Exploring the Realism of AI-Generated Onomatopoeia: An Interactive Demo

Hein Christoph, 125356
christoph.peter.hein@uni-weimar.de

ABSTRACT

This paper explores the comparison of human and AI-generated onomatopoeia within an interactive demo exhibited at 'summaery' 2024 at Bauhaus university Weimar. The project investigates the ability of deep learning models, specifically text-to-speech (TTS) systems, to produce natural-sounding onomatopoeia, and evaluates the human ability to distinguish these from real human-made sounds. Using open-source datasets and models such as AudioLDM2 and ElevenLabs' monolingual model, the demo involved participants listening to and voting on various sounds categorised as farm animals, instruments and domestic noises. The results show a significant challenge for participants in accurately identifying AI-generated sounds, with a near 50/50 split between correct and incorrect votes, highlighting both the advances in AI audio generation and the limitations of human recognition capabilities.

1 Introduction

As part of the course "Fake" the project focuses on the comparison of human and AI made onomatopoeia within a framework of an application suited for the "summaery" 2024. In this paper the focus lies on explaining the idea and techniques used as well as to discuss some results gained by the exhibiting and user interactions with the demo.

Deep learning models have become dominant in many areas of applied machine learning. Text-to-speech (TTS), the process of synthesising artificial speech from a text prompt, is no exception. Deep models that produce more natural-sounding speech than traditional sequential approaches began to emerge in 2016. Much of the research since then has focused on making these deep models more efficient, sounding more natural, or training them in an end-to-end fashion. OpenAI at this point is developing for example their product called voice engine designed for generating high-quality, natural-sounding synthetic voices. Developed since late 2022, it powers applications such as ChatGPT's Voice Mode and a text-to-speech API, offering features like emotion and intonation recognition, multiple languages and accents, and high-quality audio output (OpenAI, 2024). The idea to focus on onomatopoeia and the comparison of human to AI made noises imitating different things assumes that when replicating human voices or producing an artificial voice it lacks characteristic which can be then identified by humans listening to the end-product (cf. Inamdar et al, 2023, Shi, 2023). Onomatopoeia introduces a rich variety of sounds that might help the model better understand and reproduce nuanced audio patterns. This variety and nuances make onomatopoeia a suitable alternative to other audio snippets which could be used to filter artificial voices from real ones. Additionally, the incorporation of the mimicked sounds has a higher appeal to visitors and are more inviting to experiment and therefore more fitting for the purpose of exhibiting. As the used datasets are opensource available it can be assumed that the imitations have already been incorporated into the learning data set by LLM. Therefore, this demo aims for identifying to which extent onomatopoeia can be produced from the available TTS models and if test persons will identify these in a listening scenario more often than not.

Even though the aforementioned data was used in the training, there are still possibilities that fine tuning existing models with onomatopoeia can lead to more expressive and realistic synthetic voices, improving applications in entertainment, education, accessibility, and more. For instance, this training could improve how the model conveys emotions and sound effects, making interactions with AI more engaging and lifelike.

An additional area which could be improved is the amount of data which is needed to produce reliable voice clones. "*Voice cloning models require large datasets (thousands of hours) of high-quality recordings in order to learn the unique characteristics of a person's voice. However, obtaining such datasets can be time-consuming and expensive, especially to clone individuals with no available data*" (Espinosa, 2023). When using onomatopoeia efficiently and its unique attributes the dataset could get significantly smaller.

2 Related Work

This section focuses on similar scientific work that evaluates the identification of AI-generated sounds, for example in a Turing test-like manner, and the development and training of LLM with onomatopoeia.

Researchers envision future versions of algorithms being used to automatically produce sound effects for movies and TV shows, as well as to help robots better understand objects' properties. In their paper "Visually indicated sounds" Owens et al., 2016 use deep learning techniques to train a model to produce sounds based on seeing a video of a drumstick hitting different objects. To test how realistic the fake sounds were, the team conducted an online study in which subjects saw two videos of collisions one with the actual recorded sound, and one with the algorithm's and were asked which one was real. The result: Subjects picked the fake sound over the real one twice as often as a baseline algorithm. They were particularly fooled by materials like leaves and dirt that tend to have less "clean" sounds than, say, wood or metal (cf. Owens et al., 2016). This is one of the first instances which utilizes a similar

