IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Proposal to Use GAN For Speech Recognition in Natural Language Processing

[1]Vijetha Ringu, [2]Aileni Eenaja

[1,2]Assistant Professor

[1,2]Computer Science and Engineering

[1]Nalla Malla Reddy Engineering College, [2]Holy Mary Institute of Technology & Science, Hyderabad, India

Abstract: Usage of advanced technology in our daily routine has been increased. NLP has been more needful to humans. Automatic speech recognition, translating of spoken words into text, is still a challenging task due to the high viability in speech signals. So here we are discussing the Deep Learning usage in the future with TTS. NLP, and speech applications in many areas (including Finance, Healthcare, and Government) there is a growing need for one comprehensive resource. Deep Learning for NLP and Speech Recognition explains recent deep learning methods applicable to NLP and speech, provides state-of-the-art approaches, and offers real-world case studies. NLP has various tools that are used in speech recognition process such as speech tagging, sentiment analysis, semantics and general understanding of the speech. NLP enables the classification and location of entities into various categories, while processing the language, using the named-entity technology Even IT industries have transformed the way they perform speech recognition process with the use of natural language processing. NLP has allowed the industries to process more accurate and automated understanding of speech and text. Natural Language Processing plays a critical role in supporting machine-human interactions. We propose to use Generative Adversarial Network (GAN) along with the idea of ”Professor Forcing” in training. A discriminator in GAN is jointly trained to equalize the difference between real and the predicted data. The idea with GAN is to have a Generator and Discriminator that are adversaries in a learning game. The concept applies to any type of problem for which a generator and a discriminator can be constructed. In NLP there are many types of Generators that could be considered, depending on the problem to be solved. As more research is being carried in this field, we expect to see more breakthroughs that will make machines smarter at recognizing and understanding the human language. [1] [2]

***Index Terms* - NLP, TTS, GAN, Deep Learning**

I. Introduction

Natural Language Processing,, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language. The objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable. Most NLP techniques rely on machine learning to derive meaning from human languages.

A typical interaction between humans and machines using Natural Language Processing could go as follows:
1. A human talks to the machine
2. The machine captures the audio
3. Audio to text conversion takes place
4. Processing of the text’s data
5. Data to audio conversion takes place
6. The machine responds to the human by playing the audio file

NLP usage:

Natural Language Processing is the driving force behind the following common applications:
Language translation applications such as Google Translate Word Processors such as Microsoft Word and Grammarly that employ NLP to check grammatical accuracy of texts.

- Interactive Voice Response (IVR) applications used in call centers to respond to certain users’ requests.
- Personal assistant applications such as OK Google, Siri, Cortana, and Alexa.

NLP looks difficult of:

Natural Language processing is considered a difficult problem in computer science. It's the nature of the human language that makes NLP difficult.

The rules that dictate the passing of information using natural languages are not easy for computers to understand.

Some of these rules can be high-leveled and abstract; for example, when someone uses a sarcastic remark to pass information. On the other hand, some of these rules can be low-levelled; for example, using the character "s" to signify the plurality of items.Comprehensively understanding the human language requires understanding both the words and how the concepts are connected to deliver the intended message. While humans can easily master a language, the ambiguity and imprecise characteristics of the natural languages are what make NLP difficult for machines to implement.

NLP Working:

NLP entails applying algorithms to identify and extract the natural language rules such that the unstructured language data is converted into a form that computers can understand. When the text has been provided, the computer will utilize algorithms to extract meaning associated with every sentence and collect the essential data from them. Sometimes, the computer may fail to understand the meaning of a sentence well, leading to obscure results.

For example, a humorous incident occurred in the 1950s during the translation of some words between the English and the Russian languages.

Here is the biblical sentence that required translation:
"The spirit is willing, but the flesh is weak."

Here is the result when the sentence was translated to Russian and back to English:
"The vodka is good, but the meat is rotten."

Techniques used in NLP:

Syntactic analysis and semantic analysis are the main techniques used to complete Natural Language Processing tasks.

Here is a description on how they can be used.

1. Syntax

Syntax refers to the arrangement of words in a sentence such that they make grammatical sense.

In NLP, syntactic analysis is used to assess how the natural language aligns with the grammatical rules.

Computer algorithms are used to apply grammatical rules to a group of words and derive meaning from them.

