

Article

Building a Speech and Text Corpus of Turkish: Large Corpus Collection with Initial Speech Recognition Results

Huseyin Polat *  and Saadin Oyucu 

Department of Computer Engineering, Faculty of Technology, Gazi University, 06560 Ankara, Turkey; saadinoyucu@gazi.edu.tr

* Correspondence: polath@gazi.edu.tr

Received: 5 February 2020; Accepted: 13 February 2020; Published: 17 February 2020



Abstract: To build automatic speech recognition (ASR) systems with a low word error rate (WER), a large speech and text corpus is needed. Corpus preparation is the first step required for developing an ASR system for a language with few argument speech documents available. Turkish is a language with limited resources for ASR. Therefore, development of a symmetric Turkish transcribed speech corpus according to the high resources languages corpora is crucial for improving and promoting Turkish speech recognition activities. In this study, we constructed a viable alternative to classical transcribed corpus preparation techniques for collecting Turkish speech data. In the presented approach, three different methods were used. In the first step, subtitles, which are mainly supplied for people with hearing difficulties, were used as transcriptions for the speech utterances obtained from movies. In the second step, data were collected via a mobile application. In the third step, a transfer learning approach to the Grand National Assembly of Turkey session records (videotext) was used. We also provide the initial speech recognition results of artificial neural network and Gaussian mixture-model-based acoustic models for Turkish. For training models, the newly collected corpus and other existing corpora published by the Linguistic Data Consortium were used. In light of the test results of the other existing corpora, the current study showed the relative contribution of corpus variability in a symmetric speech recognition task. The decrease in WER after including the new corpus was more evident with increased verified data size, compensating for the status of Turkish as a low resource language. For further studies, the importance of the corpus and language model in the success of the Turkish ASR system is shown.

Keywords: automatic speech recognition; speech corpus; text corpus; data acquisition; multi-layer neural network; natural language processing

1. Introduction

The primary function of an automatic speech recognition (ASR) system is to automatically convert human speech into transcribed text. Command and control systems, dictation software, broadcast news transcription (for indexing news content), and telephone speech transcriptions are at the forefront of the applications of ASR systems [1]. In addition, ASR can also be used to analyze social media data. The social data contains not only textual data, but also a large number of audio, video, and image data. ASR can be used to transcribe various audio recordings and videos on social media [2].

Typical ASR works with models trained by machine learning algorithms based on statistical pattern classification [3]. Machine learning algorithms use two main approaches: supervised and unsupervised learning. Supervised learning is usually used for classification and uses labelled data as a training set. Unsupervised learning is usually used for clustering with unlabelled data. ASR uses supervised

learning to training speech classifiers, such as hidden Markov models (HMMs) [4]. In training, the statistical pattern classification technique Gaussian mixture model (GMM) can be used with HMM. In the early development era of ASR, GMM with HMM were used successfully. Despite the success of GMM and HMM approaches, the accuracy of ASR systems still lags behind human-level performance. This implies that more research is required.

The first use of neural networks in speech applications in the 1990s never produced improved performance compared to the traditional GMM and HMM technology [5]. Some key problems appeared with the use of artificial neural networks (ANNs), such as the vanishing gradient, lack of sufficient amounts of training data, lack of significant computation power, and weak temporal correlation structure in the neural predictive models. To overcome these problems, studies have been conducted in different fields [6]. As a result of these studies, ANN-based approaches have shown accurate performance in many domains of research, such as image processing, speech recognition, language modelling, parsing, information retrieval, speech synthesis, and speech translation [7,8]. When an ANN is applied to these domains, the results obtained exceed those of other state-of-the-art approaches. The reason why an ANN-based approach produces this high performance is the capability to find and learn compound structures in a large amount of data [9]. These deep neural nets have been used in speech recognition to develop acoustic models (AMs). However, ANN-based approaches require large amounts of data to produce accurate results. Therefore, to develop a speaker-independent and high-quality performance (i.e., low word error rate) ASR, an extensive collection of speech samples must be gathered from various speakers along with the transcriptions of those speeches [10].

Developing a useful and practical process to create a Turkish transcribed speech corpus, consisting of audio and text data, poses a substantial challenge. The major problem with the existing commonly used methodologies to collect transcribed speech data is that the process is usually costly, both in terms of time and budget, as well as cumbersome. In the classical approach, native speakers of a particular language transcribe pre-recorded speech documents. Williams et al. showed that six hours of work is needed on average for transcribing one hour of speech [11].

Unfortunately, most of the spoken languages, including all Turkish-based Asian languages, are classified as low-resource for ASR applications due to the limited amount of available transcribed speech, which is required for training AMs and language models (LMs). Turkish language ASR systems also suffer from a low accuracy rate; hence, it is very crucial to create a symmetric Turkish transcribed speech corpus according to the corpus of the high resources languages.

The accuracy rate of ASR is not the same for different languages. Even for the same language, this rate may widely vary for different environments with different acoustic characteristics. However, even under the same conditions, ASR performance for English is known to be superior to that of Turkish due to the two main constraints in Turkish ASR. The first and more restrictive is the above-mentioned shortage of transcribed speech. The second reason is the agglutinative nature of the language with free word order and sequence, which leads to the generation of a relatively high number of new words through the use of suffixes. These characteristics complicate accurate prediction under conditional probabilities.

In this article, the two fundamental problems described above are addressed by developing a Turkish corpus with a large vocabulary. The Turkish speech corpus was obtained with three different approaches. The first approach used a feature film, the second one, a mobile application, and the third one, the transfer learning method. Also, the obtained Turkish speech corpus was presented for the approval of real users, so that more accurate data were obtained. The tests conducted by comparing the speech recognition performance of a system trained with the newly collected corpus to that of a system trained on an existing corpus showed that the results are comparable, which validates the effectiveness of the newly collected corpus. We completed studies to solve the difficulties in the processing of Turkish. We illustrate the procedures and the steps involved in ASR and present how different choices in the design can influence Turkish ASR performance. Finally, we show that ANN approaches could be more profitable.

The remainder of this paper is organized as follows. In the next section, a brief literature review of speech recognition research, with an emphasis on the Turkish-language-related efforts, is provided. Section 3 provides the details of the corpus collection process and the characteristics of the Turkish language. Section 4 describes the experimental setup. Section 5 outlines the joint results on the progress and evaluation of test sets, followed by our conclusions in Section 6.

2. Related Works

Numerous studies have aimed to recognize speech using a computer. The most important advances in this domain, however, were obtained after the use of HMMs, especially between the 1960s and 1970s. The theoretical bases of HMMs used in contemporary speech recognition systems were described in [12]. The historical developments of speech recognition systems were detailed in [13]. A tutorial on HMM and selected applications in speech recognition was published [4].

The first LM approach for improving speech recognition results was reported [14]. The authors introduced a component called the store of linguistic knowledge, which can be considered a precursory LM. Speech analysis results obtained from an acoustical analyzer were combined with the information provided by this LM using a computer, and a text output or transcript was produced.

