June 2020

Boosted Audio Recall for Music Matching

Hanna Maria Pasula

Martin Seiler

Boosted Audio Recall for Music Matching

ABSTRACT

Music is frequently modified from an original version before uploading to music-sharing or video-sharing websites or social media networks. For example, the music can be remixed, have voice-overs added, or have other edits. In several use cases, e.g., search, deduplication, ensuring fair use, etc. it is of interest to determine if an uploaded audio track is substantially similar to existing audio tracks in a database. However, modifications made to the original version can in some instances be enough that a match is not obtained with any track in the reference database even when the tracks match substantially. This disclosure describes neural network based techniques to ignore modifications, e.g., voice-overs, from an audio track such that a match, if any, with a reference audio track is easier to detect.

KEYWORDS

- Music matching
- Media matching
- Content identification
- Content matching
- Plagiarism detection
- Audio recall
- Triplet loss
- Fair use
- Voice-over

BACKGROUND

Music is frequently modified from an original version before uploading to music-sharing or video-sharing websites or social media networks. For example, the music can be remixed; have voice-overs added; modified in speed, pitch, or instrument ensemble; mixed with distracting sounds (e.g., a person talking); or have other edits.

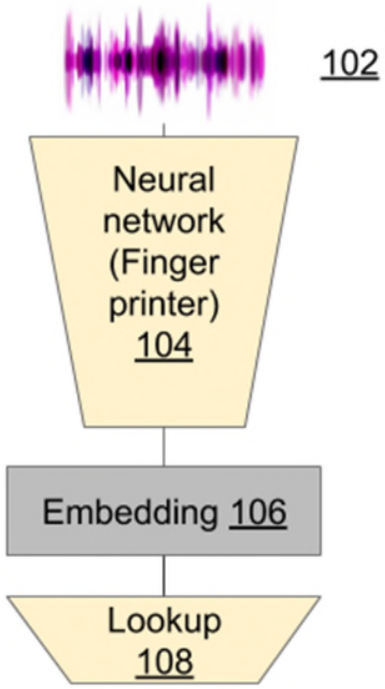

Fig. 1: Using a neural network to generate an embedding that enables matching a probe audio track to audio tracks in a database

In several use cases, e.g., search, deduplication, ensuring fair use, etc. it is of interest to determine if an uploaded audio track is substantially similar to existing audio tracks in a database. Fig. 1 illustrates using a neural network to generate an embedding that enables matching an uploaded (probe) audio track to audio tracks in a database.

A probe audio track (102), e.g., in the form of a spectrogram, is fed to a neural network (104), which can be, e.g., a convolutional neural network. The neural network generates from the probe audio track an embedding (106), which is a compact and robust representation, e.g., a 128-

entry vector, of the audio track. The embedding is used to look up (108) a database of audio tracks. Because the neural network effectively characterizes the audio track by an embedding, it is referred to as an (audio) fingerprinter.

