CORPUS PREPROCESSING

ADAPTIVE ARTIFICIAL INTELLIGENT QUESTION ANSWER SYSTEM

17-107

Saad Sahibjan

IT14109072

Bachelor of Science (Honours) in Information Technology
(Specialization in Software Engineering)

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

October 2017

DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. Also I hereby grant to Sri Lanka Institute of Information Technology the non-exclusive right to reproduce and distribute my dissertation in whole or part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as article or books).

Signature: Date:

Signature of the Supervisor: Date:

ACKNOWLEDGEMENT

I would like to thank my supervisor Mr. Yasas Mallawarachi, head of Software Engineering Services at Sri Lanka Institute of Information Technology for his guidance and for helping me putting this project together.

Furthermore I would like to thank my external supervisor PhD student Anupiya Nugaliyadde for his guidance, help and for always being available through the project process.

ABSTRACT

This final report is for the component titled ‘Corpus Preprocessing’ of project titled ‘Adaptive Artificial Intelligent Question Answer’. The goal of the component Corpus Preprocessing is to have proper representation of data which will be input for the training model. The goal of this project is to build a Question Answer system that is capable of processing the information in a large dataset and allows the user to gain knowledge from this dataset by asking questions in natural language form. The system is capable of understanding this question responds to the user’s query in natural language form as well. The goal is to make the user feel as if they were interacting with a person. In this report we do a literature survey to understand the current status of the QA domain and present some popular solutions that are being used currently. Then identify and prepare the case for a deep learning alternative in this field and explain why that approach would help negate some of the issues being faced by the other systems.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

The information growth rate in the world is increasing at a rapid rate. It has become impossible to keep up with the amount of information that is being added to any given domain. A regular human cannot keep up with all new that gets generated. Due to the wide availability of digital input and output devices and the ease of use of these devices people are creating more and more raw data. If we are able to process this data into meaningful information fast, it would be possible for people to become more productive and to get the information they want faster.

A key idea in social sciences is that a rational human being makes the best decision when he/she is able to access and use all the available relevant information. However this is not very practical. The best way to access all the available information for a given domain would be the internet, but a human is not able to go through all this information and understand and process it in a timely manner to make a good decision.

To a certain extent search engines have been able to tackle this problem. Search engines have become the center of the internet because that is what we use as the gateway to the internet. Rarely do we actually type in a specific web address. Instead we would type in a query into a search engine and use the information that is provided by the search engine to access the data that we want. Over the years search engines have become better and are using much more powerful techniques than simple keyword matching and page link ranking. Still search engines only provide us resources through which we have to sort through and find the answers that we need to find. We cannot use them to give us a direct answer.

We think that this situation can be improved much more and by using deep learning techniques we will be able to create a platform that is able to learn from given data set and then produce direct answers that users can rely on. For the platform to be effective it is important that users can interact with it as naturally as possible. So the user must be able to type in a generic question like they would be talking to a person and the platform should produce an answer that is both factually and grammatically correct. This means that we

would not need to have a specific query language or we would not need to structure the corpus (dataset) manually.

The report will illustrate the purpose and the main areas to be focussed throughout the component Corpus Preprocessing and also about Adaptive Artificial Intelligence Question and Answer system. Also it will explain on literature survey, research problem, research gap, research objectives and methodology. Further indicates implementation, research findings, results and discussion.

1.1 LITERATURE REVIEW

Machine learning is a field of Artificial Intelligence that has been gaining a lot of prominence in the current era. It is specifically to do with building systems that are able to learn by themselves without having to be programmed. Deep learning is a subset of techniques of machine learning. Deep learning allows multiple processing layers to breakdown the given data into smaller parts and learn the representations of these data [1]. Deep learning is the state of the art in areas such as speech recognition, natural language understanding, visual object recognition, etc. Convolutional neural networks have brought many breakthroughs in areas such as processing images, pictures and speech, whereas Recurrent networks have been extremely successful in areas such as processing text and speech.