During the early years of speech recognition activities, the comparison of speech recognition systems was tricky as there were no standards for database (corpus) creation or for the properties of the corpus. To solve these standardization issues, researchers from the Massachusetts Institute of Technology (MIT) and Texas Instruments (TI) joined efforts to create a recorded corpus to be used for training speech recognition systems. The corpus created as a result of this collaboration was named TIMIT and has since become a standard for benchmarking speech recognition results in English [15]. The Defense Advanced Research Projects Agency was also interested in collecting a corpus for speech recognition and recorded a corpus based on the Wall Street Journal (WSJ) [16]. The goal was to recognize read speech from the WSJ with a vocabulary size as large as 60,000 words.

Traditionally, two approaches are employed for collecting a transcribed speech corpus to be used for training speech recognition models. In the first approach, selected texts extracted from newspapers or books are read aloud by different speakers. In the second and more pervasive approach, pre-recorded television and radio programs are collected and manually transcribed [17–19]. The former approach is faster as it does not require a transcription effort; however, it has two main drawbacks. The first disadvantage is that spoken texts may not include enough samples for all the phonemes in the language. The second disadvantage is that only a limited number of speakers are used, and the environment is usually clean without any noise, which does not represent the test conditions well. The lack of speech samples from different types of speakers and different environments leads to poor recognition results for large vocabulary tasks. The second approach, transcription, is usually a costly and a rather slow process that is conducted by professional transcribers. Such experts demand approximately six hours of work for one hour of speech and the hourly cost of transcription is between USD \$90 and \$150 in the USA [20,21]. The slow pace and high expense of acquiring transcriptions are significant deterrents to further improvements in ASR for low-resourced languages. Additionally, procuring as many expert transcribers as needed for transcription projects of high-volume is difficult, mainly when the volume of work tends to fluctuate. On the other hand, more natural and spontaneous speech fragments from a large number of speakers can be collected in the corpus using the transcription approach.

Since the beginning of the 2010s, the speech recognition activity for Turkish has increased [22–24]. Attempts have been made to develop text and speech corpora for Turkish. The first transcribed Turkish speech corpus published by the Linguistic Data Consortium (LDC) contains eight hours of text, speech, and alignments [23]. This corpus contains 120 speakers (60 men and 60 women, all native Turkish speakers whose ages ranged between 19 and 50 years, with an average of 24 years) delivering 40 sentences each (approximately 300 words per speaker), which required approximately 500 min of clean and noise-free speech in total. The 40 sentences were selected randomly for each speaker from a triphone balanced set of 2462 Turkish sentences. In a more recent work, Oflazoglu and Yildirim

concentrated only on detecting emotions by creating a corpus from movies and labelling them with emotions, which is unsuitable for large vocabulary speech recognition tasks [25].

Recently, crowdsourcing has risen as a new method for the large-scale economic transcription of spoken documents [20–25]. In one study, large spoken documents were divided into smaller utterances, which were then distributed to a pool of staff through a coordinating web service, such as Amazon's Mechanical Turk [26]. The staff worked concurrently, accelerating the culmination of the original transcription task. The economics of such work incurred costs ranging from USD \$2.25 to \$22.50 per hour of transcription. Several studies used closed captions for the collection of speech databases from broadcast news [27–29]. Several other studies used movie subtitles for building parallel text datasets for aligning texts between different language pairs [30].

When previous studies were examined, we found that many field-specific studies were conducted. Some of these studies focused on a particular dialect. The study on the Goalparia dialect is one of them [31]. Records of spontaneous talks were taken from 27 speakers (19 men, 8 women). In total, six hours and eight minutes of speech data were collected for the Assam language. For the Bengali language, in which the dialect of Goalparia was spoken, records were collected from 30 speakers (20 men, and 10 women). In a study on South Korea, about 40 h of speech were obtained from 40 speakers [32].

Another study aimed to label a 20-h conversation by creating a model with an 11-h speech corpus, previously tagged in Slovakia [33]. In a study on Urdu, a speech corpus was created for travel. An interactive response system was used when constructing a corpus consisting of a total of 250 words, such as city names, days, and times [34]. In another study, a corpus was prepared for the Bahasa Indonesian language [35], records being captured in a studio environment. A corpus was created for the Japanese language to understand what older adults were saying. Recordings were collected from old people with the help of a microphone [36]. A comprehensive study was carried out on Hindi [37]. In this study, recordings were captured in a studio environment. However, the texts required for registration were subdivided and presented to the speaker. Thus, a balanced distribution was provided for different topics in the dataset. In a different study, the need for tag social media was first explained, and then a dataset containing ASR transcripts and social media data information was presented [2]. The data is predominantly English conversation records, but there are also small numbers of Czech, Dutch, French, Italian, German, and Spanish present.

When the literature was examined, we observed that speech data for different purposes, different language origins, and different dialects have been collected. When collecting speech data, different sources were used: movie subtitles, a studio environment, social media data, an interactive response system, and portable microphones, and audio files in different areas were transferred and labelled using existing speech recognition systems. When the Turkish language was considered, we found that the existing datasets were insufficient. For this reason, we conducted a comprehensive speech corpus creation work on the Turkish language. Film subtitles, mobile application, and transfer learning methods were used together when creating the corpus. In the selection of the film subtitles, the content of the genre and speech were considered. Studies were conducted to ensure that sex distribution was equal. All data were presented to real users through a web interface for approval. Developed using a combination of different methods, this study presents an innovative approach that may be useful for other languages.

3. Description of the Corpus and Collecting Process

In this study, the film subtitles, mobile application, and transfer learning methods were used together to create a speech corpus. The use of these methods is explained in detail under separate headings.

3.1. Use of Film Movies and Time-Bound Subtitle Documents

In this study, we used the audio collected from Turkish movies and the text extracted from the corresponding time-aligned subtitle documents as sources for spoken data and transcriptions, respectively. Using our proposed process, it is possible to collect massive amounts of transcribed speech for any given language, provided that a large collection of movies and their subtitles in that language are accessible.

Figure 1 shows a block diagram of the main modules for the movie speech corpus (MSC) collection process presented in this work. Our process has four main modules that were implemented as MSC collection tools.

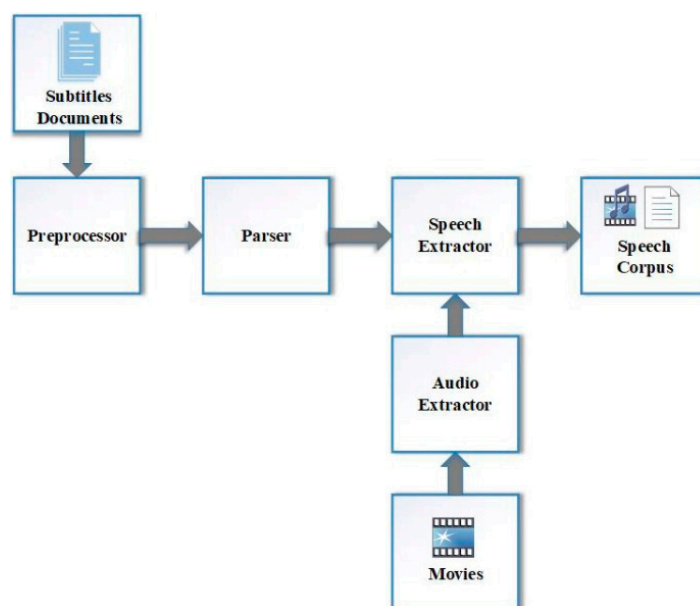


Figure 1. Block diagram of the movie speech corpus (MSC) collection process.

