Research Article GU J Sci 34(4): 1035-1049 (2021) DOI: 10.35378/gujs.816499

Gazi University

Journal of Science

http://dergipark.gov.tr/gujs

Turkish Speech Recognition Techniques and Applications of Recurrent Units (LSTM and GRU)

Burak TOMBALOGLU* , Hamit ERDEM

Başkent University, Electrical and Electronics Engineering, Ankara, Turkey

Highlights

- This paper focuses on Automatic Speech Recognition techniques for Turkish Language.
- Long Short Term Memory is applied for increasing the recognition performance.
- Using Gated Recurrent Units decreased the computation time.

Article Info

Received: 27 Oct 2020
Accepted: 21 Jan 2021

Keywords

Turkish
Speech recognition
Lstm
Gru

Abstract

A typical solution of Automatic Speech Recognition (ASR) problems is realized by feature extraction, feature classification, acoustic modeling and language modeling steps. In classification and modeling steps, Deep Learning Methods have become popular and give more successful recognition results than conventional methods. In this study, an application for solving ASR problem in Turkish Language has been developed. The data sets and studies related to Turkish Language ASR problem are examined. Language models in the ASR problems of agglutative language groups such as Turkish, Finnish and Hungarian are examined. Subword based model is chosen in order not to decrease recognition performance and prevent large vocabulary. The recogniton performance is increased by Deep Learning Methods called Long-Short Term Memory (LSTM) Neural Networks and Gated Recurrent Unit (GRU) in the classification and acoustic modeling steps. The recognition performances of systems including LSTM and GRU are compared with the the previous studies using traditional methods and Deep Neural Networks. When the results were evaluated, it is seen that LSTM and GRU based Speech Recognizers performs better than the recognizers with previous methods. Final Word Error Rate (WER) values were obtained for LSTM and GRU as 10,65% and 11,25%, respectively. GRU based systems have similar performance when compared to LSTM based systems. However, it has been observed that the training periods are short. Computation times are 73.518 and 61.020 seconds respectively. The study gave detailed information about the applicability of the latest methods to Turkish ASR research and applications.

1. INTRODUCTION

Speech is the most common communication method**.** The advancement of technology has allowed machines to process human speech and increased the use of communication between human and machine. As a result, the use of technology by everyone will become widespread and it will be easier for people with disabilities to meet their needs.

An Automatic Speech Recognition (ASR) system basically translates speech into text. The system extracts features of speech and clasifies the phonemes and word components. Some of the application areas are call centers, security, gaming, support for people with disabilities, in cars, devices control, robotic, dictation, mobile communication applications and home automation.

Current speech recognition systems include feature extraction, acoustic model, Language Modeling (LM), vocabulary dictionary and classification sections. In order to recognize the words of sentences, the sound components which form the words must be modeled acoustically. Acoustic analysis is performed by Gaussian Mixture Models (GMM) and posterior probabilities are generated. Acoustic Models are created

*Corresponding author, e-mail: buraktombaloglu@hotmail.com using Hidden Markov Models (HMM) and processed by Deep Learning methods with the development of computers and advanced microprocessors in recent years. By this way, words or sentences are able to be predicted.

The traditional recognition method used in ASR applications is the use of HMM and GMM. With the development of computer technology and using GPU (Graphics Processing Unit) for computing in recent years, Deep Learning has replaced GMM in ASR applications and provided significant performance increases. Classifiers within this scope can be grouped as GMM-HMM, Deep Neural Networks (DNN)-HMM. DNN and GMM provide status information for HMM, representing each phonemic track. DNNs provide more status information to HMM, which better represents the differences between phonemes. Replacing GMM with DNN has been proposed by many researchers to estimate the probabilities of HMM states [1-6].

Various voice assistants such as, "Apple-Siri" and “Google Voice Transcription” are used on smart communication devices. These applications use a Deep Neural Network (DNN) to convert the acoustic pattern of your voice at each instant into a probability distribution over speech sounds. The ASR implementation of these applications are in the cloud. The cloud servers can provide large storage facilities and updates to the acoustic models used by the ASR [7-9].

DNNs can model only fixed size sliding windows of acoustic frames but cannot model different speech rates. Recurrent Neural Networks (RNNs) is another class of network that contains loops in the hidden layers. The information at the previous time step is retained and the value at the current step is predicted by the help of these loops. By this way RNNs can handle different speaking rates [1]. Temporary dependencies pose a problem during the solution of the speech recognition problem. Temporal dependencies can occur in the long or short term, depending on ASR. RNNs only account for short-term dependencies, depending on the vanishing / exploding gradient problem. RNNs have been applied to speech recognition problems in recent years. Since RNNs can handle the dynamic process in speech better, it is a good choice compared to the traditional feed forward network [10]. Compared to DNNs, RNNs have additional recurrent connections and memory cells that allow them to recall data. Based on the words and rankings previously determined, the next word is predicted.

Two kinds of RNN networks which are called LSTM and GRU are used. RNN networks are widely used in many areas, including speech recognition, natural language processing, image recognition. RNNs have become the state-of-the art character recognition methods. The gates in RNNs such as forget gate (for LSTM) or update gate (for GRU) can keep longer contextual dependencies. Back propagation of error information and a highway channel to provide shortcuts for the smooth propagation of history information are built by the gates. The architecture of the GRU is less complex than that of the LSTM. The forget gate and the input gate are combined into a single update gate. There is not a separate “cell” to store intermediate information in the GRU which has fewer parameters. With the widespread use of GRU in many sequential learning tasks, memory or computation time savings have been achieved [1-3, 11-14]. Studies about Turkish language also show that RNNs and LSTMs are advantageous in speech recognition compared to DNN sensitive to changes in input characteristics over time and give more successful results [15-17].