QA is a well researched area from the point of NLP (Natural Language Processing) research. QA has mostly been used to develop intricate dialogue systems such as chat-bots and other systems that mimic human interaction [2]. Traditionally most of these systems use the tried methods of parsing, part-of-speech tagging, etc that come from the domain of NLP research. While there is absolutely nothing wrong with these techniques, they do have their limitations. [3] W.A. Woods et al. shows how we can use NLP as a front end for extracting information from a given query and then translate that into a logical query which can then then be converted into a database query language that can be passed into the underlying database management system. In addition to that there needs to be a lexicon that functions as an admissible vocabulary of the knowledge base so that it is possible to filter out unnecessary terminology. The knowledge base is processed to an ontology that breaks it down into classes, relations and functions [4]. Natural Language Database Interfaces (NLDBIS) are database systems that allow users to access stored data using natural language requests. Some popular commercial systems are IBM's Language Access and Q&A from Symantec [5].

When it comes to Corpus Preprocessing Information Extraction (IE) [6] is also another technique. In simple IE identifies the keywords and relationship between those. It does this by a process called pattern matching, by looking for predefined sequences in the text. IE infers the relationship between places, people and time to provide the user with meaningful

information. This technique is useful when handling with large volume of data. To have the best outcome through this technique traditional data mining process expects the information to be mined already in the form of a relational database. Unfortunately dataset/corpus are available (most of the time) in the form of free natural language documents rather than structured database [7]. This process is depicted in Figure 1.0.

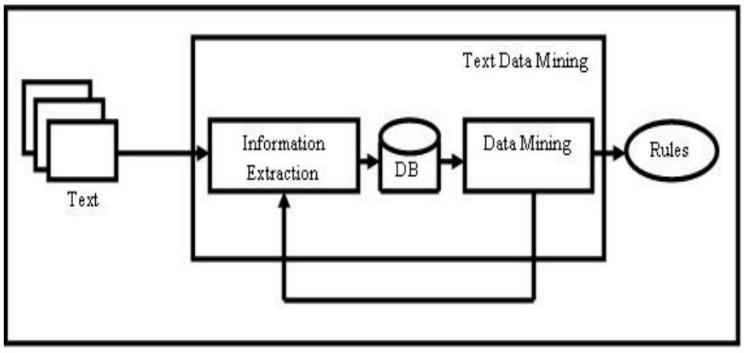

Figure 1.0 Process of Text Extraction

Information Retrieval (IR) is another technique that has been used to address Corpus Preprocessing. With IR systems pay attention to the organisation, representation and storage of information artifacts such that when a user makes a query the system is able to return a document or a collection of artifacts that relate to the query [8]. Recent advances in OCR and other text scanning techniques have meant that it is possible to retrieve passages of text rather than entire documents. However IR is still widely seen as from the document retrieval domain rather than from the QA domain.

Categorization is also another technique which involves in identifying the major subjects of a document through inserting the document into a predefined set of topics. It actually does not try to process the actual information. Rather, it only counts words that appears and form that count it identifies the main topics that the document covers. So that categorization technique often relies on a glossary for which topics are predefined and relationships are identified by looking for large terms, narrow terms, synonyms and related terms [9]. The major drawback in this technique of corpus preprocessing is that it misses out the syntactical meanings of the words.

Template based question answering is another technique that has been used for QA and is currently being used by the START system which has answered over a million questions since 1993 [21]. START uses natural language annotations to match questions to candidate answers. An annotation will have the structure of 'subject-relationship-object' and when a user asks a question, the question will be matched to all the available annotation entries at the word level (using synonyms, IS-A, etc) and the structure level. When a successful match is found, the annotation will point to an information segment which will be returned as the answer. When new information resources are incorporated into the SMART system, the natural language annotations have to be composed manually [22]. START uses Omnibase as the underlying database system to store information and when the annotation match is found, the database query must be used to retrieve the information. While this system has been relatively successful, it requires a lot of preprocessing which must be done manually.

There is a wealth of information on the internet. For any given domain we are able to find a huge amount of information. However to use this information effectively, there needs to be a system to process the data and extract out the meaningful information. Further it is important to provide a simple and seamless way of interacting with this data. This has given rise to the field to natural language question answering where a user must be able to ask a question in everyday language and receive a factually correct answer quickly. In the literature survey we discussed few different techniques that have been used to tackle this problem, NLP, information retrieval and information extraction, categorization. All the methods have its own flaws where either the accuracy is not high enough or it may take a lot of manual processing and so on. This has meant that while this is a significant problem domain due to the high costs there haven't been any commercially viable solutions yet.