The preprocessor module is responsible for optimizing subtitle documents for automatic alignment. It performs pre-processing steps in the subtitle documents for this task. Each subtitle document is converted into a format that can be used in the training of the ASR system. The transcription of each speech section in the subtitle document is just below the time interval tag. The punctuation marks in the transcription prepared for each section were removed. Also, the pronunciation of the numbers and symbols were presented in written form. The transcriptions in the subtitle documents are ready for the parser module after these processes.

The parser module extracts the text that corresponds to the speech in subtitles. In this module, each speech section is transcribed in the subtitle document. Then, it is saved as a different text file. The ID provided during this process is saved to be provided to the speech file. The speech files corresponding to the text units obtained from the subtitle documents are created using the movie audio extractor.

The output of the parser and the movie audio extractor are used as inputs for the speech-extractor module, which cull speech segments from the given movie's audio. Before this process, audio information is extracted from film videos in the audio extractor module. In the speech extractor module, the speech information corresponding to the text information obtained by the parser module is extracted. The time interval of the speech information is provided at the beginning of each speech section in the subtitle document. The speech extractor process is performed according to this time interval information. The ID provided to the text file saved in the parser output is also given to the audio file in the speech extractor output. Thus, the necessary speech–text matching for the corpus was achieved.

To build a transcribed speech corpus, a total of 150 Turkish movies were collected. The film collection procedure consisted of choosing movie data by checking the summaries and the genre of movies. However, not all of the collected movies were used in the corpus collection procedure because the speech data extracted from some of these movies were found to be unsuitable due to a low speech-to-noise ratio or numerous speakers. After the screening process, 120 of these movies were selected for building the speech corpus.

Generally, the length of a speech corpus in hours is regarded as an indicator of how useful it may be to train an AM. The system developed here extracted 90 h of Turkish speech from subtitled Turkish movies, and the transcription of this extracted speech was collected from the prealigned subtitles associated with each movie. Table 1 summarizes some of the statistics of the speech data extracted from the selected movies.

Table 1. Statistics about movies used in the study.

Movie No.	No. of Words	No. of Unique Words	Length (min)
M1	4056	1308	55.0
M2	2306	900	62.0
...
M120	5440	1605	44.0
Avg.	4842	1229	45.0

As shown in Table 1, an average of 45.0 min of speech and transcription per processed movie were extracted using our MSC tools in as quickly as a few seconds. Table 1 also presents the number of words and the number of unique words (vocabulary size) to demonstrate the word diversity per movie. The number of words and vocabulary size together with the length of speech parts were used during the pre-selection process for eliminating movies with little speech content.

3.1.1. Pre-Processing of Subtitles

The goal of the preprocessor module is to convert each subtitle into a format that can be used by the speech recognition training tools for which the corpus is being prepared. Figure 2 presents an example of these subtitles for a Turkish movie.

```

1
00:01:31,540 --> 00:01:32,256
Evet burası çok güzel !!

2
00:01:33,860 --> 00:01:38,058
Peki siz ne zaman geliyorsunuz??

3
00:01:49,060 --> 00:01:54,373
[MUSIC]

4
00:01:55,420 --> 00:02:02,019
Aslında ben arabayı alıp gitmek istiyordum.

5
00:02:02,380 --> 00:02:02,892
<i>Buyurun gidelim.</i>

```

Figure 2. Sample subtitle document for Turkish movies.

As shown in Figure 2, each segment (time interval) of transcribed speech was numbered, and a new line was placed between successive segments. The transcription of each speech segment was immediately below its corresponding time interval tags.

The Turkish language also makes frequent use of umlaut letters, which are particularly prone to spelling, encoding, and formatting errors. For example, different forms of time interval specification,

unnecessary new lines, misspelled words, text encoding errors, and a varying number formats are the most frequently observed inconsistencies in the subtitles. In addition, subtitle documents are prepared in different text editors with varying Unicode standards. These differences cause some Turkish characters to be displayed incorrectly. For example, the Turkish characters Ü, Ç, Ö, İ, Ş, and Ğ have different encoding standards in different text editors. Therefore, for consistency and accuracy in the present study, all subtitle documents were re-encoded in a single Unicode standard: UTF-8. Also, in the pre-processing step, non-alphanumeric characters that were not spoken were removed from the given text during the pre-processing phase. All numbers in a subtitle document were converted to their official word forms because AM training requires a phonetic representation of numbers. For example, the number 86 was converted to “seksen altı”, which is the corresponding written form in Turkish.

3.1.2. Misspelled Words in the Subtitle Document

Misspelled words distort the statistical distributions that are used for training AM and LM. Hence, we had to detect and remedy misspellings during pre-processing in our study. Three different categories of frequent spelling errors occurred in the subtitle documents:

- Insertion errors: One or more extra letters in the word. For example, in the misspelled word “merrhabaa,” there are two letter insertion errors and the correct form should be “merhaba” (hello).
- Deletion errors: One or more letters are missing from the correct form of the word. For example, in “arbala,” there are two letter deletions, and the correct form should be “arabalar” (cars).
- Substitution errors: One or more letters of the original word are replaced with other letters. For example, in “çanda,” there is one substitution error, and the correct form is “çanta” (bag).

A proportion of these errors were fixed automatically using a Turkish spell checker framework. To correct the misspelled words, we used the Zemberek tool [38]. Devised for Turkic languages, mainly Turkish, Zemberek is a set of open-source natural language processing libraries and tools. Table 2 shows the statistics for the transcription corpus extracted from subtitles.

Table 2. Misspelled word statistics.

Corpus Name	Total Number of Words	Vocabulary Size	Misspelled (%)
MSC	670,546	100,128	5.2

As shown in Table 2, 5.2% of the total non-unique words were misspelled; therefore, fixing misspelled words enhanced the quality of transcription corpus by improving the accuracy of the transcription.

3.1.3. Speech Segmentation for Movies

Once the pre-processing steps were completed, time-aligned subtitle texts were ready to be used for the segmentation of speech data. First, the audio extracted from each movie was saved as a separate sound file in WAV format. The sampling rate and sampling size were set to 16 kHz and 16 bits, respectively, avoiding the need for re-adjusting the sampling rate since 16 kHz is the most common sampling rate used for speech recognition.

All the extracted audio files along with their subtitles needed to be checked to identify and correct misalignments between speech segments and their matched transcriptions in subtitle documents. Misalignment occurred when the start and end times of a speech segment in an audio file did not match the start and end times of the corresponding annotated transcription in the subtitle document. For example, a sentence started at the 120th second of the audio file and ended at the 150th second, but, in the subtitles document, this sentence was annotated between the 125th and 155th seconds. Misalignments were detected by randomly picking sample subtitles and verifying if start times were correctly marked by applying a speech recognition test that used our baseline AM and checked if

the error rate was above a certain threshold. Movies that failed this test were examined carefully by manually conducted a listening test to determine the time shift causing this misalignment. If the time shift was consistent throughout the movie, the correction of the subtitle time annotation was necessary. If not, the movie was marked as unreliable and excluded from the corpus.

3.2. Collection of Speech Data with a Mobile Application

Turkish is an agglutinating language, and the vocabulary is very large. A corpus containing only speech from films would not be sufficient for general-purpose ASR. For this reason, a mobile application was developed to increase the size of the corpus and to obtain speech and text data in different areas (magazine, sports, technology, agenda, etc.). Figure 3 shows a block diagram of the operation of the main modules for mobile data collection applications.

