Imagine to Hear: Auditory Knowledge Generation can be an Effective Assistant for Language Models

Suho Yoo[1*] **Hyunjong Ok**[1,2*] **Jaeho Lee**[2†]
[1]HJ AILAB [2]POSTECH
{uso7d0, hyunjong.ok}@gmail.com, jaeho.lee@postech.ac.kr

Abstract

Language models pretrained on text-only corpora often struggle with tasks that require auditory commonsense knowledge. Previous work addresses this problem by augmenting the language model to retrieve knowledge from external audio databases. This approach has several limitations, such as the potential lack of relevant audio in databases and the high costs associated with constructing the databases. To address these issues, we propose Imagine to Hear, a novel approach that dynamically generates auditory knowledge using generative models. Our framework detects multiple audio-related textual spans from the given prompt and generates corresponding auditory knowledge. We develop several mechanisms to efficiently process multiple auditory knowledge, including a CLAP-based rejection sampler and a language-audio fusion module. Our experiments show that our method achieves state-of-the-art performance on AuditoryBench without relying on external databases, highlighting the effectiveness of our generation-based approach.

1 Introduction

Can language models (LMs) understand auditory signals like humans? Recent studies suggest that the answer is negative. Although LMs do exhibit some understanding of audio signals (Ngo and Kim, 2024), they perform poorly in answering questions that require auditory commonsense knowledge (Ok et al., 2025). For example, LMs pretrained on text-only datasets do not know which object is more likely to make a higher-pitched sound or which animal is most relevant to the given onomatopoeia.

While such limitations can be partially addressed by augmenting language models with auditory representations retrieved from external audio database (Ok et al., 2025), this approach suffers from two key limitations: (1) The database may not contain any audio that is directly relevant to the given query. (2) Constructing such databases requires a large computational cost.

To address these challenges, we present a novel approach, coined *Imagine to Hear* (ITH), that leverages the imaginative capabilities of audio generative models (Evans et al., 2024; Hung et al., 2024; Liu et al., 2023). That is, ITH acquires auditory knowledge directly related to the given task by generating it instead of retrieving it from a database.

ITH works in three steps. First, we extract textual spans from the given input prompt that may contain auditory knowledge. Unlike visual imagination literature (Lu et al., 2022), we extract multiple short spans instead of a single long span so that we can generate cleaner audio without auditory interference. Next, we generate audio knowledge for each span, ensuring their relevance via CLAP-based rejection sampling (Elizalde et al., 2023). Finally, we inject generated audio into the LM with a newly developed fusion module that can efficiently handle an arbitrary number of auditory features.

The proposed ITH achieves state-of-the-art results on AuditoryBench (Ok et al., 2025), an auditory commonsense benchmark. The primary contributions of our work can be summarized as follows:

- We develop a new generation-based framework that leverages imaginative capabilities to enhance the auditory understanding of language models.
- We propose a new algorithm (ITH) that generates and utilizes multiple auditory knowledge from the given query, with new architectural components to process multiple audio signals while ensuring their relevance effectively.
- ITH sets the new state-of-the-art on AuditoryBench without relying on external databases.

[*] equal contribution
[†] corresponding author

Findings of the Association for Computational Linguistics: ACL 2025, pages 14182–14193
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

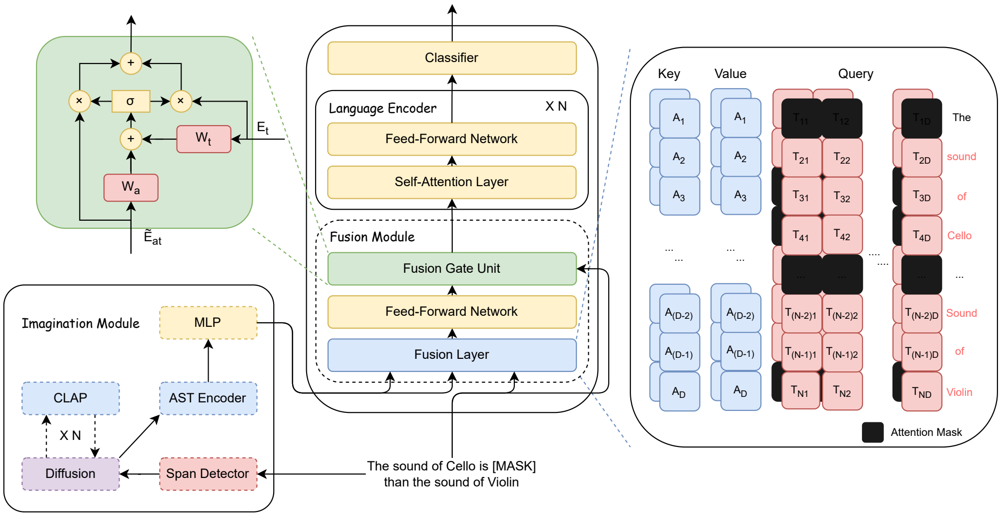


Figure 1: An illustration of the overall framework of the proposed Imagine to Hear (ITH), consisting of three components: (1) An imagination module, which detects multiple audio-related spans from the given prompt and generates multiple corresponding audio knowledge. (2) A fusion module, which combines the (variable-length) auditory and textual information. (3) A language encoder, which processes the output of the fusion module.