According to the literature survey QA domain has an active community of researchers and many different approaches have been tried to tackle this problem. While the problem of QA is a very old one, the origins of the problem can be traced back as far as the 1960's. Also it clearly identified through the techniques identified, corpus preprocessing plays a major role since the preprocessed corpus will be used to train and build model using which the question will be mapped and answers will be generated. Using access to cheaper and better

computational power and newer techniques in data processing, identified flaws are addressed using different set of tactics.

1.2 RESEARCH GAP

As explained in the introduction, there is a wealth of information on the internet. For any given domain we are able to find a huge amount of information. However to use this information effectively, there needs to be a system to process the data and extract out the meaningful information. Further it is important to provide a simple and seamless way of interacting with this data. This has given rise to the field to natural language question answering where a user must be able to ask a question in everyday language and receive a factually correct answer quickly. In the literature survey we discussed three different techniques that have been used to tackle this problem, NLP, Information Retrieval and Template based question answering. All three methods have flaws where either the accuracy is not high enough or it may take a lot of manual processing and so on. This has meant that while this is an important problem domain due to the high costs there haven't been any commercially viable solutions yet.

The solution that we are proposing for this problem domain is one that is based on deep learning. Deep learning can be defined a subset of machine learning techniques that uses non-linear information processing to identify and extract features and patterns in data, classification and transformations. There are three key reasons that deep learning has become so popular in the recent past. First, the hugely increased processing abilities and availability of general purpose GPU's, the vast amount of training data that has become available and the many advances made in the recent past in the field of deep learning that has made the task of training artificial neural networks more efficient [23].

Deep learning makes extensive use of Artificial Neural Networks (ANN) to complete a given task. ANN's are inspired from the neural networks found in the human brain. The brain consists of an intricate network of neuron cells. Researchers have found success in trying to replicate this structure on silicon chip. Each neuron would consist of what is known as an activation function and the neurons would be connected to each other via connections called tensors [24]. The entire ANN would consist of an input layer of neurons, multiple hidden layers and an output layer. Biases are assigned to neurons and weights are assigned to the tensors. These values influence what eventually becomes the output of the ANN. When

training an ANN we will be adjusting these weights and biases to get the desired output for a known input data set and known scenario [25]. When working with a large ANN with several tens of neurons it can become a tedious task to adjust the weights and biases manually. There are algorithms we can use such as backpropagation to help us adjust these values automatically until we get to the desired output.

Since deep neural networks are exemplary at recognising patterns and processing data extremely fast, we believe that by applying deep learning techniques to this problem domain we will be able to overcome many of the drawbacks of the other approaches. We will be able to have a higher accuracy because of the state of the art neural network training paradigms, reduce manual tasks by allowing an ANN to process and structure the corpus and automatically extract the required features and we will not need to use an underlying database engine so therefore we will not need to adopt or develop a different query language.

1.3 RESEARCH QUESTION

Corpus Preprocessing is one of primary component in the research project. Initially the corpus will be unstructured text data, which can be understood by humans, not by machines. It simply implies that, there is a necessary for a transformation of such data because many machine learning algorithms including deep neural networks, require inputs to be vectors of continuous values; they won't just work on plain text or strings.

Since the amount of information is rapidly growing and it is becoming difficult to keep manually created knowledge bases and ontologies uptodate. Than being relying on manually created knowledge bases, applying deep learning techniques to unstructured data or the corpus to identify relationship of words and providing the proper transformation of unstructured data to a representation as an input to the neural network training model, will provide an immense help in extracting the needed answer.

To summarize on the research question, the corpus or the dataset will be of unstructured data and cannot be understood by machines or the machine learning algorithms. Therefore the corpus need to be preprocessed into a form where it can be understood by the deep neural networks prevailing the semantic and syntactic relationship of the words.

1.4 RESEARCH OBJECTIVES

When it comes to deep learning QA systems there are two broad categories that can target. There are open datasets such as the Allen AI Science and Quiz Bowl datasets, and closed datasets such as the ones provided by Facebook (bAbI) [10]. Open QA systems require using the information provided in the dataset as well as any additional available knowledge. This requires some Information Retrieving techniques. In real world applications this would be the most likely required solution, however for the purposes of this research, have chosen to focus on a closed QA system, where the answer to a question would depend on the given dataset.

