Voice-based Road Navigation System Using Natural Language Processing (NLP)

Pooja Withanage[1], Tharaka Liyanage[1], Naditha Deeyakaduwe[1], Eshan Dias[1], Samantha Thelijjagoda[2]

[1] Faculty of Computing, [2] Faculty of Business, Sri Lanka Institute of Information Technology
New Kandy Rd, Malabe, Sri Lanka

***Abstract*—In a highly technological era, voice-based navigation systems play a major role in bridging the gap between man and machine. To overcome the difficulty in understanding the user's voice commands and natural language simulations, process the path with the user's turn by turn directions with the mention of key entities such as street names, landmarks, points of interest, connections and path mapping in an interactive interface, we propose a user-centric roadmap navigation mobile application called "Direct Me". To generate the user's preferred path, the system will first convert audio streams to text through ASR using the Pocket Sphinx library, followed by Natural Language Processing (NLP) by taking advantage of Stanford CoreNLP Framework to retrieve navigation-related information and handle the path in the map using the Google Map API at the user's request. This system is used to provide an effective approach to translating natural language commands into a format that can be fully understood by machine and will benefit in the development of human-machine-oriented interface.**

Keywords—Natural Language Processing; Speech Recognition; Road Navigation; Route Mapping; Navigational Information Extraction

I. INTRODUCTION

Navigation Dialog Systems have become very popular among Human Machine Interfaces in recent years. This voice-based human-machine interaction creates a high value when the user interacts with the system while driving. Because it helps to keep the driver's eyes focused on the road and hands on the wheel. Therefore, the demand for these interactive systems has become very high. Voice-enabled navigation is more commonplace than ever, as the average portable navigation device price keeps getting lower and lower, and high-quality navigation apps are available for most smartphones [1]. Today, smartphones come with voice-enabled navigation applications that provide turn-by-turn navigation between two locations.

When considering this kind of similar application developments, we can find many features including the main feature that provides navigation to the destination provided by the user. In current navigation applications in the market, users need to specify the starting point and the endpoint of the journey as the input to the system, and the system gives all possible paths between these two locations. However, if the user can not accurately determine the destination, they cannot get specific usage of these applications. Although the user can not accurately determine the destination, he may know the directions of the route to that destination (e.g. To someone's home). Therefore, if the navigation application can map the route when user describes the directions to where he wants to navigate it will address the above limitation in existing navigation apps. But current prevailing Navigation Systems are not provided dynamic route mapping upon user's customized own preferred route which giving as voice commands.

In the proposed "Direct Me" system, our main goal is to address this limitation and provide facilities for the user to obtain the best use of the navigation system. The focus of this paper is on the implementation of our proposed system that will accept the user provided route directions in natural language, address the directives using natural language processing and will map the specified route to the endpoint defined by user. In order to identify the user's turn by turn directions, which include junctions, street names, landmarks, distances, and points of interest, our system uses automatic speech recognizer. Then using natural language processing, the system will understand and handle the directions provided by the user. In route pre-processing, the information necessary to map directions will be processed. After that, the specified route will be map accordingly in the Map API. As the final output, the map will show the route given by the user in voice commands.

The remainder of this paper is organized as follows. In Section II, we discuss previous approaches and existing systems in the domain of road navigation. The methodology for implementing the proposed system and the core components are discussed in Section III. Section IV presents the results and discussion. Finally, we conclude the paper and discuss our future work in Section V.

II. LITERATURE REVIEW

Navigation applications are used in day to day life frequently and allow positioning and guidance for the user. Route guidance applications are already a well-established product and there are two main navigation system classifications. Mainly in-vehicle navigation systems and standalone applications running on a mobile device. GPS-enabled High-quality standalone navigation apps are available for most smartphones with the support of internet access for real-time updates.

History of Car GPS Navigations and Portable GPS & Smartphones – Satellite-based global positioning technology was there since early 1960s and the first-ever GPS navigation system for a production vehicle was, GM's Guide Star system for the 1995 Oldsmobile Eighty-Eight

978-1-5386-9418-3/18/$31.00 ©2018 IEEE

[1]. And when the advancement of technologies took place later 3D map views and text-to-speech conversions were added to the systems.

Standalone GPS devices were another revelation and by the mid-2000s, Garmin, Mio, Navigation, Magellan, TomTom, and others flooded the market with devices across multiple price points [1].

Today smartphones come with navigation apps which provide turn-by-turn navigation between two locations.