Turkish is a phonemic-based language, and it is also included in the agglutinative language groups such as Finnish and Hungarian. There are suffixes which are added to word endings. Since the suffixes added to the end of the word derive new words, the vocabulary size increases considerably. In ASR applications, the common solution for agglutinative languages is to use subword-based modeling [18]. The proposed system recognizes the words in the vocabulary and separates the words that are not in the vocabulary into phonemes. Sub-word (morpheme) based LM is applied in the system. Each phoneme unit of Turkish Language is modeled as a sub-word in the model. Sub-word (morpheme) based LM is widely used to avoid excessive vocabulary size in agglutinative languages [4].

In this paper, the performances of the proposed LSTM and GRU-based ASR systems were compared with the traditional recognition method, GMM-based and DNN-based systems. The ASR systems are applied to “Turkish Microphone Speech Corpus (METU 1.0)” database [19]. In the proposed system, Subword-based

LM was used for training and Kaldi was used for GMM-HMM and Deep learning based ASR coding [20]. Regarding performance measurements, the recognition rates of the applied methods have been compared with previous studies [4,19,21] using the same dataset.

The continuation of the article is organized as follows. In Chapter 2, information about ASR and its architecture is given. Chapter 3 Information about Deep Learning Methods used in ASR and their choices has been conveyed. In Chapter 4, the structure and modeling of the Turkish language is examined. The performance of the LSTM and GRU based ASR system proposed for Turkish is compared with the traditional recognition method, GMM based HMM in Chapter 5. In Chapter 6, the results are evaluated. Regarding performance measurements, the recognition rate of the Turkish language has been improved over previous studies.

2. ASR PROBLEM AND CONVENTIONAL SOLUTION METHODS

Generally, the purpose of the ASR system is to take the speech data and to translate what is voiced into sentences or words. Prediction functionality is accomplished by feature extraction and language analysis steps. The feature extractor obtains acoustic feature sequences from the speech signal. The probabilities of phonemes in the features are calculated and passed to the classifier that predicts the phonemes. Two basic modeling is used when estimating. These are Acoustic Modeling and LM. Phonemes probabilities are used to recognize speech together with LM and HMM which represent the acoustic model. Words and sentences are predicted with the help of LM.

ASR is a classification and pattern recognition problem. In this field, as well as classical methods, machine learning based applications are also used. To apply classical methods in ASR, the following steps are applied regardless of language:

1. Feature Extraction
2. Feature Classification
3. Acoustic Modeling
4. Language Modeling (LM).

2.1. Feature Extraction

Obtaining Mel Frequency Cepstrum Coefficients (MFCC) used for features is based on hearing perception in humans. The human ear perceives sounds with a frequency lower than 1 kHz linearly and those with higher values as logarithmic. Mel filter bank consists of two types of filters with linear range up to 1000 Hz and logarithmic range at frequencies higher than 1000 Hz. Using this filtering, the desired phonetic features in the speech signal are obtained [22].

2.2. Feature Classification

After the feature extraction step, Gaussian Mix functions consisting of Gauss probability functions are applied to the obtained coefficients. GMM is used to model the distribution of states to be used in an HMM Based Recognizer. This structure of GMM- HMM Classifier is shown in Figure 1.

The phonemes in the speech record are predicted using the GMM-HMM model. Then the spoken word or continuous words are determined [19].

The spectral shape can be represented by an M component mixture model with parameters θ_m and $P(\omega_m)$ component weights in the GMM. The mixture model can be expressed as in Equation (1)

$$p(x|\theta) = \sum_{m=1}^{M} P(\omega_m) p(x|\omega_m, \theta_m), \tag{1}$$

$p(x|\omega_m, \theta_m)$ is the prior probability of componet ω_m and θ_m. Here the mixture components are Gauss and are expressed as in Equation (2)

$$p(x|\omega_m, \theta_m) = \frac{1}{\sqrt{2\pi\sigma_m^2}} exp\left[-\frac{(x-\mu_m)^2}{2\sigma_m^2}\right], \qquad (2)$$

σ_m is the standard deviation for component ω_m and μ_m the mean. Creating a continuous PDF based on spectral representation is the first step in calculating a GMM from the speech. Optimum GMM parameters are obtained by minimizing the distance between GMM and spectral PDF [23].

