Auto-Derivation of Homophones-Ambiguity of Chinese-Language in Hidden Tool-Kit for Automatic-Speech-Recognition (ASR)

Muhammad Habib
Department of Computer Science
University of Central Punjab
Faisalabad, Pakistan
ch.muhammadhabib@gmail .com

Musrrat Parveen
Department of Computer Science
Govt. College Women University
Faisalabad, Pakistan
mussratgh@gmail.com

Muhammad Raheel Zafar*
Department of Computer Science
National Textile University,
Faisalabad, Pakistan
raheelzafar@ntu.edu.pk

Shahbaz Ahmad
Department of Computer Science
National Textile University,
Faisalabad, Pakistan
shahbazsahi@ntu.edu.pk

***Abstract*— The paper discusses the problems of Automatic Speech Recognition (ASR) systems, as there is still a considerable gap between human and machine performance due to their lack of robustness against speech variability especially in noisy environment such as in background music, background speech and individual accents. The focus of this paper covers the most important problem "Homophones Dis-ambiguity" occurring in almost every language. However, it effects in Chinese language are notified as every character in Chinese language is pronounced as a syllable. The Architecture that caters with the matter of Chinese Homophone Disambiguation using hidden toolkit is also proposed.**

Keywords— Adaptive Learning, Corpus, Homophone Disambiguation; Automatic Speech Recognition; Hidden Markov Toolkit

I. INTRODUCTION

The issue of automatic speech recognition with the assistance of a PC is a matter of concern, and the explanation behind this is the intricacy of the human language. The way toward working of structure for mapping the acoustic signs to a series of words can be considered as speech recognition or generally automatic speech recognition (ASR).[2]

People while listening utilize more their ears, they then work on the basis of the information they have regarding the subject and the speaker. Words are not randomly sequenced together, there is a syntactic structure and repetition that people use to foresee words not being spoken. Moreover, figures of speech and how we "as a rule" say things makes the prediction significantly easier. We just have the signal of speech in ASR. We can obviously build a model for the syntactic structure and utilize some sort of measurable model to enhance the chances, yet there are as yet the issue of how to model world learning, encyclopedic information and the information of the speaker.[11] We can, obviously, not demonstrate entire knowledge of the world comprehensively, but rather the query is the amount that ASR will work on to measure up the human comprehension.[4]

II. DIFFICULTIES WITH SPEECH RECOGNITION APPLICATIONS

Speech recognition software has progressed significantly since it was first developed, but then again it still has some huge complications that inhibit it from being used completely as a technique of transcription. Some of the speech recognition problems that are hard to resolve.

It consists of followings:

1. Disparities in the pronunciation of words.
2. Individual accents, homonyms and unsolicited ambient noises.
3. An additional problem incorporates the sort of equipment used to basically enter the sound, since the results can have an incredible impact in how the product will comprehend the speech. There is an issue of not knowing the foundation of the words being vocalized, which can prompt transcript that has no verbalization or loose spellings.
4. One of the furthermost elementary speech recognition difficulty is the superiority of the input devices being used. If a microphone is not delicate enough or is excessively complex, then it can make audio evidence that is hard for the software to decode. This is particularly factual when a microphone is so delicate that the speech is partial, creating the recognition software closely unworkable.
5. A related difficulty stems after background noise that can be difficult to disperse out from the foremost speech

and can effect incorrect transformations when comprised in the speech handling.

6. Alterations in pronunciation, accents and language tempo combine to arrangement of one of the more universal speech recognition difficulty. When a word can be pronounced in some ways, the software can turn out to be confused and misunderstand what did is being said. The similar can happen when an individual speaks gentler or quicker than the program imagines. There are some incomplete results, for example working out the software in the speech patterns of a user and by means of dynamic time-warping algorithms to compete the speech to the database of models, nevertheless they do not resolve all the complications [6].

III. Methodology

We sustained to push voice as a control route into devices in unrestrained noise surroundings, it became vibrant that we had grasped the bounds of a hardware-based, noise-centric method. Constructing chips that tried to recognize noise forms and strain them out wasn't functioning. In our case, we learnt early on that we wouldn't be capable to separate and chunk every casual sound. That led us to the finding and growth of a machine-learning method that allows us to separate all of the sounds of the human voice through a profound neural network and simply let that done. Because of doing so, we block practically all background noise.