Navigation App Options - Among the available navigation systems, here we discuss the best GPS apps and navigation apps options for Android. Some popular navigation apps are Google Maps, HereWeGo Maps, BackCountry Navigator, MapFactor, etc.

Google Maps is the most popular navigation app around the world. It has turn by turn directions, live traffic updates, info about public transit schedules and it contains information about virtually every business known to man based on the location [2]. Users can download maps offline and using Google street view can have even closer looks at destination, businesses and even houses.

HERE WeGo is one of only a few serious competitors to Google Maps in the navigation app space [2]. In this app also users can have traffic information, public transit maps, and can customize by saving places for quick directions later.

Map Factor is another navigation app in this domain. It has basic turn by turn navigation and GPS features. MapFactor uses OpenStreetMap and can get free offline maps. Some other features are cross-border routing, 2D and 3D modes, day and night themes and more.

In TABLE I; we have summarized some features against existing navigation apps.

TABLE I. FEATURE COMPARISION BETWEEN EXISTING NAVIGATION APPLICATIONS [3,4,5,6]

Feature	Google Maps	BackCountry Navigator	HERE WeGo	Waze	Komoot
Turn by turn voice directions	+	+	+	+	+
Cycle Routes	+	+	+	-	+
Traffic Reports	+	-	+	+	-
Offline Maps	+	+	+	-	+
Add Stops	+	+	+	+	+
Map User Defined Directions	-	-	-	-	-

According to the literature review, it is evident that the current prevailing navigation systems do not provide dynamic route mapping upon user-defined directions which are given as voice commands. Therefore, in our proposed system, we provide the functionality of mapping the directions described by the user in voice commands.

III. METHODOLOGY

The proposed system "Direct Me" consists of the four main components. The four components of this system depend on each other. In brief the Automatic Speech Recognition (ASR) module takes the user input as voice commands in natural language and the identified voice commands are the inputs to the Natural Language Processing (NLP) component. Then the extracted navigational related information are the inputs to the Route Processing Middleware component, and the sequence of route requests are sent as the inputs to the Map API component. Mapped down route in an interactive user interface is the final output of the system.

The main goal of the proposed system is to capture the natural voice accurately and extract navigational commands from the voice input. To develop the system, we have used Pocketsphinx library, Stanford Core NLP library inside our components development with added configurations and modifications to accommodate our requirements as well as the newly developed components.

The following overall system diagram can be utilized in order to better explain the operations of the system.

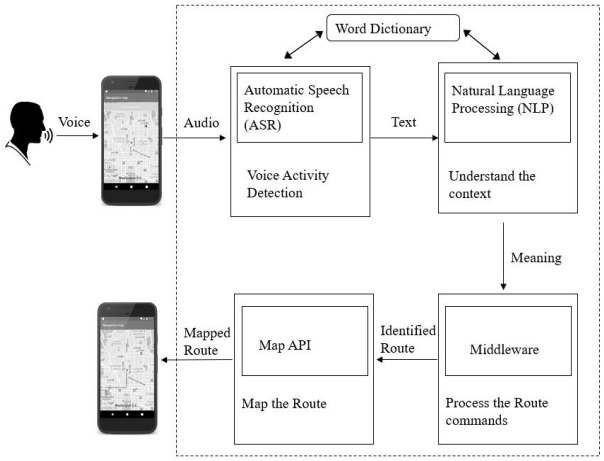

Fig. 1. High-Level Architecture of the System

Automatic Speech Recognition (ASR) – This module is responsible for identifying the human voice commands which include street names, landmarks, point of interests, distances, etc. Speech recognition does the speech to text conversion where the final output is the text corresponding to the recognized speech.

Speech is a continuous audio stream where rather stable states mix with dynamically changed states [7]. When considering the basic model of speech recognition there are

many steps taken place during the speech recognition process.