Here are some syntax techniques that can be used:

- **Lemmatization**: It entails reducing the various inflected forms of a word into a single form for easy analysis.
- **Morphological segmentation**: It involves dividing words into individual units called morphemes.
- **Word segmentation**: It involves dividing a large piece of continuous text into distinct units.
- **Part-of-speech tagging**: It involves identifying the part of speech for every word.
- **Parsing**: It involves undertaking grammatical analysis for the provided sentence.
- **Sentence breaking**: It involves placing sentence boundaries on a large piece of text.
- **Stemming**: It involves cutting the inflected words to their root form.

2. **Semantics**

Semantics refers to the meaning that is conveyed by a text. Semantic analysis is one of the difficult aspects of Natural Language Processing that has not been fully resolved yet.

It involves applying computer algorithms to understand the meaning and interpretation of words and how sentences are structured.

Here are some techniques in semantic analysis:

- **Named entity recognition (NER):** It involves determining the parts of a text that can be identified and categorized into preset groups. Examples of such groups include names of people and names of places.
- **Word sense disambiguation:** It involves giving meaning to a word based on the context.
- **Natural language generation**: It involves using databases to derive semantic intentions and convert them into human language.[3]

NLP in Real Life

- Information Retrieval(Google finds relevant and similar results).
- Information Extraction(Gmail structures events from emails).
- Machine Translation(Google Translate translates language from one language to another).
- Text Simplification(Rewordify simplifies the meaning of sentences). Shashi Tharoor tweets could be used(pun intended).
- Sentiment Analysis(Hater News gives us the sentiment of the user).
- Text Summarization(Smmry or Reddit's autotldr gives a summary of sentences).
- Spam Filter(Gmail filters spam emails separately).
- Auto-Predict(Google Search predicts user search results).
- Auto-Correct(Google Keyboard and Grammarly correct words otherwise spelled wrong).
- Speech Recognition(Google WebSpeech or Vocalware).

- Question Answering(IBM Watson’s answers to a query).
- Natural Language Generation(Generation of text from image or video data.) [4]

GAN

In domain adaptation, it has been successfully applied to help learning a domain-invariant representation to improve predictions and model generalization in the target domain. Professor Forcing describes a GAN-based adversarial training for generative autoregressive model, which can make predictions with features that cannot be discriminated between real and model distributions. Inspired by it, we propose a new GAN-based, end-to-end TTS training algorithm to introduce generated sequence in training to avoid exposure bias in the autoregressive decoder.

We are proposing Professor Forcing first, then present our proposed training framework and its training algorithm. Finally, we compare the performance of the training algorithms in different aspects with two subjective evaluation methods. The experimental results show that GAN-based training algorithm can significantly improve the model, including the naturalness of the generated speech and model generalization. We compared it with scheduled sampling to show it is more effective and proper for end-to-end TTS training. There have been some studies on GAN in TTS in the past two years, such as GAN-based post filter, and GAN-based multi-task for TTS. These algorithms focus on the generated output sequence of acoustic model, trying to make the outputs be closer to the real data.

One kind of generator used in NLP are description systems. These are NLP systems that describe a scene, an image, a problem, etc. An example might be a scene description system: given a scene, generate a description. The discriminator has to determine if a given description is reasonable. The starting point for the networks can be anything you want - you simply need a learning function that back propagates error if the generator or discriminator are incorrect. You might be able to come up with a heuristic that can figure out how wrong a given description is. This could accelerate that training.

The most important aspect of GANs is that they provide a way to generate sample data. Whereas classical training requires pre-determined training data, self-supervised learning techniques (like GAN) generate their own training data.[5]

Automatic Speech Recognition:

Automatic speech recognition, translating of spoken words into text, is still a challenging task due to the high viability in speech signals. For example, speakers may have different accents, dialects, or pronunciations, and speak in different styles, at different rates, and in different emotional states. The presence of environmental noise, reverberation, different microphones and recording devices results in additional variability.

Conventional speech recognition systems utilize Gaussian mixture model (GMM) based hidden Markov models to represent the sequential structure of speech signals. HMMs are used in speech recognition because a speech signal can be viewed as a piecewise stationary signal or a short-time stationary signal. In a short time-scale, speech can be approximated as a stationary process. Speech can be thought of as a Markov model for many stochastic purposes.Typically, each HMM state utilizes a mixture of Gaussian to model a spectral representation of the sound wave. HMMs-based speech recognition systems can be trained automatically and are simple and computationally feasible to use. However, one of the main drawbacks of Gaussian mixture models is that they are statistically inefficient for modeling data that lie on or near a non-linear manifold in the data space.