As we recognize, somewhat exact natural language may identifiable around tens accents. Despite the indistinguishable word phonemic arrangement, if it is pronounced in dissimilar accents, therefore, we drive sound waves, which are diverse from each other. Alterations in pronunciation, accent and pitch of speech in overall, produce one of the utmost mutual problems of speech recognition. If there are a ration number of accents in language we must produce the acoustic model for each distinctly. When the word is pronounced inversely, then the software developed can be confused and misinterpret (insight) also properly what is pronounced. The similar can also happen, if the human speaks gradually or vice versa rapidly, then the program imagines. Acoustic model (such as hidden Markov models) whose parameters are assessed by means of speech information from a great set of speakers. There are two major alterations which happen among speakers: acoustic alterations which are linked to the scope and form of the vocal tract, and articulation alterations which are usually signified to as accent.

A. *Complex and Important Problem*

The challenge of the speech recognition problems is the recognizing the background of the words being vocalized. Computer software is incapable to classify the proposed meaning of a gathering of words, leading to an amount of problems by the recorded text. Words that have sound alike, for instance "their" and "there", can simply be exactly implied when the situation of usage is recognized. For this similar motive, correct punctuation is closely unbearable for the software to place created exclusively on significant order of words. There is efficient transcription software that is used in grounds for instance medicine, nevertheless the outcome is frequently a block of words without any type of parting, meaning it still receipts a human transcriptionist to manage the document and produce a legible ultimate copy.

These issues lie in almost all dialect however we consider here a level-headed discussion and accentuation for Chinese language. Because in Chinese dialect every single character is articulated as a syllable. In Chinese homophone disambiguation the fundamental issue is how to decision the furthermost similar homonym, for example, character, for each syllable when a request of syllables is moved in. In other word we can describe syllable as one or other letter on behalf of a large word containing solitary uni-intermittent sequence of speech sound.so to resolve the difficulty in Hsin-Hsi Chen and Yue-Shi Lee plan system design for Chinese homophone disambiguation.

IV. The Hidden Markov Toolkit

Several toolkits are accessible now for dealing with the algorithms being used in speech recognition, letting a recognizer to accentuate on the imperative matters required in building a speech recognizer for e.g. recognizers estimation and their preparing along with information planning. Any of these toolkits can be displayed as the Hidden Markov Toolkit (HTK).

HTK develops frameworks for regularly practicing the strength of Hidden Markov models being a software toolkit. It has been industrialized by the Speech Individual by Cambridge University Engineering Section. In any case, HTK is basically made for building Hidden Markov Model to deal with speech processing, specifically speech recognizers. It can be used to maintain wide variety of tasks here and also used in models in perspective of entire word or sub-word portions through connected speech recognition. In spite of the way that HTK contains 19 instruments that achieve a few tasks that is treatment of translations, Viterbi deciphering, coding data, different styles of Hidden Markov and what's more Baum-Welch re-estimation, results examination and across the altering of Hidden Markov Model descriptions up to an extent.[9]

A. *HTK Software Architecture*

The working of HTK is shape into library parts, they guarantee that each device interfaces the outer world in precisely the comparative way and a programming surrounding for the arrangement of custom apparatuses or the consolidation of recognizer working in a sales is setup. Figure 1.1 showcases the library parts and their constancy. Input/output of the user and cooperation with the working framework is measured by the library part.

1. HMem is used to organize The HShell and altogether memory management.

2. HNet aims at lattices or networks,

3. HDict intended for dictionaries, HVQ on behalf of VQ codebooks

4. Every file type in HTK has a devoted interface component necessarily.

5. HLabel delivers the interface designed for label files, HLM on behalf of language model records,

6. HModel on behalf of Hidden Markov Model descriptions. Altogether speech input and output by the wave form side by side through HWave and by the parameterized close through HParm.

7. In addition to given that a reliable interface, Hwave and HLabel maintains several file arrangements letting data to be trade in from supplementary systems.

8. Overall mathematical provision is delivered through Hmath and the signal dispensation operations desirable for LPC and MFCC speech exploration remain in HSigP.