Primary objective of corpus preprocessing is that the preprocessed corpus is the input for training model and the answer is extracted through the training model based on the preprocessed corpus. Therefore the preprocessed corpus should contain a proper representation of data as it is the input for the training model. Intention behind the corpus preprocessing is the transformation of raw text into a representation of vectors in a low dimensional vector space along with maintaining the words which are similar in close proximity through understanding the contextual similarity of words and mapping the semantic relationship of words.

To further narrow down the scope, have chosen to build a question answer system for the multiple domains**.** As a case study and proof of concept, implemented a QA system that focuses only on answering questions related to a specific domain and not addressing open domain questions. The scope has been thus narrowed to, first, increase the accuracy of answers provided and second, to ensure that the project can be completed in the allocated time.

In the aspect of multiple domains, the QA system will answer question based on the domain. Implemented system tested based on domains such cornell (movie dataset), ubuntu and medical. Basically the objective here is to have a QA system which is able to train datasets of multiple domains and plug the trained model into the system and then ask questions related to

domain and get answers in natural form. Sole idea is to have a simple system with an easy to switch domain and to generate answers appropriately.

In analysing the multiple domain approach, corpus preprocessing and the training of preprocessed corpus should not be done each time the domain is switched as it consumes considerable amount of time and resources. Focus here is to reduce the resource usage and consumption of time to the maximum and achieve domain switchability in fraction of seconds with the usage of existing technologies.

The end product would be a system that allows the user to ask related questions based on the plugged model (i.e domain) in natural language form and the platform would find the most accurate answer and provide that answer in natural language form as well. The idea is to simulate a situation where the user is interacting with a person as close as possible. The accuracy of the answers will largely depend upon the accuracy of the data in the data set and therefore we cannot guarantee that this will be able to replace an actual human. However the goal in this research is to show that using deep learning techniques we are able to reduce some of the complexities and barriers that are present at the moment and are stopping QA systems from becoming mainstream products.

2.0 RESEARCH METHODOLOGY

2.1 METHODOLOGY

Corpus Preprocessing is one of primary component in the research project. Initially the corpus will be unstructured text data, which can be understood by humans, not by machines. It simply implies that, there is a necessary for a transformation of such data because many machine learning algorithms including deep neural networks, require inputs to be vectors of continuous values; they won't just work on plain text or strings.

Since the amount of information is rapidly growing and it is becoming difficult to keep manually create knowledge bases and ontologies uptodate. Than being relying on manually created knowledge bases, applying deep learning techniques to unstructured data or the corpus to identify relationship of words and providing the proper transformation of unstructured data to a representation as an input to the neural network training model, will provide an immense help in extracting the needed answer.

One of the major application of such transformation of data [11], is Word Embedding which is a natural language technique. It is used to map words or phrases from a vocabulary to a corresponding vector of real numbers. Word embedding aims to create a vector representation [12] with a much lower dimensional space. In contrast Bag of Words [13] approach, which often results in huge and sparse vectors. In Bag of Words approach the dimensionality of the vectors representing each document is equal to the size of the vocabulary.

Word Embedding [14] is also used for semantic parsing, to extract the meaning from text to enable Natural Language Understanding. For a language model to understand the meaning of a word, it need to know the contextual similarity of words. For example if we tend to find diseases in sentences, where diabetes, diarrhea, HIV should be of close proximity. So the vectors created by word embedding preserves these similarities along with the words that regularly occur nearby in text will also be in close proximity.

Word embedding is all about building a low dimensional vector representation from corpus, which preserves the contextual similarity of words. There are word embedding techniques such as,

- 1-of-N vec (one-hot-vec)
- GloVe (Global Vectors)
- Word2vec

In a simple 1-of-N [15] encoding transforms categorical features to a format that works better with classification and regression algorithms. In simple terms every element in the vector is associated with a word in the vocabulary. The encoding of a given word is simply the vector in which corresponding element is set to one and all others are set to zero. Suppose we consider a vocabulary of five words; diabetes, diarrhea, HIV, obesity, paralysis. We could encode the the word obesity as [0, 0, 1, 0, 0]. In such a scenario, the only comparison that can be made between vectors is equality testing or it engages all features and tell which is present and which is absent for a particular set of output.