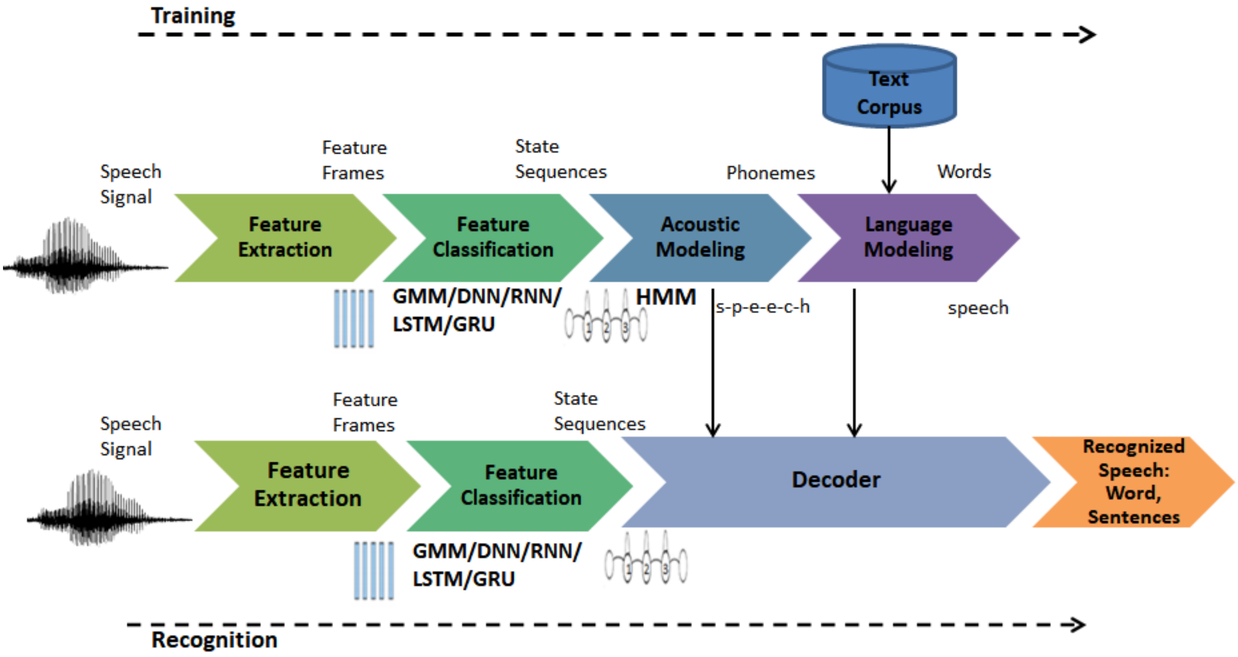

Figure 1. *ASR System*

2.3. Acoustic Modeling

In classical ASR systems, acoustic modeling is done by HMM. State transition probabilities of the states obtained after the GMM stage are estimated. The word is predicted by comparing the acoustic models built here.

In order to recognize the words that make up the sentences, the sound components or phonemes that form the word must be modeled acoustically. In acoustic modeling, the probability of each phoneme is calculated by using GMM during pronunciation or speech. Thus, the variation of the speech signal over time and its spectral diversity are modeled.

The smallest acoustic unit in voice recognition is the phoneme. Voice-based modeling is built with 3 HMM states. Single state (monophone) modeling provides a voice recognition system independent of other phonemes. In tri-state modeling (triphone), each phoneme is modeled with neighboring units (right and left). By using this model, the negative effects of acoustic differences and irregularities in the sound segments on the classification are reduced and recognition success is increased. HMMs assume that the speech signal is stable at small time intervals. HMMs can manage speech signal variability well and are good at speech modeling [24].

2.4. Language Model (LM)

Based on examples of the text, LM learns the probability of occurrence of the word and develops probabilistic models that can predict the next word in the order of the given words. It contains a large number of words and possibilities for its occurrence. Larger models can predict sentences or paragraphs. The prediction takes place using the Bayes equation. The relevant Bayes equation is given in Equation (3)

$$P(W/A) = \frac{P(A/W)P(W)}{P(A)}, \tag{3}$$

The hypothesis sentence is represented by *W*. *W* is the sentence with the highest probability of acoustical agreement with possible sentences. *P(W|A)* represents the probability of the sentence *(W)* in the database to be compatible with the acoustic sequence according to the acoustic data *(A)* obtained. Bayes rule is applied in the second part of the equality. The probability of occurrence of the *W* sentence is *P(W)* and it is calculated according to LM. The probability of occurence of the acoustic sequence in the sentence is *P(A|W).* Since *P(A)* is independent of *W,* it is abbreviated [25].

3. DEEP LEARNING BASED ASR

In recent years, the development of computer technology and the shortened training period have enabled the development of systems based on Deep Learning. Deep Learning algorithms as an advanced multi layer ANN have improved performance in solving many problems, including ASR. The Deep Learning approach outperformed GMM's performance for ASR. DNNs have been widely applied in acoustic model training and outperformed statistical methods.

The Deep Learning Approach has successfully replaced the Gaussian Mixture step in speech recognition applications [26]. DNN and GMM provide status information for HMM, representing each sound segment. DNN provides more status information to HMM, which better represents the differences between phonemes. Thus, the success for larger voice data and vocabulary recognition is higher. The structure of the DNN-HMM system is shown in Figure 1.

Deep Learning is an extension of ANN. Multiple hidden layers are used to learn about patterns and features. Many complex signal patterns such as video, image and speech are learned successfully through many layers. These layers have nodes with nonlinear processing functions. Recurent Neural Networks (RNN), one of the deep learning methods, performs both feature classification step in speech recognition applications (Figure 1).

3.1. Recurrent Neural Network (RNN)

Due to its recursive structure, RNN is suitable for the solution of time-varying problems such as speech recognition. RNNs expand the traditional feed forward concept by having a secret recurrent state. Activation of the hidden recurrent state depends on the previous one. Thus, unlike conventional neural networks, timing information in processes, which is an important consideration for speech recognition, is recorded [27].

RNNs are designed to work with array prediction problems. It is used for classification of text, speech data and regression prediction problems. RNN and its expansion are shown in Figure 2. When we think of a RNN representing a sentence consisting of an N-word, this network can also be referred to as a N-layered Neural Network with a layer for each word.

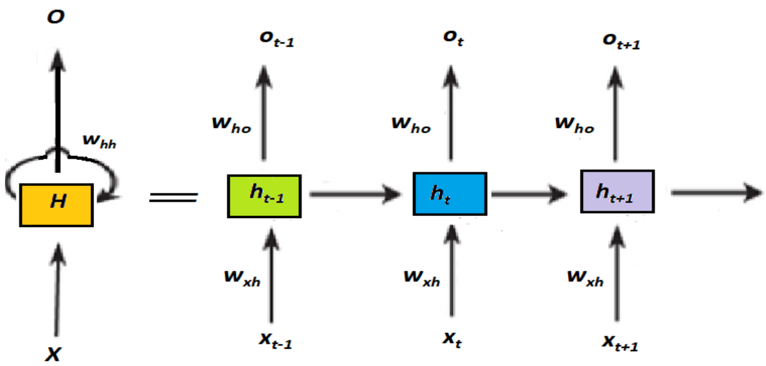

***Figure 2.** Unfolding of Recurrent Neural Network*

However, training of RNNs can be complicated by the exploding and the vanishing gradients, and that can hinder learning long-term dependencies. The basic idea of RNNs is the introduction of a gate mechanism to better control the flow of information at various time steps. By the help of the gated RNNs, vanishing gradient problems are alleviated by building effective "shortcuts", where multiple temporal steps can be skipped by gradients. LSTMs are the most popular in the gated RNNs group. In machine learning tasks, especially speech recognition, state-of-the-art performance is often achieved by the LSTMs. In LSTM architecture input, output and forget gates control the memory cells. Despite their effectiveness, such a complex gate mechanism can result in an overly complex model. Another issue is that computational efficiency is a very important for RNNs and alternative architectures have been tried to be developed [14].

