Neural Reasoning Model For Question Answering

K.S.D.Ishwari[#1], A.K.R.R.Aneeze[#2], S.Sudheesan[#3], H.J.D.A. Karunaratne[#4], A. Nugaliyadde[#5], Y. Mallawarrachchi[#6]

[#]*Faculty of Computing, Sri Lanka Institute of Information Technology*
Sri Lanka

[1]it15067098@my.sliit.lk
[2]it15060372@my.sliit.lk
[3]it15109668@my.sliit.lk
[4]it15047748@my.sliit.lk
[5]anupiya.n@sliit.lk
[6]yashas.m@sliit.lk

***Abstract*— Question Answering has recently received high attention from artificial intelligence communities due to the advancements in learning technologies. Early question answering models used rule-based approaches and moved to the statistical approach to address the vastly available information. However, statistical approaches are shown to underperform in handling the dynamic nature and the variation of language. Therefore, learning models have shown the capability of handling the dynamic nature and variations in language. This paper presents a new network, named NRMQA(Neural Reasoning Model for Question Answering), which explores Reasoning, Attention and Memory factors in a Deep Neural Network. The model uses a LSTM for improved memory and RL is used for reasoning and attention. NRMQA is capable of complex reasoning and dynamic question-fact interaction (attention) on the complex memory it holds. The proposed network is tested on the bAbI 1K dataset and it has outperformed methods such as DMN and RRN. Particularly, NRMQA has achieved near-human performance when tested on the dataset.**

***Keywords*— Question Answering, Long Short-Term Memory Network, Reinforcement Learning**

I. INTRODUCTION

Question Answering (QA) has been a challenging task in natural language understanding [1]. The key components in QA require the capability of understanding the question and the context in which the question is generated. QA has been deemed challenging due to dynamic nature of natural languages [1]. This has resulted in the application of data-driven methods in question answering. The idea is to allow the data, instead of the methods, do most of the work in question answering. This is due to a large number of text repositories that is available [2].

Rule-based approach was one of the initially used most prominent methods for QA systems. These systems utilized rules devised from grammatical semantics to determine the correct answer for a given question. These rules are usually handcrafted and heuristic, relying on lexical and semantic hints on context [3]. These rules exploit predefined patterns that classify questions based on the answer type. These grammatical rules represent the context in the form of decision trees and this was used to find the path that leads to the correct answer [4].

A major drawback of rule-based question answering systems was that the heuristic rules needed to be manually crafted. To devise these rules an in-depth knowledge of the semantics of a language was a necessity [5]. With the rapid growth of text material available online the importance of statistical approaches for QA has also increased. These approaches lean on predicting answers based on data. As these methods are capable of addressing the heterogeneity of data and free from structured query languages they have been adapted to various stages of QA [6].

Statistical approaches require the formation of a hypothesis before proceeding to build the model. This hypothesis sets the tone for the creation of the model. With the advancements in machine learning systems gained the capability to navigate the direction that the data dictates [7]. These inducted the self-learning capability to the QA systems. These systems are capable of building a knowledge base (taxonomy) from the training data it is provided and then use it to answer the actual questions. This brought a level of independence to the systems that were not quite there in rule-based or statistical approaches. Furthermore, as these systems are capable of optimizing itself over time it became one of the most lucrative approaches for QA [8].

Induction of Neural Networks for QA systems opened up a plethora of possibilities. Conventional machine-learning techniques were limited in their ability to process natural data in their raw form [34]. For decades, constructing a machine-learning system required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data into a suitable internal representation from which the learning subsystem, often a classifier, could detect or classify patterns in the input [34]. Deep learning methods are representation learning methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification [34]. These are with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level [34]. With the composition of enough such transformations, very complex functions can be learned. For classification tasks, higher layers of representation amplify aspects of the input that are important for discrimination and suppress irrelevant variations [34].

Networks such as Dynamic Memory Networks [9], Reinforced-Memory Networks currently provide the state of the art results in Question Answering bringing Artificial Intelligence closer to human perception.

