Polish Speech Dictation System as an Application of Voice Interfaces

Grażyna Demenko[1,2], Robert Cecko[1], Marcin Szymański[1], Mariusz Owsianny[1], Piotr Francuzik[1], and Marek Lange[1]

[1] Poznań Supercomputing and Networking Center, Polish Academy of Sciences, Poznań, Poland
[2] The Institute of Linguistics, Adam Mickiewicz University, Poznań, Poland
{grazyna.demenko,marcin.szymanski,mariusz.owsianny, piotr.francuzik,marek.lange}@speechlabs.pl, cecko@man.poznan.pl

Abstract. This paper presents the results of the project realized at PSNC and supported by The Polish Ministry of Science and Higher Education – "Integrated system of automatic speech-to-text conversion based on linguistic modeling designed in the environment of the analysis and legal documentation workflow for the needs of homeland security", aiming at developing a Polish speech dictation (or Large Vocabulary Continuous Speech Recognition, LVCSR) system designed with the use of a phonetically controlled large vocabulary speech corpus and a large text corpora. The functions of the resulting system are outlined, the software architecture is presented briefly, then the example applications are demonstrated and the recognition results are discussed.

Keywords: speech recognition, dictation, voice interfaces.

1 Introduction

The objective of the speech-to-text project is the implementation of the speech recognition technology based on a very large corpora, especially for dedicated end-users in Poland – for the System of Justice, for the Police, the Border Guard and other services responsible for public security. The solution can be applied in making suit documentation, writing notes, protocols, recording inspections/post-mortem examinations, formal texts, legal documents, police reports, in dictating verdicts and verdict justifications, preparing stenographic records from meetings, sessions and materials from police operations, making summaries and in voice-controlled technical systems. The project is strictly correlated with the activity of the Polish Platform for Homeland Security within the area of creating innovative technology and computer tools to support law and public security institutions.

The structure of the remaining parts of the paper is as follows: Section 2 outlines the architecture of the system; Section 3 introduces the standalone software as well as example integration with a third-party application; in Section 4 the system accuracy is evaluated; the paper is concluded in Section 5, where possible future directions are also signaled.

A. Dziech and A. Czyżewski (Eds.): MCSS 2012, CCIS 287, pp. 68–76, 2012.

2 Features and Architecture

The Section 2.1 briefly lists the features and requirements of the dictation system. In Section 2.2 the overview of the system architecture is given.

2.1 Environment and Main Features

The present dictation system for Polish is designed to run on Microsoft Windows operating systems (XP or later). It was entirely written in C# and developed under the Microsoft .NET Framework 4.0 platform, with the use of Task Parallel Library (TPL). The system can work in two modes: off-line – the speech signal is read from a file, or on-line – the speech signal is acquired directly from an audio device (microphone). Depending on the quality/speed preset used (cf. Section 4), it is possible to obtain real-time recognition, which is understood as the behavior in which recognition is not slower than the playback of an utterance (the resulting sentence is presented with a small delay).

The recommended hardware resources for the software are: Intel Core i5 processor running Microsoft Windows 7 Pro 64-bit with 4GB of RAM. It is also possible to use the dictation system under 32-bit system with 2GB of RAM, at the expense of the lower quality of language modeling. Because of the intensive use of parallelism as well as the size of acoustic and linguistic models, the present system tends to use a maximum of hardware resources, with respect to both CPU and memory.

The dictation system can be used through one of two general interfaces: (1) as a standalone application (editor) that was designed to offer a maximum of features while remaining user-friendly and reasonably simple, thus hiding many internal details, or (2) through operating system integration, which allows to enter the recognized text directly into an active edit control of any third-party software. Section 3 gives more details.

In addition to its main feature which is, naturally, the textual presentation of a recorded sentence, the software also performs rudimentary text formatting which includes dates and numbers as well as capital letters and punctuation on sentence boundaries.

Since it is not possible for any speech recognition solution to contain all possible words that any user can utter, the system offers a possibility to add custom words.

It is well known that speech decoding is performed faster and more accurately once an original, speaker-independent acoustic model is tailored to a specific voice. For this reason, the presented dictation system also offers the speaker adaptation procedure which demands a target user to record ca. 250 fixed sentences.

2.2 System Architecture

The overview of the architecture of the Polish dictation system is presented in Fig. 1. It may be seen as one following general client-server architecture (although both sides

can, and usually do, run on the same physical machine). The client is responsible only for delivering the audio signal and then presenting the recognition result. The server detects the speech, performs the recognition and formatting. The following subsections describe the acoustic and linguistic parts of the server.