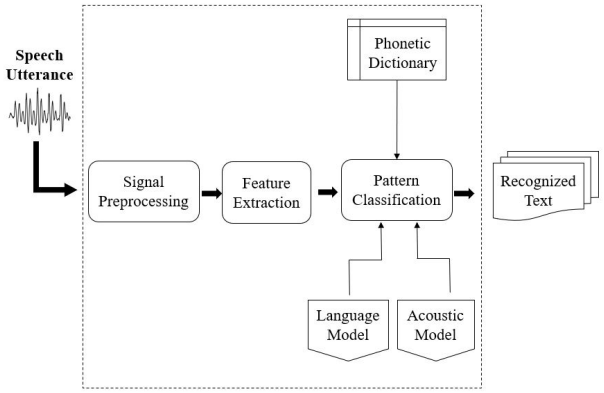

Fig. 2. Speech Recognition Module

In order to provide the recognized output, speech utterance goes through signal preprocessing, feature extraction and pattern classification with the assist of Phonetic Dictionary, Language Model and Acoustic Model. Phonetic Dictionary contains a mapping of words to phones and words are considered as to be built of phones. Pronunciation of a word is provided through this model. The Language Model is developed for detecting the connections between the words in a sentence with the help of pronunciation dictionary [8]. An acoustic model is a file that contains statistical representations of each of the distinct sounds that makes up a word [9]. Using these models, the recognition process is being done in the recognition engine. To get the functionality of speech recognition, we used Pocketsphinx library developed by Carnegie Mellon University which is designed for speed and portability.

As discussed in the introductory sections, our system needs to identify human voice commands which include junctions, street names, landmarks, distances, points of interests, etc. To identify those local names, some modifications are done to the ASR toolkit. Along with the localization modifications, some further techniques and modifications have been done inside this module in order to improve the accuracy of speech recognition.

After configuring the Pocketsphinx library, above modifications are done relating to the models mentioned in Fig. 2; Phonetic Dictionary, Language Model and Acoustic Model. Since this speech recognition model is designed for identifying navigation commands, the phonetic dictionary is created using commonly used navigation commands, local street names, junctions, landmarks, etc. In order to identify those local names, they needed to be included in the dictionary. The reduced size, navigation-oriented dictionary improves words recognition accuracy and processing speed. And also, the language model is generated to highlight the connections between words reflected in the phonetic dictionary.

Speech Recognition accuracy is the challenging part in this process. Due to different pronunciation, speaker, style of speech the recognition accuracy varies. To improve accuracy, Maximum-A-Posteriori(MAP) adaptation has been done with the acoustic model. For that adaptation process, recorded audio data containing navigation voice commands have been used. Using that methodology, the fit between the data and the model is improved.

After the recognition process, the recognized text corresponding to the input voice commands will be generated in this model. Then, that output text will be sent to Natural Language Processing Component.

Natural Language Processing Component – Application of Natural Language Processing(NLP) is one of the main interest areas in the research. Natural language processing is the process of enabling the computers to identify and understand the natural language which is a mainly explored area in artificial intelligence and machine learning [10].

This module enables the computer to understand natural language and extract the navigational related information from the direction commands. In brief NLP module takes the recognized text of the user's voice commands coming from the ASR module as the main input and process it to define the grammar structure with rules and relation of sentence construction.

Text preprocessing includes analyzing the natural language against the grammar and the rules, which consists of Morphological Analysis and Syntactical Analysis.

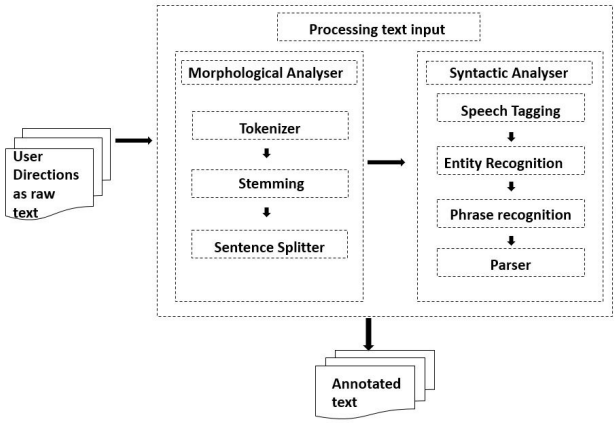

Fig. 3. Natural Language Processing Module

To deal with small word units for the identified text we use Morphological analysis. The input text is going through few processes called tokenization, sentence segmentation and stemming to do the separation of the sequence of words in a sentence, separation of sentences from a text or paragraph and reduction of words into their root words respectively.