II. Background

A. Rule-based Approach

The initial question answering systems were logical representations of decision trees. The decision trees were linguistic structures that mirrored the way humans understand text based on grammatical rules. At the very beginning, all these rules were written by hand. These systems relied on the constant extension of functionality by the addition of rules that opened up new paths in the decision tree. QA systems based on such decision trees used lexical and semantic heuristics to see find evidence whether a sentence contains an answer to a question. These systems require rule sets that define paths for questions based on the question type. The path taken by the answer extraction process for "where is Tajmahal" would be different from "Who is on the one dollar bill". To improve the answer matching syntactic analysis, morphological analysis, Part-Of-Speech tagging and Named Entity Recognition were later incorporated to these systems.

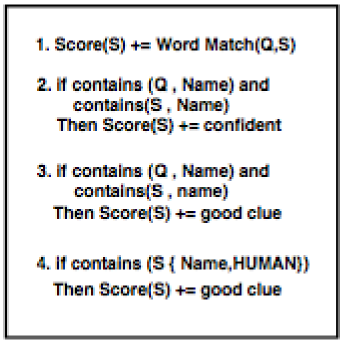

Fig. 1 Example of a rule-based algorithm

Though these systems were successful initiatives to the Question Answering domain they were not without flaws. Extensions to these systems required new rules to be introduced to the system which is cumbersome and makes these systems not suited for domains with highly volatile data [4].

As these new changes need to be explicitly programmed this slows down the development of the system considerably. These systems are highly reliant on the linguists who are creating the rules as these are unable to learn. These issues in the rule-based systems brought up the necessity for a self-learning approach that would solve the extensibility conundrum.

B. Statistical Approach

In the present research state, quick evolution in available online text repositories and web data has amplified the prominence of statistical approaches. These approaches put forward such techniques, which cannot only deal with the very large amount of data but their diverseness as well [6]. Additionally, statistical approaches are also independent of structured query languages and can formulate queries in natural language form.. Furthermore, the learned statistical program or method can be easily customized to a new domain being independent of any language form.

In general, statistical techniques have been so far successfully applied to the different stages of a QA system. Support vector machine (SVM) classifiers, Bayesian classifiers, Maximum entropy models are some techniques that have been used for question classification purpose. These statistical measures analyze questions for making a prediction about users' expected answer type. These models are trained on a corpus of questions or documents that have been annotated with the particular mentioned categories in the system [6].One of the pioneer works based on the statistical model was IBM's statistical QA [10] system. This system utilized maximum entropy model for question/ answer classification based on various N-gram or bag of words features. And in IBM's statistical QA TREC-11 [11] they incorporated a novel extension of statistical machine translation.

C. Machine Learning Approach

With the introduction of machine learning to the QA domain algorithms that can learn to understand linguistic features without explicitly being told to. The major advantage machine learning brings to the table is its learnability. This makes the system highly scalable as long as there is enough training data.

Machine learning is combined with statistical approaches often in linguistic and sentiment related fields [34]. Successful systems have been developed using basic machine learning classifiers and strong underpinning feature sets [34]. with several word dictations. Systems developed with the strongest features and sentiment lexicons usually utilize linear support vector machines to train their underlying systems [34]. Similarly, an ensemble meta-classifier based approach, was able to successfully reproduce previous implementations, where it calculates the average confidence score of individual classifiers for positive, negative and neutral classes [34]. Ensemble approach has proven to be powerful in combining several methods.

D. Deep Learning Approach

An extensive prior study of various approaches used in the Q&A domain provided us with a clear roadmap for our research. This resulted in we choosing a deep learning based approach. Deep learning is a subset of techniques of machine learning

Deep learning fundamentally differs from machine learning because of the ability it has to learn underlying features in data using neural networks. A standard neural network (NN) consists of many simple, connected units called neurons, each producing a sequence of real-valued activations [23]. Input neurons get activated through sensors perceiving the environment, other neurons get activated through weighted connections from previously active neurons [23]. Some neurons may influence the environment by triggering actions. Each input has a weight associated with it that depicts the importance of it compared to other inputs [23]. Recently, deep learning models have obtained a significant success on various natural language processing tasks, such as semantic analysis, machine translation and text summarization [24]. Neural Network architectures map textual context into logical representations that are then used for answer prediction. These neural networks utilize bi-directional Long Short-Term Memory(LSTM) units for question processing and answer classification [24].