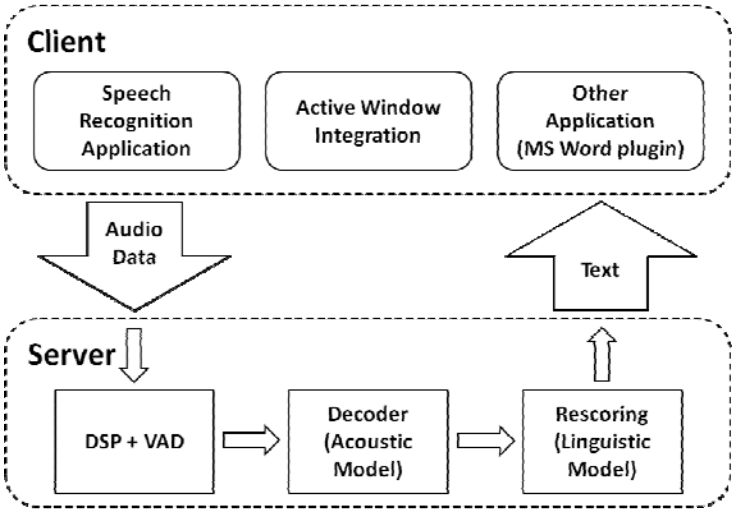

Fig. 1. Overview of Dictation System architecture

Voice Detections and Parameterization

The digital signal processing (DSP) and Voice Activity Detection (VAD) stage is presented in Fig. 2. The raw PCM audio signal from the client is passed to the DSP module, which divides it into separate observations (25 ms window with 10 ms stepping) and calculates the Mel-frequency cepstral coefficients (MFCC). The special LDA (Linear Discriminant Analysis) transformation is computed over those parameters in order to obtain VAD Parameters which are subsequently compared to statistical speech/silence/noise distributions held in VAD Acoustic Model. Once the Voice Activity Detection module determines the utterance boundaries, the list of the former parameters (i.e. MFCC) are passed to the next stage.

Decoding and Rescoring

The actual recognition stage, presented in Fig. 3, consists of two main operations: decoding and rescoring.

The Decoder is built upon Viterbi algorithm [1] (with some modifications, mainly to allow more effective parallel processing) and works over a Recognition Network being a word-loop of ca. 320-thousand dictionary entries with imposed unigram log-probabilities. As a result, it produces recognition hypotheses in form of the Word

Lattice. The decoder is also the module that uses the Acoustic Model, which is the collection of statistical distributions of Polish triphones' acoustic features. The model was trained over the material selected from the Jurisdict database designed specifically for the present dictation system whose target users are judges, lawyers, policemen and other public officers. The aforementioned database is a phonetically controlled large vocabulary corpus and contains recordings of speech delivered in quiet office environments by over 2000 speakers (a total of over 1155 hours of speech) from 16 regions of Poland.

