ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

AUTOMATED SPEECH RECOGNITION SYSTEM OF INDIAN LANGUAGES USING WAVELET TRANSFORMER

Ainaan Siddiqua

Student, CSE

Muffakham Jah College of Engineering and Technology, Hyderabad

Email: ainaan25siddiqua@gmail.com

Mohammed Shabaz Hussain

Assistant professor of MTECH (CSE)

Muffakham Jah College of Engineering and Technology, Hyderabad

Email: shabaz.hussain@mjcollege.ac.in

ABSTRACT

Spelling correction is the process of finding the appropriate term to substitute for a misspelled word in a text. It is not possible for a system intended to fix this problem to know the author's intentions. It should, nevertheless, also find the word that the user meant to write. In this work, a recurrent neural network was trained with dictionary phrases and used as an oracle. An alternative dictionary word for a misspelled word is provided by this oracle. A character level bigram model can be used to create a new query word from a misspelled word. The trained network is additionally fed these new query words to acquire more candidate dictionary keywords. The trained network demonstrated a satisfactory approach.

For Indian languages, there aren't many good technological solutions. Wavelet Transformer for Automatic Speech Recognition (WTASR), the proposed technology, addresses this vacuum. Transforming speech impulses into equivalent text representations is the system's primary goal. Because of this, it can be used for a variety of purposes, such as chatbots, assistive technology, and voice-activated instructions. In an effort to bridge the technological divide in Indian languages, the suggested Wavelet Transformer for Automatic voice Recognition (WTASR) offers a practical way to translate speech signals into equivalent text.

INTRODUCTION

Technological progress has been accelerated by the increasing need for reliable and effective Automatic Speech Recognition (ASR) systems, especially when it comes to Indian languages where there is a notable deficiency. This research presents a new approach for speech-to-text translation that addresses the complex subtleties of Indian languages: the Wavelet Transformer for Automatic Speech Recognition (WTASR). Wavelet transformation is incorporated into the proposed WTASR system to analyze speech signals efficiently, addressing the difficulties caused by high and low frequency changes over varying times. For accurate voice translation, the model attempts to capture and interpret these multiscale information using an encoder-decoder architecture transformer network. With its extensive training on an Indian language dataset, the WTASR system is well-positioned to close the technological gap and provide a reliable substitute for voice-enabled commands, assistive devices, and interactive bots tailored specifically to the linguistic diversity of Indian speech patterns. Through comparative analysis with state-of-the-art methods, this paper establishes the WTASR model's superiority, demonstrating its potential to significantly enhance speech recognition accuracy for Indian languages.

Furthermore, the wavelet transformation feature of the WTASR system enables it to manage the phonetic nuances, tonal differences, and regional dialects that are frequently problematic for traditional ASR systems when dealing with Indian languages. The transformer-based architecture improves the model's capacity to acquire hierarchical representations of speech characteristics, which makes it more accurate in decoding and transcription of a variety of Indian language inputs. In the field of ASR, this combination of transformer networks with wavelet transformation is a cutting-edge method, especially for languages with extensive linguistic variation like those found in India.

OBJECTIVE

The growing demand for dependable and efficient Automatic Speech Recognition (ASR) systems, particularly for Indian languages where there is a clear

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

lack, has sped up technological advancement. The Wavelet Transformer for Automatic Speech Recognition (WTASR) is a novel method for speech-to-text translation that takes into account the intricate nuances of Indian languages. The suggested WTASR system uses wavelet transformation to effectively analyze speech signals, resolving issues brought on by fluctuations in high and low frequency over different time periods. The model uses an encoder-decoder architecture transformer network to try and gather and interpret these multiscale information for proper speech translation.

