SYSTEM AND METHOD FOR IMPROVING THE ACCURACY OF A SPEECH RECOGNITION PROGRAM

Background of the Invention

1. Field of the Invention

The present invention relates in general to computer speech recognition systems and, in particular, to a system and method for expediting the aural training of an automated speech recognition program.

2. Background Art

Speech recognition programs are well known in the art. While these programs are ultimately useful in automatically converting speech into text, many users are dissuaded from using these programs because they require each user to spend a significant amount of time training the system. Usually this training begins by having each user read a series of pre-selected materials for several minutes. Then, as the user continues to use the program, as words are improperly transcribed the user is expected to stop and train the program as to the intended word thus advancing the ultimate accuracy of the speech files. Unfortunately, most professionals (doctors, dentists, veterinarians, lawyers) and business executive are unwilling to spend the time developing the necessary speech files to truly benefit from the automated transcription.

Accordingly, it is an object of the present invention to provide a system that offers expedited training of speech recognition programs. It is an associated object to provide a simplified means for providing verbatim text files for training the aural parameters (i.e. speech files, acoustic model and/or language model) of a speech recognition portion of the system.

In a previously filed, co-pending patent application, the assignee of the present application teaches a system and method for quickly improving the accuracy of a speech recognition program. That system is based on a speech recognition program that automatically converts a pre-recorded audio file into a written text. The system parses the written text into segments, each of which is corrected by the system and saved in an individually retrievable manner in association with the computer. In that system, the speech recognition program saves the standard speech files to improve accuracy in speech-to-text conversion. That system further includes facilities to repetitively establish

an independent instance of the written text from the pre-recorded audio file using the speech recognition program. That independent instance can then be broken into segments. Each segment in the independent instance is replaced with an individually retrievable saved corrected segment, which is associated with that segment. In that manner, applicant's prior application teaches a method and apparatus for repetitive instruction of a speech recognition program.

Certain speech recognition programs, however, do not facilitate speech to text conversion of pre-recorded speech. One such program is the commercially successful ViaVoice product sold by IBM Corporation of Armonk, New York. Yet, the receipt of pre-recorded speech is integral to the automation of transcription services. Consequently, it is a further object of the present invention to direct the output of a pre-recorded audio file into a speech recognition program that does not normally provide for such functionality.

These and other objects will be apparent to those of ordinary skill in the art having the present drawings, specification and claims before them.

Summary of the Invention

The present invention relates to a system for improving the accuracy of a speech recognition program. The system includes means for automatically converting a pre-recorded audio file into a written text. Means for parsing the written text into segments and for correcting each and every segment of the written text. In a preferred embodiment, a human speech trainer is presented with the text and associated audio for each and every segment. Whether the human speech trainer ultimately modifies a segment or not, each segment (after an opportunity for correction, if necessary) is stored in a retrievable manner in association with the computer. The system further includes means for saving speech files associated with a substantially corrected written text and used by the speech recognition program towards improving accuracy in speech-to-text conversion.

The system finally includes means for repetitively establishing an independent instance of the written text from the pre-recorded audio file using the speech recognition program and for replacing each segment in the independent instance of the written text with the corrected segment associated therewith.

In one embodiment. the correcting means further includes means for highlighting likely errors in the written text. In such an embodiment, where the written text is at least temporarily synchronized to said pre-recorded audio file, the highlighting means further includes means for sequentially comparing a copy of the written text with a second written text resulting in a sequential list of unmatched words culled from the written text and means for incrementally searching for the current unmatched word contemporaneously within a first buffer associated with the speech recognition program containing the written text and a second buffer associated with a sequential list of possible errors. Such element further includes means for correcting the current unmatched word in the second buffer. In one embodiment. the correcting means includes means for displaying the current unmatched word in a manner substantially visually isolated from other text in the written text and means for playing a portion of said synchronized voice dictation recording from said first buffer associated with said current unmatched word.

The invention further involves a method for improving the accuracy of a speech recognition program operating on a computer comprising: (a) automatically converting a pre-recorded audio file into a written text; (b) parsing the written text into segments: (c) correcting each and every segment of the written text; (d) saving the corrected segment in a retrievable manner; (e) saving speech files associated with a substantially corrected written text and used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program; (f) establishing an independent instance of the written text from the pre-recorded audio file using the speech recognition program; (g) replacing each segment in the independent instance of the written text with the corrected segment associated therewith; (h) saving speech files associated with the independent instance of the written text used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program: and (i) repeating steps (f) through (i) a predetermined number of times.

In another embodiment of the invention the means for parsing the written text into segments includes means for directly accessing the functions of the speech recognition program. The parsing means may include means to determine the character count to the beginning of the segment and means for determining the character count to the end of the segment. Such parsing means may further include the UtteranceBegin

function of Dragon Naturally Speaking to determine the character count to the beginning of the segment and the UtteranceEnd function of Dragon Naturally Speaking to determine the character count to the end of the segment.