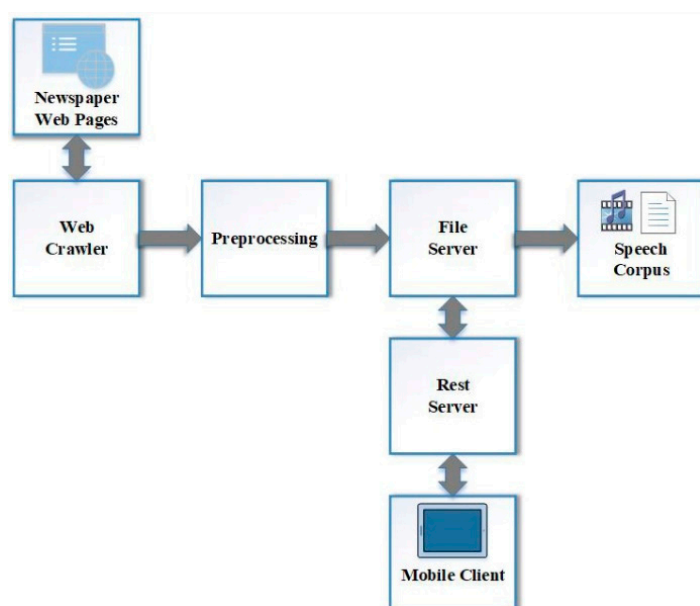


Figure 3. Collection of speech data with a mobile application.

Newly published newspaper articles were obtained with the help of a web crawler. The web crawler module had access to the published Turkish-language newspaper web pages at specific times. This procedure was limited to once a day. The identified newspaper articles were sports, magazines, and general articles. The pre-processing module analyzed the obtained articles with the help of a web crawler as a parser. Attention had to be paid to the absence of word loss at this stage. Texts separated into sentences in the preprocessor module were identified and uploaded to the file server. A representational state transfer (Rest) application programming interface (API) running on the Rest server sent previously prepared texts to a mobile client [39]. The mobile client showed the text to the user and asked the user to read the text. The created audio file created was identified and transferred to the file server. Thus, the text and voice files the users vocalized were matched and stored. A screenshot of the mobile application developed in this study and used by the real users is provided in Figure 4.

With the mobile application shown in Figure 4, the necessary text information was provided to the users. However, the user was asked to enter some information before starting the vocalization process. This information was sex, age, and name. The information received was necessary for sex distribution to be proportional when preparing the corpus. Additionally, in this study, a web interface was developed to measure the contribution of mobile application users to the corpus. Mobile users were selected from volunteer students, faculty, and staff at Gazi University (Ankara, Turkey). The native language of all mobile users was Turkish. The web interface that provides mobile user statistics is depicted in Figure 5.

Statistical information was gathered via the information from the user provided by the mobile application, such as who logs in and how many sessions they log in, the last time they sent data, how much data they processed, and how long data they vocalized. This information was followed through the prepared web interface. We found that the users created approximately 55 min of voice recordings. Also, we observed that mobile users were aged 19–52 years and the average age of mobile users was 26. The collection of the corpus via the mobile application was completed in 86 days. The developed mobile application was used by a total of 130 real people (50 men and 80 women). A total of 120 h of speech data were obtained by this method.

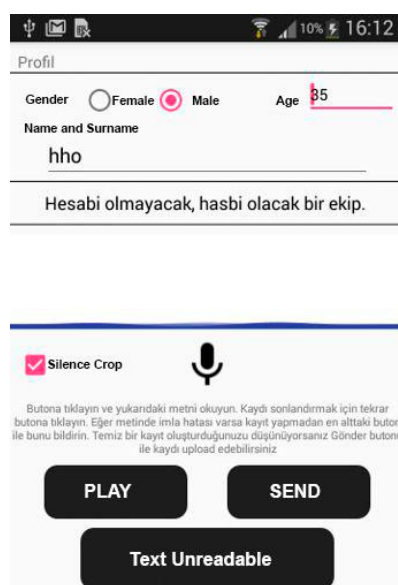


Figure 4. Mobile application interface developed for corpus collection.



Sort by	
<div style="display: flex; justify-content: space-between; align-items: center;"> 1 2 3 4 5 6 7 8 9 10 11 12 13 >> </div>	
Turk_Kubra_Yaz 	Total Number of Data Processed: 1419 Total Processed Data Length: 01:59:00.882 Last Post Time: 23.11.2019 09:01:49
Turk_Ahmet_Bugra 	Total Number of Data Processed: 602 Total Processed Data Length: 00:53:27.591 Last Post Time: 21.11.2019 18:55:02

Figure 5. Web interface for mobile usage statistics.

Some difficulties were encountered when collecting the corpus with the mobile application. Different ambient noises, the speaking style of the speaker, and the motivation of the mobile application user to use the application were among the major problems. Due to the different ambient noises in the environments in which the users were located, various acoustic information was also added to the system. However, the recordings, including inaudible speeches and high ambient noise, were not added to the corpus. Some issues occurring during corpus collection were eliminated due to the selection of voluntary users from university and training provided before the use of mobile applications.

3.3. Collection of Speech Data with Transfer Learning

Transfer learning is a standard method used in machine learning to transfer information from one model to another [40]. The purpose of transfer learning is to adapt the existing resource model

to a new task with limited target data. Transfer learning on the Grand National Assembly of Turkey (GNAT) session recording (videotext) was performed.

For the transfer learning, HMM-based AMs were trained using a seven-layer time-delay neural network (TDNN). The number of cell units in the layers was selected as 600. Rectified Linear Unit (ReLU) was used as the activation function. Acoustic model training was performed with the open-source Kaldi speech recognition toolkit. A Titan X Pascal GPU (Nvidia, Santa Clara, CA 95051, USA) was used for training. Resource model training was completed in four epochs using cross-entropy. The training of the source model was completed in 94 h. The source corpus was obtained from the film subtitles and mobile applications. The corpus consisted of a total of 210 h of speech data. For the resource LM training, a text collection with shared statistics was used, as shown in Table 2. Using this collection, a 3-g-based language model was trained and this LM was used as the source. The obtained source model was applied to the speech data of the Parliament as target data. Figure 6 shows a block diagram of the main modules used for data acquisition with the transfer learning presented in the study.

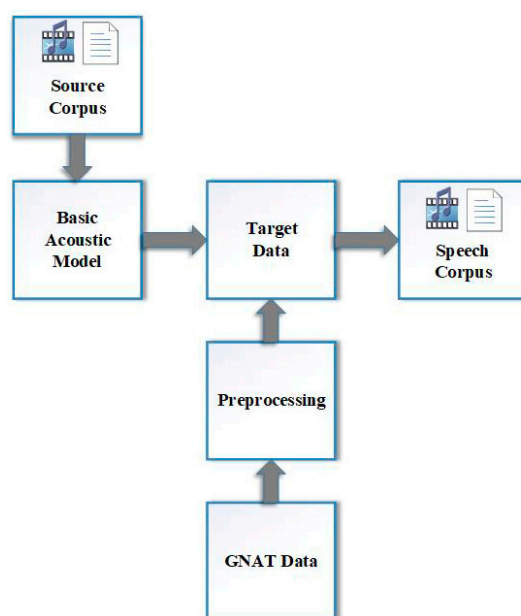


Figure 6. Collection of speech data with transfer learning.