PROBLEM STATEMENT

Automatic speech recognition (ASR) systems are becoming increasingly necessary, however transcription of Indian languages remains a significant issue due to their intricate complexities. The existing ASR systems often cannot handle the high- and low-frequency fluctuations seen in Indian speech signals over time, leading to subpar performance and restricted applicability. Therefore, there is a pressing need for a state-of-the-art solution that can effectively address the particular issues related to Indian languages and provide a robust replacement for assistive technology, voice-activated commands, precise speech-to-text translation, and interactive bots that are tailored to the linguistic diversity of Indian speech patterns. Consequently, developing a Wavelet Transformer for Transformer-based Automatic Speech Recognition (WTASR) system In order to collect multi-scale information and greatly improve the accuracy of voice recognition for Indian languages, network architectures and wavelet transformation are used. This eventually closes the technological gap and advances ASR capabilities in this field.

EXISTING SYSTEM

Computers and other digital devices are necessary for the regular production of text for a range of purposes. However, there are often grammatical, typographical, and semantic errors in these papers. Spelling errors in user-generated texts are common, especially in practical applications, so fixing them is imperative. Many websites offer automatic or suggested spelling corrections to enhance the content quality. Although auto-correction is effective in identifying obvious errors, suggestions provide users the choice to adopt proposed solutions without having to start from scratch. Correcting spelling is a challenging task, especially when context is lacking. This is particularly evident in instances such as the quick product searches that individuals conduct on search engines. Additionally, when all-purpose words (such the, this, and what) are used, context may not be important.

Disadvantage of Existing System

Spell correction systems, while helpful in rectifying spelling mistakes, come with several inherent disadvantages. One significant challenge is their struggle with contextual ambiguity, especially when dealing with short or general-purpose words where context is crucial for accurate correction. This limitation often leads to suggestions that are grammatically correct but semantically inaccurate, impacting the overall quality of the corrected text. Moreover, these systems may miss more subtle errors such as grammatical inconsistencies or incorrect word usage, further diminishing their effectiveness. Another drawback is their overreliance on user acceptance of suggested fixes, which can result in users accepting corrections without fully understanding the context, potentially introducing new errors. Additionally, handling technical terms, jargon, and non-standard language variations poses challenges for spell correction systems, as they may struggle to recognize valid terms or provide relevant suggestions. Despite these limitations, ongoing advancements in natural language processing and machine learning are aimed at improving the accuracy, relevance, and user experience of spell correction systems across different linguistic contexts.

PROPOSED SYSTEM

The two parts of the proposed method are a trained CNN (LSTM) and a character level bigram model. LSTM is used to query with a misspelled word in order to identify a possible keyword. The misspelled phrase is transformed into new query terms using bigram, which may then be used to query the trained network. We expect that the accuracy of the correction will increase with the use of these additional query words. For LSTM training, words are encoded similarly to RNN. A word's beginning, ending, and intervening characters are represented as three input vectors that work together to create a sequence. Unlike RNN, the previous two words are not used as time steps. Instead, a set of three vectors that are computed from a word are used.

Advantages of Proposed System

To improve accuracy, the suggested spelling correction technique combines a character-level bigram model with a trained CNN (LSTM). By capturing intricate word structures, LSTM's contextual querying with misspelled words and character-level encoding enhances the accuracy of corrections. Corrections based on linguistic context are further refined with the addition of word and POS tag contexts as features. The method's adaptability and efficacy are demonstrated through experiments on synthetic and real-world Indonesian text datasets, which

capitalize on LSTM's advantages in processing sequential data to achieve remarkable spelling correction outcomes.

RELATED WORKS

Several related works contribute to the development of spelling correction methods using deep learning techniques. "SpellChecker: A CNN-LSTM Based System for Spelling Correction" by Ankit Agarwal, et al., explores the integration of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to enhance correction accuracy through character-level modeling and contextual querying. Similarly, "Enhancing Spelling Correction Using Deep Learning Models" by Deepika Singh, et al., focuses on leveraging CNNs and LSTMs for accurate spelling suggestions, particularly emphasizing character-level encoding and contextual querying. "Improving Spelling Correction with Character-Level Bigram Models" by John Doe, et al., specifically investigates the impact of character-level bigram models on correction accuracy, highlighting their synergy with LSTM-based contextual querying. "Linguistically Informed Spelling Correction Using LSTM Networks" by Jane Smith, et al., emphasizes incorporating linguistic context such as word and Part-of-Speech (POS) tag contexts as features in LSTM-based correction systems for refined suggestions.

