database answer has been displayed from knowledge base.

The complex question needs a descriptive answer by combining multiple sentences summarization from related documents. Examples of complex questions are Differentiate between volatile and nonvolatile memory (analysis question), Who is Gandhi? When he was born? (Multiple sentence question) etc.

Many of the existing summarization techniques do not use semantics of terms. A deep learning technique helps in deep analysis on answer summarization system with less error rate.

### 4. PROPOSED METHODOLOGY

The complex questions are tough to handle because the answer should be in a descriptive manner to satisfy the user query. Also the system has to deal with appropriate remembering previous

questions to answer appropriate. For e.g. who is Mahatma Gandhi? Where he is born? The proposed system uses deep learning for answer selection in QA systems using LSTM (Long short-term memory) with attentive model.

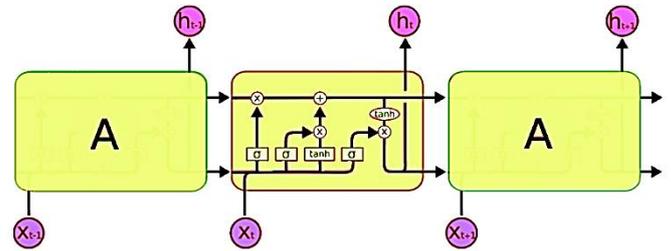


Fig.2. LSTM Architecture [9]

LSTM is an artificial recurrent neural network (RNN) architecture used in the field of deep learning for training the concept based on neural functions. The representation of question words is stored in low-dimensional vectors [11]. The meaning of the question and question type identification are found with use of hidden layers ( $h_i$ ) in the neural networks. Do recursively for all words such as  $h_1...h_4, h_1...h_3, h_1...h_2$  etc. Recursive network is used with help of dependency tree.

The LSTM attentive model is chosen rather than RNN due to its disadvantage of remembering the meaning and sequence of flow of questions. The deep learning is considered rather than RNN because the RNN is unable to store more words, match the keywords and not considered previous questions and synonymous of words. More over LSTM has the time stamp for each word such as  $t_1, t_2, ... t_n$ . In each layers the words with timestamp is passed with the help of dimensions hidden layers which get processed and pass to the next layer for computing the answer at last word.

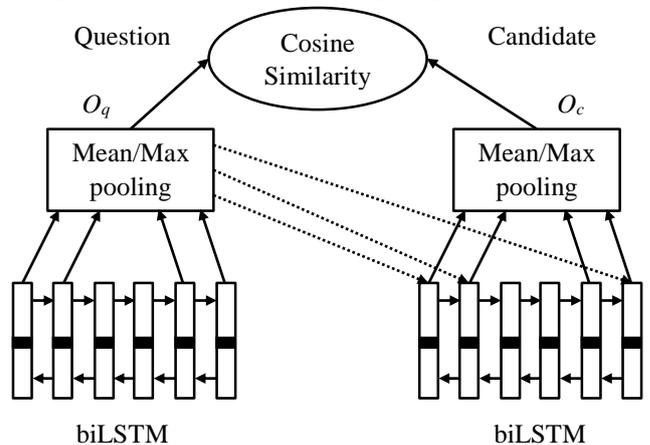


Fig.3. QA- LSTM with attention

The following Fig.3 describes the QA-biLSTM with attentive model for accurate answer selection from the retrieved answers from the datasets. The source for Fig.3 source is adapted from [12].

First, the biLSTM hidden vectors of answers  $h_a(t)$  are multiplied by  $s_{a,q}(t)$ , which is computed from the question average pooling vectors  $o_q$ , and updated to  $h_a(t)$ . Then, the original question and updated answer sentence is passed as an input word vector. The question context has been used to evaluate with softmax weights.

From the experiments, it is observed that performance improvement on different datasets are different due to its space and time complexity. Cosine similarity finds similarity between the user query and answer selection sentence from a document for answer appropriateness.