GloVe [16] is a 'count based' model. Which means GloVe learn their vectors through collecting word co-occurrence statistics in a form of word co-occurrence matrix **X**. Each element $\mathbf{X}_{ij}$ of such matrix represents how often word ***i*** appears in context of word ***j*** in a large corpus. The number of "contexts" is of course large, since it is essentially combinatorial in size. Then factoring this matrix to yield a lower-dimensional matrix, where each row now yields a vector representation for each word.

In specific, the creators of GloVe illustrate that the "ratio of the co-occurrence probabilities of two words (rather than their co-occurrence probabilities themselves) is what contains information and look to encode this information as vector differences". But when it comes for computation GloVe will be taking more memory [17], because it precomputes the large word into word co-occurrence *(large word x word co-occurrence)* matrix in memory[18]. Also sometimes there is a restriction on vocabulary since GloVe requires memory quadratic in the number of words: it keeps that sparse matrix of all *word x word co-occurrences* in RAM.

Word2vec [19] is a predictive model which is one of the most popular word embedding model. This simply learns relationship between words without any prior knowledge about the domain. The output are vectors, one vector per word with an exceptional linear relationship. Once a vector model is created out of the corpus, word2vec provides two basic tools namely *analogy* and *distance* to use.

- Distance - tool provide a list of words which are closely related to a particular word from the vector model.
- Analogy - tool is provides the ability to query for textual regularities captured in the vector model.

For example, let us assume that we use word2vec to create a vector model of the words appearing in a corpus of medical domain. If the resulting vector space represents diseases and cause of disease is projected in a two dimensional vector space, we can observe a relationship between each disease and the cause of the disease, and also similar diseases are placed closed to each other in vector space.

Further word2vec is based on two architecture: *continous bags of words (CBOW* and *skip-gram (SG)*. Since word2vec as no built-in functionalities for term normalisation, the corpus needs to processed before they could be used for word2vec. Unprocessed corpus contains syntactic variations, stopwords, punctuations which has a negative impact on on how word2vec indexes the term and which will affect the quality of vector space representation.

Before providing the corpus to the word2vec it needs to be processed as follow,

- All punctuations and unnecessary white spaces needs to be removed.
    - Eg: the term *obesity* and *obesity. (with full stop)* can be indexed as two separate terms.
- Transforming all words to lower-case.
    - Eg: the terms *Obesity obesity OBESITY* can indexed as three separate terms.
- Merging muti-word terms.

- Eg: the terms *Human Immunodeficiency Virus* transformed to *human_immunodeficiency_virus* so that word2vec can identify it as a single term.

(Merging multi word terms is not yet finalized, decision will be taken during the implementation phase. To achieve this should create a separate dictionary of multi word terms for the selected domain using knowledge source.)