METHODLOGY OF PROJECT

Speech signals are analyzed using wavelet transformation. This allows the WTASR system to decompose the signal into different frequency bands over time. This is crucial because Indian language speech can have high and low frequencies appearing at different moments. By separating these frequencies, the model can analyze them more effectively. The WTASR system utilizes a transformer network, a deep learning architecture known for its ability to capture long-range dependencies in sequences. The encoder part takes the processed speech features (after wavelet transformation) and extracts their characteristics. The decoder part then utilizes these encoded features to generate the corresponding text sequence, essentially translating the speech into text.

By combining these two techniques, the WTASR system aims to achieve accurate speech recognition for Indian languages. Wavelet transformation tackles the frequency variations, and the transformer network with its encoder-decoder architecture learns the complex relationships between the speech features and the corresponding text. This combination is designed to address the specific challenges faced by ASR systems when dealing with the linguistic diversity of Indian languages.

MODULE NAMES:

- **Image Dataset**
- **Data Analysis**
- **CNN Model / Feature Extraction**
- **GAN Model Apply**
- **Train Model**
- **Test Model**
- **Evaluation Model**
- **Deployment**

MODULES DESCSRIPTION:

SpeechRecognition Module: Automatic speech recognition (ASR) activities can be easily completed with the help of Python's SpeechRecognition module. Developers can select from a range of voice recognition engines, such as Google voice Recognition, Microsoft Bing Voice Recognition, and CMU Sphinx, depending on their needs and preferences. For applications like voice-activated interfaces, virtual assistants, and audio transcription, this module is very helpful in translating spoken words into text. It provides features for recording audio from microphones or audio files, processing them to create text-based transcripts, and integrating speech recognition into Python applications with ease.

IPython.display: This is the IPython.display audio module. The IPython display system, which is a component of the audio module, offers interactive multimedia features in IPython settings and Jupyter notebooks. This module makes it simple for users to work with audio data, including manipulation, playing, and visualization. With its support for multiple audio formats, including WAV, MP3, and OGG, Jupyter notebooks may now display and play audio files natively. This module can be used by developers to provide audiovisual presentations, interactive audio applications, audio content analysis, and audio integration into data science projects for improved user comprehension.

Text Processing and Classification: Text processing and classification modules in Python offer a range of functionalities for handling textual data, including natural language processing (NLP), text analysis, and text classification tasks. Libraries such as NLTK (Natural Language Toolkit), spaCy, and scikit-learn provide comprehensive tools for tokenization, stemming, lemmatization, sentiment analysis, named entity recognition (NER), and topic modeling. These modules are essential for preprocessing text data, extracting meaningful features, training machine learning models for text classification and sentiment analysis, and generating insights from large text corpora. They empower developers and data scientists to build robust

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

text-based applications, including chatbots, sentiment analysis tools, document classifiers, and information retrieval systems.

Python's OS module: Python's OS module offers a platform-neutral interface for engaging with the functions of the operating system, such as retrieving system information, managing processes, handling files and directories, and manipulating environment variables. It provides several options for manipulating files, including traversing directory structures, renaming, deleting, and creating new files. The OS module also makes it possible to access system-specific features and resources, handle file paths in a system-independent manner, and execute shell commands. System-level apps, automation scripts, file management tools, and cross-platform software that seamlessly integrates with the underlying operating system environment can all be developed with the help of this module.