2 Method: Imagine to Hear

The proposed ITH consists of three components (Figure 1): (1) Imagination module for audio feature generation (Section 2.1); (2) Fusion module for multimodal integration (Section 2.2); (3) Language encoder for text feature generation (Section 2.3).

2.1 Imagination Module

The imagination module (IM) works in three steps: span detection, audio generation with rejection sampling, and feature extraction.

Span detection. First, IM detects audio-related spans within the textual context given as an input. Concretely, let $\mathbf{x}$ be a sequence of tokens in the textual context. We find subsequences $(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots)$ of $\mathbf{x}$ consisting of audio-related tokens. This is done by training a model that classifies each token of $\mathbf{x}$ as audio-related or not. Each contiguous set of audio-related tokens forms a span, and thus, we get more than one span per each textual context.

To get this classifier, we fine-tune the pretrained BERT-base (Devlin et al., 2019) on the training split of the AuditoryBench dataset.

Audio generation with rejection. Next, IM uses pretrained text-to-audio diffusion models to generate audio corresponding to each auditory span. To ensure the relevance of the generated audio, we reject and resample the audio until the desired similarity to the text span has been met. The semantic similarity is measured using CLAP (Elizalde et al., 2023). Precisely, we resample until the cosine similarity exceeds a certain threshold τ, *i.e.*,

$$\frac{\mathrm{CLAP}_a(\mathbf{a}_{\text{gen}})^\top \mathrm{CLAP}_t(\mathbf{x}_{\text{span}})}{\|\mathrm{CLAP}_a(\mathbf{a}_{\text{gen}})\|\|\mathrm{CLAP}_t(\mathbf{x}_{\text{span}})\|} > \tau \qquad (1)$$

has been satisfied, where CLAP_a and CLAP_t denotes the audio and text encoders of CLAP, $\mathbf{x}_{\text{span}}$ denotes the text span extracted from the input query, and $\mathbf{a}_{\text{gen}}$ denotes the generated audio clip.

For efficiency, we limit the maximum number of generations; if the generative model fails to generate a well-aligned audio clip for a span in n trials, we simply ignore the span. Empirically, $n = 2$ has been sufficient (see Section 3).

Feature extraction. The generated audio is transformed into dense embedding with an AST encoder (Gong et al., 2021) followed by a two-layer MLP.

2.2 Fusion Module

The fusion module (FM) consists of a fusion layer, a feed-forward network, and a fusion gate unit.

Fusion layer. The Fusion layer is a cross-attention module that processes textual tokens by attending to the audio tokens. In particular, the layer computes the queries of the text tokens that belong to the audio-related spans only and computes the output using only the keys and values of the audio tokens corresponding to the span.

Methods	Animal sound recognition			Sound pitch comparison		
	Dev.	Test	Wiki Test	Dev.	Test	Wiki Test
BERT$_{\text{base}}$ (Devlin et al., 2019)	15.51	13.46	3.05	59.42	60.41	48.06
RoBERTa$_{\text{base}}$ (Liu, 2019)	14.67	14.04	2.54	54.50	55.84	47.45
Gemma2$_{\text{2B}}$ (Team et al., 2024)	14.33	15.11	6.60	59.25	60.45	47.86
LLaMA3.1$_{\text{8B}}$ (Dubey et al., 2024)	23.10	21.80	16.24	61.46	62.55	47.72
AudioBERT (Ok et al., 2025)	38.28	36.63	14.32	73.18	74.83	55.31
Ours	**39.36** ± 1.11	**41.55** ± 1.06	**19.09** ± 1.78	**79.82** ± 0.45	**78.96** ± 0.48	**76.74** ± 0.62

Table 1: Experimental results on AuditoryBench. The figures for baselines are taken from Ok et al. (2025).

Feed-forward network. The output features from the fusion layer are processed with a two-layer MLP (same as in a typical transformer block).

Fusion gate unit. The fusion gate unit combines the original input textual tokens $\mathbf{x}$ and the output of the FFN module $\mathbf{z}_{\text{FFN}}$ to generate the output via gating mechanism. More concretely, the output of the module is computed as

$$\mathbf{z}_{\text{fused}} = \mathbf{g} \odot \mathbf{x} + (\mathbf{1} - \mathbf{g}) \odot \mathbf{z}_{\text{FFN}}, \qquad (2)$$

where $\mathbf{g}$ is the gating signal:

$$\mathbf{g} = \sigma\big((\mathbf{W}_1\mathbf{x} + \mathbf{b}_1) + (\mathbf{W}_2\mathbf{z}_{\text{FFN}} + \mathbf{b}_2)\big), \quad (3)$$

with $\sigma(\cdot)$ denoting the sigmoid activation function and $\mathbf{W}_i, \mathbf{b}_i$ denoting trainable weights and biases. This gate adaptively weighs the contributions of two modalities for each token embedding. When $\mathbf{g}$ is large, more information from the raw text will be passed to the output. When $\mathbf{g}$ is small, more information from the audio will affect the output.