The means for automatically converting a pre-recorded audio file into a written text may further be accomplished by executing functions of Dragon Naturally Speaking. The means for automatically converting may include the TranscribeFile function of Dragon Naturally Speaking.

The system may also include, in part, a method for directing a pre-recorded audio file to a speech recognition program that does not normally accept such files, such as IBM Corporation's Via Voice speech recognition software. The method includes: (a) launching the speech recognition program to accept speech as if the speech recognition program were receiving live audio from a microphone; (b) finding a mixer utility associated with the sound card; (c) opening the mixer utility, the mixer utility having settings that determine an input source and an output path; (d) changing the settings of the mixer utility to specify a line-in input source and a wave-out output path; (e) activating a microphone input of the speech recognition software; and (f) initiating a media player associated with the computer to play the pre-recorded audio file into the line-in input source.

In a preferred embodiment, this method for directing a pre-recorded audio file to a speech recognition program may further include changing the mixer utility settings to mute audio output to speakers associated with the computer. Similarly, the method would preferably include saving the settings of the mixer utility before they are changed to reroute the audio stream and restoring the saved settings after the media player finishes playing the pre-recorded audio file.

The system may also include, in part, a system for directing a pre-recorded audio file to a speech recognition program that does not accept such files. The system includes a computer having a sound card with an associated mixer utility and an associated media player (capable of playing the pre-recorded audio file). The system further includes means for changing settings of the associated mixer utility, such that the mixer utility receives an audio stream from the media player and outputs a resulting audio stream to the speech recognition program as a microphone input stream.

In one preferred embodiment, the system further includes means for automatically opening the speech recognition program and activating the changing means. The system also preferably includes means for saving and restoring an original configuration of the mixer utility.

Brief Description of the Drawings

Fig. 1 of the drawings is a block diagram of the system for quickly improving the accuracy of a speech recognition program;

Fig. 2 of the drawings is a flow diagram of a method for quickly improving the accuracy of a speech recognition program;

Fig. 3 of the drawings is a plan view of one approach to the present system and method in operation in conjunction with DRAGON NATURALLY SPEAKING software;

Fig. 4of the drawings is a flow diagram of a method for quickly improving the accuracy of the DRAGON NATURALLY SPEAKING software;

Fig. 5 of the drawings is a flow diagram of a method for automatically training the DRAGON NATURALLY SPEAKING software;

Fig. 6 of the drawings is a plan view of one approach to the present system and method showing the highlighting of a segment of text for playback or edit;

Fig. 7 of the drawings is a plan view of one approach to the present system and method showing the highlighting of a segment of text with an error for correction;

Fig. 8 of the drawings is a plan view of one approach to the present system and method showing the initiation of the automated correction method;

Fig. 9 of the drawings is a plan view of one approach to the present system and method showing the initiation of the automated training method;

Fig. 10 of the drawings is a plan view of one approach to the present system and method showing the selection of audio files for training for addition to the queue;

Fig. 11 of the drawings is a flow chart showing the steps used for directing an audio file to a speech recognition program that does not accept such files; and

Figs. 12A and 12B of the drawings depict the graphical user interface of one particular sound card mixer utility that can be used in directing an audio file to a speech recognition program that does not accept such files..

Best Modes of Practicing the Invention

While the present invention may be embodied in many different forms, there is shown in the drawings and discussed herein a few specific embodiments with the understanding that the present disclosure is to be considered only as an exemplification of the principles of the invention and is not intended to limit the invention to the embodiments illustrated.

Fig. 1 of the drawings generally shows one potential embodiment of the present system quickly improving the accuracy of a speech recognition program. The system must include some means for receiving a pre-recorded audio file. This audio file receiving means can be a digital audio recorder, an analog audio recorder, or standard means for receiving computer files on magnetic media or via a data connection; preferably implemented on a general-purpose computer (such as computer 20), although a specialized computer could be developed for this specific purpose.

The general-purpose computer should have, among other elements, a microprocessor (such as the Intel Corporation PENTIUM. AMD K6 or Motorola 68000 series); volatile and non-volatile memory; one or more mass storage devices (i.e. HDD, floppy drive, and other removable media devices such as a CD-ROM drive, DITTO. ZIP or JAZ drive (from Iomega Corporation) and the like); various user input devices, such as a mouse 23, a keyboard 24, or a microphone 25; and a video display system 26. In one embodiment, the general-purpose computer is controlled by the WINDOWS 9.x operating system. It is contemplated, however, that the present system would work equally well using a MACINTOSH computer or even another operating system such as a WINDOWS CE, UNIX or a JAVA based operating system, to name a few. In any embodiment, the general purpose computer has amongst its programs a speech recognition program, such as DRAGON NATURALLY SPEAKING. IBM's VIA VOICE. LERNOUT & HAUSPIE"S PROFESSIONAL EDITION or other programs.

Regardless of the particular computer platform used, in an embodiment utilizing an analog audio input (such as via microphone 25) the general-purpose computer must include a sound-card 27. Of course, in an embodiment with a digital input no sound card would be necessary to input the file. However, sound card 27 is likely to be necessary for playback such that the human speech trainer can listen to the pre-recorded audio file toward modifying the written text into a verbatim text.

