Research article Available online www.ijsrr.org ISSN: 2279–0543

International Journal of Scientific Research and Reviews

Ask Gayatri v2: A Speech Based Question Answering System

B. Aparna[*1]

[1]Department of Computer Science, Gayatri Vidya Parishad College of Engineering, Vizag.

ABSTRACT

The best possible way of communication is always through speech. Because it is through voice only we completely express what we want. And machines are no exception. Instead of typing strenuously we can directly interact with our computer. Interaction made easy at minimal cost is our motto. In order to realize our dream we started off with a basic Speech based Question Answering System. When voice can be directly transformed into text and text is transformed to speech then its uses are immense. Our experimentation starts from a Question Answering System. We use "English" as a means to achieve our goal of interacting. Further this concept is employed to Speech based Question Answering System. This system allows asking questions in speech and retrieving the answers in speech. This contains three phases: one is Speech Recognition, second is Question Answering System and third is Speech Synthesis.

KEYWORDS: Question Answering System, Speech Recognition, Speech synthesis

[*]Corresponding Author:

B.Aparna

Department of Computer Science Engineering,

Visakhapatnam, Andhra Pradesh.

Email:botchu.aparna@gmail.com, mob: 7396585623

INTRODUCTION

In Information Retrieval (IR) and Natural Language Processing (NLP), a **Question Answering System** (QAS) automatically answers a question posed in natural language. Question Answering System is one of the emerging areas of research in Natural Language Processing applications of Artificial Intelligence. Question Answering System (QAS) is a man machine communication system. The basic idea of QA Systems in Natural Language Processing (NLP) is to provide more appropriate responses to the questions in a human like manner giving short and accurate answers.[6]

With the wide spread use of information in the internet exploration era, there is a recently renewed interest for retrieving short and accurate answers to the questions. QA System aims to retrieve point-to-point answers rather than flooding with documents or even matching passages as most of the information retrieval systems do. For example, the exact answer expected by the user for a query like "Who is the first prime minister of India?" The answer should not expect by user to dig bundles of documents that match with the words like first, prime minister, India etc.

The major challenging issues in Question Answering System is to provide accurate answers from tremendous data available on the web. The processing of time based information to answer temporal queries still remains as a challenge.

LITERATURE REVIEW

As the richness of information in the World Wide Web grows and users get used to the wealth of information, the need for an automated answering system becomes more urgent. We need a system that allows a user to ask questions in a specific language and receive a satisfactory answer quickly. In addition, the system has to validate if the given answer matches the requirements of the user. The problem of current search engines is that they return ranked lists, but do not give answers to the user.

Users

The system has to be designed in such a way that both first time or casual users as well as "power users" should be able to use such a system and the system should be able to cater to the needs of both categories of users. These users need different functionality, ask different questions and expect answers.

Questions

Generally questions are distinguished by their answers. Answers can be factual answers, opinions or summaries. Generally it is difficult to differ between these three types of answers. Furthermore the type of question makes no difference to the answer since the comparison and the storage of a question and answer set is the same. Next detect the different kind of questions like questions, which can be answered with “yes or no”, or the so-called “Wh” questions. These are questions which either begins with “Who is the president of India?” or with “How much did Manchester United spend on players in 1993?” There is an evidence that why and how questions tend to be more difficult to answer, because they require understanding causality or instrumental relations and these are typically expressed as clauses or separate sentences.

Answers

Answers can be short or long, a list, a summary or just a diagram or a picture. There are also different methodologies for constructing an answer. It is possible to give an exact answer or just to extract snippets from a document. The second solution is utilized by many search engines in the internet. If the answer is drawn from multiple sentences or various documents you have to take into account that the interconnection of the answer is low and maybe obscure. In that case the user has to trace out the necessary parts of the answer, which is relevant for him. The question is what makes an answer satisfactory! The point is if an answer is derived from an external resource, which was generated automatically, the system should present multiple answers. This allows the user to find a correct answer out of some available ones or out of a whole document. The hit ratio for an exact and correct answer given by an automatically generated answering system is very low. On the other hand, an answer given from a natural person should, if possible, be short and precise.

EXISTING SYSTEM

Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine-readable format. Rudimentary speech recognition software has a limited vocabulary of words and phrases and may only identify these if they are spoken very clearly. More sophisticated software has the ability to accept natural speech. Speech recognition applications include call routing, speech-to-text, voice dialing and voice search[1].

The terms "speech recognition" and "voice recognition" are sometimes used interchangeably. However, the two terms mean different things. Speech recognition is used to identify words in spoken language. Voice recognition is a biometric technology used to identify a particular individual's voice.