2.3 Language Encoder

After the fusion module integrates the textual features with the audio features to get $\mathbf{z}_{\text{fused}}$, we apply a pretrained text encoder to generate the output. Following Ok et al. (2025), we use BERT-base as our text encoder and attach a single linear layer at the end of the model as a classifier.

2.4 Training

The model is trained end-to-end with the training split of the AuditoryBench dataset. All weights in the model are trained, except for the audio diffusion model, CLAP, and the span detector—which has been separately trained using the AuditoryBench.

3 Experiments

We conducted our experiments on the AuditoryBench with further experiments. More details are provided in Appendix D.

Task	Dev.	Test	Wiki
Animal sound	92.47 ± 0.77	92.66 ± 0.35	100.00 ± 0.00
Sound pitch	94.50 ± 0.32	94.75 ± 0.29	90.24 ± 3.00

Table 2: F1 scores of span detection.

Experimental results. To evaluate the effectiveness of our approach, we compare it with recent studies (Ok et al., 2025) on AuditoryBench. As shown in Section 2.1, most language models perform poorly due to the absence of auditory knowledge. While AudioBERT performs well by leveraging external databases, it struggles with inter-domain, as reflected in the wiki test results. In contrast, our method leverages imagination to dynamically generate relevant auditory representations, establishing a new state-of-the-art AuditoryBench. This result highlights the enhanced generalization ability of our approach. We additionally report F1 scores of our span detector, in Table 2. We observe that even a simple span detector can achieve high accuracy across all subsets.

Ablation studies. We first examine the effect of removing the rejection system. As shown in Figure 2 (a), the rejection threshold of 0 is optimal for animal sound recognition. In contrast, for sound pitch comparison, the optimal threshold is 0.6, as illustrated in Figure 2 (b). For sound pitch comparison, to further analyze the impact of the iteration count n, we conducted an additional ablation study by fixing the rejection threshold at 0.6 (see Figure 2 (c)). The results indicate that accuracy improves as n increases, reaching its peak at $n = 2$. Detailed performance gains are reported in Section 3.

The fusion gate plays a crucial role in maintaining performance by selectively incorporating auditory knowledge while preventing interference.

A lack of dynamic knowledge injection generates audio for the entire sentence. In tasks like sound pitch comparison, where multiple auditory spans exist, generating audio for the entire sentence results in overlapping sounds, which significantly

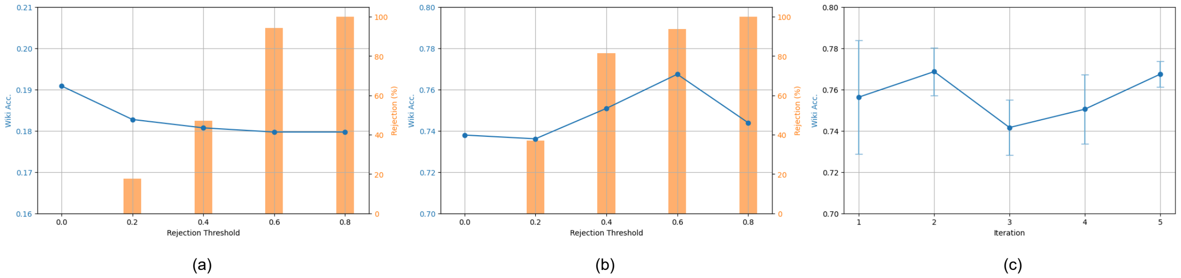
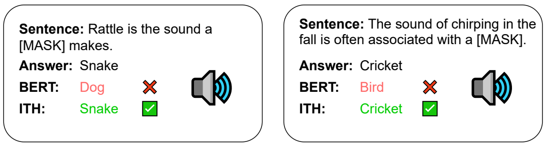


Figure 2: Ablation study of our rejection system. Each (a) and (b) are ablation results of CLAP threshold on animal sound recognition and sound pitch comparison. (c) Ablation results of iterative refinement on sound pitch comparison. The light blue line represents accuracy, while the orange bars in (a) and (b) indicate the proportion of instances where rejection occurs even after n iterations, excluding the corresponding audio input.

Models	Animal sound recognition		Sound pitch comparison	
	Test	Wiki Test	Test	Wiki Test
ITH (Ours)	41.55 ± 1.06	19.09 ± 1.78	78.96 ± 0.48	76.74 ± 0.62
w/o Rejection	-0.00	-0.00	-0.49	-2.95
w/o FG	-0.66	-2.14	-0.78	-0.87
w/o DKI	-0.84	-0.92	-0.13	-1.29
w/o (DKI + FG)	-0.58	-1.43	-0.28	-2.32

Table 3: The result of the ablation study. DKI denotes dynamic knowledge injection, FG denotes fusion gate.



Sentence: Rattle is the sound a [MASK] makes.
Answer: Snake
BERT: Dog ✗
ITH: Snake ✓

Sentence: The sound of chirping in the fall is often associated with a [MASK].
Answer: Cricket
BERT: Bird ✗
ITH: Cricket ✓

Figure 3: ITH case study. To listen to the generated sounds, visit our project page.