The GNAT data module contained two different files. These files were GNAT session video recordings and session reports. The GNAT manually matched session reports and related videos, facilitating the process. In the pre-processing module, audio files were first extracted from video images. The separated audio files were 15–25 s. These audio files were used as input to the previously trained ASR system. The obtained texts from the ASR system output were investigated in the GNAT reports and the related texts (the original transcriptions) were found. A total of 250 h of speech data were obtained by the transfer learning method.

3.4. Corpus Verification Procedures

The all corpus (MSC, mobile-acquired speech and GNAT speech) obtained in this study was verified manually by real users. A web interface was created for this process. Figure 7 shows a block diagram of the main modules of corpus verification.

Audio-to-text mappings on the file server were received one by one and were transferred to a web interface with the help of Rest web service. Through the web interface, real users could control the received audio-to-text mappings. In this control process, users listened to the incoming audio data. They also controlled text information that was received with the audio file. The web interface created for the verification process is depicted in Figure 8.

As seen in Figure 8, the users could check and verify the audio–text mapping. The users could correct the text provided to them and send it back to the file server. In addition, if noise, overlap, and uncertainty existed within the audio data, they could label the data according to the specified situation. As a result of the verification process, we obtained a corpus that was controlled by real users.

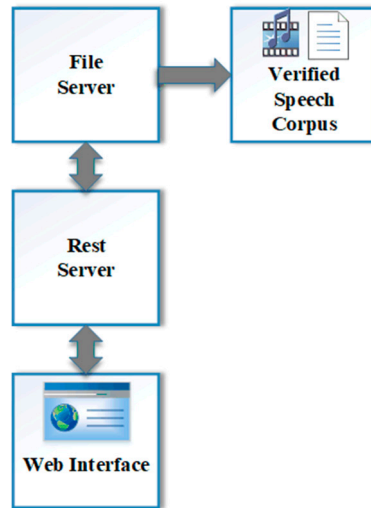


Figure 7. Corpus verification processes.



Figure 8. Web interface for the corpus verification process.

The human-based verification process requires extra time. However, it is necessary to perform the verification task to construct a totally error-free corpus. A web interface was created to obtain statistical information regarding the verification process, similar to the tracking of mobile application users. According to the statistical information obtained, an average of 2 h and 46 min of study was required to confirm one hour of speech. This indicates that the time spent on data verification was less than with the conventional transcription process. We also considered the possibility that users made mistakes. Some samples were selected from corpus data verified by each user and these samples were sent to real users again. The performance of the user performing the verification process was evaluated through this process.

4. Experimental Setup

This section explains the development of our Turkish ASR system in detail. We conducted experiments on an existing corpus and the newly collected corpus. We aimed to overcome the difficulties in the processing of Turkish. We illustrate the procedures and the steps involved for ASR and present how different choices in the design influence the Turkish ASR system performance.

4.1. Speech Corpus for Experiments

The first Turkish speech corpus was prepared by the Middle East Technical University (METU) in 2006 [23]. The METU corpus is formed from the speech of 120 different people, including 60 women and 60 men. Each person spoke 40 sentences on average and approximately 300 words. The average age of the speakers was 24 years. Another Turkish Corpus was prepared by Bogazici University in 2012 [17]. Both corpora were set to 16 kHz sampling frequencies with a 16-bit sample size. Thus, the acoustic information similarity was provided to produce more accurate results. The obtained corpus information and corpus data in this study are detailed in Table 3. The corpus developed within the scope of this study is called the HS corpus. When creating the HS corpus, three different methods were used. The real users using the mobile application included 50 men and 80 women. Since the target in the transfer learning process was corpus GNAT data, we had 490 male and 96 female speakers. Since the age of the deputies working at the GNAT was confidential, that information was unavailable. Sex distribution is a difficult task for the MSC because the same actor may act in different films. Therefore, identifying the number of different speakers was difficult. For this task, the number of actors in the films was obtained. We found that eight different people spoke in each movie on average. Classification studies on the HS corpus in the future will have the necessary statistical information.

Table 3 provides information about the Turkish corpora considered in the study. The verified HS corpus refers to a corpus verified by real users. The unverified HS corpus refers to a corpus that had not passed through the filter of real users. The speech signals were sampled at 16 kHz and parsed up to a maximum of 15 s. A text file with the transcription corresponding to each speech file was also provided.

Table 3. Information on Turkish corpora.

Corpus Name	Duration (h)	Total Number of Utterances
METU	8.33	8002
Bogazici	94.44	82,033
Unverified HS	460.12	780,014
Verified HS	350.27	565,073

The Mel frequency scale was used in feature extraction operations. The Mel frequency cepstral coefficient (MFCC) is a feature extraction technique commonly used in speech recognition systems [41]. The frequency bands are logarithmically located in the MFCC. The MFCC calculation is based on short-term analysis. In this study, for MFCC feature extraction, speech signals were divided into overlapping segments and each segment is windowed. Each segment's length is 25 ms and the overlapping ratio was taken as 40%. We calculated 13 MFCCs per segment using a Mel-scaled filterbank with 23 triangular filters distributed between 20 and 16,000 Hz.

4.2. Development of the Turkish ASR System for Experiments

The Kaldi toolkit was used for the development of the Turkish ASR system. Kaldi is an open-source toolkit for speech recognition applications written in C++ and licensed under Apache License v2.0 [42]. The Kaldi toolkit is basically connected to two external libraries. The first one is OpenFst and the other is the digital algebra library. The digital algebra library is divided into: BLAS and LAPACK. The Kaldi library modules are positioned so that each module is connected to only one of the external libraries. Access to library functions is provided by code snippets written in C++. The code snippets and libraries prepared in Kaldi are called by the scripting language to create and run the ASR system.

A Turkish ASR system was developed using version 5.0 of the Kaldi toolkit. The AM in the classical GMM-HMM-based ASR system was prepared using Gauss blend models. Based on phoneme units, all basic GMM models were created and their initial values were assigned. Then, following the coding of the texts in the training corpus, these models were properly integrated. The classical GMM model to be created from a training sample, from which phoneme models are combined, is determined

by the word-based analysis of the text on the training sample and the phoneme analysis corresponding to the words in a pronunciation lexicon. GMM model parameters for all phonemes must be trained on a large number of observations. The datasets prepared for training AM in this study were large enough to estimate model parameters.

Acoustic models can be generated in many ways using HMM, GMM, and deep neural networks (DNNs). In this study, the subspace Gaussian mixture model (SGMM)-based acoustic models were used. SGMM is an efficient method for GMM acoustic modelling by means of strong a speaker adaptation. The SGMM uses mixtures of Gaussians as the underlying state distribution. In addition, it contains the state-dependent parameters, like mean, covariance, and mixture weights. The state-dependent parameters are estimated for low dimensional subspace. In this study, the mean of the Gaussian components was calculated to create a low dimensional subspace. A maximum likelihood linear transform (MLLT) and linear discriminant analysis (LDA) for speaker adaptation training were used. To create an LDA-MLLT model, MLLT was applied on top of LDA features. Then, speaker adaptation training was performed on top of the LDA-MLLT models. Our GMM models have a total of 3500 context-dependent (CD) triphone states. In addition, they have a mean of 14 Gaussian components per state. Every triphone was modelled with three CD states.