approach as the Turing test to characterise natural from artificially made sounds.
Another team of researchers propose an environmental-sound-extraction method using an onomatopoeic word. The proposed method they utilize is described as follows: An onomatopoeic word is used to specify the sound to extract from a mixture sound. They used a U-Net encoder-decoder architecture which has been used in various source-separation and sound-extraction studies to estimate the time-frequency mask of the target sound (cf. Okamoto et al., 2022). In their evaluation the proposed method outperformed conventional methods that use a sound-event class as a condition. Both instances can be regarded as the basis for the design and development of the project described in this text.

3 Design Process for the Demo

For the interactive demo which was exhibited at the "summaery" the basic foundation was a mixture of prompt generated AI sounds and real ones, retrieved from an open-source, recorded by test persons which can be found under the identifier: "VocalImitationSet 1.1.3" (Kim et al., 2018). As this demo focuses on identifying artificial voices from human ones it is important to guarantee the quality of the generated sounds. When sifting out possible sounds and categories, examples with clicking sounds of a mouse, uncommon and rarely used noises and background noises from other people or from the environment were excluded. Additionally, recordings with low quality and noticeable length in between the start of the recording and making the noises were excluded in the filtering process for the final set of onomatopoeia. Ending up with the categories farm animals, instrument and domestic sounds with multiple sounds for each item in their respective category. Animals were made up of chicken, cow, goat, pig, rooster and duck sounds having three imitations each from the "VocalImitation" set. The same applies to the instruments made up from electric guitar, cello, drumkit and trumpet as well as the domestic sounds category containing a computer keyboard, toilet flush, vacuum cleaner and microwave. Some of the selected imitations in the final set were from the same voice actor due to the lack of usable samples. This could have been noticed by participants and influenced their decision on the voting.

Different approaches and language models were utilized to maintain a satisfying creation of the desired noise generated by AI. To classify the generated sound as usable a few performance indicators were identified beforehand which are presented in the list underneath.

Clearly distinguishable from the real noise – A clear imitation
Noticeable difference from other previously generated sounds for the same category and item.
No disruptive noises or lots of white noise
Somewhat similar to the noises provided in the database for each category – e.g. A cow sound is normally produced starting with M and o´s in-between a W at the end.

On the basis of this PI each item received at least two up to five AI generated sounds which would be then played while listening to the respective category.
The listening and voting process was possible through an application based on JavaScript, HTML and CSS.
The artificial sounds were generated with applying techniques known from prompt engineering and in some cases also speech to speech. The models which were used during this process were AudioLDM2 and eleven_monolingual_v1. While AudioLDM2 was first run locally on a virtual machine utilizing the own CPU: Intel(R) Core(TM) i7-6700HQ CPU @ 2.60G it would be later used within Google Colab to achieve a satisfying working speed. The second model is utilized by ElevenLabs in their application making it possible to produce a certain number of sounds within the free plan.

Prompt Engineering:

After determining the goal what the expected output should sound like, the next step was specifying the output requirements. The target was having 5 seconds long audio files in the wav or mp3 format. The style was supposed to mimic as close as possible a human sounding voice with no additional specifics. The first experiments with prompting were done on the assumption that giving instructions like: "A woman imitating the sound of [item]" is already sufficient for the model. However, with some refinement and iterating of the prompts with additions like "with a childish voice, just their voice, imitating with their voice" better results were possible. These were then used as templates to generate most of the needed sounds. At certain points it was necessary to deviate from this scheme and describe the expected sound directly. This was done for the keyboard sound which needed to be prompted in this fashion: "A man making fast clicking noises with his voice." This approach needed to be taken because when mentioning the subject computer keyboard the sounds were all realistic ones. In a few cases speech to speech was utilized, which meant the sound was given as the prompt for example from the data base or self-recorded to obtain a similar version output by an artificial voice.
In the end at least two up to five fake versions were generated for each item which would be then available to be played in their respective category.