9. Haudio is used to maintain Direct audio input.

10. And decent collaborating graphics is delivered through HGraf. Hutil offers many utility practices for operating HMMs although Htrain and HFB comprise provision for the countless Hidden Toolkit training tools.

11. Provision for the several HTK adaptation tools are offered by HAdapt. Lastly, Hrec comprises the foremost recognition dispensation utilities.

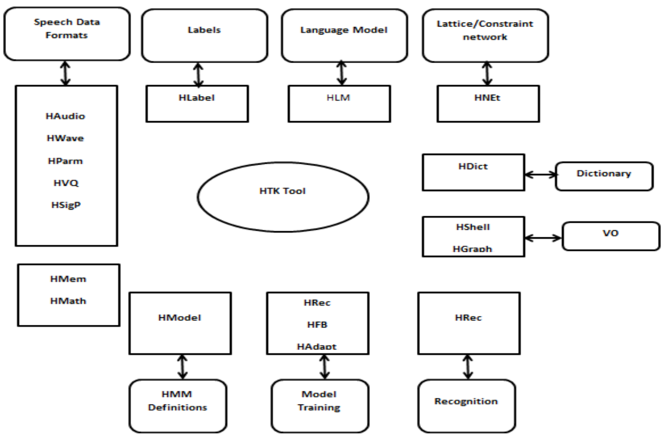

Figure 1.HTK Libraries

Control in overabundance of the execution of these library parts is handled through setting design factors or through setup documents that involve an extensive list of configuration factors.10]

V. Overview of the HTK Tools

The Figure 1.2 provides a general idea of the HTK tools is distributed into groups conferring to the different processing stages. The general process is distributed into four phases.

- Data Preparation
- Training Phase
- Testing Phase
- Analysis Phase

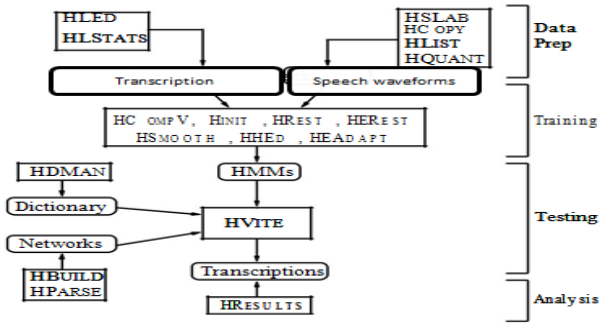

Fig 2. HTK TOOLS

A. *Data Preperation*

An arrangement of discourse information documents and their related interpretations are mandatory with a specific end goal to manufacture an arrangement of Hidden Markov Models. Even though discourse information will be achieved from database records. It is utilized as a part of preparing to be changed into the reasonable parametric frame and any related interpretations to have the exact configuration and utilize the word names must be considered. The instrument HSLab can be used to physically decode it with any essential translations. Albeit whole HTK instruments can parameterize waveforms on-the-fly, in setting it up is regularly well to parameterize the data in certain events at least for once while recording speech. The HCopy tool is utilized for this reason, this tool can imitate a couple source documents to a yield record. For the most part, HCopy tool duplicates the entire document, in any case a scope of systems are conveyed for mining segments of records and connecting records. Through set the appropriate setup factors, entire documents can be changed to parametric shape as they are just perused in. So, essentially duplicating each document in this comparative way accomplishes the required encoding. With the assistance of HList tool that can be utilized to monitor the substances of all speech record and in the interim it can likewise use to interpret contribution on-the-fly, it can also help to check the results of all changes as preparing massive measures of information. Regularly the names are utilized as a part of the interesting source translations and it won't be precisely as basic, for example, for the reason that of alterations utilized as a part of the telephone sets. Moreover, Hidden Markov Model preparing may include the names which is considered as setting ward. The HLEd tool is intended to make the fundamental changes to name records it is a script-driven name corrector. HLEd can likewise be utilized for single Master Label File MLF is regularly more appropriate for progressive planning. In conclusion to information planning stage, HLStats can gather and show measurements including files being labelled and where basic, HQuant device can be utilized to shape a VQ codebook in readiness of the data for the development of discrete probability through Hidden Markov Model system.