In this study, the ASR system was developed primarily based on the classical GMM-HMM and that improved with a DNN-based system. The ASR system using neural networks is different from the classical GMM-based ASR system. In the ASR system using classical GMM-HMM, the probability of observation of each HMM state was calculated using GMM. However, in the DNN-based AM, the probability of observation of each HMM triphone state was calculated using a DNN. In the DNN-based AM, a phoneme tag was assigned to these features by taking the sound attributes. Technically, when training the GMM-HMM system, the monophone model was trained from the expression-level transcripts. To train the DNN, we obtained phoneme-sound alignments produced by the traditional GMM-HMM system. Therefore, DNN acoustic modelling depends on the properties used to train a GMM-HMM as well as the decision tree produced by the GMM-HMM. In this study, acoustic feature frames in DNNs were placed on the first layer, and the neural network assigned a phoneme label to each acoustic frame.

The DNN-based ASR was developed by taking MFCC as input into the neural network, phoneme state posteriors per frame (HMM states) were estimated, and then stochastic gradient descent (SGD) was applied to compare the estimated phoneme state posteriors with the HMM state obtained from GMM system. Then, the weights of the neural network were adjusted. To speed up the training of our neural network, we computed the gradient on a small portion of training data, known as a mini-batch, then we updated the weights soon after every mini-batch. The DNN network used five hidden layers containing 500 neurons each. In our experiment, we used either a hyperbolic tangent activation function or a P-norm generalized maxout function. Also, a 3-g LM was applied for ASR. We trained our DNN system using a Nvidia Titan X Pascal GPU.

5. Experimental Results

Performance is expressed in terms of word error rate (WER) for different experiments. Several measurement methods are available to evaluate ASR performance. The most accurate measurement method is to evaluate the differences between the hypothesis and the reference words [43]. For this reason, we did WER as a performance metric in this study. WER is calculated as

$$WER = \frac{D + S + I}{N} \times 100 \quad (1)$$

where N is the total number of symbols in the reference word, D represents the number of deleted symbols in the hypothesis with respect to the reference word, S is the number of changed symbols, and I represents the number of additional symbols.

In the study, two approaches were compared: GMM and DNN. However, we also compared the multiplicity and variety of data. Two different ASR systems were developed, and similar vocabularies and language models were used, but their acoustic models were different. In all the tests, the DNN-based system performed better than the GMM-based system. This result shows that DNN is a better choice than GMM for acoustic modelling in Turkish ASR systems. DNN performed better because the DNN was able to capture current context information by adding adjacent frames. Table 4 shows the effect of HS corpus and other corpora regarding the GMM-based Turkish ASR system.

Table 4. Results of Gaussian mixture model (GMM) based Turkish automatic speech recognition (ASR) systems.

Corpus Name	Word Error Rate (WER%)
METU	70.71
Bogazici	27.70
Unverified HS	55.27
Verified HS	24.70

For all experiments, one-third of the corpus was reserved for test purposes. As shown in Table 4, the METU corpus had the worst WER in the tests performed. The verified HS corpus produced the best WER. When the details of the corpus were examined, we observed many trivial examples within the METU corpus. These pointless examples and limited information adversely affected the performance of the ASR system. When we analyzed the case of the Bogazici corpus, we concluded that the acoustic diversity was insufficient even though the content was smooth.

The performance of the ASR system depends not only on the phonetic information obtained from the AM but also on the representation of the lexicon and syntax obtained by the LM. Therefore, in addition to the preparation of the text corpus, each developed model was tested using an LM with different n-gram values. Tests were carried out with 2-, 3-, 4- and 5-g. The most successful results were recorded at 3- and 4-g values. Table 5 lists the effect of different n-gram values on ASR systems developed on different datasets. Experiments were conducted without changing the GMM-based AM.

Table 5. Results of different n-gram values on Turkish ASR systems.

Corpus Name	N-Gram	Word Error Rate (WER%)
METU	2-Gram	78.06
METU	3-Gram	70.71
METU	4-Gram	72.46
METU	5-Gram	74.92
Bogazici	2-Gram	31.79
Bogazici	3-Gram	27.70
Bogazici	4-Gram	28.23
Bogazici	5-Gram	30.02
Unverified HS	2-Gram	63.14
Unverified HS	3-Gram	55.27
Unverified HS	4-Gram	58.93
Unverified HS	5-Gram	61.02
Verified HS	2-Gram	29.41
Verified HS	3-Gram	24.70
Verified HS	4-Gram	26.19
Verified HS	5-Gram	28.01

The results demonstrated that the phonetic content of the HS corpus performed well for balanced and different language models. The nature of Turkish explained the decline in the word-error rate for larger n-gram language models. The use of a larger set of texts for language model training led to better results.

In the tests of the DNN-based ASR system, the size of the corpus is crucial. We observed that different parameters affect the system. The depth of a DNN has a significant effect on performance. However, both the training and the decoding processing of large models are slow. Therefore, we recommend selecting the appropriate amount of hidden layers rather than choosing a large model size. Theoretically, deeper DNNs must be capable of modelling more complex functions than simple neural networks. However, optimizing the size of models in real applications is considered a problem.

In addition to the number of hidden layers and the hidden layer size, the learning rate, which is an important parameter, also affects the performance of the system. However, the size of the corpus is important in the selection of the learning rate. A high learning rate for a large corpus will require a longer training period. Another important parameter for DNN is the size of the minibatch. The size of the minibatch should be selected in intervals such as 128, 256, or 512. Using a large minibatch is useful because it considers interaction with the optimizations used in matrix multiplication, especially when a GPU is used. In the case of CPU usage, a large minibatch size can cause instability. For these reasons, the size of the minibatch was set to 128 for multi-threaded CPU-based training, and 512 for GPU-based training. The performance of DNN-based ASR systems, trained and tested on different datasets, is outlined in Table 6. The ASR system was trained on 158 h when using the verified HS corpus.

Table 6. Results of deep neural networks (DNN)-based Turkish ASR systems.

Corpus Name	Word Error Rate (WER%)
METU	64.55
Bogazici	22.63
Unverified HS	49.20
Verified HS	18.70

The results presented in Table 6 show that results on the LDC database are comparable with those obtained on the HS corpus. Experiments on the METU corpus led to a much higher WER rate. This situation can be explained by two reasons: the size and the quality of the corpus. Turkish is a language with an extensive vocabulary. Therefore, the size of the corpus and the number of unique words should be high. The speech files are very short in the METU corpus. Therefore, sufficient acoustic information in the METU corpus has not been completely captured; on the other hand, the Bogazici corpus contains news broadcasts. Therefore, the vocabulary of the corpus is small and most of the voice recordings are presenter speeches, limiting the variety of the acoustic environments.

The HS corpus includes the data obtained from different environments. Therefore, more information about the acoustic environment is provided in the HS corpus than in the other two corpora. The HS corpus not only contains political or news speeches but also those from other fields. For this reason, when an ASR system developed with HS corpus is applied to different areas, its recognition performance will be higher than that of other available corpora. A new set of experiments indicated the generalization ability of HS corpus is much better. In these experiments, ASR models developed with the HS corpus were tested on METU and Bogazici. In the experiments, one-third of both corpora was reserved for test purposes. The results obtained are provided in Table 7.