Generally, this pre-recorded audio file can be thought of as a ".WAV" file. This ".WAV" file can be originally created by any number of sources, including digital audio recording software; as a byproduct of a speech recognition program; or from a digital audio recorder. Of course, as would be known to those skilled in the art, other audio file formats, such as MP2, MP3, RAW, CD, MOD, MIDI, AIFF, mu-law or DSS, could also be used to format the audio file, without departing from the spirit of the present invention. The method of saving such audio files is well known to those of ordinary skill in the art.

In one embodiment, the general purpose computer may be loaded and configured to run digital audio recording software (such as the media utility in the WINDOWS 9.x operating system, VOICEDOC from The Programmers' Consortium, Inc. of Oakton, Virginia, COOL EDIT by Syntrillium Corporation of Phoenix, Arizona or Dragon Naturally Speaking Professional Edition by Dragon Systems, Inc.) In another embodiment, the speech recognition program may create a digital audio file as a byproduct of the automated transcription process.

Another means for receiving a pre-recorded audio file is dedicated digital recorder 14, such as the Olympus Digital Voice Recorder D-1000 manufactured by the Olympus Corporation. Thus, if a user is more comfortable with a more conventional type of dictation device, they can use a dedicated digital recorder in combination with this system. In order to harvest the digital audio text file, upon completion of a recording, dedicated digital recorder would be operably connected toward downloading the digital audio file into that general-purpose computer. With this approach, for instance, no audio card would be required.

Another alternative for receiving the pre-recorded audio file may consist of using one form or another of removable magnetic media containing a pre-recorded audio file.

With this alternative an operator would input the removable magnetic media into the general-purpose computer toward uploading the audio file into the system.

In some cases it may be necessary to pre-process the audio files to make them acceptable for processing by the speech recognition software. For instance, a DSS or RAW file format may selectively be changed to a WAV file format, or the sampling rate of a digital audio file may have to be upsampled or downsampled. Software to accomplish such pre-processing is available from a variety of sources including Syntrillium Corporation and Olympus Corporation.

In some manner, an acceptably formatted pre-recorded audio file is provided to a first speech recognition program that produces a first written text therefrom. The first speech recognition program may also be selected from various commercially available programs, such as Naturally Speaking from Dragon Systems of Newton, Massachusetts, Via Voice from IBM Corporation of Armonk, New York, or Speech Magic from Philips Corporation of Atlanta, Georgia is preferably implemented on a general-purpose computer, which may be the same general-purpose computer used to implement the pre-recorded audio file receiving means. In Dragon Systems' Naturally Speaking, for instance, there is built-in functionality that allows speech-to-text conversion of pre-recorded digital audio. Accordingly, in one preferred approach, the present invention can directly access executable files provided with Dragon Naturally Speaking in order to transcribe the pre-recorded digital audio.

In an alternative approach, Dragon Systems' Naturally Speaking is used by running an executable simultaneously with Naturally Speaking that feeds phantom keystrokes and mousing operations through the WIN32API, such that Naturally Speaking believes that it is interacting with a human being, when in fact it is being controlled by the microprocessor. Such techniques are well known in the computer software testing art and, thus, will not be discussed in detail. It should suffice to say that by watching the application flow of any speech recognition program, an executable to mimic the interactive manual steps can be created.

In an approach using IBM Via Voice – which does not have built-in functionality to allow speech-to-text conversion of pre-recorded audio – the system preferably includes a sound card (such as sound cards produced by Creative Labs, Trident, Diamond, Yamaha, Guillemot, NewCom, Inc., Digital Audio Labs, and Voyetra Turtle

Beach. Inc.). The key to the this embodiment is the configuration of sound card 27 to "trick" IBM Via Voice into thinking that it is receiving audio input (live audio) from a microphone or in-line when the audio is actually coming from a pre-recorded audio file. As an example. rerouting can be achieved using a SoundBlaster Live sound card from Creative Labs of Milpitas, California.

Fig. 11 is a flowchart showing the steps used for directing an audio file to a speech recognition program that does not accept such files, such IBM ViaVoice. In particular. the following steps are used as an example implementation: (1) speech recognition software is launched; (2) the speech recognition window of the speech recognition software is opened in the same as if a live speaker were using the speech recognition software; (3) find mixer utility associated with the sound card using operating system functionality; (4) open mixer utility (see the depiction of one of mixer's graphical user interface in Fig. 12A); (5) (Optional) save current sound card mixer settings; (6) change sound card mixer settings to a specific input source (i.e. "line-in") and the output path to wave-out (via "What U Hear" in the case of SoundBlaster Live Card.); (7) (Optional) change the sound card mixer settings to mute the speaker output; (8) activate the microphone input of the speech recognition software; (9) initiate the media player device to play ".WAV" file into the line-in specified in step 6; (10) open the speech recognition window such that the speech recognition program receives the redirected audio and transcribe the document; (11) (Optional) restore the sound card mixer settings saved in step 5.