The cosine similarity is calculated for finding the semantic relatedness between the words with the summing of the vectors of all words in the text. The equation for computing the cosine similarity with comparing vectors  $u$  and  $v$  is given below,

$$sim(u, v) = \frac{\sum_{i=1}^N u_i * v_i}{\sqrt{\sum_{i=1}^N u_i^2} * \sqrt{\sum_{i=1}^N v_i^2}} \quad (1)$$

LSTM attentive model information flow is passed with the question and knowledgebase (processed dataset) for processing in the layers. The word2vector is used to represent the words; each word is represented as a column vector with 128 to 200 dimensions with bag-of-words and skip grams. Each row has a value position when it is stored.

The training phase has the input of questions  $q_i$  and list of sentence retrieved such as  $C\{c_{i1}, c_{i2}, c_{im}\}$  are used to calculate the score by using the empirical formula.

$$Score_j = h_0(q_i, c_{ij}) \quad (2)$$

$$S = \text{softmax}([Score_1, Score_2, \dots, Score_m]) \quad (3)$$

Softmax function as the output layer is used for calculating the accuracy of answer along with question co-occurrence in the same vector space. In that, recurrent neural networks will calculate the question information and predict the answer with softmax.

The answer can also be predicted with recursive neural networks which predict the answer selection through parse tree and evaluate with softmax. The proposed system uses advanced techniques Deep Averaging Network (DAN) for answer selection prediction with Softmax.

The added information is passed to the LSTM attentive model vectors for training and matching the question related with answers. The Attentive architecture uses the input sentence information to compute other form of answer representations.

Finally it is passed through softmax output layer for generation of answer predictions. To calculate the final prediction, sum of output layer  $o$ , input layer  $u$  are passed with weight matrix  $W$  and the softmax is produced [1].

$$Output = \text{softmax}(W(o+u)) \quad (4)$$

The attention model in the neural networks works for more data and networks is considered for answering complex questions. Proposed system was trained and tested with the benchmark dataset for answer selection.

Error function is determined for the following features such as Softmax for answer and question co-occurrence in same vector space and Max with (question  $q$ , correct answer  $a$ , incorrect answer  $b$ ). Extracted answers with high score will be displayed to the user through interface.

## 5. RESULT AND PERFORMANCE ANALYSIS

The dataset bAbI simple QA tasks [7] is pre-processed with NLTK Stanford parser was done in Java. To build the model deep learning libraries such as keras, pycamp, and tensorflow was used. The knowledge base is generated from the benchmark dataset SimpleQuestions (BABI Facebook) which are collected for research in automatic question answering with human generated questions [8].

Question types include yes/no questions, simple questions, and complex questions, so on. This dataset consists of a totally 108,442 questions written in natural language by human English-speaking annotators. It not only provides the answer but also a complete explanation with facts. From the dataset 70% taken as training set i.e. 75910 questions, 10% as validation set i.e. 10845 questions, and the remaining 20% as test set i.e. 21687 questions.

Some tweaks were made to the already defined models while training in order to obtain better results. The datasets TREC QA, Insurance QA, Yahoo! Answers are processed to form answer selection from the given knowledge bases. Different dataset yields different results. TREC- QA is taken from [18] which were submitted to Microsoft Encarta encyclopedia. TREC - QA consists of newspaper and newswire documents collection from various sources such as APnewswire, financial times, Los Angeles times etc. The TREC-9 QA dataset consist of attributes such as question ID, questions, document ID and judgment answer string.

The Mean Average Precision (MAP) is one of the popular performance measures in the field of information retrieval which is used to evaluate of ranked relevant documents retrieved with the average precision values

$$MAP = \frac{1}{n} + \sum_{Q_i} \frac{1}{R_i} \sum_{D_j \in R_i} \frac{j}{r_{ij}} \quad (5)$$

where  $r$  is the rank of the  $j^{\text{th}}$  relevant document in  $Q_i$ ,  $n$  is the number of test questions and  $R$  is the relevant document for  $Q_i$ . MAP is also calculated for various datasets like TREC QA, Insurance QA, Yahoo! Answer [17] with the number of documents for the given user query.

