INTERSPEECH 2021: Show & Tell Contribution
30 August – 3 September, 2021, Brno, Czechia

Beey: More than a Speech-to-Text Editor

Lenka Weingartová, Veronika Volná and Ewa Balejová

NEWTON Technologies, Prague, Czech Republic
{lenka.weingartova, veronika.volna, ewa.balejova}@newtontech.cz

Abstract

We present Beey, a newly developed web-based multimedia platform for producing Automatic Speech Recognition (ASR) and editing its output. In addition to ASR, Beey employs modules for speaker diarization and identification, text formatting, automatic punctuation insertion, subtitling, automatic translation, transcription of stream and more.

The platform and its development are focused on user experience and fast document creation. Our aim is to transfer research results in the field of speech recognition and signal processing into practice and enable Beey's users to make their production processes faster and cheaper by minimizing human effort and costs.

Index Terms: speech recognition, text editing, subtitling

1. Motivation

It is a long road from the raw output of a Speech-to-Text or Automatic Speech Recognition (ASR) engine to a text which is comfortably readable by a human user. While for speech recognition researchers, Word Error Rate (WER) is often the most important attribute of the recognized text, in production applications, the actual content and its form is what matters the most.

Currently, there is a number of different use cases for ASR output, such as subtitling, media monitoring, various speech analytics, dictating systems, etc. Moreover, European state administration and other official bodies are required to publish transcripts from their meetings and sessions. Only a minimum of those applications can employ raw ASR output, most of them require further processing, which is predominantly manual error correction.

The general public, which may have very little to no experience with ASR, often has a hard time accepting that the ASR output is never going to be 100% correct (WER = 0). In most production environments, ASR accuracy generally oscillates between 75 - 90%, lower values being only of little use in practice, higher values requiring specific conditions, such as studio-quality acoustic input or a general vocabulary. But even 90% accuracy means that on average, every 10^{th} word requires correction - that would amount to one error on every line in this article. In addition, a significant number of corrections is required for punctuation as well, even if the particular ASR system has an automatic punctuation feature. Many production cases also require inserting speaker information, non-speech events or other metadata not included in the transcription.

All this makes raw ASR output highly impractical without a user interface which enables users to correct the output in alignment with sound, to format it, to insert speaker names and other information not included in speech. Every correction that a human editor needs to make is translated into time that they spend working, which in turn translates into additional costs.

The motivation for developing Beey (beey.io), a web-based multimedia platform for editing transcriptions, was to reduce human effort as much as possible, to speed up corrections, and to enable the use of ASR output for most use cases, such as subtitling or creating verbatim transcripts for various purposes.

2. Platform overview

Beey's user interface enables users to proofread and edit the text as soon as the first words are recognized. Not having to wait until the entire transcription is complete saves a great deal of time and resources, especially in those use cases where speed is of the essence.

The majority of Beey's features were and continue to be developed in collaboration with users who work with the platform daily. The result is a modern, user-centered system which anticipates user needs and offers solutions to accelerate their workflow and therefore reduce costs.

The user is able to make all changes with the help of keyboard shortcuts and without having to use the mouse. An example of this is word uppercasing, which can be done with a single keyboard shortcut, regardless of the cursor position in the word. Some of the shortcuts are language-specific, so for instance in Czech, there is a possibility to correct common misspellings regarding the letters i/y which have an identical phonetic realization. Moreover, there are some automatic corrections built in, e.g. uppercasing the following word when inserting sentence punctuation.

During the session, full Beey functionality will be demonstrated. Participants will be given demo accounts to test the platform on their own computers, and sample multimedia files will be provided.

2.1. Main features

Beey's basic features include:

- **Automatic transcriptions of recordings.** We use the ASR system developed by the SpeechLab team at the Technical University in Liberec, see [1], [2] and [3]. It currently supports 19 languages and includes an x-vector-based voice activity detection algorithm [4], [5] that filters out non-speech (music, noise, etc.). The transcribed text appears in the editor window continuously, without needing to wait until the entire transcription process is finished. The platform supports all standard audio and video formats.
- **Editing while transcription is running.** The user is able to edit the transcribed text from the moment the first words appear on the screen.