Neural networks trained by back-propagation error derivatives emerged as an attractive acoustic modeling approach for speech recognition in the late 1980s. In contrast to HMMs, neural networks make no assumptions about feature statistical properties. When used to estimate the probabilities of a speech feature segment, neural networks allow discriminative training in a natural and efficient manner. However, in spite of their effectiveness in classifying short-time units such as individual phones and isolated words, neural networks are rarely successful for continuous recognition tasks, largely because of their lack of ability to model temporal dependencies. Thus, one alternative approach is to use neural networks as a pre-processing e.g. feature transformation, dimensionality reduction for the HMM based recognition.

Deep learning , sometimes referred as representation learning or unsupervised feature learning, is a new area of machine learning. Deep learning is becoming a mainstream technology for speech recognition and has successfully replaced Gaussian mixtures for speech recognition and feature coding at an increasingly larger scale. In the course project, we focus on deep belief networks (DBNs) for speech recognition. The main goal of this course project can be summarized as:

1) Familiar with end-to-end speech recognition process.
2) Review state-of-the-art speech recognition techniques.
3) Learn and understand deep learning algorithms, including deep neural networks (DNN), deep belief networks (DBN), and deep auto-encoders (DAE).
4) Applying deep learning algorithms to speech recognition and compare the speech recognition performance with conventional GMM-HMM based speech recognition method.[7]

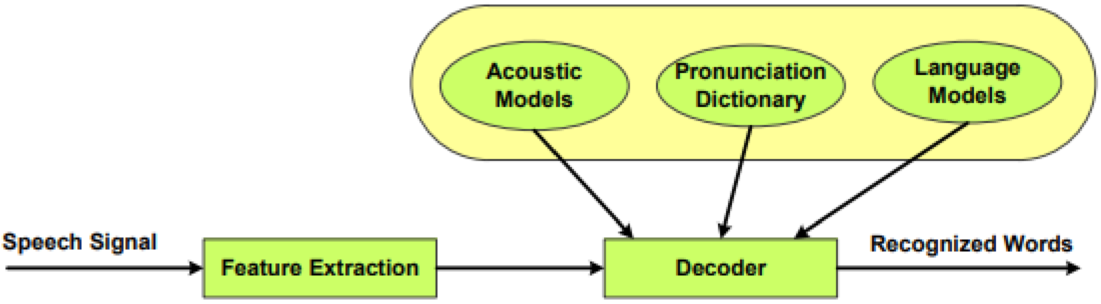


Fig. 1 A typical system architecture for automatic speech recognition

Text-to-speech synthesis -TTS - is the automatic conversion of a text into speech that resembles, as closely as possible, a native speaker of the language reading that text. Text-to-speech synthesizer (TTS) is the technology which lets computer speak to you. The TTS system gets the text as the input and then a computer algorithm which called TTS engine analyses the text, pre-processes the text and synthesizes the speech with some mathematical models. The TTS engine usually generates sound data in an audio format as the output.

The text-to-speech (TTS) synthesis procedure consists of two main phases. The first is text analysis, where the input text is transcribed into a phonetic or some other linguistic representation, and the second one is the generation of speech waveforms, where the output is produced from this phonetic and prosodic information. These two phases are usually called high and low-level synthesis . A simplified version of this procedure is presented in figure 1 below. The input text might be for example data from a word processor, standard ASCII from e-mail, a mobile text-message, or scanned text from a newspaper. The character string is then pre-processed and analyzed into phonetic representation which is usually a string of phonemes with some additional information for correct intonation, duration, and stress. Speech sound is finally generated with the low-level synthesizer by the information from high-level one. The artificial production of speech-like sounds has a long history, with documented mechanical attempts dating to the eighteenth century.[6]

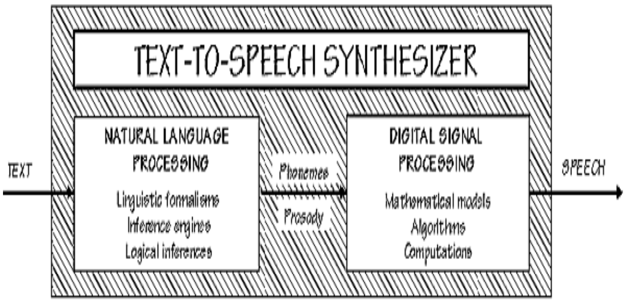


Figure 2: TTS Functioning

II. Literature Survey:

Speech Recognition is one of the challenging areas in computer science, a lot of pattern recognition methodology tried to resolve a good way and higher percentage of recognition. The challenges of deep learning are – making intelligent machines capable of not only hearing (speech) and seeing (vision), but also of thinking with a mind, i.e. reasoning and interface over complex, hierarchical relationships and knowledge sources that comprise a vast number of entities and semantics concept in the real world based in part on multi-sensory data from user. To this end, language and multimodal processing – joint exploitation and learning from text, speech (audio), and image (video) – is evolving into a new frontier of deep learning, beginning by mixture of research communities speech and spoken language processing, natural language processing, computer vision, machine learning. Main challenges in speech recognition in deep learning methods are –

A. The multi-model learning – For advancing the ability of representation learning systems to discover semantic spaces that underlie multiple kinds of sensory input.