The result obtained for accuracy of sentence retrieval by the proposed system on various datasets has been evaluated by the standard metrics is as shown in the Table.1.

Table.1. MAP for Answer Selection Retrieval

No. of Documents	MAP based on Sentence retrieval			
	TREC QA	Insurance QA	Yahoo! Answer	bAbI
10	0.375	0.378	0.392	0.381
50	0.378	0.389	0.395	0.372
100	0.397	0.381	0.395	0.513
100	0.385	0.412	0.413	0.417
200	0.381	0.405	0.413	0.355

The comparison of results on existing model with the proposed model for various datasets is considered for evaluation. From the obtained results, it is proved that the proposed method is outperforming. The results are as shown in the Table.2.

Table.2. Comparison of Deep learning models

Method	MAP			
	TREC-QA	Yahoo! answers	Insurance QA	bAbI
QA-LSTM	0.682	0.701	0.691	0.671
LDC Model	0.771	0.751	0.699	0.717
IWAN	0.822	0.812	0.842	0.852
MCAN	0.838	0.842	0.827	0.839
Proposed QA-biLSTM with attention model	0.845	0.851	0.853	0.857

The results of proposed system have measured with the standard metrics for correct answer selection by top 1 accuracy. Threshold values analyzes point of change for retrieving the answers at various levels of abstraction up to 0.5 with 0.1 interval scale. The accuracy ranges of the answers from the retrieved resources 10, 15, 20, 25 and 50 have calculated for the threshold value  $t$ . The following Fig.4 shows the results obtained from various retrieved resources for answer generation with respect to threshold values.

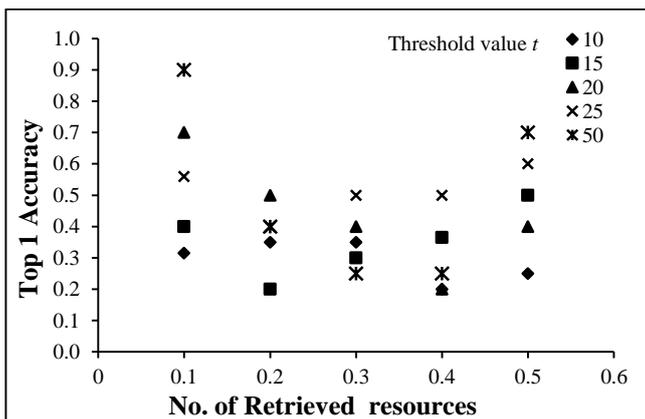


Fig.4. Top 1 accuracy on Threshold for Answer Selection

## 6. CONCLUSION

The proposed system objective is to develop a conversation agent using biLSTM attentive model work focused on analyzing, implementing and improving answer selection process in the field of Question Answering. The question type such as factoid or non-factoid questions is identified by POS-tagger based Question Pattern analysis (T-QPA). The semantic and cosine similarity is calculated the match between question and answer sentence generated, the highest score sentence is collected as a list. To retrieve the appropriate answer from the listed source, the system proposes the deep learning method bi-LSTM attentive model. The advantage of using this approach is to answer a complex question by remembering previous utterances and semantic meaning of the words for the given context analysis. For training and experimenting various bench mark datasets are used such as TREC- QA, Insurance QA, Yahoo! Answers and bAbI (Facebook simple QA tasks). The proposed system has produced the better results which was analyzed by the standard metrics SUCH AS MAP and top 1 accuracy. The further enhancement is to optimize

the results to increase better response and analysing emoticons inputs in a profound manner.