The Sinhala Question Answering System “Mahoshada” intends to obtain accurate information from a Sinhala tagged corpus [21]. It is a QA system for the Sinhala language which provides answers to any question within the context. “Mahoshada” has produced the capability to adopt any specified domain by the corpus provided. The system can take any annotated Sinhala text document under a specific domain [21]. The system summarizes the tagged corpus and utilizes it to generate answers for a given query. Summarized corpora are classified to enable fast retrieval of information. The system consists of four modules which are Document Summarizing, Document categorizing, Question processing, and Answer processing [21]. Document summarization is important to summarize multiple documents input by the user to reduce the number of terms and increase efficiency [21]. Document processing does the organization of the documents to retrieve answers conveniently by categorization. Question processing is responsible to analyze the question type and identify the question type. Answer processing involves in identifying and retrieving the most suitable answer [21].

Recently, Dynamic Memory Networks (DMNs) have shown success in Question Answering [9]. It is a neural network architecture that can process input sequences, form episodic memories and produce appropriate answers [13]. Questions posed by the user initiates an iterative attention process, where the model focuses attention on the inputs and previous iteration results [13]. DMN has achieved fine results on sentiment analysis and attained state-of-the-art results of the Facebook bAbI dataset [9][28].

E. Neural Models

RNNs have been able to make a significant improvement in contrast to other machine learning techniques due to their ability to learn and carry out complicated transformations of data its ability maintaining long-term as well as short-term dependencies. They are said to be ‘Turing-Complete’ [16], therefore having the capacity to simulate arbitrary procedures. RNN contains an interplay of Reasoning, Attention, and Memory, commonly referred to as the ‘RAM model’ in Deep Neural Networks (DNN). Researchers have been keen on building models of computation with various forms of explicit storage. Google’s DeepMind project released Neural Turing Machines that extend the capabilities of neural networks by coupling them to external memory resources, which they can interact with by attentional processes [16]. Neural Training Machines were more expensive than previously considered due to the utilization of an external memory. Thus, Reinforcement Learning Neural Turing Machines (RLNTM) was introduced [18]. RLNTMs use a Reinforcement Learning Algorithm to train Neural Network that interacts with interfaces such as memory tapes, input tapes, and output tapes, to solve simple algorithmic tasks. In addition to that, Neural Machine Translation is another approach to machine translation, which attempts to build and train a single large neural network that reads and outputs a correct translation instead of having small sub-components that are tuned separately [19]. Most machine translators are encoder-decoder models where the encoder neural network reads and encodes a sentence to a fixed length vector and the decoder output the translation from the encoded vector.

Neural networks have been applied in several languages specific question answering systems. Among these, a novel approach has been introduced to answer arithmetic problems using the Sinhala language [21]. A methodology is presented to solve Arithmetic problems in Sinhala Language using a Neural Network. This system comprises of keyword identification, question identification and mathematical operation identification and these are combined using a neural network[21].

III. METHODOLOGY

Question Answering techniques constitute of complex natural language understanding capabilities like word sense disambiguation, inference resolution and most importantly complex reasoning. Reasoning, Attention and Memory(RAM) are key requirements to a model understanding text for Question Answering. Memory-based deep neural networks, DMN, MemNet and RRN have shown to achieve sections of RAM for the bAbI 1K dataset with 20 different types of tasks to achieve a task the test accuracy must be above 95%. But these networks are in capable of addressing complex reasoning. DMN and RRN are contrasting in their approaches. DMN are good at basic induction but fails in positional reasoning and pathfinding. RRN fails at complex reasoning but is very good in simple reasoning. RNN’s main focus is on Reasoning, while DMN’s main focus is on Attention. LSTMs have also been tested for similar tasks but fails to compare to RRN and DMN. This is due to the strong memory an LSTM holds which overloads memory that in turn results in failing to achieve RAM due to its poor memory management capabilities. Addressing this issue would provide more context to be held in memory that would then in turn improve on the RAM.

Reinforcement Learning(RL) are not usually used to address RAM or QA based tasks, but it has exceeded human performance. It is capable of utilizing reasoning and attention to achieve a clearly stated goal. Because of this RL has the capability of achieving RAM. Combination of RL and LSTM, RL-LSTM has shown promise in reasoning utilizing memory to predict future state. In our approach the LSTM predicts the answer for the question derived from the context. . RL then replicates the response from the real environment (correct answer). If the RL decides the answer given by the LSTM is wrong, then it would act on the vector space and identify the goal state (the answer) for the given question. The answers are traversed through the available environment of words to find the related answer.