Speech Recognizer is a multipurpose project whose uses are many. In this project, the most widely used language “English” is considered as a means to achieve our goal of interacting with the computer and devices through voice and also to convert uttered words into its textual form. Since it is the most used language for translating a person’s thoughts, this language is chosen.

Speech Recognition software supports many functions such as recording of speech, controlling of devices with speech, etc. The function of speech recognition software (also called speech recognizer) include

- Conversion of uttered words to text
- Control of computer using voice commands
- Control of devices using voice commands
- Recording of speech and playback

These four are the basic functions of speech recognition. It can later be extended in a variety of ways, since it has a capacity to convert spoken words to their textual form.

Speech Recognition is used to translate the spoken words into textual form. Few speech recognition systems use training so that it can recognize the specific person’s voice. The system, which uses the training, is context dependent and which do not use training is context independent. We are developing a context independent system.

Speech recognition has few parameters, which vary from speech to speech.

One dimension is vocabulary size: If the vocabulary size is small the recognition is easy. For example recognizing yes or no words and recognizing zero to ten number sequences are easy.

Second dimension is how fluent, natural, or conversational the speech is: Individual words are easy to recognize than a continuous speech as individual words contain pauses make that recognition easy.

Third dimension is channel and noise: The speech captured using high quality and head mounted microphone is easy to recognize. The speech with noise makes recognition harder. Speech captured with quite environment increases accuracy than speech captured in a noise.

Final dimension is accent or speaker-class characteristics: If the speaker speaks the data on which the system was trained on, then the recognition is easy. Recognition is harder on children’s speech and foreign accented speech.

Table 1 shows the rough percentage of incorrect words (the Word Error Rate, or WER,) from state-of-the-art systems on different ASR tasks.

Table 1 Error rate

Task (Input)	Vocabulary (number of words)	Error rate %
11 Digits	11(zero-nine, oh)	0.5
Wall Street Journal read speech	5,000	3
Wall Street Journal read speech	20,000	3
Broadcast News	64,000+	10
Conversational Telephone Speech	64,000+	20

Variation due to noise and accent increase the error rates quite a bit. The word error rate on strongly Japanese-accented or Spanish-accented English has been reported to be about 3 to 4 times higher than for native speakers on the same task. And adding automobile noise with a 10db SNR (Signal to Noise Ratio) can cause error rates to go up by 2 to 4 times.

In this project, a package called sphinx is used, which uses Hidden Markov Model[2] and text to speech package called Free TTS and Question Answering System. We integrate the above three systems so that a system which takes the queries through speech and produces the result as speech.

PROPOSED SYSTEM

Speech recognition involves takes an acoustic waveform as an input and gives text words as output. HMM-based speech recognition systems view this task using the metaphor of the noisy-channel. The acoustic waveform is passed through this noisy channel which produces a noisy version of original sentence. Our goal is to build a model of the channel so that we can find out how the channel modified this “true” sentence and hence recover the sentence.

In figure1 we search through a huge space of potential “source” sentence and choose the one, which has the highest probability of generating the “noisy” sentence.

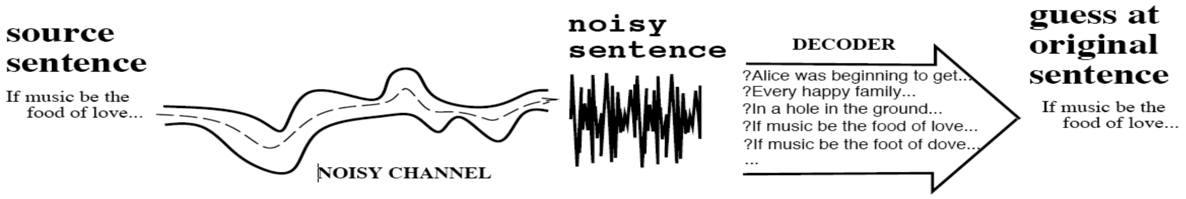
source sentence
If music be the food of love...
NOISY CHANNEL
noisy sentence
DECODER
?Alice was beginning to get...
?Every happy family...
?In a hole in the ground...
?If music be the food of love...
?If music be the foot of dove...
...
guess at original sentence
If music be the food of love...

Figure 1: The Noisy-Channel Model[1]

Implementing the noisy-channel model requires solution to two problems.

One is to find the sentence that best matches the input, so all the metrics are considered to find the “best match”.

As the speech is variable, an acoustic input sentence will never match any model for that sentence. Various probabilities are combined to get a complete estimate for the probability of a noisy acoustic observation-sequence given a candidate estimate for the probability then search through the space of all sentences, and choose the source sentence with the highest probability.