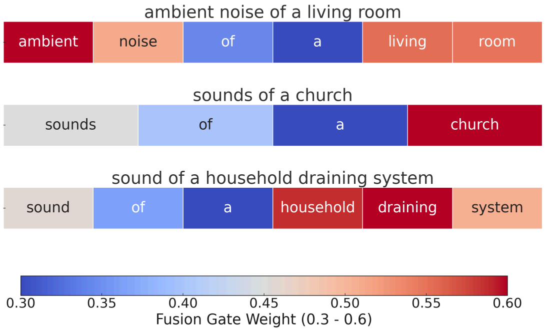


Figure 4: Fusion gate case study, where a larger fusion gate weight indicates a higher degree of audio knowledge injection into the corresponding token.

affects comprehension and model performance. Removing dynamic knowledge injection and the fusion gate further amplifies this issue, as indiscriminate audio injection leads to uncontrolled modality integration.

Case study. We conducted a case study to evaluate the effectiveness of our method. As shown in Figure 3, the first example describes the sound of a "rattle," strongly associated with a snake. Without auditory information, it can be challenging to make the correct prediction. However, leveraging imagined auditory knowledge, our method correctly identifies "snake" as the answer. In the second example, the sentence mentions "chirping in the fall," a sound commonly associated with crickets rather than birds. While "chirping" might semantically align with avian species, our model, by incorporating auditory reasoning, correctly predicts "cricket," demonstrating its ability to refine predictions based on imagined auditory context.

Role of fusion gate. To analyze the role of the fusion gate in cross-modal integration, we visualize the token-wise fusion gate weights in Figure 4. The fusion gate operates at the embedding level. However, for visualization purposes, we compute the average fusion gate values per token and represent them in a heatmap.

Our analysis shows that the fusion gate effectively prioritizes auditory-relevant tokens while minimizing the influence of function words. In phrases like "ambient noise of a living room" and "sound of a household draining system," key auditory descriptors such as "ambient," "living," "room," "household," "draining," and "system" receive the highest weights. This indicates that the model assigns more importance to tokens that directly describe sound characteristics while suppressing function words like "of" and "a," which contribute less to auditory integration.

Dynamic knowledge injection. To deepen our understanding on the role of dynamic knowledge injection (DKI), we analyze cases where injecting auditory knowledge at the span level improves

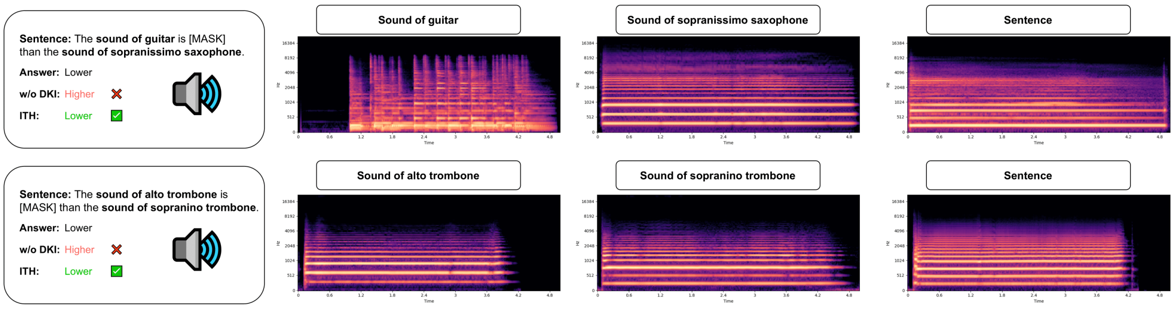


Figure 5: Dynamic knowledge injection (DKI) case study. To listen to the generated sounds, visit our project page.

model predictions, as illustrated in Figure 5.

In the first example, the sentence compares the sound of a guitar with that of a sopranissimo saxophone. When generating audio representations separately for each instrument, their mel-spectrograms exhibit distinct frequency characteristics, with the saxophone producing significantly higher frequencies than the guitar. However, when generating a single audio representation for the entire sentence, this distinction is lost, leading to an incorrect prediction of "higher." In contrast, our method, by dynamically injecting span-specific auditory knowledge, correctly predicts "lower," maintaining the frequency contrast observed in the spectrograms.

In the second case, the sentence contrasts the sound of an alto trombone with that of a sopranino trombone. Similarly, generating audio separately for each instrument produces clear frequency differences, with the sopranino trombone having higher frequencies. However, sentence-level generation fails to capture this difference, leading to an incorrect prediction of "higher." By selectively injecting auditory spans, our model correctly predicts "lower," aligning with the frequency distribution observed in the mel-spectrogram analysis.

More details. We also illustrate more ablation studies for each method to demonstrate their contributions in Appendix B.

4 Conclusion

In this work, we proposed *Imagine to Hear*, a novel generation-based approach for injecting auditory knowledge into language models. By dynamically generating audio, our method overcomes the limitations of retrieval-based systems. With a hierarchical architecture and a rejection system, we ensure high-quality, context-relevant audio knowledge integration. Experiments demonstrate significant improvements on AuditoryBench with audio imagination. This work underscores the value of imagination in enhancing multimodal capabilities in language models.