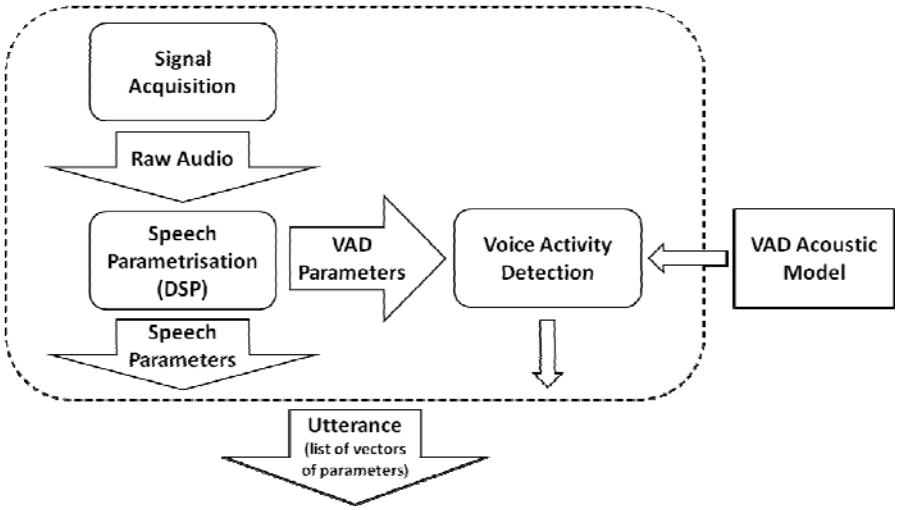

Fig. 2. Voice detection and speech parameterization stage

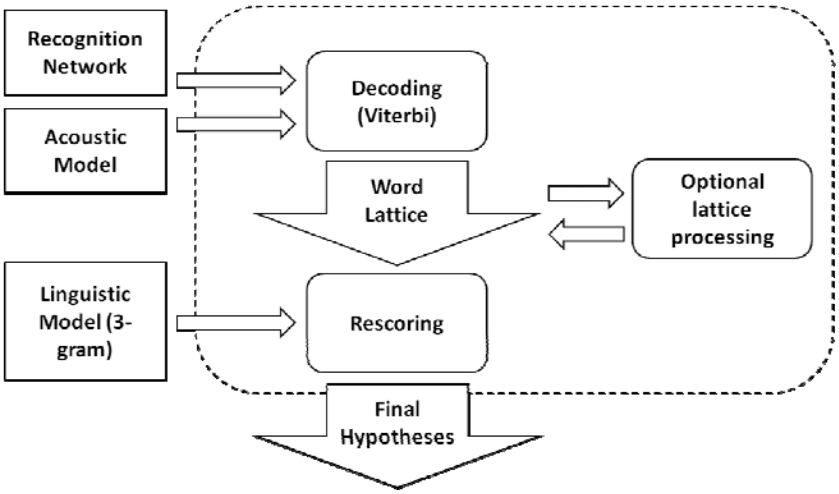

Fig. 3. Speech decoding and hypotheses rescoring

The Acoustic Model was estimated using HTK [2], with the standard training procedure for Triphone Continuous Density Hidden Markov Model. See [3-5] for more details on database design and [4,5] for details on Acoustic Model training.

The Word Lattice elements are attributed with appropriate acoustic probabilities. All lattice hypotheses are then evaluated using N-Gram Linguistic Model in the Rescoring module (see eg. [6]). The hypothesis with the best probability is returned as the recognized text. The Linguistic Model is currently an interpolation of two back-off trigram models (i.e. models that assign a word probability based a word in question and two preceding words) built with SRILM toolkit [7], estimated on over 4GB of automatically normalized text, including newspaper articles and legal-domain texts (judgments, law acts, briefs, contracts etc.).

Optional lattice processing module, presented in the Fig. 3, stands for the operation in which non-linguistic events are removed from the Word Lattice.

3 Applications

As already stated, the dictation system can be used through one of two general interfaces.

The standalone application, presented in Fig. 4, is software resembling a simple editor. It allows user to: select the input device (a microphone) for the on-line or disk audio file for the off-line recognition, to record a sentence (or a longer section of speech) and to play it back, to recognize an utterance on one of the 7 possible quality/speed presets and to manually correct the output sentence. The manual correction can be performed by, naturally, typing actually-spoken words, or, more importantly, by selecting one of alternative recognition hypotheses which are available within the context menu for most of output tokens. The recognition speed is presented in the status bar (both nominal and in terms of a real-time ratio); also, for testing purposes, it is possible to load a reference file (i.e. a text file with the proper orthographic transcription), that allows to calculate the recognition accuracy.

The Fig. 5 demonstrates the other interface, which is the operating system integration. This grants the possibility to enter the recognized text directly into an active edit control of any third-party software. The actual application presented in Fig. 5 is e-Posterunek (Polish for “e-PoliceStation”). The dictation system integration widget can be seen in the bottom-right corner.

However, the latter case does not allow to use alternative hypotheses unless a special plug-in is implemented (such a plug-in is being prepared for Microsoft Word).

4 Recognition Results

The speech recognition evaluation is commonly based on a recognition accuracy that is the ratio of the number of correctly recognized words minus the number of inserted words to the total number of words in the reference (perfect recognition result).