Second, since the sentences in English language is huge, we need an efficient algorithm that does not require to search through all the sentences but only finds the sentence that have a good chance of matching the input. This is done in decoding or searching phase that is done by Viterbi algorithm in HMMs. The search space is large in speech recognition so an efficient algorithm is needed to minimize the searches.

The probabilistic or Bayesian model for speech recognition that are introduced for part-of-speech tagging. Then introduce the various components of a modern HMM-based ASR system. The goal of the probabilistic noisy-channel architecture for speech recognition can be summarized as follows:

What is the most likely sentence out of all sentences in the language L given some acoustic input O?

The acoustic input O is treated as a sequence of individual “symbols” or “observations”, by slicing the input signal every 10 milliseconds and representing each input slice by a floating-point value of energy or frequencies of that slice. Each index then represents some time interval, and successive o_i indicate temporally consecutive slices of the input:

$$O=o_1 ,o_2 ,o_3,\ldots,o_t \qquad (1)$$

Similarly, we treat a sentence as if it were composed of a string of words:

$$W=w_1 ,w_2 ,w_3 ,\ldots,w_n \qquad (2)$$

Both of these are simplifying assumptions: for example, dividing sentences into words is a fine division sometimes and sometimes too gross. Usually in speech recognition a word is defined by orthography (after mapping every word to lower case).

The probabilistic implementation of our intuition above can be expressed as

$$W'=argmax_{W€L}P(W/O) \qquad (3)$$

The above equation gives the optimal sentence W. For a given sentence W and acoustic sequence O we need to compute P(W/O). Any probability P(x/y), we can use Bayes’ rule to break it down as follows:

$$P(x/y)= P(y/x)P(x)/P(y) \qquad (4)$$

We can substitute Eq. 4 into Eq. 3 as follows:

$$W'= argmax_{W€L} P(O/W)P(W)/P(O) \qquad (5)$$

In above equation we can ignore P(O) as it doesn’t change for each sentence. Thus,

$$W'= argmax_{W€L} P(O/W)P(W) \qquad (6)$$

To summarize, the most probable sentence W given some observation sequence O is computed by taking the product of two probabilities for every sentence and choosing the sentence for which this product is greatest. The components of speech recognizer that compute these two terms have names; P(W), the prior probability, is computed by the language model. While P(O/W), the observation likelihood, is computed by the acoustic model.

$$W' = argmax_{W\text{€}L}\ \overset{\text{likelihood}}{P(O/W)}\quad \overset{\text{prior}}{P(W)} \qquad (7)$$