The foregoing steps are automated by running an executable simultaneously with the speech recognition software that feeds phantom keystrokes and mousing operation through WIN32API, such that the speech recognition software believes that it is interacting with a human being, when in fact it is being controlled by the microprocessor. It is appreciated that these techniques are well known by those skilled in the computer software testing art. It should suffice to say that by watching the application flow of the foregoing steps. an executable to mimic the interactive manual steps can be created.

One example of code to effect the change of the mixer settings to redirect the audio of a Sound Blaster Live card from Creative Labs in a WIN9x environment with IBM ViaVoice software is shown in Appendix A.

In a preferred embodiment, the transcription errors in the first written text are located in some manner to facilitate establishment of a verbatim text for use in training the speech recognition program. In one approach, a human transcriptionist establishes a transcribed file, which can be automatically compared with the first written text creating a list of differences between the two texts, which is used to identify potential errors in the first written text to assist a human speech trainer in locating such potential errors to correct same. Such effort could be assisted by the use of specialized software for isolating or highlighting the errors and synchronizing them with their associated audio.

In another approach for establishing a verbatim text, the acceptably formatted pre-recorded audio file is also provided to a second speech recognition program that produces a second written text therefrom. The second speech recognition program has at least one "conversion variable" different from the first speech recognition program. Such "conversion variables" may include one or more of the following:

(1) speech recognition programs (e.g. Dragon Systems' Naturally Speaking, IBM's Via Voice or Philips Corporation's Speech Magic);

(2) language models within a particular speech recognition program (e.g. general English versus a specialized vocabulary (e.g. medical, legal));

(3) settings within a particular speech recognition program (e.g. "most accurate" versus "speed"); and/or

(4) the pre-recorded audio file by pre-processing same with a digital signal processor (such as Cool Edit by Syntrillium Corporation of Phoenix, Arizona or a programmed DSP56000 IC from Motorola, Inc.) by changing the digital word size, sampling rate, removing particular harmonic ranges and other potential modifications.

By changing one or more of the foregoing "conversion variables" it is believed that the second speech recognition program will produce a slightly different written text than the first speech recognition program and that by comparing the two resulting written texts a list of differences between the two texts to assist a human speech trainer in locating such potential errors to correct same. Such effort could be assisted by the use of specialized software for isolating or highlighting the errors and synchronizing them with their associated audio.

In one preferred approach, the first written text created by the first speech recognition is fed directly into a segmentation/correction program. (See Fig. 2). The segmentation/correction program utilizes the speech recognition program's parsing system to sequentially identify speech segments toward placing each and every one of those speech segments into a correction window – whether correction is required on any portion of those segments or not. A speech trainer plays the synchronized audio associated with the currently displayed speech segment using a "playback" button in the correction window and manually compares the audible text with the speech segment in the correction window. If one of the pre-correction approaches disclosed above is used than less corrections should be required at this stage. However, if correction is necessary, then that correction is manually input with standard computer techniques (using the keyboard, mouse and/or speech recognition software and potentially lists of potential replacement words).

Sometimes the audio is unintelligible or unusable (e.g., dictator sneezes and speech recognition software types out a word, like "cyst"--an actual example). Sometimes the speech recognition program inserts word(s) when there is no detectable audio. Or sometimes when the dictator says a command like "New Paragraph." and rather than executing the command, the speech recognition software types in the words "new" and "paragraph". One approach where there is noise or no sound, is to type in some nonsense word like "xxxxx" for the utterance file so that audio text alignment is not lost. In cases, where the speaker pauses and the system types out "new" and "paragraph." the words "new" and "paragraph" may be treated as text (and not as command). Although it is also possible to train commands to some extent by replacing such an error with the voice macro command (e.g. "\New-Paragraph"). Thus, it is contemplated that correction techniques may be modified to take into account the limitations and errors of the underlying speech recognition software to promote improved automated training of speech files.

In another potential embodiment, unintelligible or unusable portions of the pre-recorded audio file may be removed using an audio file editor, so that only the usable audio would be used for training the speech recognition program.

Once the speech trainer believes the segment in the correction window is a verbatim representation of the synchronized audio, the segment is manually accepted and

the next segment automatically displayed in the correction window. Once accepted, the corrected/verbatim segment from the correction window is pasted back into the first written text. In one approach, the corrected verbatim segment is additionally saved into the next sequentially numbered "correct segment" file. Accordingly, in this approach, by the end of a document review there will be a series of separate computer files containing the verbatim text, numbered sequentially, one for each speech segment in the currently first written text.

In Dragon's Naturally Speaking these speech segments vary from 1 to, say 20 words depending upon the length of the pause setting in the Miscellaneous Tools section of Naturally Speaking. If you make the pause setting long, more words will be part of the utterance because a long pause is required before Naturally Speaking establishes a different utterance. If it the pause setting is made short, then there are more utterances with few words. In IBM Via Voice, the size of these speech segments is similarly adjustable, but apparently based on the number of words desired per segment (e.g. 10 words per segment).