4 Findings

This part presents the measured votes while exhibiting the demo. While it is interesting and sensible to look at the acquired data it must be mentioned that the participants were mostly unsupervised while interacting with the demo and it was their free choice which sounds to listen to. The distribution of votes is also influenced by the arrangement and structure of the application. Moreover, to keep the demo easily understandable the participants could only vote for if the track they listened to is AI generated. All in all, this means the votes might be not as representative and manipulated to a certain extent.
With that in mind Fig.1 shows the category farm animals received the most votes across all items, most likely due to listening count and its position being shown first on the main page. There are multiple tracks for chicken sounds, with "Chicken-Cluck-Track 5" receiving the highest votes (20), indicating it was most often

perceived as AI-generated. Other chicken tracks have significantly fewer votes, suggesting a varied perception of AI generation among them. However, Chicken-Cluck-Track 6 has no votes at all, hinting it was not played at all or perceived as human made. The items chicken and cow received more overall votes, likely due to being played more frequently. "Chicken-Cluck-Track 5", "Cow-Moo-Track 1 ", Electric Guitar-Track 4 and “Computer Keyboard-Track 3” all received nine to eleven votes making them the most as Ai generated perceived Tracks. "Cow-Moo-Track 1" stands out in its category with 11 votes, potentially indicating it was played frequently, but other cow sounds have fewer votes with up to five marking it as the most suspected track. The tracks under the item goat have very even distributed votes however track five did not receive any votes making it another by user’s unidentified track. Similar to this the tracks inside the item duck received evenly distributed votes with the exception of Track 4 which went also as unidentified. These so-called unidentified tracks are AI generated but did not receive any votes. This could be a result of them being played not at all, less frequent or the users listening to them being fooled. The other categories have a lower listening count and therefore lower votes. The more evenly distributed votes across instrument tracks suggest a general difficulty in distinguishing these sounds as AI-generated.
“Cello-Track 2” and “Electric Guitar” received in their category and for their item the most votes while the other votes are mostly even distributed. The category domestic sound was played the least however “Computer Keyboard-Track 3 received wrongfully a lot of votes for being AI generated. Similar to instruments, the even distribution of votes implies a challenge in identification.

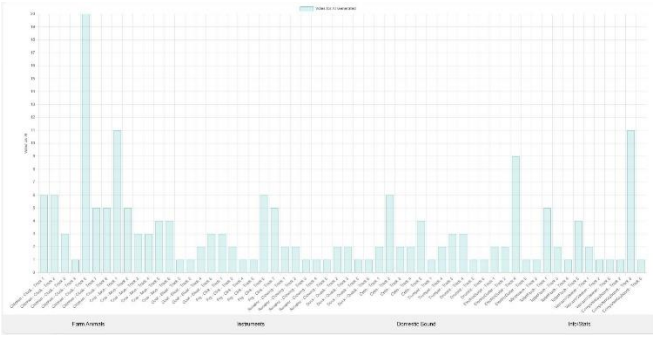

Fig. 1 Bar Chart with the most voted tracks

Fig. 2 suggests that, on average, people were slightly better than chance at identifying AI-generated sounds. However, the near 50-50 split also indicates a significant challenge in distinguishing AI-generated sounds from real ones.
One other noticeable thing is that the sounds from the “VocalImitation Set” received across all categories 87 votes which poses the question if the recorded sounds were not made to the expectation of the participants based on cultural differences and mother tongue and thereby receiving votes as AI generated.

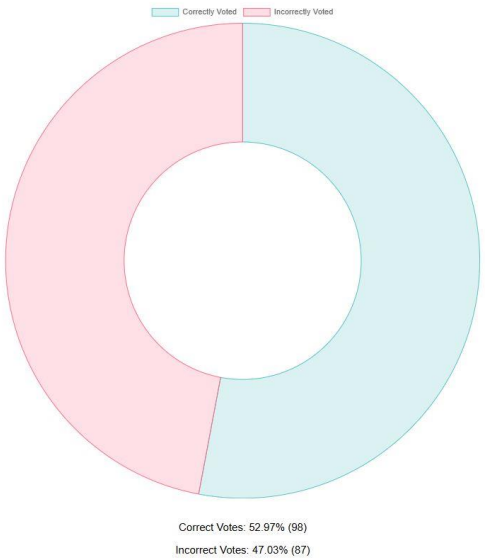

Fig. 2 Donut Chart portraying the amount of correct/incorrect votes

The close split between correct and incorrect votes highlights the current limitations of human ability to reliably distinguish AI-generated sounds from real ones.
Certain categories, such as farm animals (especially chickens and cows), show higher variability in identification, possibly due to the distinctiveness or frequency of certain sounds.