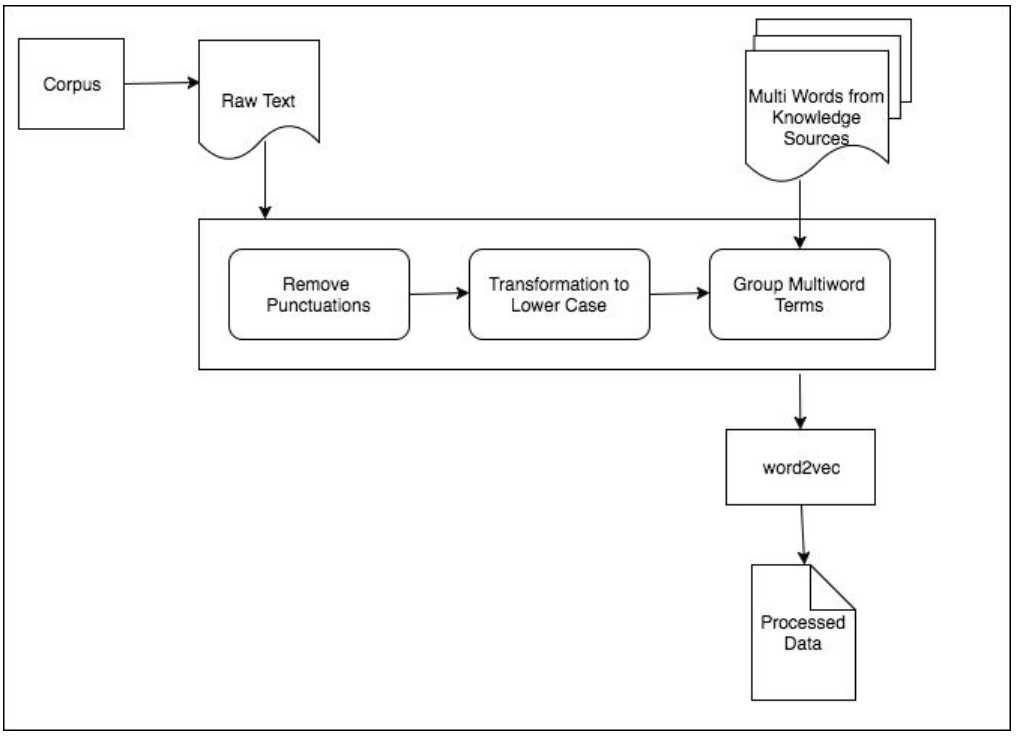

Figure 2.0 Workflow of Corpus Preprocessing

The preferred choice of technology for this component will be Python along with Tensorflow. Tensorflow is a popular machine learning platform with a lot of community support and ease of use, along with Tensorflow keras.io will also be used. Keras is a superset of Tensorflow. Python enables us to carry out string manipulation easily unlike other programming languages since corpuses are text based it will be the preferred choice of language.

2.2 TESTING AND IMPLEMENTATION

//TODO

2.3 RESEARCH FINDINGS

Having the research component of Corpus Preprocessing in the Adaptive Artificial Intelligent QA Platform is vital. Since the preprocessed corpus is the input for the training neural network model/component. Further the answer generation component/model depends on the preprocessed corpus to extract the answers. Thus to provide the users with the most accurate and a benefitted answer the corpus needs to be well preprocessed. So that the user will be highly benefitted with the answer than misleading with an incorrect answer.

Also if the corpus is to be trained without preprocessing via training neural network model, the outcome of that model will not be the expected and will never provide an accurate result. Further if the corpus is not to be preprocessed, answer extracted for the question asked can provide an inaccurate result to the user. Hence it proves that the corpus preprocessing is one of the major component in Adaptive Artificial Intelligent QA Platform and it will benefit the end user immensely in providing the accurate result.

By taking place of this research component of Corpus Preprocessing will add up as a contribution to the body of knowledge under deep neural networks. As the research carrying out currently on this component, have understood the major drawbacks of the existing corpus preprocessing techniques as explained in the Literature Review chapter. So this research is focussed on overcoming such drawbacks and coming up with a good corpus preprocessing technique with the use of existing techniques and algorithms along with new enhancements.

So that this research component will provide a benefit for the body of knowledge where as the team has an idea to publish a research paper on Adaptive Artificial Intelligent QA Platform once the research is completed. In the research paper to be published, this specific component will also be described. Therefore anyone who are referring in the context of corpus preprocessing can gain knowledge through that paper to be published hence it will go as a contribution made to the body of knowledge.

3.0 RESULTS AND DISCUSSION

3.1 RESULTS

//TODO

3.2 DISCUSSIONS

//TODO

4.0 CONCLUSION

//TODO

REFERENCES

[1] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning", Nature, vol. 521, pp. 436 – 444, 2015

[2] Silvia Quarteroni, “A Chatbot-based Interactive Question Answering System”, 11th Workshop on the Semantics and Pragmatics of Dialogue, 2007

[3] W.A Woods, R.M. Kaplan and B. Nash-Webber, “The lunar sciences natural language information system”, BBN Rep. 2378, Bolt Beranek and Newman, Cambridge, Mass., USA, 1977

[4] T.R. Gruber, “A translation approach to portable ontology specifications”, Knowledge Acquisition, 5 (2), 1993.

[5] R. Dale, H. Moisl and H. Sommers, Handbook of Natural Language Processing, 1st ed. New York: Marcel Dekker AG, 2006, pp. 215 - 250.