One potential user interface for implementing the segmentation/correction scheme is shown in Fig. 3. In Fig. 3, the Dragon Naturally Speaking program has selected "seeds for cookie" as the current speech segment (or utterance in Dragon parlance). The human speech trainer listening to the portion of pre-recorded audio file associated with the currently displayed speech segment, looking at the correction window and perhaps the speech segment in context within the transcribed text determines whether or not correction is necessary. By clicking on the "Play Back" button the audio synchronized to the particular speech segment is automatically played back. Once the human speech trainer knows the actually dictated language for that speech segment, they either indicate that the present text is correct (by merely pressing an "OK" button) or manually replace any incorrect text with verbatim text. In either event, in this approach, the corrected/verbatim text from the correction window is pasted back into the first written text and is additionally saved into the next sequentially numbered correct segment file.

In this approach, once the verbatim text is completed (and preferably verified for accuracy), the series of sequentially numbered files containing the text segments are used to train the speech recognition program. First, video and storage buffer of the speech

recognition program are cleared. Next, the pre-recorded audio file is loaded into the first speech recognition program, in the same manner disclosed above. Third, a new written text is established by the first speech recognition program. Fourth, the segmentation/correction program utilizes the speech recognition program's parsing system to sequentially identify speech segments and places each and every one of those speech segments into a correction window – whether correction is required on any portion of those segments or not – seriatim. Fifth, the system automatically replaces the text in the correction window using the next sequentially numbered "correct segment" file. That text is then pasted into the underlying Dragon Naturally Speaking buffer (whether or not the original was correct) and the segment counter is advanced. The fourth and fifth steps are repeated until all of the segments have been replaced.

By automating this five-step process, the present system can produce a significant improvement in the accuracy of the speech recognition program. Such automation would take the form of an executable simultaneously operating with the speech recognition means that feeds phantom keystrokes and mousing operations through the WIN32API, such that the first speech recognition program believes that it is interacting with a human being, when in fact it is being controlled by the microprocessor. Such techniques are well known in the computer software testing art and, thus, will not be discussed in detail. It should suffice to say that by watching the application flow of any speech recognition program, an executable to mimic the interactive manual steps can be created. This process is also automated to repeat a pre-determined number of times.

This selection and replacement of every text segment within the buffer leads to an improvement in the aural parameters of the speech recognition program for the particular speech user that recorded the pre-recorded audio file. In this manner, the accuracy of first speech recognition program's speech-to-text conversion can be markedly, yet quickly improved.

Alternatively, in another approach to correcting the written text, various executable files associated with Dragon Systems' Naturally Speaking may be directly accessed. This allows the present invention to use the built in functionality of Naturally Speaking to transcribe pre-recorded audio files. Fig. 4 is a flow diagram of this approach using the Dragon software developer's kit ("SDK"). A user selects an audio file (usually ".wav") for automatic transcription. The selected pre-recorded audio file is sent to the

TranscribeFile module of Dictation Edit Control of the Dragon SDK. As the audio is being transcribed. the location of each segment of text is determined automatically by the speech recognition program. For instance, in Dragon, an utterance is defined by a pause in the speech. As a result of Dragon completing the transcription. the text is internally "broken up" into segments according to the location of the utterances by the present invention.

In this alternative approach, the location of the segments is determined by the Dragon SDK UtteranceBegin and UtteranceEnd modules which report the location of the beginning of an utterance and the location of the end of an utterance. For example, if the number of characters to the beginning of the utterance is 100, and to the end of the utterance is 115. then the utterance begins at 100 and has 15 characters. This enables the present system to find the text for audio playback and automated correction. The location of utterances is stored in a listbox for reference. Once transcription ends (using the TranscribeFile module), the text is captured. The location of the utterances (using the UtteranceBegin and UtteranceEnd modules) is then used to break apart the text to create a list of utterances.

Each utterance is listed sequentially in a correction window (see Fig. 6). The display may also contain a window that allows the user to view the original transcribed text. As in the other approach, the user then manually examines each utterance to determine if correction is necessary. Using the utterance locations. the present program can play the audio associated with the currently selected speech segment using a "playback" button in the correction window toward comparing the audible text with the selected speech segment in the correction window. As in the other approach, if correction is necessary, then that correction is manually input with standard computer techniques (using the keyboard, mouse and/or speech recognition software and, potentially. lists of potential replacement words) (see Fig. 7).

Once the speech trainer believes the segment in the correction window is a verbatim representation of the synchronized audio. the segment in the correction window is manually accepted and the next segment automatically displayed in the correction window. Once the erroneous utterances are corrected. the user may then have the option to calculate the accuracy of the transcription performed by Dragon. This process compares the corrected set of utterances with the original transcribed file. The

percentage of correct words can be displayed, and the location of the differences is recorded by noting every utterance that contained an error. In the approach using the Dragon SDK, the corrected set of utterances may then be saved to a single file. In this embodiment, all the utterances are saved to this file, not just corrected ones. Thus, this file will contain a corrected verbatim text version of the pre-recorded audio.