3.2. Long Short-Term Memory (LSTM) Neural Networks

One of the most used RNN types is the LSTM Neural Network. It is designed to solve the problem of long-term dependence [28]. It is used to better model dependencies over time and to solve the Vanishing Gradient Problem.

In the recurrent cell, there is not only one neural network gate, but three interacting gates, as shown in Figure 3. Input, forget and output ports define the behavior of LSTM. Information can be stored or read in the cell, depending on whether the doors are open or closed. The previous cell values are multiplied by the forget gate. Thus, the reset function is realized. Doors pass or block the information on them according to the strength and weight of the signal they receive. Weights in LSTM are adjusted according to recursive neural network learning [29].

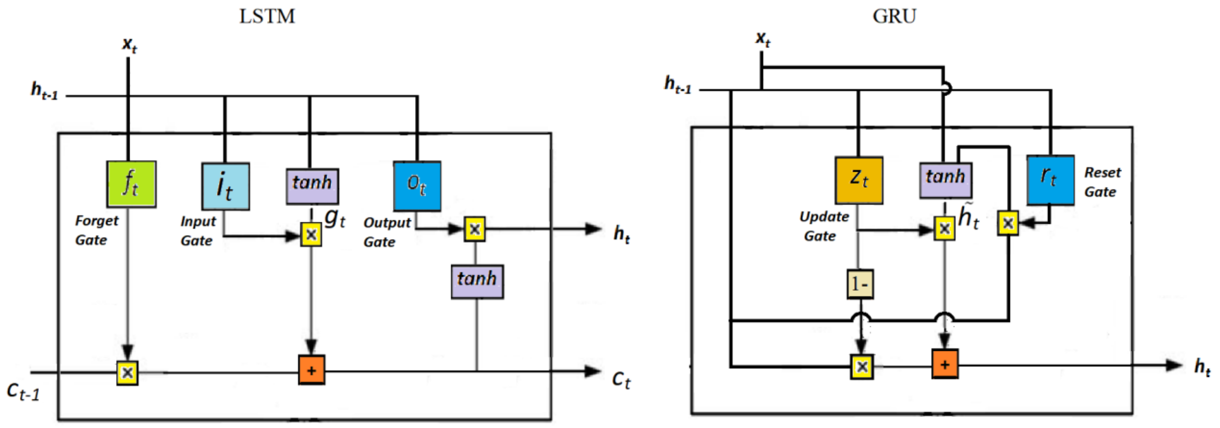

***Figure 3.** LSTM and GRU Cell Structures*

The vector calculation of the LSTM layer is given below [15]:

$$i_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + b_i) \tag{4}$$

$$f_t = \sigma(w_{xf}x_t + w_{hf}h_{t-1} + b_f) \tag{5}$$

$$g_t = tanh(w_{xg}x_t + w_{hg}h_{t-1} + b_g) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{7}$$

$$o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o) \tag{8}$$

$$h_t = o_t \odot \tanh(c_t). \tag{9}$$

RNNs in general and LSTMs in particular have been successful in working with word and paragraph sequences, often called natural language processing.

3.3. Gated Recurrent Units (GRU):

Recently, the design of a new model called the Gated Repeating Unit (GRU) based on only two multiplicative gates has been a notable attempt to simplify LSTMs. It combines the forget gate and the input gate into a single update gate. The GRU does not have a separate “cell” to store intermediate information and has a reset gate and an update gate, which control the memory flow as shown in Figure 3. Therefore, GRU has slightly fewer parameters than LSTMs. Due to its simplicity, GRUs conserve memory or computation time and have been widely used in many sequence learning tasks [2,3]. In particular, the standard GRU architecture is defined by the following equations, where r_t and z_t are the vectors of the reset and update gates, respectively, while h_t represents the state vector for the current time frame *t*

$$z_t = \sigma(w_z x_t + u_z h_{t-1} + b_z) \tag{10}$$

$$r_t = \sigma(w_r x_t + u_r h_{t-1} + b_r) \tag{11}$$

$$\tilde{h}_t = tanh(w_h x_t + u_h(h_{t-1} \odot r_t) + b_h) \tag{12}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t)\odot\tilde{h}_t. \tag{13}$$

The “ʘ” function denotes Element-wise multiplications. The activation functions of both gates are logistic sigmoid σ. By this way, z_t and r_t are constrained to take values ranging from 0 and 1. The candidate state $\widetilde{h_t}$ is processed with a hyperbolic tangent. The current input vector x_t (e.g., speech features) feeds the network and the matrices, w_z, w_r,w_h (the feedforward connections) and u_z, u_z, u_h (the recurrent weights) represents the parameters of the model. Trainable bias vectors, b_z, b_r and b_h, are added before the non-linearities are applied [14].

4. LANGUAGE MODELING TYPES FOR TURKISH LANGUAGE

Most commonly used Language models in ASR problems are:

1. Word Based
2. Sub-Word Based.

Word Based Model models are used for analytical and isolated languages such as English, while Sub-Word Based Model is used to model agglutinative languages such as Turkish and Finnish.

Turkish is an agglutinative language. New words can be derived by adding many suffixes to the root of the word. Prefix is not used in Turkish. Another important feature of Turkish in terms of LM is free word order. Subject-Object-Verb word order is a typical feature in Turkish, but in some conditions, other kinds of orderings are possible. Sentences can be reconstructed with different word sequences without changing their meanings, but the complexity of n-gram LM is increasing [24].