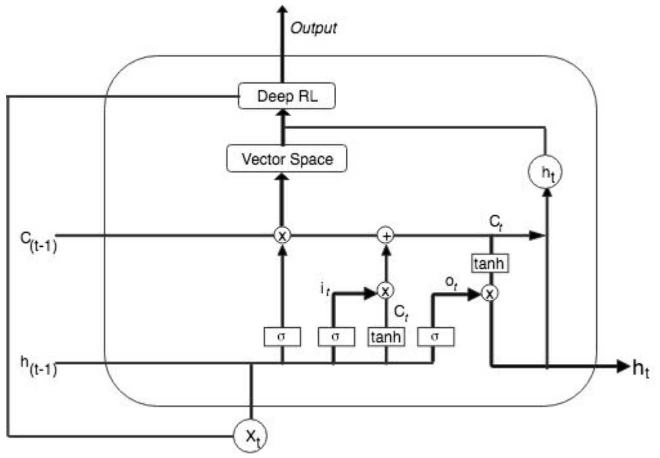

Fig. 2 An overview of the proposed network model

A. *Improvements to Embeddings*

As the knowledge sources for the embeddings PPDB and conceptnet were selected. ConceptNet is a semantic network of terms connected by labeled relations. Its terms are words or multiple-word phrases in a variety of natural languages. PPDB is another resource that is useful for learning about word similarity, providing different information from ConceptNet. It lists pairs of words that are translated to the same word in parallel corpora.

Word2vec and GloVe are two current systems that learn vector representations of words according to their distributional semantics. Given a large text corpus, they produce vectors representing similarities in how the words co-occur with other words. Word2vec is a neural network with one hidden layer is trained to recognize words that are likely to appear near each other. GloVe is an unsupervised learning algorithm that learns a set of word embeddings such that the dot product of two words' embeddings is approximately equal to the logarithm of their co-occurrence count.

B. *Improvements to Memory*

The LSTM has two purposes.

1. Generating an answer to the question.
2. Using the input gate to clear unwanted context, create the vector space and pass to the RL neural network.

The LSTM should possess the ability to utilize the whole network for Question Answering. It's trained to behave as a normal net to generate answers to the queries out of scope. The answer generated is in turn validated by the RL's model. The second aspect of the LSTM is to utilize the input gate to create the environment. The context passed through the input gate would remove unnecessary words (Removing noise). This environment is then passed to the reinforcement learning layer. The input layer thus acts as the memory factor in creating vector space. It also acts as the primary factor in the attention factor.

C. *Improvements to Attention and Reasoning*

We used a model based Reinforcement Learning layer to achieve the improvements to attention and reasoning. The model is trained to mimic the training environment and react similarly. The environment provided for the layer is the vector space. The goal assigned to the layer is generate the correct answer to a given question from the vector space. The goal state is set as the correct answer and the RL receives the only award of 1. RL goes through the vector space till it achieves the goal state.

The model learns from the environment how it should react to a given answer from the environment. LSTM places its result in the vector space and model takes this result as the initial action on the environment, and the initial state is the location of LSTM result. If the goal state is not achieved the RL acts on the environment and changes the state. Through Q-learning, the state would then be changed to the goal state.

If the environment decides that the results from the LSTM are wrong then the RL is enforced. RL holds the Q-learning structure. The environment would then act to change the state to the answer available on the vector space. If the current state is not the goal state then the state would be moved to the smallest cosine similarity word in the vector space using Q-learning.

IV. Results

A. *Results using bAbI 1K dataset*

TABLE I. BABI 1K RESULTS FOR EACH TASKS COMPARED WITH RNN AND DMN

Tasks	Accuracy (%)				
	1000 sets for Training data			*500 sets of Training data*	*450 sets of Training data*
	DMN	RRN	LSTRM	LSTRM	LSTRM
Single Support Factor	100	100	100	100	100
Two Supporting Facts	98.2	99.7	99.5	99.5	99
Three Supporting Facts	95.2	96.5	99.8	99.8	97.5
Two Argument Relations	100	100	99.7	99.5	98.2
Three Argument Relations	99.3	99.6	99.6	99.6	99.1
Yes/No Questions	100	100	100	100	100
Counting	96.9	100	99.6	99.6	98.2
Lists/Sets	96.5	100	99.6	99.6	98.6
Simple Negation	100	100	100	100	99.5
Indefinite Knowledge	97.5	100	99.9	99.9	99.2
Basic Co reference	99.9	100	100	100	100
Conjunction	100	100	100	100	100
Compound Co-reference	99.8	100	100	100	99.6
Time Reasoning	100	99.9	100	100	99.5
Basic Deduction	100	100	100	100	100
Basic Induction	99.4	45.1	97.7	97.5	96.2
Positional Reasoning	59.6	100	98.5	98.5	97.1
Size Reasoning	95.3	99.7	98.5	98.5	98
Path Finding	34.5	99.9	98.5	98.5	97
Agent's Motivations	100	100	100	100	100
Mean Accuracy	93.6	97.02	99.54	99.51	98.83