As in the other approach, the user may then choose to do an automated correction of the transcribed text (see Fig. 8). This process inserts the corrected utterances into the original transcription file via Dragon's correction dialog. After corrections are complete, the user is prompted to Save the Speech file. This correction approach uses the locations of the differences between the corrected utterances and the transcribed text to only correct the erroneous utterances. Consequently, unlike the other approach to the training of the speech recognition program, only erroneous segments are repetitively corrected. Consequently in the approach using the Dragon SDK, as the number of errors diminish, the time to incrementally train the speech recognition program will drop.

Another novel aspect of this invention is the ability to make changes in the transcribed file for the purposes of a written report versus for the verbatim files (necessary for training the speech conversion program). The general purpose of the present invention is to allow for automated training of a voice recognition system. However, it may also happen that the initial recording contains wrong information or the wrong word was actually said during recording. (e.g the user said 'right' during the initial recording when the user meant to say 'left') In this case, the correction of the text cannot normally be made to a word that was not actually said in the recording as this would hinder the training of the voice recognition system. Thus, in one embodiment the present invention may allow the user to make changes to the text and save this text solely for printing or reporting, while maintaining the separate verbatim file to train the voice recognition system.

One potential user interface for implementing the segmentation/correction scheme for the approach using the Dragon SDK is shown in Fig. 6. In Fig. 6, the program has selected "a range of dictation and transcription solutions" as the current speech segment. As in the other approach, the human speech trainer listening to the portion of pre-recorded audio file associated with the currently displayed speech segment, looking at the correction window and perhaps the speech segment in context

within the transcribed text determines whether or not correction is necessary. By clicking on the "Play Selected" button the audio synchronized to the particular speech segment is automatically played back. As in the other approach, once the human speech trainer knows the actually dictated language for that speech segment, they either indicate that the present text is correct or manually replace any incorrect text with verbatim text. In this SDK-based approach, in either event, the corrected/verbatim text from the correction window is saved into a single file containing all the corrected utterances.

Once the verbatim text is completed (and preferably verified for accuracy), the file containing the corrected utterances can be used to train the speech recognition program (see Fig. 9). Fig. 5 is a flow diagram describing the training process. The user has the option of running the training sequence a selected number of times to increase the effectiveness of the training. The user chooses the file on which to perform the training. The chosen files are then transferred to the queue for processing (Fig. 10). Once training is initiated, the file containing the corrected set of utterances is read. The corrected utterances file is opened and read into a listbox. This is not a function of the Dragon SDK, but is instead a basic I/O file. Where the SDK is used, the associated pre-recorded audio file is sent to TranscribeFile method of DictationEditControl from the Dragon SDK. (In particular, the audio file is sent by running the command "FrmControls.DeTop2.TranscribeFile filename;" FrmControls is the form where the Dragon SDK ActiveX Controls are located; DeTop2 is the name of the controls.) Transcribe File is the function of controls for transcribing wave files. In conjunction with this transcribing, the UtteranceBegin and UtteranceEnd methods of DragonEngineControl report the location of utterances in the same manner as previously described. Once transcription ends, the location of the utterances that were determined are used to break apart the text. This set of utterances is compared to the list of corrected utterances to find any differences. One program used to compare the differences (native to Windows 9.x) may be File Compare. The location of the differences are then stored in a listbox. Then the locations of differences in the list box are used to only correct the utterances that had differences. Upon completion of correction, speech files are automatically saved. This cycle can then be repeated the predetermined number of times.

Once training is complete. TranscribeFile can be initiated one last time to transcribe the pre-recorded audio. The location of the utterances are not calculated again

in this step. This transcribed file is compared one more time to the corrected utterances to determine the accuracy of the voice recognition program after training.

The foregoing description and drawings merely explain and illustrate the invention and the invention is not limited thereto. Those of the skill in the art who have the disclosure before them will be able to make modifications and variations therein without departing from the scope of the present invention.

WHAT IS CLAIMED IS:

1. A system for improving the accuracy of a speech recognition program operating on a computer, said system comprising:

- means for automatically converting a pre-recorded audio file into a written text;
- means for parsing said written text into segments;
- means for correcting each and every segment of said written text;
- means for saving each corrected segment in a retrievable manner in association with said computer;
- means for saving speech files associated with a substantially corrected written text and used by said speech recognition program towards improving accuracy in speech-to-text conversion by said speech recognition program; and
- means for repetitively establishing an independent instance of said written text from said pre-recorded audio file using said speech recognition program and for replacing each segment in said independent instance of said written text with said corrected segment associated therewith.

2. The invention according to Claim 1 including means for saving said corrected segment in an individually retrievable manner in association with said computer.

3. The invention according to Claim 1 wherein said parsing means includes means for directly accessing functions of said speech recognition program.

4. The invention according to Claim 3 wherein said parsing means further include means to determine the character count to the beginning of each of said segments and means to determine the character count to the end of each of said segments.