B. The black-box learning – For reducing the usefulness of having a human researcher working in the loop with the training algorithm.

TTS

Speech synthesis can be described as artificial production of human speech. A computer system used for this purpose is called a speech synthesizer, and can be implemented in software or hardware. A text-to-speech (TTS) system converts normal language text into speech . Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or diphones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output . The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood. An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written works on a home computer.

A text-to-speech system (or "engine") is composed of two parts: a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called text normalization, pre-processing, or tokenization. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences. The process of assigning phonetic transcriptions to words is called text-to-phoneme or grapheme-to-phoneme conversion. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-end—often referred to as the synthesizer—then converts the symbolic linguistic representation into sound. In certain systems, this part includes the computation of the target prosody (pitch contour, phoneme durations), which is then imposed on the output speech.[8]

III. Exsisting System

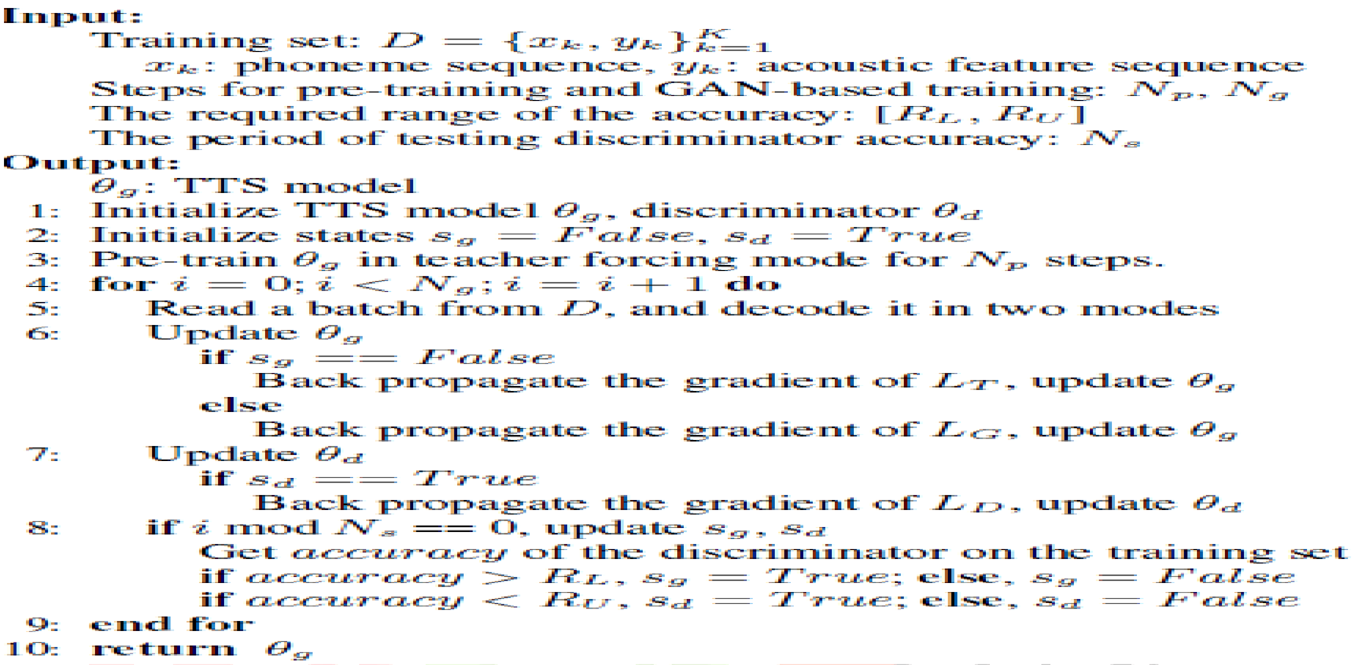
When comparing the functionality of different methods for converting speech to text, it is saying that neural networks are replacing the Gaussian Mixture Model as the main method. They are more effective at giving the correct output and are able to compute more difficult sentences by comparing the words and using its memory to understand the input better. These methods are still needed to be updated due to fast recognition, where the Neural networks provide a promising future for speech recognition but not still.