As a result, sub-words should be used differently from words to solve the coverage problem in Turkish ASR and a large amount of training data should be used to reliably train LM parameters.

It is possible to add another subword followed by a subword in agglutinative languages. Each subword carries certain morphological information such as time, situation and agreement. This feature of agglutinative languages leads to wide vocabulary of many words with the same root but different suffixes. Since there are too many words in the dictionary, there will be a lot of non-modeled words out of the dictionary [30]. The larger dictionary size reduces the speed of the decoder in word-based speech recognition [31]. Due to the large number of non-vocabulary words, speech recognition methods used for English give low recognition success results for Turkish language.

4.1. Word Based Model

The word-based model is the most basic LM approach that uses words as recognition units. Words are selected as dictionary entries for speech recognition, and LM probabilities are removed from the training corpus using words as units [32]. The structure of the word-based system is shown in Figure 4. Word-based LM is preferred when modeling analytical and isolated languages with low sub-words per word. For example, English and Mandarin Chinese are in this group.

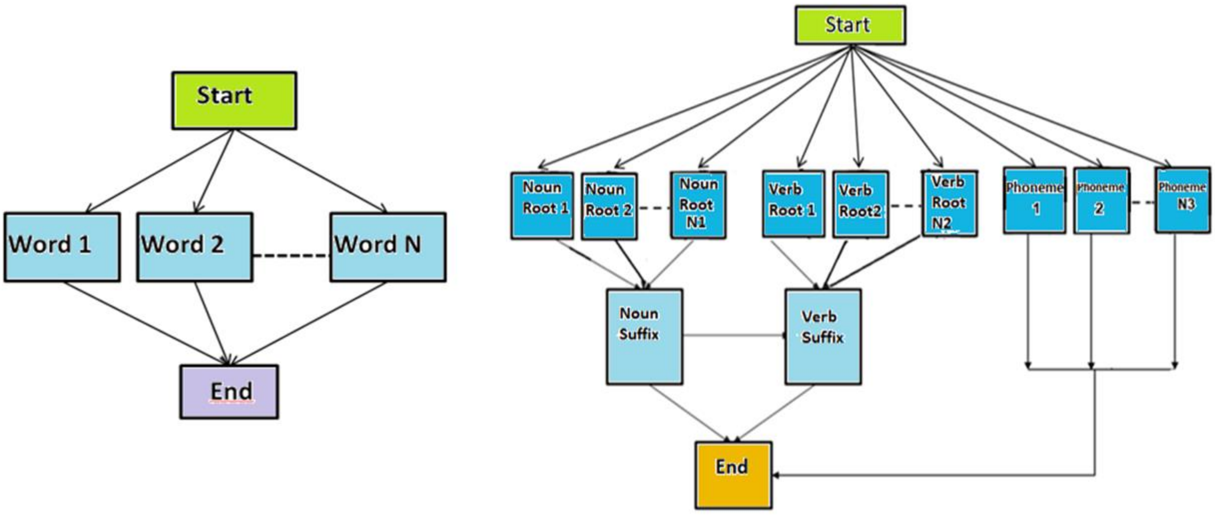

Figure 4. *Word and Sub-Word based model*

4.2. Sub-Word Based Model

Sub-Word (morpheme) based LM improves word recognition success in speech recognition systems of agglutinative languages [33-35].

Sub-word based LM uses sub-words as the basic units of speech recognition. Words, as shown in Figure 4, consist of the root and the attachments in accordance with the spelling rules and morphology of Turkish. Modeling is done by considering this structure. Transitions between sub-words are weighted with bigram probabilities [36].

Phonetic rules are used to establish connections between the root and suffixes. The last phoneme and the last vowel of the root determine the suffixes that can follow a particular root [37].

5. APPLICATIONS OF LSTM and GRU NEURAL NETWORKS FOR TURKISH ASR

In this study, Turkish language speech recognizer was developed. The proposed system recognizes vocabulary words and finds phonetic components of words that are out of the vocabulary. Subword based LM is applied to the system. Each phoneme of Turkish is modeled as a subword in the model. Sub-word-based LM is widely used for agglutinative languages to prevent excessive growth in vocabulary. The performance of the proposed LSTM-based ASR system is compared with the traditional recognition method, GMM-based HMM. Regarding performance measurements, the recognition rate of the Turkish language has been improved over the authors' previous study.

5.1. Audio-Text Corpus

“Turkish Microphone Speech Corpus (METU 1.0)” database is used to realize the system proposed in this study. METU 1.0-Turkish Microphone Speech Database contains speech records collected from 193 people at the Middle East Technical University. It consists of reading the texts with 40 different sentences by each speaker and recording 2482 sentences in total [19]. This is the method which is used to form TIMIT corpus. Each sentence is uttered once by the speakers. The corpus was accepted by Linguistic Data Consortium (LDC) in 2005 [19,38,39]. There have been studies [4,19,40,41]. using “Turkish Microphone Speech Corpus”. Voice recordings belonging to 120 people are used in the analyses. The speech data of 100 persons were used for the training and the data of 20 persons were used for the test.

5.2. Kaldi ASR Toolbox

The Kaldi ASR toolbox is an open speech recognition tools set up by Daniel Povey. Kaldi speech recognition toolkit performs training and decoding operations. Kaldi provides building acoustic models and LMs. Kaldi is written in C++ programming language and it is an open source speech recognition toolkit. The Kaldi toolkit includes several shell scripts and C++ executables. The codes are easy to understand, very modern and flexible. This is available on both Linux and Microsoft Windows operating systems [42].

Feature extraction, label/alignment computation, and decoding are performed with Kaldi. Acousting modeling is realised by GMM, DNN, RNN, LSTM and GRU toolkits. The posterior probabilities generated for each frame by the GMM and the Deep Learning toolkits are normalized by their prior probabilities. A HMM-based decoder processes the obtained likelihoods and finally estimates the sequence of words after integrating the acoustic, lexicon and language model information [13]. For Language Modelling SRILM Toolkit for Kaldi is used.