5. The invention according to Claim 4 wherein said means to determine the character count to the beginning of each of said segments includes UtteranceBegin function from the Dragon Naturally Speaking, and said means to determine the character



count to the end of each of said segments includes UtteranceEnd function from the Dragon Naturally Speaking .

6. The invention according to Claim 1 wherein said means for automatically converting includes means for directly accessing functions of said speech recognition program.

7. The invention according to Claim 6 wherein said means for automatically converting further includes TranscribeFile function of Dragon Naturally Speaking.

8. The invention according to Claim 1 wherein said correcting means further includes means for highlighting likely errors in said written text.

9. The invention according to Claim 8 wherein said written text is at least temporarily synchronized to said pre-recorded audio file. said highlighting means comprises:

- means for sequentially comparing a copy of said written text with a second written text resulting in a sequential list of unmatched words culled from said copy of said written text, said sequential list having a beginning, an end and a current unmatched word, said current unmatched word being successively advanced from said beginning to said end;

- means for incrementally searching for said current unmatched word contemporaneously within a first buffer associated with the speech recognition program containing said written text and a second buffer associated with said sequential list: and

- means for correcting said current unmatched word in said second buffer. said correcting means including means for displaying said current unmatched word in a manner substantially visually isolated from other text in said copy of said written text and means for playing a portion of said synchronized voice dictation recording from said first buffer associated with said current unmatched word.

* C L M S *

10. The invention according to Claim 9 wherein said second written text is established by a second speech recognition program having at least one conversion variable different from said speech recognition program.

11. The invention according to Claim 9 wherein said second written text is established by one or more human beings.

12. The invention according to Claim 9 wherein said correcting means further includes means for alternatively viewing said current unmatched word in context within said copy of said written text.

13. The invention according to Claim 1 further including means for directing said pre-recorded audio file to said speech recognition program, wherein said speech recognition program does not accept such files, said means for directing comprising:

- said computer having a sound card with an associated mixer utility;

- a media player operably associated with said computer, said media player capable of playing said pre-recorded audio file; and

- means for changing settings of said associated mixer utility, such that said mixer utility receives an audio stream from said media player and outputs a resulting audio stream to said speech recognition program as a microphone input stream.

14. The invention according to Claim 13 further including means for automatically opening said speech recognition program and activating said changes.

15. The invention according to Claim 14 further including means for saving and restoring an original configuration of said mixer utility.

16. A method for improving the accuracy of a speech recognition program operating on a computer comprising:

(a) automatically converting a pre-recorded audio file into a written text;

(b) parsing the written text into segments;

(c) correcting each and every segments of the written text;

* C L M S *

(d) saving the corrected segment in a retrievable manner:

(e) saving speech files associated with a substantially corrected written text and used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program;

(f) establishing an independent instance of the written text from the pre-recorded audio file using the speech recognition program;

(g) replacing each segment in the independent instance of the written text with the corrected segment associated therewith;

(h) saving speech files associated with the independent instance of the written text used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program; and

(i) repeating steps (f) through (i) a predetermined number of times.

17. The invention according to claim 16 including saving said corrected segment in an individually retrievable manner in association with said computer.

18. The invention according to Claim 16 further comprising highlighting likely errors is said written text.

19. The invention according to Claim 18 wherein highlighting includes:

- comparing sequentially a copy of said written text with a second written text resulting in a sequential list of unmatched words culled from said copy of said written text, said sequential list having a beginning, an end and a current unmatched word, said current unmatched word being successively advanced from said beginning to said end;

- searching incrementally for said current unmatched word contemporaneously within a first buffer associated with the speech recognition program containing said written text and a second buffer associated with said sequential list; and

- correcting said current unmatched word in said second buffer. said correcting means including means for displaying said current unmatched



word in a manner substantially visually isolated from other text in said copy of said written text and means for playing a portion of said synchronized voice dictation recording from said first buffer associated with said current unmatched word.

20. The invention according to Claim 16 wherein said computer includes a ssound card and further comprises directing said pre-recorded audio file to said speech recognition program, wherein said speech recognition program does not accept said pre-recorded files, and said computer has a sound card.

21. The invention according to Claim 20 wherein directing said pre-recorded audio file to said speech recognition program includes:

(a) launching the speech recognition program to accept speech as if the speech recognition program were receiving live audio from a microphone;

(b) finding a mixer utility associated with the sound card the computer;

(c) opening the mixer utility, the mixer utility having settings that determine an input source and an output path;

(d) changing the settings of the mixer utility to specify a line-in input source and a wave-out output path;

(e) activating a microphone input of the speech recognition software; and

(f) initiating a media player associated with the computer to play the pre-recorded audio file into the line-in input source.

22. The invention according to Claim 21 further including changing the mixer utility settings to mute audio output to speakers associated with the computer.

23. The invention according to Claim 21 further including:

- saving the settings of the mixer utility before changing the settings of the mixer utility to specify a line-in input source and a wave-out output path; and

- restoring the saved sound card mixer settings after the media player finishes playing the pre-recorded audio file.