Syntactical Analysis improves the precision of the navigation information retrieval by using the knowledge about the syntax of sentences in natural language. Part of Speech tagging (POS tagging) used to recognize the elements of a sentence like nouns, verbs, adjectives etc. This

is important to know the word types to analyze the grammar of the text. Named Entity Recognizer (NER) helps in identifying named and numerical entities. NER helps to identify the usage of nouns in a sentence comparatively to the context in which it is used.

e.g. Go straight to Kaduwela from Malabe

```
Dependency Parse (enhanced plus plus dependencies):
        root(ROOT-0, go-1)advmod(Kaduwela-4, straight-2)
        case(Kaduwela-4, to-3)advcl:to(go-1, Kaduwela-4)
        case(Malabe-6, from-5)nmod:from(go-1, Malabe-6)

Extracted the following NER entity mentions:
        Kaduwela LOCATION
        Malabe   LOCATION
```

Fig. 4. Location Entities Extracted from the Navigational Command

The phrase recognition (PR) caters locating groups of words, the phrases. PR is needed to maintain relationships between word groups that would lose their meaning if separated. This step gives the first approach to identify the relationships among the words. Phrases include the prepositional phrase, noun phrase, verb phrase, adjectival phrase and adverbial phrase.

Parsing is the process of structuring a sentence with the given grammar. Therefore, the sentences are fractionalized into the grammatical units. The structure of a sentence is represented in a tree structure [11]. The tree structure is very useful to find specific parts of a sentence as shown. The tree allows to narrow the information down from the sentence to its single words [10]. Each word of a sentence gets annotated with its type of grammar. This process allows the extraction of information from chosen syntactical units.

e.g. Go straight to Kaduwela from Malabe

```
Constituency parse:
(ROOT
  (S
    (VP (VB go)
      (PP (RB straight)
        (PP (TO to)
          (NP (NNP Kaduwela))))
      (PP (IN from)
        (NP (NNP Malabe))))))
```

Fig. 5. The Generated Tree Structure for the Given Navigational Command

Stanford Core NLP is contained a machine learning (ML) model itself, which can be used to extract the meaning of the words. ML model is trained using Conditional Random Field (CRF) algorithm for extracting the meaning of the input sentences given to the model.

Customized ML model was developed by modifying the existing model for training and taking out navigational meanings of the words. Using this trained model our system is capable of identifying whether the input word is a start-location, end-location, point of interest, landmark, street name, distance or a turn. The extracted useful information is placed in the predefined template for further processing. The training process required large data sets of different routes to increase the accuracy of identifying user-defined routes. After aggregating the ML model with the NLP component, the system identified the navigational words in the sentences effectively.

Finally, with help of the natural language processing module, our system extracts navigation-related information in users' voice commands and send that information to the route processing middleware to process the user intended route.

Route Preprocessing Middleware – This middleware component is inside the DirectMe web server which is developed using Java Spring-boot. The component is responsible for taking and further processing the identified route commands coming from the natural language processing module and sending those important route information to the route mapping module, in order to map it in an interactive interface.

Route handler is capable of understanding the input and check whether the route is contained a single destination or complex route with turn by turn directions. If it is a complex query, Route divider will separate information into segments based on the destinations, landmarks, point of interest etc. while interacting with the NLP module for extracting required data.

Route generator is responsible for arranging that processed information in a JavaScript Object Notation (JSON) format which contains all necessary data about starting points, destinations, turns, distances etc. Collected data are used to generate requests for map API and Route Sequence Generator will store a sequence of requests which are ready for sending to the map API. Subsequently, in a sequential manner, this module will send the requests one by one to the map API. Application of this mechanism will make the system very much eligible to map the user preferred route. This separation of the internal components is promoting the single responsibility principle in the application which is very important to this kind of development. Low coupling and high cohesion are maintained between the internal components.

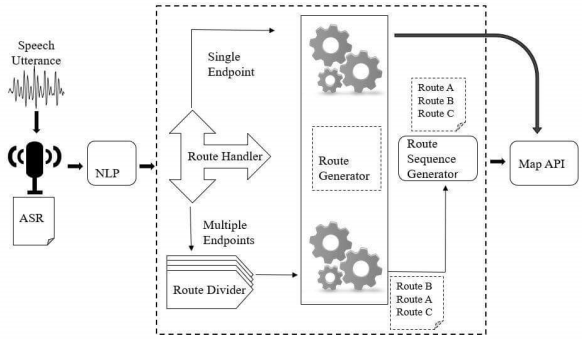

Fig. 6. Route Processing Middleware Architecture

To handle the security in DirectMe server our system uses OAuth2 mechanism along with the JWT tokens for authentication. Only if the authentication header in API request contains a valid JWT token it will be processed in the server. For the token generation system will use login function with API endpoint and client should call to the login and get the token for requests.