ITH extends beyond its established advantages over retrieval methods in the multimodal era. Its utility is notable in resource-constrained settings reliant on unimodal models where multimodal inputs are infeasible. Furthermore, even for advanced multimodal systems, ITH can provide a practical pathway when audio is absent, ambiguous, or subtle—parallel to visual imagination modules that enhance reasoning without direct visual input (Lu et al., 2022; Zhu et al., 2023; Liu et al., 2024).We expect that our approach will remain effective and complementary in the era of multimodal language models.

Limitations

One limitation is that the effectiveness of our method is influenced by the quality of input prompts. If the input lacks sufficient auditory cues, the generated audio may not always align perfectly with the intended knowledge (Appendix E). Additionally, while our rejection-based generation ensures high-quality auditory integration, it introduces some computational overhead.

Acknowledgments

This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Ministry of Science and ICT (MSIT)(No.RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH)), and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(No. RS-2023-00213710, neural network optimization with minimal optimization costs). We also thank our colleagues at EffL Lab and others for their help.

References

M. Alper, M. Fiman, and H. Averbuch-Elor. 2023. Is bert blind? exploring the effect of vision-and-language pretraining on visual language understanding. In *CVPR*.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2024. Simple and controllable music generation. *Preprint*, arXiv:2306.05284.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

B. Elizalde, S. Deshmukh, M. Al Ismail, and H. Wang. 2023. Clap: Learning audio concepts from natural language supervision. In *ICASSP*.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. 2024. Stable audio open. *arXiv preprint arXiv:2407.14358*.

Y. Gong, Y. Chung, and J. Glass. 2021. AST: Audio Spectrogram Transformer. In *Interspeech*.

Hangyu Guo, Kun Zhou, Wayne Xin Zhao, Qinyu Zhang, and Ji-Rong Wen. 2023. Visually-augmented pretrained language models for nlp tasks without images. In *ACL*.

Chia-Yu Hung, Navonil Majumder, Zhifeng Kong, Ambuj Mehrish, Rafael Valle, Bryan Catanzaro, and Soujanya Poria. 2024. Tangoflux: Super fast and faithful text to audio generation with flow matching and clap-ranked preference optimization. *arXiv preprint arXiv:2412.21037*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pretraining for unified vision-language understanding and generation. In *ICML*.

Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*.

Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. 2023. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *arXiv preprint arXiv:2308.05734*.

Jingming Liu, Yumeng Li, Boyuan Xiao, Yichang Jian, Ziang Qin, Tianjia Shao, Yao-Xiang Ding, and Kun Zhou. 2024. Enhancing visual reasoning with autonomous imagination in multimodal large language models. *Preprint*, arXiv:2411.18142.

X. Liu, D. Yin, Y. Feng, and D. Zhao. 2022. Things not written in text: Exploring spatial commonsense from visual signals. In *ACL*.

Y. Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

I. Loshchilov. and . Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

Y. Lu, W. Zhu, X. Wang, M. Eckstein, and W. Y. Wang. 2022. Imagination-augmented natural language understanding. In *NAACL*.

J. Ngo and Y. Kim. 2024. What do language models hear? probing for auditory representations in language models. In *ACL*.

Hyunjong Ok, Suho Yoo, and Jaeho Lee. 2025. Audiobert: Audio knowledge augmented language model. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

P. Rahmanzadehgervi, L. Bolton, M. R. Taesiri, and A. T. Nguyen. 2024. Vision language models are blind. *arXiv preprint arXiv:2407.06581*.

H. Tan and M. Bansal. 2020. Vokenization: Improving language understanding with contextualized, visual-grounded supervision. In *EMNLP*.

Tianyi Tang, Yushuo Chen, Yifan Du, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Learning to imagine: Visually-augmented natural language generation. In *ACL*.

G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, Lé. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Alex Wang. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

W. Wang, L. Dong, H. Cheng, H. Song, X. Liu, X. Yan, J. Gao, and F. Wei. 2023. Visually-augmented language modeling. In *ICLR*.

A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

C. Zhang, B. Van Durme, Z. Li, and E. Stengel-Eskin. 2022. Visual commonsense in pretrained unimodal and multimodal models. In *NAACL*.

Wanrong Zhu, An Yan, Yujie Lu, Wenda Xu, Xin Eric Wang, Miguel Eckstein, and William Yang Wang. 2023. Visualize before you write: Imagination-guided open-ended text generation. In *EACL Findings*.

A Related work

Auditory knowledge in language model. Pretrained language models (LMs) lack auditory commonsense knowledge due to their text-only training, limiting their ability to reason about sound-related concepts. Recent studies have investigated whether LMs encode latent auditory information by aligning text embeddings with representations from pretrained audio models. While these studies suggest some degree of alignment, the learned representations remain insufficient for effectively understanding and utilizing auditory information (Ngo and Kim, 2024).

Retrieval-based approaches attempt to address this limitation by retrieving relevant audio-text pairs from external databases and integrating them into LMs. AudioBERT (Ok et al., 2025) follows this paradigm, detecting spans that require auditory knowledge and injecting retrieved information through lightweight adaptation layers. However, these methods inherently depend on the availability of existing data, making them unreliable when relevant audio is missing.