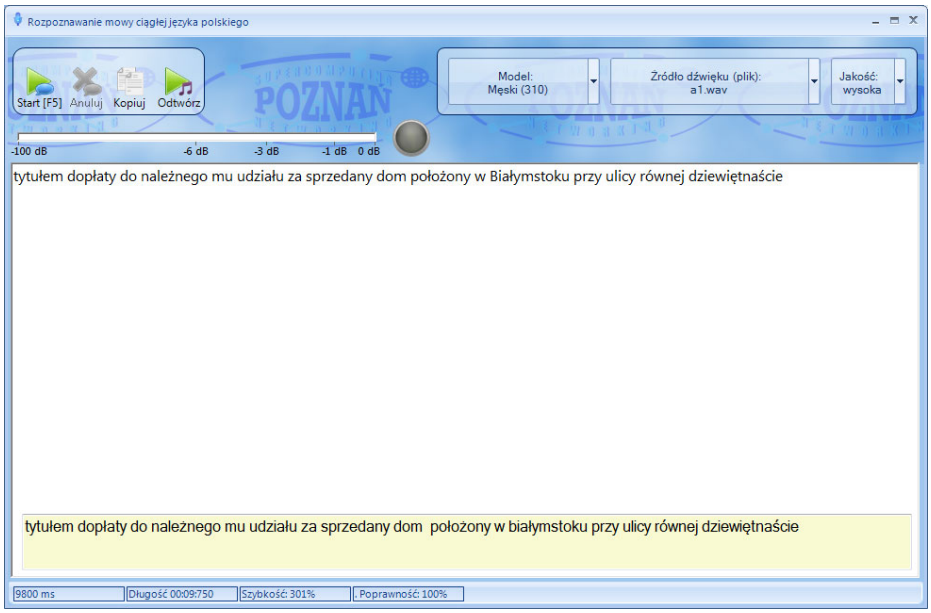

Fig. 4. Standalone speech dictation application. See Sec. 3 for description.

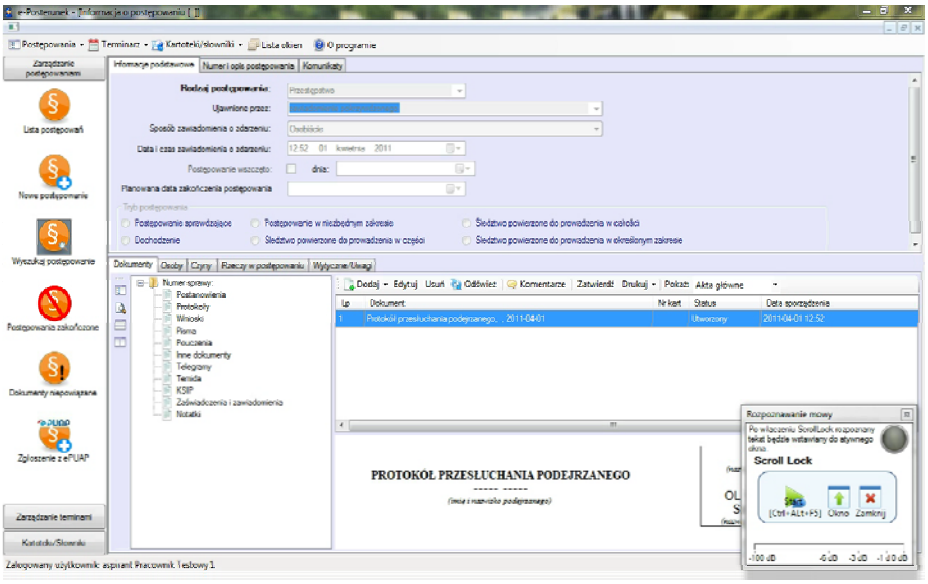

Fig. 5. Speech dictation with third-party software, e-Posterunek (e-PoliceStation), as an example of active window integration. Speech dictation widget is visible in the lower-right corner.

Table 1. Recognition results for 97-speaker testing set. **Acc** is the mean percentage of correctly recognized words minus inserted words, **T** is the recognition time compared to recording duration (ie. playback time; 100% is the real-time recognition). Quality level depends on internal decoder parameters. The evaluation was run on Intel® Core Duo E8400 3.00GHz, 4GB RAM.

Quality level	Acc[%]	T[%]
The highest	88,7	745,14
Higher	88,1	370,42
High	87,3	197,5
Mean	**86**	**117,1**
Low	84,1	74,95
Lower	82,2	56,9
The lowest	79,6	45,42

Here, we show the recognition accuracy for two distinct test corpora. The Table 1 presents the speaker-independent system performance for a 6-hour set with 97 speakers. The Table 2 presents the results for the 3-hour set with 13 speakers. In the latter case, however, the subjects separately recorded the adaptation sets, which allowed the evaluation of the speaker-dependent scenario. The accuracies and the recognition times are presented separately for each of the 7 quality/speed presets. The accuracy figures have been calculated using the Sclite tool [8].

Table 2. Recognition results for 13-speaker testing set, speaker-independent (left) and speaker-dependent (i.e. adapted, right). Cf. Table 1 for the explanation of captions.