5.3. Experiments and Results

In the LSTM/GRU-HMM experiments, the LSTM and GRU are trained to predict context-dependent phone targets. Feature extraction involves partitioning the signal into 25 ms frames with a 10 ms overlap and estimating the feature coefficients. These frames are represented by 13 parameters consisting of 12 MFCC parameters and frame energy. The experimental activity is conducted considering different acoustic features, i.e., 39 MFCCs (13 static+derivatives+second derivatives).

Subword-based LM was used for training. Since Turkish is a phoneme based language, all phonemes in Turkish are also trained as a subword and entered into the dictionary. In this way, phoneme components of the words out of the vocabulary can also be recognized.

Kaldi has been used for GMM-HMM and Deep Learning based ASR coding. When the Word-Based and Sub-word-based LM is applied for Turkish in the GMM-HMM-based system, the Word Error Rate (WER) was examined and the results are given in Table 1.

Table 1. *WER Comparison of Word Based and Sub Word Based LMs*

System	LM	WER
GMM-HMM	Word Based	%18,67
GMM-HMM	Sub-Word Based	%17,21

When the performances of the two used LMs are analyzed, the contribution made by the Sub-Word based LM, which gives more successful results for the agglutinative languages including Turkish, is confirmed. This model was used in the later steps of the study.

Two RNN type models are trained (LSTM, GRU). The following parameters used for LSTM and GRU training are shown in Table 2.

Table 2. *LSTM and GRU Parameters*

Dropout Rate	20%
Number of Epochs	24
Training and Test Batch Size	8
Activation Function	tanh
Neuron Count in Hidden Layers	550
Optimum ε	1e-8
Learning Rate	0.0004

PER values of GMM-HMM, LSTM and GRU speech recognizers working on Kaldi are given in Figure 5. The GMM-HMM PER value we obtained is 30,64%. This value is consistent with the value of the study in [19] using the same corpus.

Feed forward ANNs such as DNN, DBN do not carry or process the information of previous states which are important for the ASR problem. In LSTM and GRU which are RNN types, information can be stored or read in the cell, depending on whether the doors are open or closed.

When we run the LSTM-based speech recognizer with the same corpus, we got a PER value of 14.41% and improved recognition performance. When the word recognition performance was analyzed, the Word Error Rate (WER) of GMM-HMM was measured as 17,21%. In the proposed LSTM based structure, WER performance has improved and the value decreased to 10,65%, which is lower than GMM-HMM, as seen in Figure 5. LSTM gave more successful results than modeling with GMM. Additionally, the same setup is performed with GRU. PER and WER results are obtained as 15,23% and 11,25% respectively.

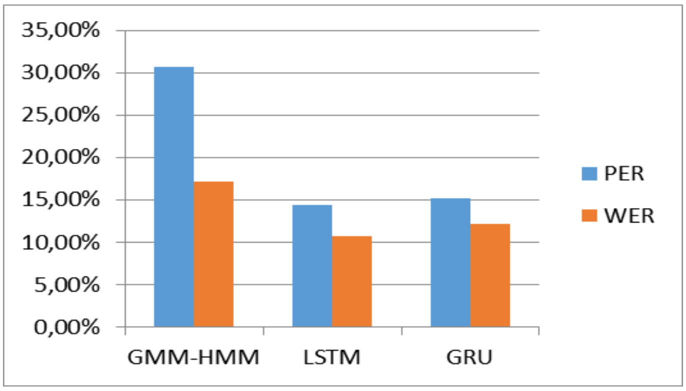

Figure 5. *PER and WER Values for applied methods*

The elapsed time of computation is 73.518 seconds for LSTM and 61.020 seconds for GRU. GRU beats LSTM with a shorter execution time. As obtained here, in GRU, 17% of time is saved and WER results are similar to LSTM (Table 3).

Table 3. *LSTM and GRU Computation Time Comparison*

	Computation Time (sec)	PER (%)	WER (%)
LSTM	73.518	14,41	10,65
GRU	61.020	15,23	11,25

Application and comparison results show that recognition performance improves considering the PER and WER criteria.

In the proposed study, the performance of the LSTM and GRU based recognizers using Turkish Microphone Speech Corpus (METU 1.0) are compared with the GMM-HMM and DBN (Deep Belief Network)-HMM (Deep Learning Method) recognizers using the same corpus in [4,19,21] are compared according to Phoneme Error Rate (PER) and WER (Tables 3-4).

Table 3. *Comparison according to Phoneme Error Rate (PER)*

Speech Recognizer	Corpus	PER
SONIC Speech Recognizer in [19]	METU 1.0	29,3%
Kaldi Speech Recognizer (GMM-HMM)	METU 1.0	30,64%
Kaldi Speech Recognizer (DBN-HMM) [4]	METU 1.0	24,8%
Kaldi Speech Recognizer (LSTM)	METU 1.0	14,41%
Kaldi Speech Recognizer (GRU)	METU 1.0	15,23%

Table 4. *Comparison according to Word Error Rate (WER)*

Speech Recognizer	Corpus	WER
HTK Speech Recognizer (GMM-HMM) in [21]	METU 1.0	21,46%
Kaldi Speech Recognizer (GMM-HMM)	METU 1.0	17,21%
Kaldi Speech Recognizer (DBN-HMM) in [4]	METU 1.0	13,04%
Kaldi Speech Recognizer (LSTM)	METU 1.0	10,65%
Kaldi Speech Recognizer (GRU)	METU 1.0	11,25%

In the similar study [5], both DNN-HMM and GMM-HMM based recognizers are trained. A corpus which is recorded and prepared by authors is used for training and testing. The WER of the systems are measured as 14,65% and 17,40% respectively. The obtained results are similar to the GMM-HMM recognizer of our presented study. The used speech corpus and the applied Deep Learning method are the differences between two studies. The Recurrent Units, LSTM and GRU are used instead of Feed Forward Deep Neural Network (DNN) in our study. As mentioned in [5], researchers have proven that RNN and LSTM are advantageous over other deep learning techniques and capable of explaining the temporal evolution of input features. The results of our study show that, using LSTM and GRU has approximately 3,7% higher performance effect on the system. Unlike the study [5], the corpus “Turkish Microphone Speech v1.0” used in our presented study is a standard dataset and accepted by LDC. Researchers can access an approved dataset. Therefore, the evaluability, accessibility and comparability of studies using similar database are improved.