B. Training Phase

HTK lets HMMs to be developed with any expected topology. Portrayals can be stored as better than average text records and from this time it is possible to right them with any appropriate word processor. Through the exclusion of the probabilities in transition, altogether HMM parameters accepted in the model significance are disregarded. Model clarification and its persistence is just to distinguish the topology and regular appearances of the Hidden Markov Model. The definite parameters will be figured in advanced through the training tools.

Functional standards for the transition probabilities requisite to be assumed but the training procedure is very unresponsive to these as satisfactory and modest plans for selecting these likelihoods is to create all the transitions out of slightly state similarly probable. If segmented transcriptions are accessible, then the tools HInit and HRest offer inaccessible word style training by means of the completely labeled data in place with bootstrap data.

HInit can be utilized to offer unique estimations of word models that are complete in which the perception courses of action are understandings of the reliable vocabulary word. HInit can likewise be utilized to deliver unique estimations of seed Hidden Markov Models for sub-unit assembled speech acknowledgment. In this case, the perception plans will contain areas of consideration. HInit will be modifying this accessible information mechanically by simply giving it an area mark. In above applications HInit for the most part continues by methods for a model HMM, that depicts the fundamental Hidden Markov Mode topology, for example, it requires the type of the basic Hidden Markov Model barring means, hence any fluctuations are ignored. The model of the transition matrix will be required in both that admissible transitions and in their unique probabilities. Transitions, that are allocated zero probability will continue zero and from this time signify non-allowed changes. HInit estimations transition likelihoods through totaling the amount of times every state is visited throughout the arrangement procedure.

HRest implements simple Baum-Welch re-estimation of the limits of a solitary Hidden Markov Model with a set of observation arrangements. HRest is planned to function on HMMs with original parameter standards assessed by HInit.

HERest is utilized to make a single re-estimation of the parameters of an arrangement of HMMs by methods in preparing type of the Baum-Welch algorithm. Preparing information contains of various articulations of each of which commits an interpretation in the system of an average mark document (area limits are disregarded). In the interest of each preparation articulation, a consolidated model is effectively made through linking the phoneme models accepted through the translation. Each telephone set has the accumulators in an indistinguishable arrangement allocated to it which are utilized as a part of HRest. However, in HERest they are effective means of achievement in a normal Baum-Welch ignoring each trained sound by means of the consolidated model.

The HHEd tool is a Hidden Markov Model description corrector which will clone models into context-dependent groups, put on a variety of parameter tying and increase the amount of combination mechanisms in stated distributions. To progress performance for exact speakers the tools HEAdapt and HVite can be used to familiarize Hidden Markov Models to improved model the features of specific speakers using a minor quantity of training and adaptation of data.

C. Testing Phase

HTK offers a solitary recognition tool known as HVite, that uses a token passing algorithm corresponding to the one and only defined to achieve Viterbi-based speech recognition. HVite takes as input a network relating the permissible word arrangements, a dictionary describing in what way each word is pronounced and a set of Hidden Markov Models. It functions by transforming the word network to a phone network and then assigning the suitable Hidden Markov Model explanation to each phone occurrence. Recognition can then be achieved on either on direct audio input or on a list of stored speech files. HVite can maintain cross-word tri-phones or it can route with several tokens to produce lattices comprising several hypotheses. It can also be organized to rescore lattices and achieve compulsory arrangements. The word systems desirable to drive HVite are put in storage using the HTK average lattice arrangement. This is a text-based arrangement and from this time word systems can be formed straight by means of a text-editor. Conversely, this is somewhat monotonous and from now HTK offers two tools to support in making word systems.

Initially, HBuild sub-systems to be formed and utilized under advance level systems. In this way, however the same small level representation is worked on and significant redundancy is removed. HBuild can also help to create word loops and it reads an upheld off bigram language model and modify the word circle moves to join the bigram probabilities.

As a substitute to identifying a word system straight, an advanced level grammar representation can be used. This representation is created on the Lengthy Backus Naur Form (EBNF) used in compiler description. The tool HParse is provided to transform this representation into the corresponding word system. Either technique is selected to produce a word system, it is beneficial to be able to see samples of the language that it describes. The tool HSGen is delivered to do this. It takes as input a network and then accidentally crosses the system outputting word cords. These strings can then be reviewed to guarantee that they resemble to what is essential. HSGen can also work out the experiential perplexity of the assignment. As a final point, the building of great dictionaries can include integration of a variety of alterations on respectively areas. Dictionary organization tool HDMan is provided to support with the related procedure.