* C L M S *

24. A system for improving the accuracy of a speech recognition program operating on a computer. said system comprising:

- means for automatically converting a pre-recorded audio file into a written text;
- means for parsing said written text into segments;
- means for correcting each and every segment of said written text;
- means for saving said corrected segment in a retrievable manner in association with said computer;
- means for saving speech files associated with a substantially corrected written text and used by said speech recognition program towards improving accuracy in speech-to-text conversion by said speech recognition program; and
- means for repetitively establishing an independent instance of said written text from said pre-recorded audio file using said speech recognition program and for replacing each erroneous segment in said independent instance of said written text with said corrected segment associated therewith.

25. The invention according to Claim 24 wherein said parsing means includes means for directly accessing functions of said speech recognition program.

26. The invention according to Claim 25 wherein said parsing means further include means to determine the character count to the beginning of each of said segments and means to determine the character count to the end of each of said segments.

27. The invention according to Claim 26 wherein said means to determine the character count to the beginning of each of said segments includes UtteranceBegin function from the Dragon Naturally Speaking, and said means to determine the character count to the end of each of said segments includes UtteranceEnd function from the Dragon Naturally Speaking .

* C L M S *

28. The invention according to Claim 24 wherein said means for automatically converting includes means for directly accessing functions of said speech recognition program.

29. The invention according to Claim 28 wherein said means for automatically converting further includes TranscribeFile function of Dragon Naturally Speaking.

30. The invention according to Claim 24 wherein said correcting means further includes means for highlighting likely errors in said written text.

31. The invention according to Claim 30 wherein said written text is at least temporarily synchronized to said pre-recorded audio file, said highlighting means comprises:

- means for sequentially comparing a copy of said written text with a second written text resulting in a sequential list of unmatched words culled from said copy of said written text, said sequential list having a beginning, an end and a current unmatched word, said current unmatched word being successively advanced from said beginning to said end;
- means for incrementally searching for said current unmatched word contemporaneously within a first buffer associated with the speech recognition program containing said written text and a second buffer associated with said sequential list; and
- means for correcting said current unmatched word in said second buffer, said correcting means including means for displaying said current unmatched word in a manner substantially visually isolated from other text in said copy of said written text and means for playing a portion of said synchronized voice dictation recording from said first buffer associated with said current unmatched word.

32. The invention according to Claim 31 wherein said second written text is established by a second speech recognition program having at least one conversion variable different from said speech recognition program.



33. The invention according to Claim 31 wherein said second written text is established by one or more human beings.

34. The invention according to Claim 31 wherein said correcting means further includes means for alternatively viewing said current unmatched word in context within said copy of said written text.

35. A method for improving the accuracy of a speech recognition program operating on a computer comprising:

(a) automatically converting a pre-recorded audio file into a written text;

(b) parsing the written text into segments;

(c) correcting each and every segment of the written text;

(d) saving said corrected segment in a retrievable manner;

(e) saving speech files associated with a substantially corrected written text and used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program;

(f) establishing an independent instance of the written text from the pre-recorded audio file using the speech recognition program;

(g) replacing each erroneous segment in the independent instance of the written text with the corrected segment associated therewith;

(h) saving speech files associated with the independent instance of the written text used by the speech recognition program towards improving accuracy in speech-to-text conversion by the speech recognition program; and

(i) repeating steps (f) through (i) a predetermined number of times.

36. A method for directing a pre-recorded audio file to a speech recognition program that does not accept such files. the speech recognition program being stored on a computer that has a sound card, said method comprising:



(a) launching the speech recognition program to accept speech as if the speech recognition program were receiving live audio from a microphone;

(b) finding a mixer utility associated with the sound card;

(c) opening the mixer utility, the mixer utility having settings that determine an input source and an output path;

(d) changing the settings of the mixer utility to specify a line-in input source and a wave-out output path;

(e) activating a microphone input of the speech recognition software; and

(f) initiating a media player associated with the computer to play the pre-recorded audio file into the line-in input source.

37. The invention according to Claim 36 further including changing the mixer utility settings to mute audio output to speakers associated with the computer.

38. The invention according to Claim 36 further including:

- saving the settings of the mixer utility before changing the settings of the mixer utility to specify a line-in input source and a wave-out output path; and

- restoring the saved sound card mixer settings after the media player finishes playing the pre-recorded audio file.

39. A system for directing a pre-recorded audio file to a speech recognition program that does not accept such files, said system comprising:

- a computer having a sound card with an associated mixer utility; said computer executing said speech recognition software;

- a media player operably associated with said computer, said media player capable of playing said pre-recorded audio file; and

- means for changing settings of said associated mixer utility, such that said mixer utility receives an audio stream from said media player and outputs a resulting audio stream to said speech recognition program as a microphone input stream.

* C L M S *

40. The invention according to Claim 39 further including means for automatically opening said speech recognition program and activating said changes.

41. The invention according to Claim 40 further including means for saving and restoring an original configuration of said mixer utility.

42. The invention according to Claim 41 further including means for saving and restoring an original configuration of said mixer utility.