bAbI 1K dataset has 20 different tasks. Each task has sets of context, question, and answer. There are 1K training sets and 1K testing sets for each task. Each task has a separate goal to be achieved. Human performance for the bAbI 1K has an accuracy of 100%. NRMQA which is composed using hybrid method of RL and LSTM has the ability of attaining RAM for the QA tasks and has been tested on bAbI 1K dataset achieving all the tasks and having an accuracy of

99.51 %. The Table I shows that NRMQA achieves better results and is the closest to achieving human-level performance compared to DMN which focuses on Attention and does not achieve Reasoning and RRN. NRMQA does not require the whole 1K data to achieve its overall accuracy of 99.51%. Therefore, NRMQA requires less training data to achieve this accuracy is a clear indication of it's improvements to RAM model.

V. Conclusions And Future Work

Though QA systems have evolved over the years they are still in need of improvements. Usually QA systems show an upper bound performance close to a 60% of the questions correctly answered. Because of these evaluation campaigns have been conducted to improve on the self-confidence of QA systems to reduce incorrect answers. For example, ResPubliQA introduced c@1 as the main evaluation measure. c@1 rewards QA systems that reduce incorrect answers while maintaining correct ones. Additionally, some researchers suggested simple and individual tasks with the goal of helping to develop better QA systems. On the other hand, some systems reached promising results in isolated tasks out of the scope of traditional campaigns, as for example Watson, the IBM's QA system which defeated humans at the Jeopardy TV show. Despite some of these individual efforts, there has not been a clear improvement in systems' results evaluated under the same conditions. One such major area is answering complex questions. These questions refer to different pieces of information that has to be gathered from several sources. These questions require the application of several inference steps, where each step might collect information demanded by the following steps. Another area that needs further improvement is identifying the answer even if appears in different wordings. These questions refer to the information implicitly represented in the documents. Another way of improving QA systems is introducing strategies adopted by humans in answering questions. One such method is rejection, that is selecting the correct answer by discarding the incorrect answers.

References

[1] L. Kodra and E. Kajo, "Question Answering Systems: A Review on Present Developments, Challenges and Trends", *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 9, 2017 [Online]. Available: https://thesai.org/Downloads/Volume8No9/Paper_31-Question_Answering_Systems_A_Review_on_Present_Developments.pdf. [Accessed: 22- May- 2018].

[2] E. Brill, J. Lin, M. Banko, S. Dumais and A. Ng, "Data-Intensive Question Answering", *Trec.nist.gov*, 2018. [Online]. Available: https://trec.nist.gov/pubs/trec10/papers/Trec2001Notebook.AskMSRFinal.pdf. [Accessed: 22- May- 2018].

[3] H. Madabushi and M. Lee, "High Accuracy Rule-based Question Classification using Question Syntax and Semantics", *Aclweb.org*, 2018. [Online]. Available: http://www.aclweb.org/anthology/C16-1116. [Accessed: 23- May- 2018].

[4] E. Riloff and M. Thelen, "A Rule-based Question Answering System for Reading Comprehension Tests", 2018. [Online]. Available: https://pdfs.semanticscholar.org/4454/06b0d88ae965fa587cf5c167374ff1bbc09a.pdf. [Accessed: 23- May- 2018].

[5] S. Humphrey and A. Brownea, "Comparing a Rule Based vs. Statistical System for Automatic Categorization of MEDLINE Documents According to Biomedical Specialty", 2018. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2782854/. [Accessed: 23- May- 2018].

[6] S. K. Dwivedia and V. Singh, "Research and reviews in question answering system," in *Proceedings of International Conference on Computational Intelligence: Modeling Techniques and Applications*, 2013, pp. 417 – 424 .