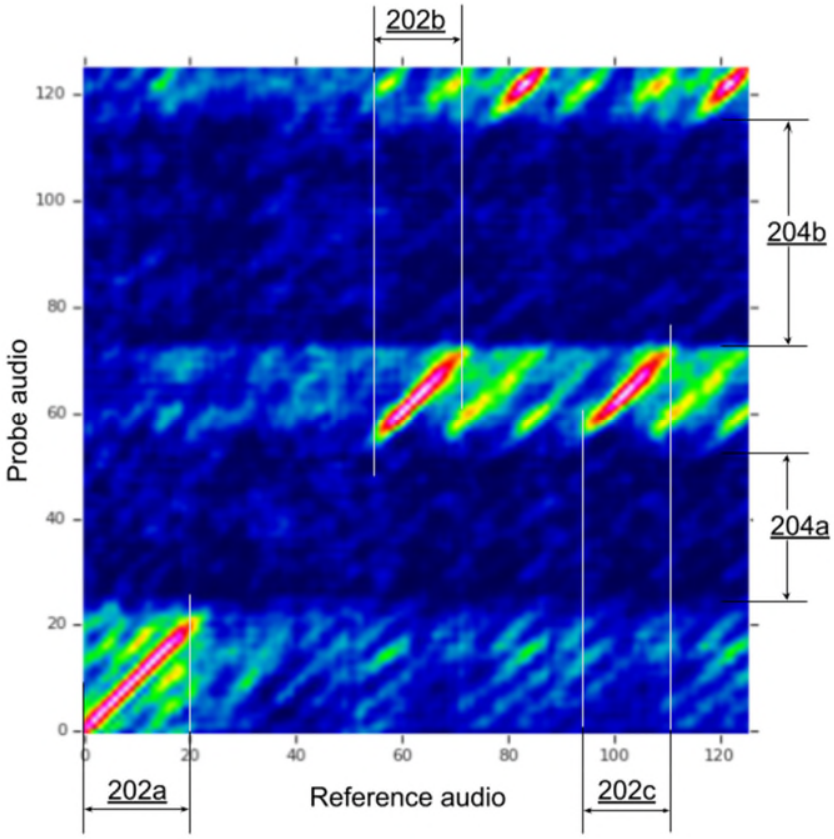

Fig. 2: Heatmap of correlation between probe and reference audio tracks

Fig. 2 illustrates an example heatmap of correlation between a probe audio track and an example matched reference audio track in the database. The blue regions indicate time spans of low correlation, while green, red, and white regions represent time spans of progressively higher correlation.

From the figure, it can be seen that the probe and the reference audio have some correlation between zero and twenty seconds (202a) and between fifty-five and seventy seconds (202b). The probe audio between ninety-five and one hundred and ten seconds also bears

resemblance to the reference audio between fifty-five and seventy seconds (202c), possibly indicating the presence of a chorus or repetition of the main melody/theme reverberation in the probe.

However, there are large patches (204a-b) where there is apparently no correlation between the reference and the probe audio tracks. The lack of a continuously bright (red- or white-hot) correlation line between reference and probe makes it difficult to conclude with certainty that the reference and the probe match each other. Indeed, the probe may actually match the reference in its entirety, but, in the blue patches, the probe may simply be dominated by noise-like signals that are intentionally or unintentionally mixed in.

DESCRIPTION

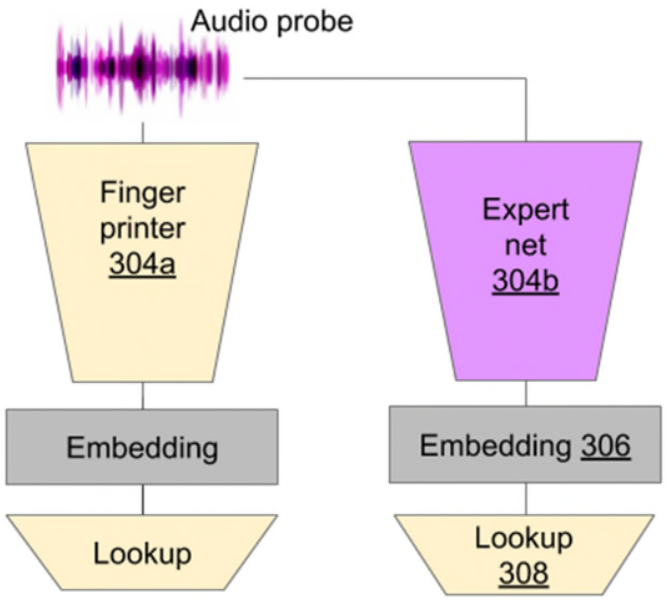

Fig. 3: Boosted audio recall using an expert neural network

Fig. 3 illustrates techniques that enable the matching of an audio probe to a probe in a database of reference probes, where the audio probe may be a substantially corrupted or modified version of a reference probe. Per the techniques, a fingerprinter neural network (304a)

produces an embedding of the audio probe conventionally. In addition, an auxiliary neural network, known as an expert net (304b), is trained to home into the musical content of the audio probe, effectively canceling out, or ignoring, disturbances that may be present in the audio probe.

The expert net generates an embedding (306) that is in the same vector space as the embedding generated by the fingerprinter neural network. The embedding generated by the expert net is looked up (308), or matched, against the same database of reference audio tracks used by the fingerprinter neural network.