Our approach instead generates auditory knowledge directly from context using a text-to-audio diffusion model, removing the need for external retrieval. By dynamically incorporating auditory information, this method offers a generalizable solution that can be applied across various tasks requiring auditory understanding.

Visual knowledge in language model. Beyond auditory knowledge, LMs also lack visual commonsense knowledge (Zhang et al., 2022; Liu et al., 2022; Alper et al., 2023; Rahmanzadehgervi et al., 2024), which limits their ability to reason about concepts such as object shapes, colors, and spatial relationships. For this problem, previous studies have explored methods for integrating visual knowledge, primarily by generating images from textual input and incorporating them into LMs.

Most existing approaches generate images based on the entire input context or sentence level generation, assuming that all textual content requires visual grounding. While this allows for broad visual augmentation, it often results in irrelevant or redundant images, as these methods do not selectively identify which parts of the text actually require visual support (Wang et al., 2023; Tang et al., 2023; Lu et al., 2022). A different line of work detects visually informative spans within the text before aligning them with visual representations (Guo et al., 2023), but these methods lack mechanisms to verify the relevance of generated images, leading to potential inconsistencies when integrating visual knowledge.

Models	Animal sound recognition		Sound pitch comparison	
	Test	Wiki Test	Test	Wiki Test
Stable	**41.55** ± 1.06	**19.09** ± 1.78	**78.47** ± 0.44	73.79 ± 0.99
AudioLDM 2	40.89 ± 0.57	18.38 ± 2.77	78.44 ± 0.90	73.42 ± 1.05
MusicGen	41.01 ± 1.27	17.77 ± 1.19	78.05 ± 0.70	**74.06** ± 1.27

Table 4: diffusion ablation

Our approach addresses these limitations by detecting specific spans and generating representation only for those spans. Additionally, we introduce a rejection system that filters out irrelevant representation before integration, ensuring that only contextually aligned generated knowledge is incorporated into the model.

Generation models. Diffusion models have become the dominant paradigm for generative tasks, yet recent studies have increasingly focused on improving efficiency in training and inference. In text-to-audio generation, models like AudioLDM2 (Liu et al., 2023), Tango (Hung et al., 2024), and Stable Audio (Evans et al., 2024) optimize stable diffusion-based architectures, leveraging latent representations to reduce computational overhead while maintaining high-quality generation.

B More ablation study

Various generation models. To evaluate the impact of different audio generation models, we conducted an ablation study comparing Stable Audio (Evans et al., 2024), AudioLDM 2 (Liu et al., 2023), and MusicGen (Copet et al., 2024) in Appendix B.

Overall, Stable achieved the best performance across most tasks, demonstrating its effectiveness in generating relevant auditory representations. It outperformed other models in both animal sound recognition and sound pitch comparison, particularly in the Test sets. This suggests that Stable provides the most robust auditory features, contributing to better downstream task performance.

Interestingly, MusicGen performed slightly better than other models in the sound pitch comparison

Preference	Ratio (%)
Real Audio	41.16
Generated Audio	50.30
Equal	8.54

Table 5: Pairwise comparison between real and generated audio.

Setup	Chose Real (%)	Chose Generated (%)
2 Reals, 1 Gen	65.00	**35.00**
2 Gens, 1 Real	**44.03**	55.97

Table 6: We design a 3-way identification task to evaluate the realism of generated audio. In the first setup, participants are presented with two real audio samples and one generated sample and are asked to identify which one is the generated audio. In the second setup, the inverse task is given: among the three audio samples (two generated and one real), participants are asked to identify the real audio.

wiki test. This dataset primarily involves musical instrument pitch comparison, aligning well with MusicGen's domain specialization in music generation. The results indicate that MusicGen can generate more domain-relevant auditory representations for musical contexts, leading to an improvement in this specific evaluation.

C Human Evaluation

Generated Audio. To assess perceptual quality of the generated audio samples, we have conducted human evaluations comparing our generated audio with real audio retrieved from LAION-Audio-630K. Sixteen participants have evaluated the samples under both in pairwise and 3-way identification settings, with the total of 40 questions. The results, shown in Table 5 and Table 6, indicate that the generated audio is often indistinguishable from the real recordings. Specifically, Table 5 shows that, when comparing retrieval-based audio (in AudioBERT) and generated audio (in ours) conditioned on the same input text from AuditoryBench, human evaluators exhibit a clear preference for the generated audio. Table 6 presents a 3-way identification task where participants are asked to identify the real sample among two generated and one real audio; the results suggest that participants struggle to distinguish real from generated audio reliably.

We have further evaluated on the wiki set, which includes real audio aligned with the prompts. As shown in Table 7, ten out of sixteen annotators have judged the generated audio to be of sufficiently high quality and fidelity to serve as plausible substitutes; we have tested with 20 questions sampled from the wiki set. These findings confirm that our method produces realistic and contextually appropriate auditory knowledge.

Evaluation Criterion	Result
Acceptable as substitute for real audio	77.42%
Quality and semantic fidelity (1–5 scale)	3.78

Table 7: Human evaluation on the Wikipedia subset of AuditoryBench.