## REFERENCES

- [1] Yashvardhan Sharma and Sahil Gupta, "Deep Learning Approaches for Question Answering System", *Proceedings of International Conference on Computational Intelligence and Data Science*, Vol. 132, pp. 785-794, 2018.
- [2] James O. Shea and Zuhair Bandar Keeley, "Systems Engineering and Conversation Agents", Master Thesis, School of Computing, Manchester Metropolitan University, pp. 1-232, 2011.
- [3] G.O. Sing, K.W. Wong, C.C. Fung and A. Depickere, "Towards a more Natural and Intelligent Interface with Embodied Conversation Agent", *Proceedings of International Conference on Game Research and Development*, pp. 177-183, 2006.
- [4] Chatbots, "Alice (Artificial Linguistic Internet Computer Entity)", Available at: [chatbots.org/chatbot/a.l.i.c.e/](http://chatbots.org/chatbot/a.l.i.c.e/)
- [5] M. Tan, C.Dos Santos, B. Xiang and B. Zhou, "Improved Representation Learning for Question Answer Matching", *Proceedings of 54<sup>th</sup> Annual Meeting of the Association for Computational Linguistics*, pp. 464-473, 2016.
- [6] X. Li and D. Roth, "Learning Question Classifiers", *Proceedings of 19<sup>th</sup> International Conference on Computational Linguistics*, pp. 1-7, 2002.
- [7] The Babi Project, Available at: <https://research.fb.com/downloads/babi/>
- [8] Antoine Bordes, Nicolas Usunier, Sumit Chopra and Jason Weston, "Large-Scale Simple Question Answering with Memory Networks", *Proceedings of International Conference on Computation and Language*, pp. 1-10, 2015.
- [9] Analytics Vidhya, "Essentials of Deep Learning: Introduction to Long Short Term Memory", Available at: <https://www.analyticsvidhya.com/blog/2017/12/fundamentals-of-deep-learning-introduction-to-lstm/>
- [10] M. Feng, B. Xiang, M.R. Glass, L. Wang and B. Zhou, "Applying Deep Learning to Answer Selection: A Study and An Open Task", *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 1-10, 2015.
- [11] T.L. Lai, T. Bui and S. Li, "A Review on Deep Learning Techniques Applied to Answer Selection", *Proceedings of 27<sup>th</sup> International Conference on Computational Linguistics*, pp. 2132-2144, 2018.
- [12] Ming Tan, Bing Xiang and Bowen Zhou, "LSTM-based Deep Learning Models for Non-Factoid Answer Selection", *Proceedings of International Conference on Computational Language and Machine Learning*, pp. 1-7, 2016.
- [13] Tom Young, Devamanyu Hazarika, Soujanya Poria and Erik Cambria, "Recent Trends in Deep Learning based Natural Language Processing", *IEEE Computational Intelligence Magazine*, Vol. 13, No. 3, pp. 55-75, 2017.
- [14] Lei Yu, Karl Moritz Hermann, Phil Blunsom and Stephen G. Pulman, "Deep Learning for Answer Sentence Selection", *Proceedings of International Workshop on Deep Learning*, pp. 1-9, 2014.
- [15] K. Karpagam and A. Saradha, "A Framework For Intelligent Question Answering System using Semantic

- Context Specific Document Clustering and Wordnet”, *Sadhana-Academy Proceedings in Engineering Sciences*, Vol. 44, No. 3, pp. 1-10, 2019.
- [16] K. O. Shea, “An Approach to Conversational Agent Design using Semantic Sentence Similarity”, *Applied Intelligence*, Vol. 37, No. 4, pp. 558-568, 2012.
- [17] Ming-Wei Chang, Lev Ratinov, Dan Roth and Vivek Srikumar, “Importance of Semantic Representation: Data less Classification”, *Proceedings of 23<sup>rd</sup> AAAI Conference on Artificial Intelligence*, pp. 830-835, 2008.
- [18] Ellen M Voorhees, “The TREC Question Answering Track”, *Journal of Natural Language Engineering*, Vol. 7, No. 4, pp. 361-378, 2001.