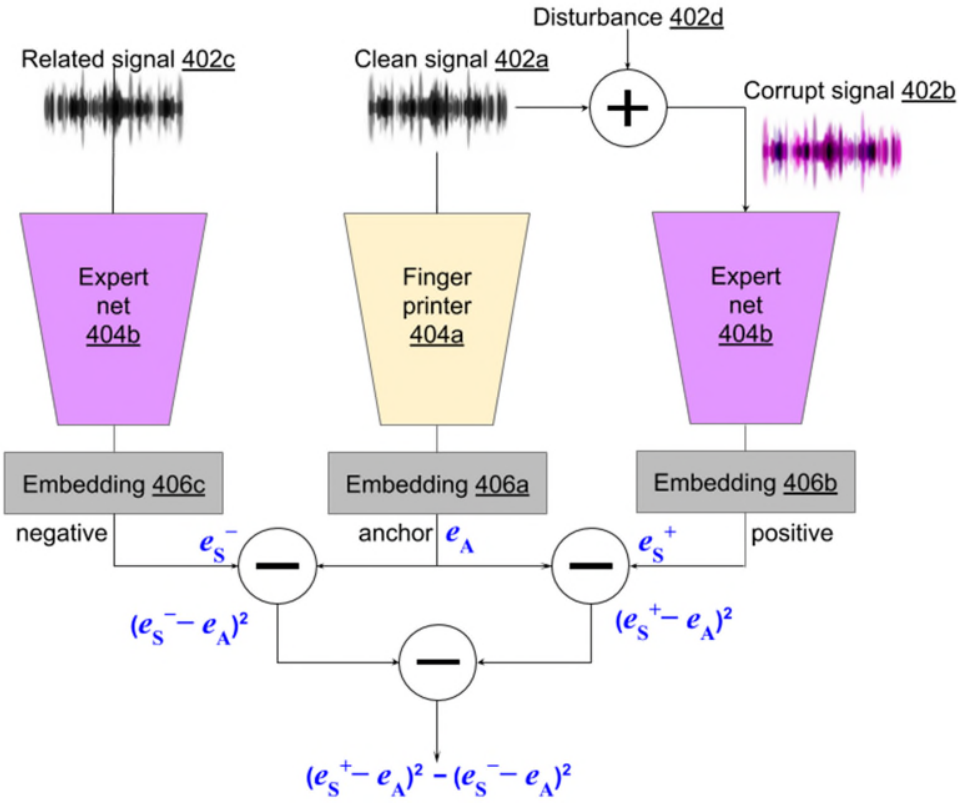

Fig. 4: Training the expert net

Fig. 4 illustrates training the expert net, per techniques of this disclosure. As explained before, the expert net is trained to isolate and create an embedding of just the musical content of the provided input. The expert net is trained using the principle of triplet loss, which minimizes

the distance between the output of the network and a positive example, and maximizes the distance between the output of the network and a negative example.

During training, a clean signal (402a) is fingerprinted using a fingerprinter (404a) to generate an embedding (406a) known as an anchor embedding e_A. The clean signal is mixed with a disturbance (402d), e.g., a voice-over, and the resulting corrupted signal (402b) is fed to the expert net (404b) to generate an embedding (406b) known as a positive embedding e^+_S. In general, the corrupted signal can be generated as a transformation of the clean signal.

A related signal (402c), described in greater detail below, is fed to the expert net (404b) to generate an embedding (406c) known as a negative embedding e^-_S (for ease of illustration, two blocks that are the same expert net are shown). To minimize the distance between the positive embedding and the anchor, and to maximize the distance between the negative embedding and the anchor, the expert net is trained to minimize the cost function

$$(e_S^+ - e_A)^2 - (e_S^- - e_A)^2$$

The purpose of the related signal is to provide a negative training example. The related signal can be chosen as an audio track that is somewhat similar to the clean signal, but not quite. For example, if the expert net is being trained to isolate musical content that is being dominated by speech, then the clean signal is the musical content, the disturbance is the speech, and a related signal can be music that includes a lot of vocals, e.g., singing. By providing music with singing as a negative example, the expert net is trained to *not filter out* singing voices.