IV Proposed System

In this Paper we are suggesting to use GAN-based end-to-end TTS training algorithm. Two TTS test sets to compare these algorithms in two aspects, speech quality and model generalization (stability). We use the common test set, which contains 50 typical sentences used in news and general conversation, to compare the performance of these models in speech quality

We use TTS model, include Wave Net as vocoder for all experiments. We use one-hot feature as input, which contains phonemes, punctuation and the blank between two adjacent words. The model output is an 80-channel Mel spectrum, one frame at a time. When we calculate LT in GAN-based algorithm, teacher forcing can also be replaced with scheduled sampling to generate sequence.

Four TTS models with 4 different
training algorithms: teacher forcing (TF), scheduled sampling (SS), GAN-based algorithm with teacher forcing (TF-GAN) and GAN-based algorithm with scheduled sampling (SS-GAN, replace TF in TF-GAN with SS). These experiments are performed based on an American English speech data set, which has 14 hours of speech, recorded by a single speaker.
A new GAN-based, end-to-end TTS training algorithm, which introduces the generated sequence to GAN training to avoid exposure bias in autoregressive decoder. Experimental results show that schedule sampling is harmful to synthesized speech quality, but can improve the model generalization capability of TTS model. [9]



```
Input:
    Training set: $D = \{x_k, y_k\}_{k=1}^{K}$
        $x_k$: phoneme sequence, $y_k$: acoustic feature sequence
    Steps for pre-training and GAN-based training: $N_p$, $N_g$
    The required range of the accuracy: $[R_L, R_U]$
    The period of testing discriminator accuracy: $N_s$
Output:
    $\theta_g$: TTS model
    Initialize TTS model $\theta_g$, discriminator $\theta_d$
    Initialize states $s_g = False$, $s_d = True$
    Pre-train $\theta_g$ in teacher forcing mode for $N_p$ steps.
    for $i = 0; i < N_g; i = i + 1$ do
        Read a batch from $D$, and decode it in two modes
        Update $\theta_g$
            if $s_g == False$
                Back propagate the gradient of $L_T$, update $\theta_g$
            else
                Back propagate the gradient of $L_G$, update $\theta_g$
        Update $\theta_d$
            if $s_d == True$
                Back propagate the gradient of $L_D$, update $\theta_d$
        if $i$ mod $N_s == 0$, update $s_g, s_d$
            Get $accuracy$ of the discriminator on the training set
            if $accuracy > R_L$, $s_g = True$; else, $s_g = False$
            if $accuracy < R_U$, $s_d = True$; else, $s_d = False$
    end for
    return $\theta_g$
```

Figure 3: GAN-based end-to-end TTS training algorithm

V. Conclusion

In this paper my idea is to use GAN algorithm with end-to-end TTS and improve the speech recognition to mingle to present situations every individual is moving towards technology. Proposed algorithm improves both output quality and generalization of the model. We can further improve the generalization of the model by maintaining the speech quality and natural ness at the same level with slight advantages. [9]

VI. References

[1] https://www.venture7.com/how-natural-language-processing-is-helping-in-speech-recognition-process/#:~:text=NLP%20has%20offered%20various%20tools,process%20human%20structure%20sentences%20effectively.

[2] https://www.quora.com/profile/Bruce-Matichuk

[3] https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32

[4] https://towardsdatascience.com/natural-language-processing-nlp-for-machinelearningd44498845d5b

[5]https://www.semanticscholar.org/paper/A-New-GAN-based-End-to-End-TTS-Training-Algorithm-Guo-Soong/fa081e91d94cefea2a6908b67bf550b3a8ba0f75

[6] https://pdfs.semanticscholar.org/1710/a33302b0587b4304fa49e8aeb2809a705fa3.pdf

[7] https://pdfs.semanticscholar.org/fbf6/2fad033af2c083bc3152066fd2cc4544da66.pdf

[8]. https://www.ijcsmc.com/docs/papers/May2015/V4I5201599a61.pdf

[9] http://www.ijmlc.org/vol8/739-DA0052.pdf

[10] Design and Implementation of Text To Speech Conversion for Visually Impaired People International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 7– No. 2, April 2014 – www.ijais.org

[11].Speech Recognition Using Deep Learning Algorithms Yan Zhang, SUNet ID: yzhang5 Instructor: Andrew Ng