Map API and Route Mapping – When considering the inputs of the map API, processed navigational information coming from the route processing middleware need to be precise and more accurate. To achieve these qualities Google Maps API was chosen for this research. The major limitation we found with Google API is that it can only give a route when there are a start and an end location. To overcome this challenge NLP was used to create commands with start and end locations as described.

To draw the route on the map, at first navigational commands were taken out from the JSON response. Then these commands are processed and sent to the respective API provided by Google Maps. The response from the map API goes through some special methods which give another reason to prove this solution is unique from the existing systems.

e.g. Distances mentioned by the users are calculated to mark location points on the map. Finally using the polyline points on the API response, the route will be drawn on the map and showed to the user.

To get the most accurate distance we search for a range of distances, mark the point and check if it matches with the user's next command. (e.g. when user says 500m we consider that point is in between 450m to 550m range and then depend on the next command we map the point). To verify the mapped location is on the road we use road API. Finally, all marked locations are taken with their geo-coordinates to send to direction API, hence route will be provided with the most accurate end location.

IV. Results and Discussion

This section mainly explains the outcomes of the developed system. The System supports user voice as input and can extract and execute navigation instructions or the commands on the Google Map API. As a result, the system will map the route, embedded in the user's voice input.

Although some existing navigation applications support voice input, they do not extract and process route instructions from the input. They only consider about the locations from the input and only show them as pinpoints. Our results showed that the proposed system is able to extract the navigation instructions, such as Locations, start point, end point, turns, etc. And from those instructions map the user defined route in the map.

In speech recognition module the reduced size, navigation-oriented dictionary is used to provide the pronunciation for commonly used navigation commands and local names. The default dictionary provided by CMU does not contain local street names, locations, landmarks and there is much non-navigation associated vocabulary. Therefore using that dictionary we could not identify those local names and the accuracy also was poor because of much non-navigation associated words. After creating the dictionary with navigation oriented words and local names, we were able to solve the problem of identifying local names. And the accuracy was improved for the navigation commands than with the default dictionary but accuracy was not that much satisfactory. To address that problem acoustic model adaptation was done. Recorded audio data from different male and female users which contain navigation commands were used for that process. By doing so we were able to achieve 65% - 75% recognition accuracy. We will be focusing on improving the accuracy furthermore.

One of the limitations identified is the accuracy of ASR module being a dependency of the NLP module as the recognized speech from ASR is taken directly as an input to the NLP module. Furthermore, if there are faults in the recognized speech, NLP module will not give the correct output due to the mistaken sentences provided by the ASR module. To avoid above dependency and increase the accuracy, Deep Neural Network (DNN) can be used.

The processing of recognized voice commands is done inside the server using Stanford core NLP library. By the default processing, the meanings of words are given only as nouns, verbs, entities (location, person, organization), etc. But as per to our system requirements we need to identify navigation related entities. After the training of the Classifier model, NLP recognizes required navigational related entities such as start-location, end-location, distances, etc. From the presence of trained model, we were able to achieve 60%-70% accuracy gain when identifying those entities.

When the ASR module recognizes the natural voice commands and converts them to the text format, client application will send a request to the DirectMe server along with the attached text. It invokes the NLP functions and using the results of NLP module, a JSON object is generated and kept the navigation oriented information in a sequential manner. Then this JSON object will be sent as the response to the frontend request. Discussing on the Distance mapping in Map API there was one constraint we had to consider. A way of the direction (e.g. towards, to the left) should be mentioned with a name of a point of interest or with any navigational entity followed by the mentioned distance.

e.g. Go straight from Malabe about 500m towards Kaduwela.

If the direction has not been mentioned, we can only take an angular measurement of a 360-degree view of the area and find the possibilities from the current location with a radius of the given distance. Therefore, the system would not know which road it should take when mapping. Due to these reasons a direction must be mentioned whenever a distance is referred.

When comparing our results with other existing systems in the navigation domain, key findings emerge that current navigation systems in the market have many different features unique to each of them as well as some common features which are in almost all the available navigation systems. Voice recognition can be named as one of the most common features these days in this type of systems, but none of the available systems support natural language route directions inputs. Even though these available systems support keyword search, some local city names, street names, point of interests will not be able to recognize due to the different pronunciations of people in Sri Lanka. Also, the distance mapping feature is not currently available in these products even though the distance measuring from point to point feature is available.