Table 7. Results of DNN-based Turkish ASR system (trained on the verified HS corpus) on different test corpora.

Training Corpus	Test Corpus	Word Error Rate (WER%)
Verified HS	METU	1.8
Verified HS	Bogazici	2.3

When the results in Table 7 are analyzed, we found that the ASR system developed with the verified HS corpus produced better results in different corpus recognition tasks. METU and Bogazici corpora

include transcriptions of broadcast news and newspaper news. Therefore, the acoustic information to be obtained from these two corpora is nearly symmetric. Most of these transcriptions have political content. The HS corpus, including particularly GNAT data, more successfully recognized political news speeches.

In this paper, a quick method was presented for creating a detailed speech recognition corpus. This method is a suitable alternative to the costly corpus preparation techniques used to construct a corpus. When tests were performed with the corpus, deep-learning-based multi-layer DNN approaches were found to be more efficient than classical GMM-based approaches. In addition, the importance of corpus size in deep learning-based multi-layer DNN approaches was demonstrated.

6. Conclusions and Future Work

In this study, we created a viable alternative procedure for collecting Turkish speech data to classical transcribed corpus preparation techniques. In the presented approach, three different steps were used. In the first step, instead of using selected pre-recorded utterances, we used movies as the source of speech utterances. Subtitle documents, which are mainly supplied for people with hearing difficulties, were then used as transcriptions for the speech utterances obtained from the movies. In the second step, speech data were collected via a mobile application from real users. Text data presented to users were taken from current newspaper texts. In the third step, a transfer learning approach was used. Thus, the most common corpus preparation and verification approaches were presented for Turkish speech recognition systems.

The main challenges during the study were the selection of appropriate movies, the construction of an appropriate LM, and creating a baseline speech recognition system. We selected appropriate movies based on their genre and their speech content inferred from the subtitle documents. We also manually verified the speech content of movies by randomly listening to parts of the movies. The results obtained from the speech recognition experiments showed that the proposed method provided an acceptable and relatively low-cost alternative for building a transcribed speech corpus. However, the cost of data collection through mobile applications increased because finding enough human resources was a difficult task. The environments in which the mobile application was used were noisy, and the inclusion of unwanted sounds complicated the process.

In transfer learning, we observed that pre-processing time on video data was a challenge. A web interface was developed to verify the obtained corpus. Actual users were asked to verify this corpus. This process was a challenging task due to the increase in human resource requirements. However, this step was crucial for training the basic model with a corpus.

The process was designed so that many other movies in Turkish could be added to increase the corpus size for further corpus improvement. For mobile applications, many articles from various fields could be collected from different web pages. Thus, diversity could be increased. The basic model could be successfully created, and the transfer learning method could be used in various areas. Thus, a corpus could be constructed for different areas and languages.

Looking at the Turkish ASR results, we observed that DNN-based approaches were better than classical GMM based approaches. In DNN-based approaches, the corpus size is quite important. To prepare the corpus, a unique approach was presented. Thus, a Turkish ASR was developed and tested with the largest available corpus. The two main outputs of this study were: the collection of the largest speech corpus in Turkish and highlighting the importance of the corpus and LM in the success of an ASR system. As a result, the Turkish corpus deficiency reported in many studies was resolved in this study. The superiority of DNN-based speech recognition approaches over classical approaches was evident in the study.

The process outlined here is also applicable for creating a domain-specific corpus for use in the training of specialized speech recognition systems. These specific systems could include the medical domain, law domain, call centres, or noisy environments. We think that this work on Turkish ASR will provide opportunities for more research on Turkish AM and LM. The designing of algorithms

for speaker and environment adaptation in DNN and designing an LM using deep neural networks are future paths to explore. Different models could be developed using the HS corpus, and with its increased use, further studies could be conducted to increase the performance of Turkish speech recognition systems.

Author Contributions: Conceptualization, H.P. and S.O.; methodology, S.O. and H.P.; software, S.O.; validation, S.O. and H.P.; formal analysis, S.O. and H.P.; investigation, S.O. and H.P.; resources, S.O. and H.P.; data curation, S.O. and H.P.; writing—original draft preparation, S.O. and H.P.; writing—review and editing, S.O. and H.P.; visualization, S.O. and H.P.; supervision, H.P.; project administration, H.P. All authors have read and agreed to the published version of the manuscript.

Funding: No additional funding was received.

Acknowledgments: We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Oyucu, S.; Sever, H.; Polat, H. Overview of automatic speech recognition, approaches and challenges: Way the future to Turkish speech recognition. *Gazi Univ. Sci. J. Part C Des. Technol.* **2019**, *7*, 834–854.
- Schmiedeke, S.; Xu, P.; Ferrané, I.; Eskevich, M.; Kofler, C.; Larson, M.; Estève, Y.; Lamel, L.; Jones, G.; Sikora, T. Blip10000: A social video dataset containing SPUG content for tagging and retrieval. In Proceedings of the 4th ACM Multimedia Systems Conference, Oslo, Norway, 27 February–1 March 2013; pp. 96–101.
- Braun, D.; Neil, D.; Liu, S. A curriculum learning method for improved noise robustness in automatic speech recognition. In Proceedings of the European Signal Processing Conference, Kos, Greece, 28 August–2 September 2017; pp. 548–552.
- Paramonov, P.; Sutula, N. Simplified scoring methods for HMM-based speech recognition. *Soft Comput.* **2016**, *20*, 3455–3460. [[CrossRef](#)]
- Bourlard, H.; Morgan, N. *Connectionist Speech Recognition a Hybrid Approach*, 1st ed.; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1994.
- Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3rd ed.; Pearson Education: London, UK, 2010.
- Siniscalchi, S.M.; Svendsen, T.; Lee, C.H. An artificial neural network approach to automatic speech processing. *Neurocomputing* **2014**, *140*, 326–338. [[CrossRef](#)]
- Jena, M.; Mishra, S. Review of Neural Network Techniques in the Verge of Image Processing. In *International Proceedings on Advances in Soft Computing, Intelligent Systems and Applications. Advances in Intelligent Systems and Computing*; Springer: Singapore, 2018.
- Mahanty, R.; Mahanti, P.K. Unleashing artificial intelligence onto big data: A review. In *Research on Computational Intelligence Applications in Bioinformatics*, 1st ed.; Dash, S., Subudhi, B., Eds.; IGI Global: Hershey, PA, USA, 2016; pp. 1–16.
- Aggarwal, R.; Dave, M. Acoustic modeling problem for automatic speech recognition system: Advances and refinements (Part II). *Int. J. Speech Technol.* **2011**, *14*, 1572–8110. [[CrossRef](#)]
- Williams, J.D.; Melamed, I.D.; Alonso, B.; Hollister, T.; Wilpon, J. Crowd-sourcing for difficult transcription of speech. In Proceedings of the IEEE Workshop on Automatic Speech Recognition Understanding, Big Island, HI, USA, 11–15 December 2011; pp. 535–540.
- Rabiner, L.R. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proc. IEEE* **1989**, *2*, 257–286. [[CrossRef](#)]
- Janet, M.B.; Li, D.; Sanjeev, K.; Chin-Hui, L.; James, G.; Nelson, M. Historical development and future directions in speech recognition and understanding. In *Report of the Speech Understanding Working Group*; NIST: Gaithersburg, MD, USA, 2007; pp. 1–21.
- Fry, D.B. Theoretical aspects of mechanical speech recognition. *J. Br. Inst. Radio Eng.* **1959**, *19*, 211–218. [[CrossRef](#)]
- John, S.; William, M.; Jonathan, G. *Darpa Timit*; National Institute of Standards and Technology Computer Systems Laboratory: Gaithersburg, MD, USA, 1993; pp. 1–99.
- Paul, D.; Douglas, B.; Baker, M. The design for the Wall Street Journal-based CSR corpus. *Assoc. Comput. Linguist.* **1992**, *6*, 357–362.