Triplet loss enables the expert net to focus on a subspace of the embedding space if necessary. For example, to isolate musical content, the expert net can home into the rhythm of the musical instruments and ignore dimensions that correspond to human voices. Effectively,

using triplet loss is tantamount to achieving a short distance between the anchor and the positive example, with deviations allowed so long as no other piece of music is closer.

Although the figure illustrates training data being synthetically generated, e.g., the disturbance being mixed with the clean signal to generate a corrupt signal, the training data can also include naturally occurring signals. For example, the audio feed of a video or movie that includes sections of pure music and music mixed with voice-over can serve as a training signal, where the pure music section serves as a clean signal and the music mixed with voice-over serves as the corrupted signal.

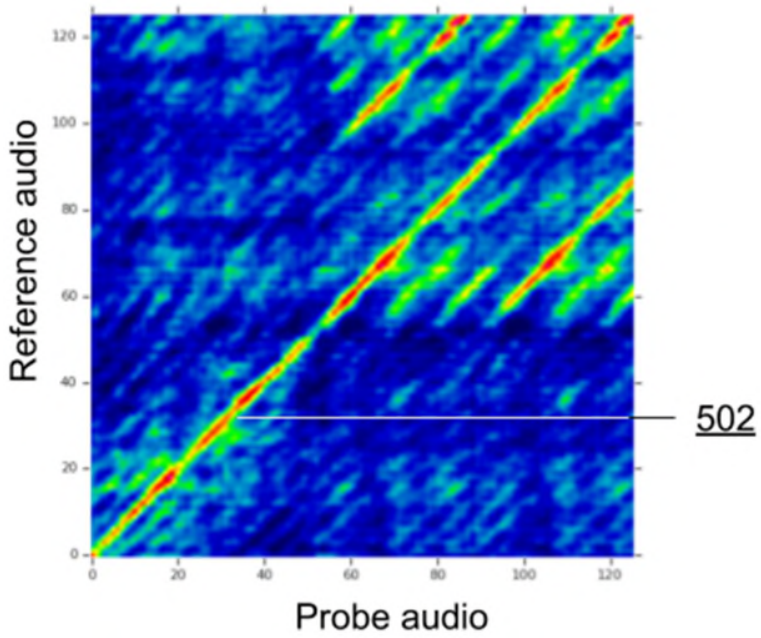

Fig. 5: Heatmap of correlation between a reference audio track and a probe audio after voice-over removal by the expert net

Fig. 5 illustrates an example heatmap of correlation between a reference audio track and a probe audio after voice-over removal by the expert net. In contrast to Fig. 2, a continuous, bright correlation (502) is clearly visible for the duration of the probe. In other words, the expert net has so isolated the music content of the probe that the presence of voice-overs in the probe does not

interfere with the matching of its music content with a reference audio track. Despite the disturbances in the probe audio, the expert net has helped establish that music content was continuously present.

Although the narrative here has assumed that the musical content of an audio track is the clean copy and the voice-over the disturbance, it is possible that it is the voice-over that is the clean copy and the music the disturbance. This can happen, e.g., if an interview is contaminated by background music. By appropriately training the expert net, the case where voice is the clean copy and music the disturbance is just as easily handled. Similarly, expert nets can be trained that specialize in particular types of deviations from the clean signal.

The described techniques can be used for content matching or content identification to determine whether a new audio track, e.g., uploaded to a music or video-sharing service, used as a podcast, etc. matches substantially with a prior audio track. The techniques are robust to the presence of various types of edits to the original audio.

CONCLUSION

Music is frequently modified from an original version before uploading to music-sharing or video-sharing websites or social media networks. For example, the music can be remixed, have voice-overs added, or have other edits. In several use cases, e.g., search, deduplication, ensuring fair use, etc. it is of interest to determine if an uploaded audio track is substantially similar to existing audio tracks in a database. However, modifications made to the original version can in some instances be enough that a match is not obtained with any track in the reference database even when the tracks match substantially. This disclosure describes neural network based techniques to ignore modifications, e.g., voice-overs, from an audio track such that a match, if any, with a reference audio track is easier to detect.

REFERENCES

1. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." In *Advances in neural information processing systems*, pp. 1097-1105. 2012.
2. Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "FaceNet: a unified embedding for face recognition and clustering (2015)." *arXiv preprint arXiv:1503.03832* (2015).