D Experiment details

D.1 Training Datasets

AuditoryBench. We conducted our experiments on the AuditoryBench—a benchmark designed to evaluate the auditory knowledge of language models which comprises two fundamental tasks: Animal Sound Recognition, which assesses the model's ability to classify various animal sounds, and Sound Pitch Comparison, which measures its capability to discern differences in pitch between sounds. To evaluate our proposed methods, we employ accuracy for AuditoryBench.

The dataset includes 6,212 sentences for Animal Sound Recognition and 15,502 sentences for Sound Pitch Comparison, with varying lengths and complexities. The average number of words per sentence is 9.21 for Animal Sound Recognition and 16.83 for Sound Pitch Comparison. More statistics info is in Appendix C.

Implementation details. The audio span detector employ a BERT-base model, trained for 5 epochs with a batch size of 16, a learning rate of 1×10^{-5}, and using the AdamW (Loshchilov. and Hutter, 2019) optimizer. For labeling, spans in the AuditoryBench dataset were directly utilized as they are included in the dataset.

For audio generation, we employed the stable-audio-open-1.0 (Evans et al., 2024). Compared to existing state-of-the-art models, this model was selected based on their strong performance demonstrated through experiments on various datasets. An AST (Gong et al., 2021) was employed to inject auditory representations into the language model.

For training, we utilized a BERT-base model to ensure a fair comparison with existing methods (Ok et al., 2025). The model was trained for 8 epochs with a batch size of 32, a learning rate of 3×10^{-4}

Category	Train	Dev	Test	Wiki	Total	Train	Dev	Test	Wiki	Total
	Animal sound recognition					Sound pitch comparison				
# Sentence	4,211	593	1,211	197	6,212	8,312	1,178	2,387	3,625	15,502
# Words/Sentence	9.27	9.39	9.30	7.01	9.21	18.18	18.12	18.19	12.41	16.83
# Total words	39,024	5,567	11,268	1,381	57,240	151,081	21,343	43,431	44,987	260,842

Table 8: Statistics for Auditorybench.

Methods	SST-2	QNLI	QQP	MNLI	MRPC	STS-B	Avg.
Multimodal text encoder							
CLIP (Radford et al., 2021)	73.3	74.5	72.8	68.4	74.3	73.8	72.85
BLIP (Li et al., 2022)	76.3	77.4	78.8	72.5	77.8	76.4	76.53
ALBEF$_{14M}$ (Li et al., 2021)	78.9	78.2	79.4	73.4	76.5	77.5	77.31
BERT$_{base}$							
Baseline (Devlin et al., 2019)	89.3	87.9	87.2	79.4	81.7	84.4	84.98
VOKEN (Tan and Bansal, 2020)	92.2	88.6	88.6	82.6	83.5	86.0	86.83
iACE (Lu et al., 2022)	91.7	88.6	89.1	82.8	**85.8**	86.6	87.43
VAWI (Guo et al., 2023)	**92.9**	89.1	89.7	83.0	**85.8**	87.2	87.80
Ours	92.7	**90.4**	**90.9**	**83.7**	85.7	**87.5**	**88.5**
RoBERTa$_{base}$							
Baseline (Liu, 2019)	89.2	87.5	86.2	79.0	81.4	85.4	84.78
VOKEN (Tan and Bansal, 2020)	90.5	88.2	87.8	81.0	87.0	86.9	87.83
iACE (Lu et al., 2022)	91.6	89.1	87.9	82.6	87.7	86.9	87.63
VAWI (Guo et al., 2023)	91.7	90.6	87.9	82.6	**88.5**	88.3	88.21
Ours	**94.6**	**91.9**	**91.4**	**87.4**	88.0	**90.1**	**90.6**

Table 9: Experimental results on GLUE. The results of prior work are reported from VAWI (Guo et al., 2023). VAWI utilizes various methods; we report the best performance of VAWI for each task.

and 4×10^{-5} for each task, and using the AdamW optimizer.

All experiments report the average score from 5 runs with different random seeds for each setting and experiments were done on a single NVIDIA L40S or NVIDIA 6000ADA GPU or Geforce RTX 4090. Additionally, we performed a search over the CLAP rejection threshold, as well as the number of iterations, to identify the optimal values.

E Experiments on GLUE

Experimental result To assess the impact of auditory knowledge in standard NLP tasks, we evaluated our method on the GLUE (Table 9). While our approach shows improvements over the baseline, it is not entirely clear whether the performance gain is directly attributable to the integration of auditory knowledge.

Indeed, audio spans appear in only 0.19% to 4.32% of all training and development samples in GLUE (see Appendix C for details), meaning that a large portion of the dataset lacks explicit auditory information. This raises the possibility that the observed improvements could stem from factors unrelated to auditory augmentation, such as increased model complexity or additional fine-tuning effects.

Implementation details. Following recent works (Wang, 2018; Guo et al., 2023), we train with a batch size of 32 and apply a weight decay of 0.01. A grid search is conducted to determine the optimal learning rate within the range of $[2 \times 10^{-5}, 5 \times 10^{-5}]$, and we set the number of epochs to 10.