Quality level	without adaptation		with adaptation	
	Acc[%]	T[%]	Acc[%]	T[%]
The highest	90,7	500,09	93	297,08
Higher	90,4	254,89	92,6	161,06
High	89,7	145,51	92,2	97,6
Mean	**87,9**	**82,83**	**90,8**	**52,72**
Low	85,6	56,95	89,2	39,51
Lower	83,5	45,6	88	33,17
The lowest	81,2	37,66	85	28,2

It can be observed that, under the selected quality/speed presets which reflect the internal decoder parameters (heuristic pruning deltas), the system accuracy tends to saturate at ca. 91% in case of non-adapted acoustic models and at ca. 93% in case of adapted-speaker models with ever-increasing recognition time. Conversely, the recognition time tends to saturate at the lower presets with ever-decreasing accuracy. The real-time recognition can be obtained with an accuracy of ca. 85%-88.5%

in case of speaker-independent and over 92% in case of speaker-dependent models. The speaker adaptation yields up to 30% of the reduction of error (adaptation method implemented within the software is the MLLR[9]).

5 Discussion and Future Development

The recognition accuracy and the general software behavior are very promising, making it an already serious candidate for the widespread release. The system is currently under intensive evaluation by a group of Police officers, with a positive overall feedback; also, training and evaluation campaigns are planned within selected lawyers' office courts, Border Guard and other services responsible for public security.

Moore[10] states that a 1000-hour database allows for building a system with a word error rate of ca. 12% when language modeling is applied; he also estimates that at least 100 000 hours of speech is needed to train an ASR system with accuracy comparable to that of a human listener. The system presented in this paper is already showing accuracy close to the former theoretical figure. However, the authors feel that some improvements can still be done, especially in terms of the recognition accuracy (that, ideally, should be improved without sacrificing speed, which is already acceptable in most presets). Some ideas include: tuning of some heuristics used by the decoder, or, possibly, re-allocation of the decoder (or some parts of it) to GPU.

From the functional point of view, the client-server architecture makes it possible to develop a speech recognition service, which would allow a remote (Internet) access, handling many recognition sessions at the same time, both live (on-line) and batch (off-line), possibly with many different acoustic models at the same time.

Acknowledgements. Work supported by grant "Integrated system of automatic speech-to-text conversion based on linguistic modeling designed in the environment of the analysis and legal documentation workflow for the needs of homeland security" (OR 00006707). The authors are currently supported by grant "Collecting and processing of the verbal information in military systems for crime and terrorism prevention and control." (OR 00017012).

References

1. Rabiner, L.R.: A tutorial on Hidden Markov Models and selected applications in speech recognition. Proc. of the IEEE 77(2), 257–286 (1989)
2. Young, S., et al.: The HTK Book (for HTK Version 3.2), Cambridge University Engineering Department (2002)
3. Klessa, K., Demenko, G.: Structure and Annotation of Polish LVCSR Speech Database. In: Proc. of Interspeech, Brighton UK, pp. 1815–1818 (2009)
4. Szymański, M., Klessa, K., Lange, M., Rapp, B., Grocholewski, S., Demenko, G.: Development of acoustic models for the needs of a speech recognition system using large lexical databases. Best Practices - Nauka w obliczu społeczeństwa Cyfrowego, Poznań (2010)

5. Demenko, G., Szymański, M., Cecko, R., Lange, M., Klessa, K., Owsianny, M.: Development of Large Vocabulary Continuous Speech Recognition using phonetically structured speech corpus. In: Proc. Intl. Congress of Phonetic Sciences, Hong Kong (2011)
6. Kneser, R., Ney, H.: Improved backing-off for M-gram language modeling. In: Proc. ICASSP, Detroit, vol. 1, pp. 181–184 (1995)
7. Stolcke, A.: SRILM - An Extensible Language Modeling Toolkit. In: Proc. Intl. Conf. Spoken Language Processing, Denver (2001)
8. Sclite tool kit on-line documentation, `http://www.itl.nist.gov/iad/mig/tools/`
9. Leggetter, C.J., Woodland, P.: Speaker Adaptation of Continuous Density HMMs Using Multivariate Linear Regression. In: Proceedings of ICSLP 1994, Yokohama, Japan (1994)
10. Moore, R.K.: A comparison of the data requirements of automatic speech recognition systems and human listeners. In: Proc. Eurospeech, Geneva (2003)