[6] "UCI Machine Learning Repository: Data Sets", Archive.ics.uci.edu, 2017. [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html?sort=nameUp&view=list.

[7] Vishal Gupta and Gurpreet S. Lehal, A Survey of Text Mining Techniques and Applications, JOURNAL OF EMERGING TECHNOLOGIES IN WEB INTELLIGENCE, VOL. 1, NO. 1, AUGUST 2009.

[8] L. Hirschman and R. Gaizauskas, “Natural language question answering: the view from here”, Natural Language Engineering, 7 (4), 2001, pp. 275-300.

[9] Saleh Alsaleem, Automated Arabic Text Categorization Using SVM and NB, International Arab Journal of e-Technology, Vol. 2, No. 2, June 2011.

[10] E. Stroh and P. Mathur, “Question Answering Using Deep Learning”, Stanford Reports

[11] "An overview of word embeddings and their connection to distributional semantic models - AYLIEN", AYLIEN. [Online]. Available: http://blog.aylien.com/overview-word-embeddings-history-word2vec-cbow-glove/.

[12] Corrado, Greg, and Jeffrey Dean. "Distributed Representations Of Words And Phrases And Their Compositionality". N.p., 2017.

[13] Y. Zhang, R. Jin and Z. Zhou, "Understanding Bag-of-Words Model: A Statistical Framework". [Online]. Available: https://ai2-s2-pdfs.s3.amazonaws.com/4eb6/00aa4071b9a73da49e5374d6e22ca46eaba6.pdf.

[14] A. Colyer, "The amazing power of word vectors", *the morning paper*, 2016. [Online]. Available: https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/.

[15] J. Collis, "What-is-one-hot-encoding-and-when-is-it-used-in-data-science", *https://www.quora.com*. [Online]. Available: https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science.

[16] A. Colyer, "GloVe: Global Vectors for Word Representation", *the morning paper*. [Online]. Available: https://blog.acolyer.org/2016/04/22/glove-global-vectors-for-word-representation/.

[17] S. Gouws, "How-is-GloVe-different-from-word2vec", *https://www.quora.com*. [Online]. Available: https://www.quora.com/What-is-word-embedding-in-deep-learning.

[18]"An overview of word embeddings and their connection to distributional semantic models - AYLIEN", *AYLIEN*. [Online]. Available: http://blog.aylien.com/overview-word-embeddings-history-word2vec-cbow-glove/.

[19] "Google Code Archive - Long-term storage for Google Code Project Hosting.", *Code.google.com*. [Online]. Available: https://code.google.com/p/word2vec/.

[20] "UCI Machine Learning Repository: Data Sets", Archive.ics.uci.edu, 2017. [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html?sort=nameUp&view=list.

[21] "The START Natural Language Question Answering System", Start.csail.mit.edu, 2017. [Online]. Available: http://start.csail.mit.edu/index.php. [Accessed: 26- Mar- 2017]

[22] B. Katz, G. Borchardt and S. Felshin, “Natural Language Annotations for Question Answering”, Proceedings of the 19th International FLAIRS Conference (FLAIRS 2006), 2006

[23] L. Deng and D. Yu, "Deep Learning: Methods and Applications", Foundations and Trends in Signal Processing, vol. 7, no. 34, pp. 197–387, 2013.

[24] S. Wang, Interdisciplinary computing in Java programming language, 1st ed. Boston, Mass.: Kluwer Academic, 2003.

[25] N. Gupta, "Artificial Neural Network", Network and Complex Systems, vol. 3, no. 1, pp. 24-28, 2013.

APPENDIX I: OVERALL ARCHITECTURE OF THE SYSTEM

The end product would be a system that allows the user to ask medical emergency related questions in natural language form and the platform would find the most accurate answer and provide that answer in natural language form as well. The idea is to simulate a situation where the user is interacting with a person in the medical profession as close as possible. The accuracy of the answers will largely depend upon the accuracy of the data in the data set and therefore we cannot guarantee that this will be able to replace an actual medical professional. However the goal in this research is to show that using deep learning techniques we are able to reduce some of the complexities and barriers that are present at the moment and are stopping QA systems from becoming mainstream products. The medical emergency situation was chosen purely out of convenience because of the availability of the dataset. It is only a proof of concept.