To detect auditory spans within GLUE, we train an audio span detector based on a BERT-base model. The detector is trained for 5 epochs with a batch size of 16, a learning rate of 1×10^{-5}, and the AdamW optimizer (Loshchilov. and Hutter, 2019). Since GLUE lacks predefined auditory span labels, we generate labels using Qwen2-72B-Instruct-AWQ (Yang et al., 2024) that can be inferred on a

Category	SST-2			QNLI			QQP		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
# Total Words	633,724	17,046	35,025	3,818,145	205,581	204,685	8,050,562	893,635	8,752,112
# Span Words Ratio	4.32	2.29	2.70	0.84	0.19	0.20	0.45	0.34	0.39
# Total Span Words	27,378	391	947	31,976	394	413	36,068	3,034	34,353
# Total Span Count	9,612	173	413	8,343	169	175	10,961	1,045	12,187
Category	**MNLI**			**MRPC**			**STS-B**		
	Train	**Dev**	**Test**	**Train**	**Dev**	**Test**	**Train**	**Dev**	**Test**
# Total Words	11,695,881	286,426	289,461	160,986	17,968	75,042	114,346	34,147	27,052
# Span Words Ratio	2.25	1.63	1.60	0.94	0.55	0.47	2.73	2.31	2.61
# Total Span Words	263,536	4,662	4,633	1,521	98	354	3,124	789	706
# Total Span Count	72,473	1,621	1,577	582	63	231	1,169	337	302

Table 10: Statistics for GLUE benchmark datasets with audio span information.

Category	SST-2			QNLI			QQP		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
# Sentence	67,349	872	1,821	209,486	10,926	10,926	727,692	80,860	781,930
# Words/Sentence	9.41	9.36	9.23	18.22	18.81	18.74	11.06	11.06	11.19
# Total words	633,724	17,046	35,025	3,818,145	205,581	204,685	8,050,562	893,635	8,752,112
Category	**MNLI**			**MRPC**			**STS-B**		
	Train	**Dev**	**Test**	**Train**	**Dev**	**Test**	**Train**	**Dev**	**Test**
# Sentence	785,364	19,630	19,592	7,336	816	3,450	11,498	3,000	2,758
# Words/Sentence	14.89	14.59	14.78	21.94	22.02	21.76	9.94	11.38	9.80
# Total words	11,695,881	286,426	289,461	160,986	17,968	75,042	114,346	34,147	27,052

Table 11: Statistics for GLUE benchmark datasets.

single A100 GPU. The model independently generates labels for auditory spans using a predefined prompt, which is detailed in Appendix G.

F Detailed dataset information of GLUE

The General Language Understanding Evaluation (GLUE) benchmark is utilized to assess the natural language understanding (NLU) capabilities of our model. We selected six tasks for evaluation, including SST-2, QNLI, QQP, MNLI, MRPC, and STS-B.

SST-2 (Stanford Sentiment Treebank): A binary classification task that involves determining the sentiment (positive or negative) of a given sentence extracted from movie reviews.

QNLI (Question Natural Language Inference): A task where the model predicts whether the context sentence contains the answer to a given question, derived from the Stanford Question Answering Dataset (SQuAD).

QQP (Quora Question Pairs): This task focuses on identifying whether a pair of Quora questions are semantically equivalent.

MNLI (Multi-Genre Natural Language Inference): A large-scale dataset for evaluating the model's ability to perform textual entailment across multiple genres, determining whether a given hypothesis is true, false, or undetermined based on a premise.

MRPC (Microsoft Research Paraphrase Corpus): A binary classification task where the model determines whether two given sentences are semantically equivalent.

STS-B (Semantic Textual Similarity Benchmark): A regression task that measures the degree of semantic similarity between two sentences on a scale from 0 to 5.

Evaluation Metrics: For the classification tasks (SST-2, QNLI, QQP, MNLI, MRPC), we utilize accuracy as the primary evaluation metric. For the regression task (STS-B), we employ the Spearman correlation coefficient to evaluate the correlation between the predicted similarity scores and the ground truth.

More details, including dataset statistics, are presented in Appendix C. The table provides an overview of total word counts, auditory span ratios, and detected span counts for each task.

G Prompts for span generation

This section describes the prompt for span labels for auditory knowledge in GLUE. Table 12 is the prompt for auditory.

Prompt for auditory knowledge

You will be given information about specific sentences.
Your task is to extract audio-related spans from the given sentences.
Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.

Extraction Criteria

1. Identify spans within the sentence that describe or represent audio-related information.
2. The spans must directly reference sounds, audio characteristics, or audible phenomena.

Extraction Steps

1. Read the provided sentence.
2. Identify the portion of the sentence that directly refers to audio-related information.
3. If no audio-related spans are found, return **Span: None**.
4. If audio-related spans are found, extract and format them as **Span: {{audio-related span}}**.

Example

Example 1:
Sentence: A dog makes a growling bowwow sound.
Span: `growling bowwow sound`

Example 2:
Sentence: The wind whispers softly through the trees.
Span: `whispers softly`

Example 3:
Sentence: The visual effects in this movie are stunning.
Span: `None`

Input Prompt:
Sentence: {{Sentence}}
Span:

Table 12: Input prompt for generating auditory knowledge span.