- **Sound-to-text alignment with the help of timestamps.** This means that users are able to listen to the exact part of the text that they are currently editing. Sound playback is triggered by keyboard shortcuts. Users are also able to adjust timestamps using shortcuts.
- **Speaker diarization.** Beey automatically separates paragraphs that are uttered by different speakers using convolutional neural network-based algorithms [6]. Repeated speaker occurrences are clustered and can be edited all at once.
- **Speaker database.** Each user is able to add a new speaker (using name and role) and save the entry to a shared database.
- **Speaker identification.** We are creating our own speaker voiceprint database, which enables us to identify individual speakers. Speaker identification algorithms are x-vector-based, language- and content-independent [7].
- **Punctuation insertion.** Beey also includes an algorithm based on recurrent neural networks with long short-term memory, which inserts commas, full stops and question marks into the ASR output [8]. We currently support this feature in 4 languages, and the development of others is in progress. The remaining languages use a simpler system based on context and rules for pauses, non-speech events, speaker changes, etc.
- **User lexicon.** Beey includes the option of adding user words which are not contained in the recognizer lexicon and therefore would never be recognized correctly. Their pronunciation can be generated automatically by a grapheme-to-phoneme algorithm or entered manually with the use of a language-specific pronunciation alphabet.
- **Sharing with multiple users.** A project (media file plus transcription) can be shared with other Beey users and successively edited by different users.
- **Document export.** A finished, corrected, and edited transcription may be exported in a number of various formats, including html, pdf, docx or xml with timestamps.
- **Full-text search** in transcriptions based on Elasticsearch.
- **Application Programming Interface (API).** All of Beey's features are also accessible through REST API.

More features, such as concurrent editing by different users, or user workspaces, which will enable advanced user administration and project sharing in groups, are currently being developed. Two of the most innovative features in Beey will be outlined in more detail in the following sections.

2.2. Subtitle mode

Based on increasingly frequent requests from clients, we have decided to include a new subtitle mode (currently in beta) that enables fast creation of subtitles and captions. According to user specifications (characters per line), an optimization algorithm divides the transcribed text into individual subtitles, while preserving the format of a continuous text for better readability. Language-specific rules for line and subtitle breaks are incorporated, e.g. no articles or prepositions at the end of a line, breaking at sentence- or clause-ends whenever possible, etc. Warnings about too short/long or fast subtitles are expressed by color highlighting.

Dividing, combining or recalculating timestamps of subtitles can again be done with the help of keyboard shortcuts. This is in stark contrast with commonly used subtitle editors, which do not work with ASR outputs directly and require significantly more user effort when editing individual subtitles.

2.3. Extension apps

Various use cases required by a wide range of users often call for more than basic functionality. Beey can be tailored to the needs of individual clients. It has a number of additional functions which are not part of its GUI but can be accessed as extensions. Two examples of these extensions are the Translate and the Stream apps.

The Translate app uses automatic translation to convert a project into another language. Currently, Google Translate and DeepL are integrated. The translated project can be once again opened in Beey editor and corrected or changed as needed.

When combined with the subtitle mode, this allows for very fast semi-automatic multi-language subtitle production with human corrections. During the translation process, timestamps are anchored at sentence level - this ensures that translated subtitles are aligned with the original utterances.

The Stream app is designed for continuous stream transcription in use cases such as Parliament minutes or media monitoring. After the user enters a URL, start time and duration, Beey automatically starts transcribing the stream at the specified time. While transcribing, full Beey functionality can be employed to edit the text.

3. Acknowledgements

The development of Beey was supported by the Technology Agency of the Czech Republic, project no. FW01010468.

4. References

[1] Šafařík, R., Nouza, J.: "Unified Approach to Development of ASR Systems for East Slavic Languages," in *Proc. of SLSP 2017*, Springer, LNCS, Vol. 1058, pp. 193-203, 2017.

[2] Nouza, J., Šafařík, R., Červa, P.: "ASR for South Slavic Languages Developed in Almost Automated Way," in *Proc. INTERSPEECH 2016*, pp. 3868-3872, 2016.

[3] Málek, J., Žďánský, J., Červa, P.: "Robust Recognition of Speech with Background Music in Acoustically Under-Resourced Scenarios," in *Proc. ICASSP 2018*, pp. 5624-5628, 2018.

[4] Matějů, L., Červa, P., Žďánský, J., Málek, J.: "Speech Activity Detection in online broadcast transcription using Deep Neural Networks and Weighted Finite State Transducers," in *Proc. ICASSP 2017*, pp. 5460-5464, 2017.

[5] Matějů, L., Kynych, F., Červa, P., Žďánský, J., Málek, J.: "Using X-vectors for Speech Activity Detection in Broadcast Streams," in *Proc. INTERSPEECH 2021*. Submitted.

[6] Matějů, L., Červa, P., Žďánský, J.: "An Approach to Online Speaker Change Point Detection Using DNNs and WFSTs," in *Proc. INTERSPEECH 2019,* pp. 649-653, 2019.

[7] Málek, J., Janský, J., Kounovský, T., Koldovský, Z., Žďánský, J.: "Blind Extraction of Moving Audio Source in a Challenging Environment Supported by Speaker Identification Via X-Vectors," in *Proc. ICASSP 2021*, pp. 226-230, 2021.

[8] Hlubík P., Španěl M., Boháč M., Weingartová L.: "Inserting Punctuation to ASR Output in a Real-Time Production Environment," in Sojka P., Kopeček I., Pala K., Horák A. (eds): *Text, Speech, and Dialogue. TSD 2020. Lecture Notes in Computer Science*, vol 12284, pp. 418-425, 2020.