[7] D. Cohn, Z. Ghahramani and M. Jordan, "Active Learning with Statistical Models", *Journal of Artificial Intelligence Research*, vol.4, 1996. [Online]. Available: https://jair.org/index.php/jair/article/view/10158. [Accessed: 22- May- 2018].

[8] X. Li and D. Roth, "Learning Question Classifiers", *Dl.acm.org*, 2018. [Online]. Available: https://dl.acm.org/ft_gateway.cfm?id=1072378&ftid=569844&dwn=1&CFID=37296624&CFTOKEN=52eb69472489ceca-89B68F45-9568-FB11-CC44151C22772CA6. [Accessed: 22- May- 2018].

[9] Raghuvanshi, A., & Chase, P., "Dynamic Memory Networks for Question Answering". [Online]. Available: https://cs224d.stanford.edu/reports/RaghuvanshiChase.pdf. [Accessed: 22- May- 2018].

[10] Ittycheriah A., Franz M, Zhu WJ, Ratnaparkhi A. and Mammone R. J., "IBM's statistical question answering system," in *Proceedings of the Text Retrieval Conference TREC-9*, 2000.

[11] Ittycheriah, A., & Roukos, S. (2006). "IBMs Statistical Question Answering System" - TREC-11. doi:10.21236/ada456310

[12] Moschitti A. "Answer filtering via text categorization in question answering systems," in *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence*, 2003, pp. 241-248.

[13] Kumar, A., Irsoy, O., Iyyer, M., Ondruska, P., Bradbury, J., Gulrajani, I., Socher, and R. (n.d.). "Dynamic Memory Networks for Natural Language Processing", [Online]. Available: http://proceedings.mlr.press/v48/kumar16.pdf. [Accessed: 22- May- 2018].

[14] A. Nugaliyadde, Kok Wai, WongFerdous, and SohelHong Xie. "Reinforced Memory Network for Question Answering," in *Proceedings of International Conference on Neural Information Processing*, 2017. [Online]. Available: https://www.researchgate.net/publication/320658588_Reinforced_Memory_Network_for_Question_Answering. [Accessed: 22- May- 2018].

[15] Siegelmann, H. T. and Sontag, E. D, "On the computational power of neural nets", *Journal of computer and system sciences*, vol. 50, 1995. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022000085710136. [Accessed: 22- May- 2018].

[16] Alex Graves, Greg Wayne and Ivo Danihelka. (2014) "Neural Turing Machines". [Online]. Available: https://arxiv.org/pdf/1410.5401v2.pdf. [Accessed: 22- May- 2018].

[17] Armand Joulin Facebook and Tomas Mikolov. (2015) "Inferring Algorithmic Patterns with Stack-Augmented Recurrent Nets". [Online]. Available: https://arxiv.org/pdf/1503.01007.pdf. [Accessed: 23- May- 2018].

[18] Wojciech Zaremba, and Ilya Sutskever. (2016) "REINFORCEMENT LEARNING NEURAL TURING MACHINES". [Online]. Available: https://arxiv.org/pdf/1505.00521.pdf. [Accessed: 23- May- 2018].

[19] Dzmitry Bahdanau, KyungHyun Cho and Yoshua Bengio. (2014) "NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE". [Online]. Available: https://arxiv.org/pdf/1308.0850.pdf. [Accessed: 22- May- 2018].

[20] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston and Rob Fergus (2015). "End-To-End Memory Networks". [Online]. Available: https://arxiv.org/pdf/1503.08895.pdf. [Accessed: 22- May- 2018].

[21] T. Jayakody, T.S.K. Gamlath, W.A.N. Lasantha, K.M.K.P. Premachandra, Y. Mallawaarachchi, and A. Nugaliyadde, "Mahoshadha, The Sinhala Tagged Corpus based Question Answering System," in *Proceedings of International Conference on Information and Communication Technologies for Intelligence Systems, 2015*. [Online] Available: https://www.researchgate.net/publication/292615736_Mahoshadha_The_Sinhala_Tagged_Corpus_based_Question_Answering_System. [Accessed: 23- May- 2018].

[22] M. Chathurika, C. De Silva, A. Raddella, E. Ekanayake, Y. Mallawarachchi and A. Nugaliyadde, "Solving Sinhala Language Arithmetic Problems using Neural Networks," in *Proceedings of 34th*

National Information Technology Conference, 2016. [Online]. Available: https://www.researchgate.net/publication/305262356_Solving_Sinhala_Language_Arithmetic_Problems_using_Neural_Networks. [Accessed: 22- May- 2018].