6. CONCLUSIONS

Considering the morphological structure of Turkish, the LSTM Deep Learning technique has been applied in recent studies to improve the performance of Turkish ASR systems. In this study, the performance of the suggested methods have been compared with the classical method using well known data set and corpus. Comparing due to WER criteria, the performance of the LSTM is higher than GMM-HMM method using the common corpus. When the results of the study are analyzed, it is observed that the LSTM-based system has increased the success of voice recognition and vocabulary recognition in the ASR problem. Deep Learning models can produce deeper and more precise feature possibilities and have a more distinctive ability in speech recognition. GRU networks saves computation time over LSTM and can be used in deeper networks having many hidden layers.

In our study, it is observed that when current deep learning approaches are used with a Turkish corpus approved by the LDC, there is an performance increase in solving Turkish ASR problem. In general, traditional methods used since the beginning of the ASR problem, current Deep Learning approaches and the differences between them are explained. The study will shed light on Turkish ASR applications researches that will be developed independently of the voice assistant applications currently used in smart devices. In order to increase performance of the study, a larger data set can be used or the hybrid methods which combine RNNs can be applied.

A common question is how high the accuracy performance of the voice assistants in smart devices such as, "Apple-Siri" and “Google Voice Transcription” is. These applications use a Deep Neural Network (DNN) to convert the acoustic pattern of your voice at each instant into a probability distribution over speech sounds. The ASR implementation of these applications are in the cloud. The cloud servers can provide large storage facilities and updates to the acoustic models used by the ASR. The accuracy of the systems is achieved using a large data sets, which are quite expensive to collect and prepare. Language models are typically trained over very large corpora of text. Having large data sets and a technological infrastructure which provides storage and processing power, allow the assistans to process complex acoustic and language models. Therefore, the performance of these voice assistants is higher than research studies that use limited processor resources and data sets.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Shewalkar, N., Nyavanandi, D., Ludwig, S. A., “Performance Evaluation of Deep Neural Networks Applied to Speech Recognition: RNN, LSTM and GRU”, Journal of Artificial Intelligence and Soft Computing Research, 9(4): 235-245, (2019).

[2] Kang J., Zhang, W., Liu, J., “Gated Recurrent Units Based Hybrid Acoustic Models for Robust Speech Recognition”, Proceedings of the 10th International Symposium on Chinese Spoken Language Processing (ISCSLP), Conference, Tianjin, (2016).

[3] Dridi, H., Ouni, K., “Towards Robust Combined Deep Architecture for Speech Recognition : Experiments on TIMIT”, International Journal of Advanced Computer Science and Applications (IJACSA), 11(4): 525-534, (2020).

[4] Tombaloğlu B., Erdem H., “Deep Learning Based Automatic Speech Recognition for Turkish”, Sakarya University Journal of Science, 24(4): 725 – 739, (2020).

[5] Kimanuka, U , Buyuk, O ., "Turkish Speech Recognition Based On Deep Neural Networks", Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 22: 319-329, (2018).

[6] Graves, A., Mohamed, A. R., Hinton, G., “Speech Recognition with Deep Recurrent Neural Networks”, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Conference, Vancouver, 6645- 6649, (2013).

[7] Arslan, R., S., Barışçı, N., “A Detailed Survey of Turkish Automatic Speech Recognition”, Turkish Journal of Electrical Engineering & Computer Sciences, 28: 3253-3269, (2020)

[8] Siri Team, “Hey Siri: An On-device DNN-powered Voice Trigger for Apple’s Personal Assistant”, machinelearning.apple.com, https://machinelearning.apple.com/research/hey-siri. Accessed date: 01.07. 2021.

[9] Beaufays, F., “The neural networks behind Google Voice transcription”, ai.googleblog.com, https://ai.googleblog.com/2015/08/the-neural-networks-behind-google-voice.html. Accessed date: 01.07. 2021.

[10] Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., Kingsbury, B., “Deep Neural Networks for Acoustic Modelling in Speech Recognition”, IEEE Signal Processing Magazine, 29(6): 82-97, (2012).

[11] Graves, A., Jaitly, N., “Towards End to End Speech Recognition with Recurrent Neural Networks”, Proceedings of the 31st International Conference on Machine Learning, Conference, Beijing, 1764-1772, (2014).

[12] Huang, K., Hussain, A., Wang, Q., Zhang, R., “Deep Learning: Fundamentals, Theory and Applications”, Springer, Edinburg, (2019).

[13] Ravanelli, M., Parcollet, T., Bengio, Y., “The Pytorch-Kaldi Speech Recognition Toolkit”, Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Conference, Brighton, (2018).

[14] Ravanelli, M., Brakel, P., Omologo, M., Bengio, Y., “Light Gated Recurrent Units for Speech Recognition”, IEEE Journal Of Emerging Topics In Computational Intelligence, 2(2): 92-102, (2018).

[15] Işık, G., Artuner, H., “Turkish Dialect Recognition In Terms Of Prosodic By Long Short-Term Memory Neural Networks”, Journal of the Faculty of Engineering and Architecture of Gazi University, 35(1): 213-224, (2020).