D. Analysis Phase

The ultimate stage of the HTK Toolkit is known as the analysis stage as soon as the HMM-based recognizer has been constructed, it is essential to assess its performance by associating the recognition consequences with the accurate reference transcriptions. Analysis tool called with the accurate reference transcriptions. An analysis tool HRESULTS is used for this persistence HRESULTS achieves the contrast of recognition outcomes and accurate reference transcriptions by using dynamic programming to support them .The valuation criteria of HRESULTS are well-matched comparatively through US National Institute of Standards and Technology (NIST).

Execution of Htk Processing States on Chinese language Model:

It is a very significant section of our thesis because in this section we adapt and modify the present architecture that is for Chinese homophone disambiguation. It is specified before that in-speech recognition homophone disambiguation is foremost problem. We draw a appropriate attention in this Architecture so that to describe a stages using hidden Markova toolkit (HTK).

VI. PURPOSE TO DEFINE PHASES

In the life cycle each phase has its specific process. Distributing an architecture into phases make it possible to lead it in the finest conceivable direction. The entire work burden of a project is distributed into minor components, therefore making it simpler to monitor. Every architectural project tracks a sequence of strong stages and an architect's professional capability transports stability and superiority to each step in the constructing process.

We state stages to rely on what we have to accomplish and how to achieve it. The goal of the Amplification of stage is to state the system architecture to offer an steady reason for most of the plan and execution e exercise [12].The design advances out of a thought of the furthermost significant prerequisites (those that have an unnecessary impact on the design of the framework) and a valuation of dangers.

This overall goal interprets into main purposes, each speaking a main area of risk. You address risks related with necessities (are you constructing the correct application?) and dangers related with architecture (are you constructing the correct solution?). Lastly it is necessary to talk about the risks related to the procedure and tool environment (do you have the correct procedure and the correct tools to prepare the work?) Speaking of these dangers guarantees that you can transfer into the subsequent phase with a least possible of threat and problems. In the architecture we describe four phases with Hidden Markov Toolkit and via HTK tool we outline data preparation phase, training phase, testing phase and analysis phase.

VII. SYSTEM ARCHITECTURE

Study of Chinese homophone disambiguation has turned out to be dynamic as of late as PC readable corpora turn out to be progressively accessible. Since each character is articulated as a syllable in Chinese dialect, the real issue of Chinese homophone disambiguation is how to choose the most likely homonym, i.e., character, for every syllable when an arrangement of syllables is entered. As a rule, two noteworthy methodologies, i.e., statistical and dictionary based can be connected to this issue. The fundamental idea regarding the dictionary-based approach is to conMandarin Chinese has roughly 1,300 syllables, 13,094 regularly utilized characters, and around 100,000 words keeping in view that every character is dealt as a syllable. In this manner, numerous syllables are shared by various characters. A few syllables really compare to more than 100 characters by a grouping of syllables into characters using dictionary. [1]

A. *Data Preparation phase*

Test sentence, NTUC2S framework, word lexicon, learning base, phonetic word reference. Test sentence which is first changed into a request of syllables through C2S converter and afterward the syllables are enhanced back to a yield message through S2C converter. The rightness of the S2C interpretation can be mechanically figured through a relationship of the info content with the yield text. In this engineering, character-to-syllable converter, for instance, NTUC2S framework, is considered to change a course of action of characters, like C =c1, c2, c3, ..., cn, into a series of syllables, e.g. S =s1, s2, s3, ..., sn. NTUC2S framework needs roughly heuristic tenets to segment a series of characters C into a series of words Withee vulnerabilities in syllables is consolidated by relating the heuristic rules. These heuristic principles are altogether stored in the learning base. A order of syllables S is then produced and directed to NTUS2C system, which is a syllable-to-character converter. It comprises of three foremost modules:

1. Lattice Creator For Characters.
2. An algorithm of Character Lattice Search.
3. Language Model or framework.

Character lattice creator incorporates a phonetic dictionary to produce homophones. Every syllable is rotten into three portions:

1. The starting or the Initial Part
2. The ending or the Final Part
3. Tone

Depending on these parts, the consistent homophones, such as characters can be take out from the phonetic dictionary. [8,15]

B. *Training Phase*

As per above model Training stage incorporate Viterbi algorithm. Training phase is depicted by taking elements utilizing vast amount of speech instances "training data." In preparing stage perceived speech is recorded pre-handled and after that enter the principal stage. In our model for preparing stage we apply Viterbi search algorithm and Markov character bigram language model. The Viterbi algorithm considered as dynamic programming calculation for taking the decision in the furthermost expected succession of Viterbi path which are the concealed states and places an impact in a group of watched occasions, particularly with regards to Markov data sources and HMMs. It is normally utilized as a part of speech recognition for example, in speech-to-text the acoustic signal is saved as the observed sequence of occasions, and a string of content is reflected to be the "hidden cause" of the acoustic signal. The Viterbi algorithm finds the most extreme expected string of content accepted the acoustic signal. In the fields of computational semantics and probability a n-gram is an arrangement of n no. of items from an expected series of content or speech.

The items can be, syllables, phonemes, words, letters and base pairs conferring to the application. The n-grams naturally are composed from a speech or text corpus. When the substances are words, n-grams may also be known as shingles. An n-gram of size one is raised to as a "unigram" size two is a "bigram" size three is a "trigram" Superior sizes are occasionally raised to by the value of n, "four-gram" "five-gram", and so on.

These two are a statistical base method in our model to regulate the furthermost likely character for each syllable. However, the two strategies have struggled from a few confinements because of the run-time conditions irregularly. That is, the rightness of disambiguation will extensively be affected through run-time circumstances, their own styles and clients. Some adaptive learning methods have been suggested in our model to deal with this problem. Over a character lattice creator, a character lattice L is produced and input to the character lattice search algorithm. Therefore, Markov character bigram model and Viterbi search algorithm are approved in this training to discover the statistically best path. Viterbi search algorithm is a famous algorithm active for optimal-finding complications. Defining the optimum path O after L is an efficient approach [13,14].

C. Testing Phase

According to above model testing phase include adaptive learning model. The model obtains these two significant bits of information, such as. C and O and informs the occurrences of the associated accesses in the uni-character or bi-character training table. Table. As testing phase is categorized by taking out features from testing data "data speech". Testing data are coordinated through model that is built from training data Self-learning techniques have been proposed in testing stage known as adaptive learning strategy. This method licenses clients to depict new words all through the information system. Even though the arrangements of characters or words is concurred animatedly giving to the practice. These client methodologies can be utilized as a part of the framework for enhancing the correctness of the disambiguation. These strategies lessen the impacts of run-time conditions. That is, when errors happen, the wrong characters will be changed by clients. Over the modification, the framework acclimates itself. It absorbs the alterations among the accurate outcome and the mistake outcome. These form the beneficial run-time response information. On the other hand, the system absorbs from the errors it types.

D. Analysis Phase

In above model analysis phase involves Automating Evaluation The automatic estimation is formed as soon as O is produced from NTUS2C system, the input sentence C is matched with the output verdict O. As analysis phase achieves the contrast of recognition outcomes and accurate reference transcriptions by means of using dynamic programming to bring it into [9].