6 Conclusion

The study shows that current AI models, when fine-tuned and effectively prompted, can produce onomatopoeic sounds that are difficult for humans to distinguish from real human-made sounds. The interactive demo presented at summaery 2024 provided some insights into the perception of AI-generated audio, revealing a general difficulty among participants in accurately identifying synthetic sounds. This suggests potential for further improvements in TTS technology, particularly in applications requiring high levels of audio realism, such as entertainment, education and accessibility. In addition, the results highlight the importance of diverse and nuanced training data, such as onomatopoeia, in improving the expressiveness and naturalness of AI-generated voices.

7 Reflection

This demo provided some interesting insights into the ability to identify artificial voices and generate onomatopoeia with prompt engineering using LLMs.
However, there are several things that can be improved and adapted to produce meaningful insights and data. Firstly, additional information needs to be collected on how often a particular track has been played in order to understand how it is perceived by subjects in relation to the number of times it has been played. In addition,

data on which tracks are most likely to be human, feedback for user voting and a proper study plan need to be put in place. This would also take care of balancing certain conditions, understanding the amount of tracks played and the users involved.
There are also some possible improvements that can be made to the application itself. For example, feedback for pressed buttons and a more advanced system for selecting the sound recording played next to the test person to improve the user experience.
A next step could be to generate voice snippets with onomatopoeia, as found in children's books, or to fine-tune existing models with onomatopoeia to learn more about the effect of specially trained models and the enhancement of human characteristics in artificially produced voices and sounds.

REFERENCES

[1] Inamdar, F. M., Ambesange, S., Mane, R., Hussain, H., Wagh, S., & Lakhe, P. (2023). Voice Cloning Using Artificial Intelligence and Machine Learning: A Review. *Journal of Advanced Zoology, 44*.

[2] Shi, F. (2023). Revolutionizing Personalized Voice Synthesis: The Journey towards Emotional and Individual Authenticity with DIVSE (Dynamic Individual Voice Synthesis Engine). *arXiv preprint arXiv:2312.17281*.

[3] Open AI (2024). Navigating the Challenges and Opportunities of Synthetic Voices. Accessed: 19.07.2024. https://openai.com/index/navigating-the-challenges-and-opportunities-of-synthetic-voices/

[4]Espinosa, M. N., (2023). State of the art in Voice Cloning: A review. Accessed: 19.07.2024. https://blog.marvik.ai/2023/03/21/state-of-the-art-in-voice-cloning-a-review/

[5] Owens, A., Isola, P., McDermott, J., Torralba, A., Adelson, E. H., & Freeman, W. T. (2016). Visually indicated sounds. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2405-2413).

[6] Okamoto, Y., Horiguchi, S., Yamamoto, M., Imoto, K., & Kawaguchi, Y. (2022, May). Environmental sound extraction using onomatopoeic words. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 221-225). IEEE.

What is needed to make the Application run on your system?

Base Path for Tracks: The base path for the tracks needs to be adjusted to the correct path on the user's system.

Javascript: L207

```
const basePath = "C:\\Users\\Christoph\\Fake\\Vocal Imitation Auswahl\\";
```
Change this to match the folder structure on the user's system.

Audio File Path: The path for playing the audio file needs to be adjusted to ensure it matches the user's system. L211

```
const filePath = `${basePath}${category}\\${category} - ${track}`;
currentAudio = new Audio(`file:///${filePath}`);
```

Play Icon Path: The path for the play icon needs to be updated to the correct path on the user's system.
All Img Paths in the HTML File need to be updated

Html:
```
<  img src="C:\Users\Christoph\Fake\Images\Chicken - Cluck.png" alt="Chicken" class="button-image">

<img src="C:\\Users\\Christoph\\Fake\\Icon\\Playbutton.png" alt="Play Icon" onclick="playTrack('${currentCategory}', '${selectedTrack}')">
```

Update this path to where the icon and images are located on the user's system.

Ensure the filePath construction is correct for the folder structure on the user's system.