With the achieved 65%-70% of accuracy level of the ASR module, route mapping gives overall 70% accurate mapping results for different route scenarios. The system "Direct Me" performs mentioned unique features to make an interactive road navigation system to respond for the user's voice inputs. Fig.6; and Fig.7; shows the mapped down route as the final result for the user provided direction commands.

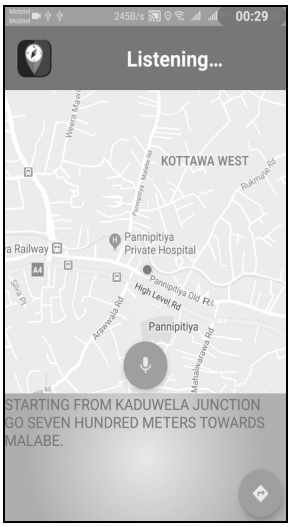

Fig. 6. Voice Commands Input UI

Fig. 7. UI for Map the User Described Route

V. Conclusion and Future Work

Direct Me is mainly focused to have the mapping of route directions as user described in voice commands. Results have shown that using a customized dictionary and adapting the acoustic model using recorded audio data from Sri Lankans can improve the recognition accuracy because when it comes to local names (locations, street names, landmarks, etc.) the pronunciation differs from those in generated models. In order to improve accuracy furthermore, we will focus on training a language model and use more adaptation data with a rich vocabulary.

Inside the NLP module, the recognized text data which includes the route directions is being processed and converted into annotated text using the customized functions written in NLP component. It gives productive results and furthermore can be optimized by developing algorithms to identify the different navigational command patterns. Basically, Tregex patterns can be used for identifying patterns in trees structure. The basic units of Tregex are Node Descriptions and Descriptions match node labels of a tree. Then the relationships between tree nodes can be specified. To increase the accuracy more and get effective results, we will be focusing on merging Deep Neural Networks (DNN) to the NLP module.

Considering the facts of the development of the country and the technical world, delivery services are getting popular day by day. In order to provide a quality service and deliver products without unnecessary delay, we need to know exact locations. Using existing systems, we can easily find the locations on the main roads, but it is proven with the help of Direct Me, these services can map the route even to a place in rural areas which cannot exactly point out in the map (e.g. a home or a building) with the closest approximate location. In the future versions of Direct Me, it will be facilitated to be used by visually impaired people and also, we will be focusing on providing an application which supports the Sinhala Language by modifying the ASR and NLP modules to reflect the localization.

References

[1] Early GPS Systems, Portables, and Cell Phones - The History of Car GPS Navigation | News & Opinion PCMag.com. (n.d.). Retrieved March 12, 2018.

[2] 10 best GPS app and navigation app options for Android - Android Authority. (n.d.). Retrieved July 25, 2018, from https://www.androidauthority.com/best-gps-app-and-navigation-app-for-android-357870/

[3] Techlicious, L. (n.d.). 5 Best Navigation Apps - Techlicious. Retrieved from https://www.techlicious.com/tip/best-navigation-apps/

[4] Turn-by-Turn Navigation Showdown: Google Maps vs. Waze. (n.d.). Retrieved March 20, 2018, from https://lifehacker.com/turn-by-turn-navigation-showdown-google-maps-vs-waze-1761550298

[5] Waze Vs. Google Maps: What's the Best Navigation App for You? | Digital Trends. (n.d.). Retrieved March 20, 2018, from https://www.digitaltrends.com/mobile/waze-vs-google-maps/

[6] 5 Android navigation apps for those who are sick of Google Maps - CNET. (n.d.). Retrieved March 20, 2018, from https://www.cnet.com/how-to/5-android-navigation-apps-for-those-who-are-sick-of-google-maps/

[7] Basic concepts of speech recognition – CMUSphinx Open Source Speech Recognition. (n.d.). Retrieved July 25, 2018, from https://cmusphinx.github.io/wiki/tutorialconcepts/

[8] Ghai, W., & Singh, N. (2012). Literature Review on Automatic Speech Recognition. International Journal of Computer Applications, 41(8), 42–50.

[9] S, K., & E, C. (2016). A Review on Automatic Speech Recognition Architecture and Approaches. International Journal of Signal Processing, Image Processing and Pattern Recognition, 9(4), 393–404.

[10] Chowdhury, G. G. (2003), Natural language processing. Ann. Rev. Info. Sci. Tech., 37: 51–89.

[11] Hoede and L. Zhang. Structural parsing. Memorandum 1527, Enschede, 2000.