[16] Arslan, R., S., Barışçı, N., “The Effect of Different Optimization Techniques on End-to-End Turkish Speech Recognition Systems that use Connectionist Temporal Classification”, Proceedings of the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Conference, Turkey, (2018).

[17] Arısoy, E., Saraclar, M., “Multi-Stream Long Short-Term Memory Neural Network Language Model”, Proceedings of the 16th Annual Conference of the International-Speech-Communication-Association (INTERSPEECH 2015), Conference, Dresden, 1413-1417, (2015).

[18] Arısoy, E., Saraclar, M., “Lattice Extension and Vocabulary Adaptation for Turkish LVCSR”, IEEE Transactıons on Audio, Speech and Language Processıng, 17(1): 183-173, (2009).

[19] Salor, O., Pellom, B. L., Çiloğlu, T., Demirekler M., "Turkish Speech Corpora and Recognition Tools Developed by Porting SONIC: (Towards multilingual speech recognition)", Computer Speech and Language, 21, 580–593, (2007).

[20] Ruan, W., Gan Z., B Liu., Guo Y., "An Improved Tibetan Lhasa Speech Recognition Method Based on Deep Neural Network", Proceedings of the 10th International Conference on Intelligent Computation Technology and Automation, Conference, Changsha, 303-306, (2017).

[21] Bayer, A. O., Çiloglu, T., Yondem, M. T., "Investigation of Different Language Models for Turkish Speech Recognition", Proceedings of the IEEE 14th Signal Processing and Communications Applications, Conference, Antalya, (2006).

[22] Muda, L., Begam M., Elamvazuthi, I., "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques", Journal of Computing, 2(3): 138-143, (2010).

[23] Stuttle, M., N., "A Gaussian Mixture Model Spectral Representation for Speech Recognition", Ph.D. Thesis, Cambridge University, 45-46, (2003).

[24] Schiopu, D., "Using Statistical Methods in a Speech Recognition System for Romanian Language", Proceedings of the 12th IFAC Conference on Programmable Devices and Embedded Systems, Conference, Czech Republic, 99-103, (2013).

[25] Aksoylar, C., Mutluergil, S., Erdoğan H., "Bir Konuşma Tanıma Sisteminin Anatomisi", Proceedings of the IEEE 17th Signal Processing and Communications Applications, Conference, Antalya, 512-515, (2009).

[26] Dhankar, A., "Study of Deep Learning and CMU Sphinx in Automatic Speech Recognition", Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI), Conference, Udupi, 2296-2301, (2017).

[27] Guan, Y., Yuan, Z., Sun, G., Cong, J., "Fpga-based accelerator for Long Short-Term Memory Recurrent Neural Networks", Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), Conference, Chiba, 629–634, (2017).

[28] Hochreiter, S., Schmidhuber, J., "Long Short-Term Memory", Natural Computation, 9(8): 1735-1780, (1997).

[29] Graves, A., Schmidhuberab J., , "Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures", Proceedings of the International Joint Conference on Neural Networks (IJCNN), Conference, Montreal, 602-610, (2005).

[30] Tunalı, V., "A Speaker Dependent Large Vocabulary Isolated Word Speech Recognition System for Turkish", Msc. Thesis, Marmara University, 25-26, (2005).

[31] Büyük, O. "Sub-Word Language Modeling for Turkish Speech Recognition", Msc. Thesis, Sabanci University, 29-30, (2005).

[32] Arısoy, E., Arslan, L., M., "Turkish Dictating System for Broadcast News Applications", Proceedings of the 13th European Signal Processing, Conference, Antalya, (2005).

[33] Aksungurlu, T., Parlak, S., Sak, H., Saraçlar M., "Comparison of Language Modelling Approaches for Turkish Broadcast News", Proceedings of the 16th Signal Processing, Communication and Applications, Conference, Aydın, (2008).

[34] Varjokallio M., , Kurimo M., , Virpioja S., , "Learning a Subword Vocabulary Based on Unigram Likelihood", IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), Workshop, Czech Republic, 7-12, (2013).

[35] Mihajlik, P., Tüske, Z., Tárjan, B., Németh B., Fegyó T., "Improved Recognition of Spontaneous Hungarian Speech-Morphological and Acoustic Modeling Techniques for a Less Resourced Task", IEEE Transactions On Audio, Speech and Language Processing, 18(6): 1588-1600, (2010).

[36] Arısoy, E., Dutagacı, H., Saraclar, M., "A unified language model for large vocabulary continuous speech recognition of Turkish", Signal Processing, 86: 2844-2862, (2006).

[37] Dutagacı, H, "Statistical Language Models for Large Vocabulary Turkish Speech Recognition", Msc. Thesis, Boğaziçi University, 20-22, (2002).

[38] Arısoy, E., Saraclar, M., "Turkish Speech Recognition", Turkish Natural Language Processing, Springer, (2018).

[39] Polat, H., Oyucu, S., "Building a Speech and Text Corpus of Turkish: Large Corpus Collection with Initial Speech Recognition Results", Symmetry, 12(2): 290, (2020).

[40] Keser, S., Edizkan, R., "Phoneme-Based Isolated Turkish Word Recognition With Subspace Classifier", Proceedings of the IEEE 17th Signal Processing and Communications Applications, Conference, Antalya, 93-96, (2009).

[41] Susman, D., Köprü, S., Yazıcı, A., "Turkish Large Vocabulary Continuous Speech Recognition By Using Limited Audio Corpus", Proceedings of the IEEE 20th Signal Processing and Communications Applications Conference (SIU), Conference, Mugla, (2012).

[42] Yadava, G T., Jayanna, H S., "Creating Language and Acoustic Models using Kaldi to Build An Automatic Speech Recognition System for Kannada Language", Proceedings of the 2nd IEEE International Conference On Recent Trends in Electronics, Information & Communication Technology (RTEICT), Conference, India, 161-165, (2017).