17. Arisoy, E.; Can, D.; Parlak, S.; Saraçlar, M.; Sak, H. Turkish broadcast news transcription and retrieval. *IEEE Trans. Audio Speech-Lang. Process.* **2009**, *17*, 874–883. [CrossRef]
18. Graff, D. An overview of broadcast news corpora. *Speech Commun.* **2002**, *37*, 15–26. [CrossRef]
19. Matsoukas, S. Advances in transcription of broadcast news and conversational telephone speech within the combined EARS BBN/LIMSI system. *IEEE Trans. Audio Speech-Lang. Process.* **2006**, *14*, 1541–1554. [CrossRef]
20. Novotney, S.; Callison-Burch, C. Cheap, fast and good enough: Automatic speech recognition with non-expert transcription. *Hum. Lang. Technol.* **2010**, *6*, 207–215.
21. Christopher, C.; David, M.; Kevin, W. The Fisher corpus: A resource for the next generations of speech-to-text. In Proceedings of the International Conference on Language Resources and Evaluation, Lisbon, Portugal, 26–28 May 2004; pp. 69–71.
22. Salor, O.; Pellom, B.; Ciloglu, T.; Hacıoglu, K.; Demirekler, M. Developing new text and audio corpora and speech recognition tools for the Turkish language. In Proceedings of the International Conference Spoken Language Processing, Denver, CO, USA, 16–20 September 2002; pp. 349–352.
23. Salor, O.; Pellom, B.; Ciloglu, T.; Demirekler, M. Turkish speech corpora and recognition tools developed by porting SONIC: Towards multilingual speech recognition. *Comput. Speech Lang.* **2007**, *21*, 580–593. [CrossRef]
24. Sak, H.; Saraçlar, M.; Güngör, T. Morpholexical and discriminative language models for Turkish automatic speech recognition. *IEEE Trans. Audio Speech-Lang. Process.* **2012**, *20*, 2341–2351. [CrossRef]
25. Oflazoglu, C.; Yildirim, S. Recognizing emotion from Turkish speech using acoustic features. *EURASIP J. Audio Speech Music Process.* **2013**, *1*, 1–11. [CrossRef]
26. Fort, K.; Adda, G.; Cohen, K.B. Amazon Mechanical Turk: Gold Mine or Coal Mine. *Comput. Linguist.* **2011**, *37*, 413–420. [CrossRef]
27. Schultz, T. Globalphone: A multilingual speech and text database developed at Karlsruhe University. In Proceedings of the Annual Conference of the International Speech Communication Association, Singapore, 16–20 September 2002; pp. 345–348.
28. Chan, H.Y.; Woodland, P. Improving broadcast news transcription by lightly supervised discriminative training. In Proceedings of the IEEE International Conference Acoustic Speech, Signal Processing, Montreal, QC, Canada, 17–21 May 2004; pp. 3–6.
29. Zhang, S.; Ling, W.; Dyer, C. Dual subtitles as parallel corpora. *Eur. Lang. Resour. Assoc.* **2014**, *5*, 1869–1874.
30. Lavecchia, C.; Smaili, K.; Langlois, D. Building parallel corpora from movies. In Proceedings of the International Workshop on Natural Language Processing and Cognitive Science, Funchal, Madeira, Portugal, 12–16 June 2007; pp. 201–210.
31. Ismail, T.; Joyprakash, L. Development of speech corpora for Goalparia dialect and similar languages. In Proceedings of the IEEE International Conference on Signal and Image Processing Applications, Kuching, Malaysia, 12–14 September 2017; pp. 12–14.
32. Weonhee, Y.; Kyuchul, Y.; Sunwoo, P.; Juhee, L.; Sungmoon, C.; Ducksoo, K.; Koonhyuk, B.; Hyeseung, H.; Jungsun, K. The Korean corpus of spontaneous speech. *J. Korean Soc. Speech Sci.* **2015**, *7*, 103–109.
33. Kočúr, T.; Ondáš, S.; Juhár, J. Speech corpus generation based on N-gram confidence measure classification. In Proceedings of the International Symposium ELMAR, Zadar, Croatia, 18–20 September 2017; pp. 149–152.
34. Qasim, M.; Rauf, S.; Hussain, S.; Habib, T. Urdu speech corpus for travel domain. In Proceedings of the Conference of the Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques, Bali, Indonesia, 26–28 October 2016; pp. 237–241.
35. Cahyaningtyas, E.; Arifianto, D. Development of under-resourced Bahasa Indonesia speech corpus. In Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, Kuala Lumpur, Malaysia, 12–15 December 2017; pp. 1097–1101.
36. Iribe, Y.; Kitaoka, N.; Segawa, S. Development of new speech corpus for elderly Japanese speech recognition. In Proceedings of the International Conference on Asian Spoken Language Research and Evaluation, Shanghai, China, 28–30 October 2015; pp. 27–31.
37. Magdum, D.; Shukla, M.; Patil, T.; Shah, R.; Belhe, S.; Kulkarni, M. Methodology for designing and creating Hindi speech corpus. In Proceedings of the International Conference on Signal Processing and Communication Engineering Systems, Guntur, India, 2–3 January 2015; pp. 336–339.
38. Akin, A.A.; Akin, M.D. Zemberek, an open source NLP framework for Turkic languages. *Structure* **2007**, *10*, 1–5. Available online: <http://zemberek.googlecode.com/> (accessed on 5 January 2020).

39. Paik, H.; Lemos, A.L.; Barukh, M.C.; Benatallah, B.; Natarajan, A. Web Services—REST or Restful Services. In *Web Service Implementation and Composition Techniques*; Springer: Cham, Switzerland, 2017.
40. Lee, J.Y.; Derroncourt, F.; Szolovits, P. Transfer learning for named-entity recognition with neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, Miyazaki, Japan, 7–12 May 2018; pp. 4470–4473.
41. Dave, N. Feature extraction methods LPC, PLP and MFCC. *Int. J. Adv. Res. Eng. Technol.* **2013**, *1*, 1–5.
42. Povey, D.; Ghoshal, A.; Boulianne, G.; Burget, L.; Glembek, O.; Goel, N.; Hannemann, M.; Motlicek, P.; Qian, Y.; Vesely, K. The Kaldi speech recognition toolkit. In *Proceedings of the Workshop on Automatic Speech Recognition and Understanding*, Big Island, HI, USA, December 2001; pp. 1–4.
43. Aksoylar, C.; Mutluergil, S.O.; Erdogan, H. The anatomy of a Turkish speech recognition system. In *Proceedings of the IEEE Signal Processing and Communications Applications Conference*, Antalya, Turkey, 9–11 April 2009; pp. 512–515.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).