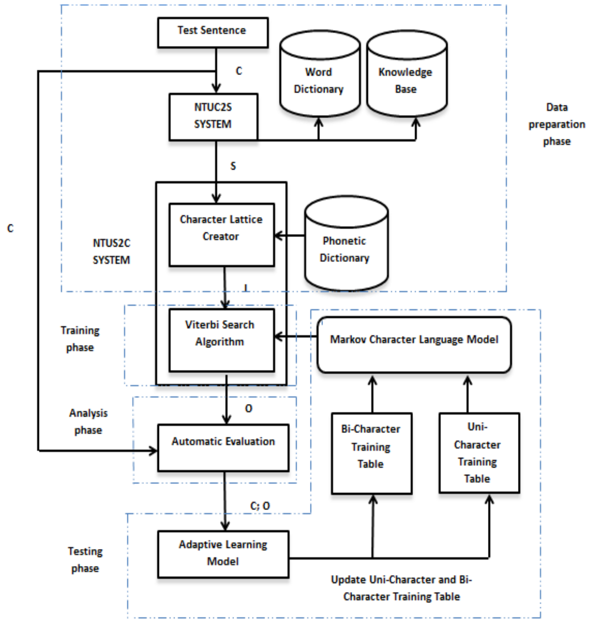

Figure 3: System Architecture for Chinese Homophone Disambiguation

VIII. Conclusion and Future work:

Speech recognition is one of the utmost integrating areas of machine intelligence. Speech recognition is a challenging problem to deal with. The performance of the system decreases as the size of the vocabulary increases. No satisfactory solution has been proposed so far so that the system can properly distinguish among similar words like “to” and “two” because they have similar sound phonemes. So, we proposed Architecture that solves the problem of Chinese Homophone Disambiguation using hidden toolkit which focused on chines language. Eventually we conclude that this project can be used for any language to solve the problem of homophones in speech recognition system.

IX. References:

[1] J. H. M. Daniel Jurafsky. Speech and Language Processing, An introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, Upper Saddle River, New Jersey 07458, 2000.

[2] E. A. Jens Allwood. Corpus-based research on spoken language. 2001.

[3] B. Schneiderman. The limits of speech recognition. Communications of the ACM, 43:63–65, 2000.

[4] Young, S., Kersaw, D., Odell, J., Ollason, D., Valtchev, V., Woodland, P. C., The HTK Book (for HTK Version 3.0), Cambridge University Engineering Department.

[5] Woodland, P.C., Odell, J., Young, S.J., Large Vocabulary Continuous Speech Recognition Using HTK, in Proc.

[6] C.H. Chang, "Bidirectional Conversion between Mandarin Syllables and Chinese Characters," Proceedings of Computer Processing of Chinese and Oriental Languages,pp. 174-181, 1992.

[7] C.H. Chang, "Design and Evaluation of Language Models for Chinese Speech Recognition," Proceedings of CMEX Workshop on Chinese Speech Recognition,pp. 248-254, 1992.ICASSP 1994. S.I. [8]Chen, C.T. Chang, et al., "The Continuous Conversion Algorithm of Chinese Character's Phonetic Symbols to Chinese Character," Proceedings of National Computer Symposium,pp. 437-442, 1987.

[9] C.K. Fan and W.H. Tsai, "Relaxation-Based Word Identification for Removing the Ambiguity in Phonetic Chinese Input," International Joint of Pattern Recognition and Artificial Intelligence, vol. 4, no. 4, pp. 651666, 1990.

[10] R. Sproat, "An Application of Statistical Optimization with Dynamic Programming to Phonemic-Input-toCharacter Conversion for Chinese," Proceedings of R.O.C. Computational Linguistics Conference III, pp. 377-390, 1990.

[11] Zafar, Muhammad Raheel, Muhammad Habib, Kashif Razzaq, and Raza Sattar. "Big Data: Bottleneck Solution for Big Companies." International Journal of Information Technology and Electrical Engineering, Vol 4 issue 3 pp: 6-12.

[12] Jan, H., Paul, A., Minhas, A. A., Ahmad, A., Jabbar, S., & Kim, M. (2015). Dependability and reliability analysis of intra cluster routing technique. Peer-to-Peer Networking and Applications, 8(5), 838-850.

[13] Jabbar, S., Ullah, F., Khalid, S., Khan, M., & Han, K. (2017). Semantic interoperability in heterogeneous IoT infrastructure for healthcare. Wireless Communications and Mobile Computing, 2017.

[14] Khurram Zeeshan Haider, Kaleem Razzaq Malik, Shehzad Khalid, Tabassam Nawaz, Sohail Jabbar, “Deepgender: real-time gender classification using deep learning for smartphones”, Journal of Real Time Image Processing, Springer, pp 1 – 15, 2017

[15] Ahmad, Shahbaz, and Muhammad Tanveer Afzal. "Combining Co-citation and Metadata for Recommending More Related Papers." 2017 International Conference on Frontiers of Information Technology (FIT). IEEE, 2017.