[23] J. Schmidhuber, "Deep learning in neural networks: An overview", 2018. [Online]. Available: https://ac.els-cdn.com/S0893608014002135/1-s2.0-S0893608014002135-main.pdf?_tid=7f19f037-5d16-4521-9163-a8b2a8f3768c&acdnat=1527478744_44e66fda06319991375c434ae2fc5b60. [Accessed: 23- May- 2018].

[24] M. Tan, C. Santos, B. Xiang and B. Zhou, “LSTM-based Deep Learning Models for Non-factoid Answer Selection”, 2015. [Online]. Available: https://arxiv.org/abs/1511.04108. [Accessed: 24- May- 2018].

[25] B. Magnini, D. Giampiccolo, P. Forner, C. Ayache, V. Jijkoun, P. Osenova, A. Pe˜nas, P. Rocha, B. Sacaleanu and R. F. E. Sutcliffe, “Overview of the CLEF 2006 Multilingual Question Answering Track,” in *Proceedings of Evaluation of Multilingual and Multi-modal Information Retrieval, 7th Workshop of the Cross-Language Evaluation Forum, CLEF 2006, Alicante, Spain*, 2006, pp. 223–256.

[26] A. Rodrigo and A. Pe˜nas, “A Study about the Future Evaluation of Question-Answering Systems”, 2017. [Online]. Available: https://www.researchgate.net/publication/319617514_A_Study_about_the_Future_Evaluation_of_Question-Answering_Systems. [Accessed: 24- May- 2018].

[27] A. Pe˜nas, P. Forner, R. Sutcliffe, A. Rodrigo, C. For˘ascu, I. n. Alegria, D. Giampiccolo, N. Moreau and P. Osenova, “Overview of ResPubliQA 2009: Question Answering Evaluation over European Legislation,” in *Proceedings of the 10th cross-language evaluation forum conference on Multilingual information access evaluation: text retrieval experiments*, pp. 174–196, 2009.

[28] J. Weston, A. Bordes, S. Chopra, T. Mikolov, “Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks”, 2016. [Online]. Available: https://arxiv.org/abs/1502.05698. [Accessed: 24- May- 2018].

[29] D. A. Ferrucci, E. W. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. M. Prager, N. Schlaefer, and C. A. Welty, “Building Watson: An Overview of the DeepQA Project”, AI Magazine 31 (3), 2010, pp. 59–79.

[30] S. Harabagiu, F. Lacatusu, and A. Hickl, “Answering Complex Questions with Random Walk Models,” in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, *SIGIR ’06*, 2006, pp. 220–227.

[31] E. Saquete, P. Mart´ınez-Barco, R. Mu˜noz, and J. L. Vicedo, “Splitting Complex Temporal Questions for Question Answering Systems,” in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, ACL ’04*, 2004.

[32] Jayasuriya, M., and Weerasinghe, A. R. “Learning a stochastic part of speech tagger for sinhala,” in *Proceedings of the 2013 International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2013. doi:10.1109/icter.2013.6761168

[33] Gunasekara, D., Welgama, W., and Weerasinghe, A. “Hybrid Part of Speech tagger for Sinhala Language,” in *Proceedings of the 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2016. doi:10.1109/icter.2016.7829897

[34] Muthutantrige, S. R., and Weerasinghe, A. “Sentiment Analysis in Twitter messages using constrained and unconstrained data categories,” in *Proceedings of the Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2016. doi:10.1109/icter.2016.7829935

[35] T. Mikolov and G. Zweig, “Context Dependent Recurrent Neural Network Language Model”, 2012. [Online]. Available: https://www.microsoft.com/en-us/research/wp-content/uploads/2012/07/rnn_ctxt_TR.sav_.pdf. [Accessed: 24- May- 2018].

[36] T. Mikolov, S. Kombrink, L. Burget, J. Cernock ˇ y and S. Khudanpur, “Extensions of recurrent neural network language model,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[37] R. Williams and D. Zipser, “A Learning Algorithm for Continually Running Fully Recurrent Neural Networks”, *Neural Computation*, 1, pp. 270-280, 1989.

[38] Y. Bengio and G. Hinton, "Deep learning", 2015. [Online]. Available: https://www.nature.com/articles/nature14539. [Accessed: 24-May-2015].