In order to achieve this, system is broken it down into four different components. Each of these components form a critical part of the system and carry out a critical function. They also have deeply integrated deep learning techniques in each component, which we have described in great detail in the methodology section. In the next section we have a brief overview of what each of the sub objectives are supposed to accomplish.

The four component of the system are,

- Corpus Preprocessing
- Question Preprocessing
- Deep Neural Network For Answer Extraction
- Answer Generation

This document contains the in-depth details of one of the research component. I.e Corpus Preprocessing.

APPENDIX II: SYSTEM ARCHITECTURE DIAGRAM

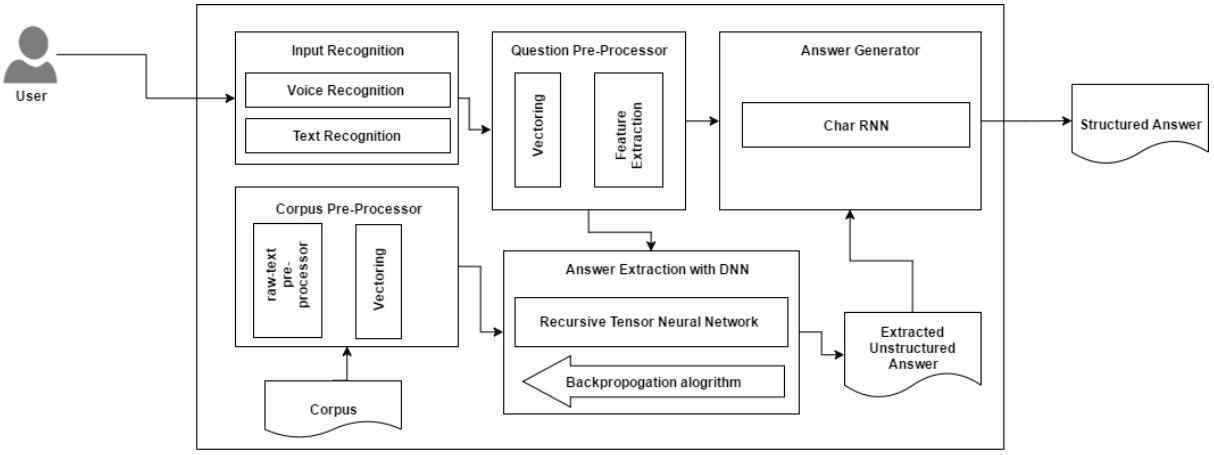

Figure 3.0 System Architecture of Adaptive Artificial Intelligent Question Answer

APPENDIX III: ACRONYM AND ABBREVIATIONS

QA	Question Answer
NLP	Natural Language Processing
IE	Information Extraction
IR	Information Retrieval
OCR	Optical Character Recognition
ML	Machine Learning
DNN	Deep Neural Network
NN	Neural Network
POC	Proof Of Concept
i.e	That is

Table 1.0 Acronyms and Abbreviations

APPENDIX IV: DEFINITIONS

Corpus	Dataset or a collection of data
Supervised Learning	Analyzes the training data and produces an inferred function, which can be used for mapping new examples
Unsupervised Learning	Is a cluster analysis, which is used for exploratory data analysis to find hidden patterns or grouping in data
End User	The person who actually uses a particular product
Preprocessing	Extract meaningful sets of data in the context of corpus preprocessing

Table 2.0 Definitions

APPENDIX V: FOCUSSED TASKS

1. Requirements Gathering
2. System Design
3. Research
4. Implementation
5. Web Interface
6. Mobile Application (Android only)
7. System Testing
8. Continuous Integration and Deployment

APPENDIX VI: DESCRIPTION OF PERSONNEL AND FACILITIES

The description of the personnel involved in this project is as follows:

Supervisor: Mr. Yashas Mallawarachi

External supervisor: Mr. Anupiya Nugaliyada

Implementation team of Adaptive Artificial Intelligent Question Answer System:

Akram M.R. (Leader)

Deleepa Perera

Singhabahu C.P.

Saad M.S.M.

The owner of this document and Corpus Preprocessing component:

Saad M.S.M.