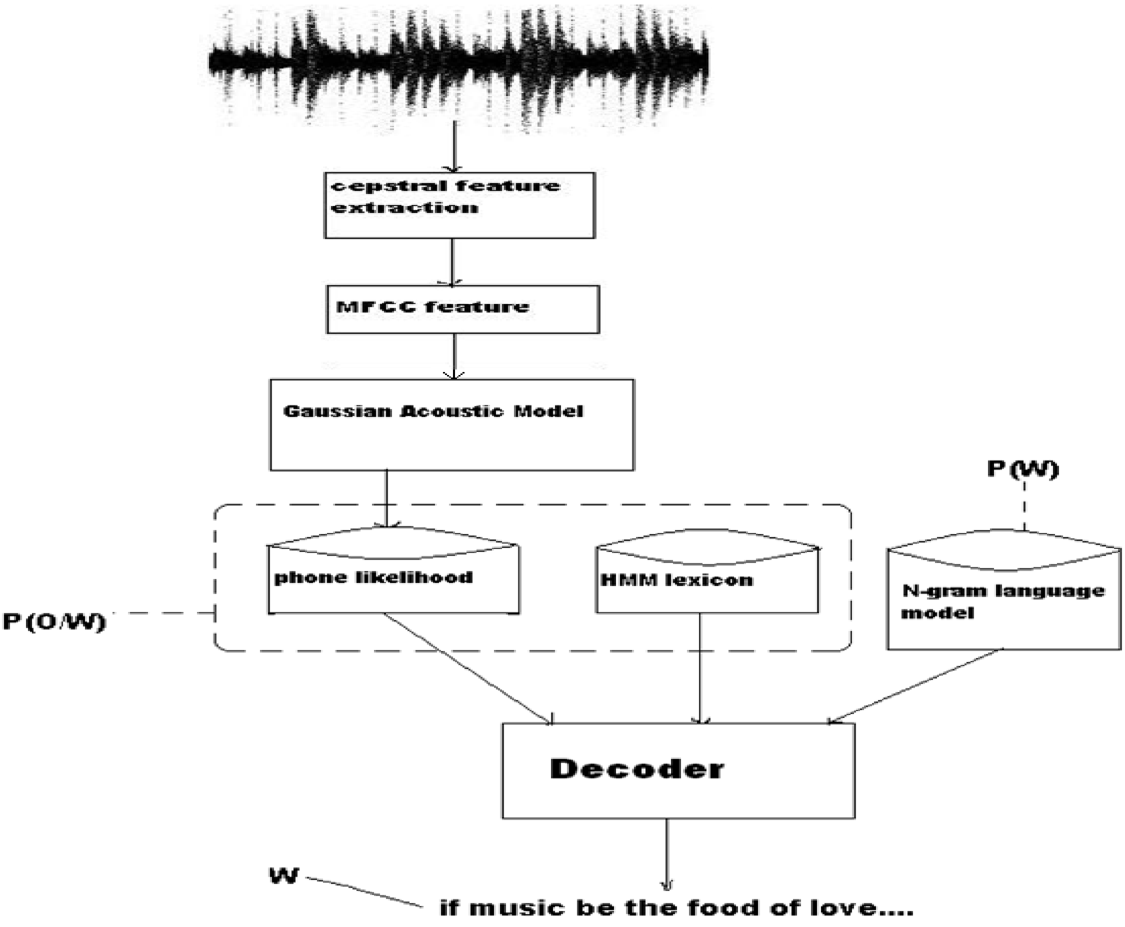

Figure 2: Architecture For Simplified Speech Recognizer Decoding Single Sentence.

Figure 2 describes the architecture of a speech recognizer decoding a single sentence. The cepstral features are extracted from the speech wave using MFCC feature extraction. The Gaussian acoustic model is constructed to the features. P(O/W) is computed using phone likelihood and HMM lexicon. P(W) is computed using N-gram language model. These probabilities are used to decode the words.

The language model

The language model prior P(W) express the probability that a given string of words is a sentence of English. To compute such a language model prior P(W) by using N-gram grammars, which assigns a probability to a sentence.

The acoustic model

The acoustic modeling compute the likelihood of the observed spectral feature vectors given linguistic units (words, phones, subparts of phones). The HMM can be used to build an acoustic model. Here we replaces the HMM with conditional random fields[3].

RECOGNITION PROCESS

There are three stages in recognition process. They are as follows:

The feature extraction stage

In this phase the acoustic waveform is sampled into frames that are transformed into spectral features. Each time window is thus represented by a vector of around 39 features representing this spectral information as well as information about energy and spectral change. ***The phone recognition stage***

In this phase an acoustic modeling is done by computing the likelihood of the observed spectral feature vectors given linguistic units. The output of this stage is as a sequence of probability vectors. One for each time frame, each vector at each time frame containing the likelihoods that each phone or sub phone unit generated the acoustic feature vector observation at that time.

The decoding Phase

In this phase, we take the acoustic model, which consists of this sequence of acoustic likelihoods, plus CRF, combined with the language model, and we output the most likely sequence of words. A dictionary is a list of word pronunciations, each pronunciation represented by a string of phones. The phones are considered in CRF and the Gaussian likelihood estimators supply the output likelihood function.

TEXT TO SPEECH

Free TTS is a speech synthesis system written entirely in the Java^{TM} programming language. It is based upon Flite: a small run-time speech synthesis engine developed at Carnegie Mellon University. Flite is derived from the Festival Speech Synthesis System from the University of Edinburgh and the FestVox project from Carnegie Mellon University.

ARCHITECTURE

In this system TREC9 Answer Selection algorithm is modified and used. This system takes query in speech as input and search in the Wikipedia. We have chosen Wikipedia because it is the authenticated information as it does not allow others to post the information as in Google. The data from the top five websites is taken and TREC9 algorithm is modified and used here to select the top ranked answer. The answer in text is converted to speech. So answer in speech is given as output.

The Architecture of the proposed system is shown in figure 3:

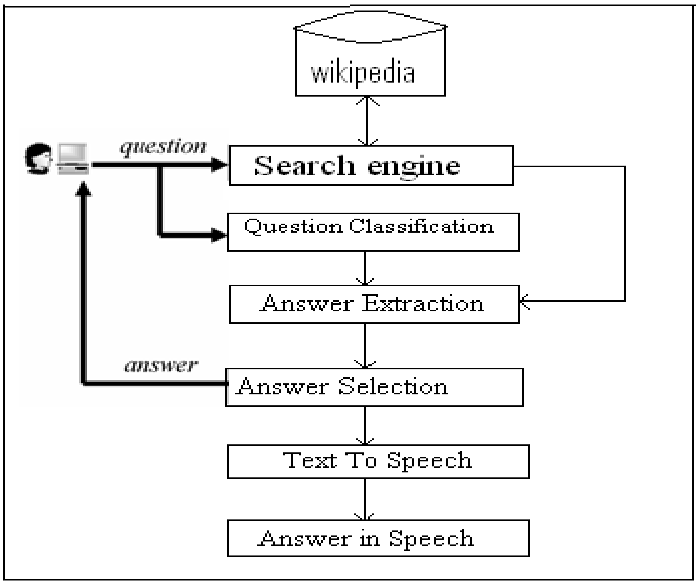

Figure 3: Architecture Of Question Answering System

CONCLUSION

This thesis presents a Question Answering System, titled, AskGayatri v2: A Speech-Based QA System.

This QAS is superior in terms of speech of answers when compared with other well known systems such as Ask Gayatri, Answer Bus, NSIR, and START, where speech interaction is absent. The results indicate that the accuracy measured as the number of matches is about double the other system’s accuracy.

Twenty test cases were run with the system. Four of them failed in the sense that the obtained answer is completely different from expected answer. The reason is that the actual answer to the query is not present in the Wikipedia Encyclopedia and the QAS picks the answer where the key word(s) is (are) located, even though the answer is irrelevant.

Gayatri QAS presents the answers in concise manner which captures the attention of the user. It answers user posed queries by the Query reformulation, Wikipedia search, Document retrieval, Answer extraction, Answer ranking, Answer selection and Answer presentation. ***Future Work***

Still work has to be done in answer extraction and query reformulation. Query reformulation may be revisited for additional modifications. The major challenging issue in question answering system is to extract accurate answers from tremendous amounts of data available on the Wikipedia and this is still a wide open field for further research. Another area which needs further investigation is to incorporate cross linguistic query facility, which allows the users to ask the questions and obtain the answers in their native language. The processing of time based information to answer temporal queries still remains a challenge. The accuracy for speech recognition must be improved.

PERFORMANCE EVALUATION

Comparative evaluation of response times of various Question Answering Systems is made and results are shown in Table 2.

Table 2: The Performance of various QASs in terms of Average Response Time

Question Answering Systems	START	AnswerBus	NSIR	AskGayatri	V2:AskGayatri(Speech based QAS)
	(Existing QA Systems)				(Proposed QA System)
Average Response Time (in seconds)	18.6	5.2*	4.6	2.1*	1.9

*Answer Bus[6] facilitates asking questions in two different languages. If two different languages are used, Answer Bus does not give the right answer though it takes a lot of processing time.

*AskGayatri cannot answer all the types of questions (how, why) and the question must start with ‘what’,’who’,’when’,’where’.

Some questions taken from TREC 9, TREC 10 are used to evaluate proposed system’s question answering performance and also compare its performance in terms of response time to that of three other similar systems (START, AnswerBus, and NSIR) and previous version of AskGayatri. It was observed that the proposed QAS is a speech based system and no other existing systems are speech based systems. The response times to each question by the systems and the lengths of the returned answers were recorded. In order to minimize the impact of network performance on the variations of the system’s response time, the answers from these systems are retrieved at the same time using the same computer.

REFERENCES

1. Jurafsky Daniel and Martin J, “Speech and Language Processing,” 2nd Edition, Pearson Prentice Hall.
2. Mc Callum.A, Freitag.D and Pereira.F, “Maximum entropy Markov models for Information extraction and segmentation” .Proc. ICML Stanford, California;June29, 2000; 591-598

3. Hifny Yasser and Renals Steve, “Speech Recognition Using Augmented Conditional Random Fields”, IEEE Transactions on audio, speech, and Language Processing; February2009;17(2)
4. Rohini Srihari and Wei Li, “InformationExtraction Supported Question Answering”, Eighth Text REtrieval Conference (TREC-8); Gaithersburg, MD; November 1999; 17-19
5. Michele Banko, Eric Brill, Susan Dumais, Jimmy Lin,”In Proceedings of 2002 AAAI SYMPOSIUM on Mining Answers from Text and Knowledge Bases”, March 2002.
6. Zhiping Zheng School of Information University of Michigan, Ann Arbor, MI 48109,” Answer Bus Question Answering System”.
7. Abraham Ittycheriah, Martin Franz, Wei-Jing Zhu, Adwait Ratnaparkhi, “IBM’s Statistical Question Answering System,”Nineth Text R Etrieval Conference (TREC-9), August 2000.
8. Adam Lally, Thomas J.Watson, “Natural Language Processing With Prolog in the IBM Watson System,” Stony Brook University, May 2011.