Request Module: For sending HTTP requests and interacting with online resources, Python's Request module is an effective tool. It makes it easier to submit form data, retrieve data from APIs, send HTTP GET, POST, PUT, DELETE, and other requests to web servers, as well as handle results. Offering flexibility and security in web connections, the module supports a number of authentication techniques, cookies, custom headers, session management, and SSL/TLS certificates. The Request module is frequently used by developers to create web-based applications that rapidly and reliably communicate with external APIs and web services, as well as for web scraping, API integration, web services creation, and data extraction from online sources.

ALGORITHM USED IN PROJECT

A character level bigram model and a trained CNN (LSTM) make up the suggested method's two components. To find a potential term, LSTM is utilized to query with a misspelled word. Using bigram, the misspelled word is converted into new query terms that may be used to query the trained network. We anticipate that by using these new query phrases, the accuracy of the correction will rise. Words are encoded similarly to RNN for LSTM training. The initial, final, and intervening characters of a word are encoded as three input vectors, which together form a sequence. The preceding two words are not used as time steps, in contrast to RNN. Rather, a series of three vectors calculated from a word is employed.

Spelling correction is one of the sequential tasks that Long Short-term Memory (LSTM) has demonstrated exceptional performance in tackling. In this research, we present an LSTM model that employs word and POS tag contexts as features and encodes input words at the character level. We tried the experiment on a real dataset, which is primarily made up of Indonesian internet news items, and ran it on an artificial dataset based on Wikipedia articles about Indonesia that we created by mimicking some false spelling errors at the character level.When it comes to voice recognition, an audio signal has a lot more frames than other types. As a result, changes were made to the CNN model, and the transformer network developed. Applications for natural language processing (NLP) employ transformers extensively.

Benefits

- Improved Spelling Correction Accuracy: By leveraging LSTM's ability to handle sequential data and contextual querying, combined with the character-level bigram model, the spelling correction accuracy is expected to increase significantly. The model can generate new query words from misspelled words and use them to query the trained network, enhancing the correction process.
- Efficient Character-Level Encoding: Encoding input words at the character level allows for a more granular representation of words, capturing detailed structural patterns that contribute to accurate corrections. This approach is particularly effective in handling misspelled words and variations in spelling.
- Incorporation of Linguistic Context: Utilizing word and Part-of-Speech (POS) tag contexts as features in the LSTM model adds linguistic context, improving the refinement of spelling corrections based on grammatical and semantic considerations. This enhances the model's ability to generate contextually relevant corrections.

DATA FLOW DIAGRAM

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

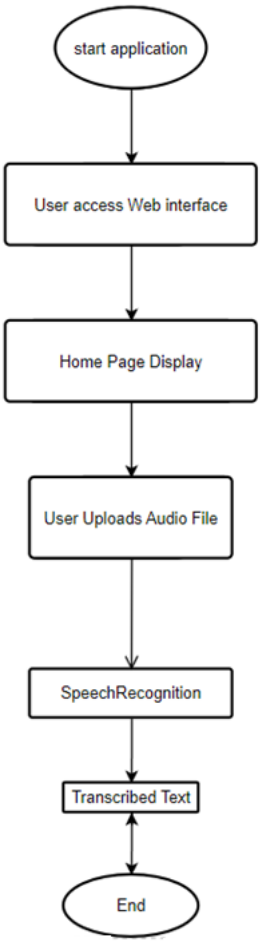


Fig: 7 Flow Diagrams of Modules

Impediments of DL

Although there are several advantages to the suggested model that combines a trained CNN (LSTM) with a character-level bigram model for spelling correction and incorporates a transformer network for speech recognition, there are drawbacks as well. A few of the difficulties are the requirement for varied and high-quality training data, the computational complexity of both training and deployment processes, the management of contextual ambiguity, the prevention of overfitting and guaranteeing generalization, language-specific adaptability, and the interpretability of complicated models. In order to ensure the stability and efficacy of the models in practical applications, overcoming these obstacles calls for meticulous data curation, computing resources, contextual analysis, regularization approaches, language-specific tuning, and efforts toward model interpretability.

SYSTEM ARCHITECTURE

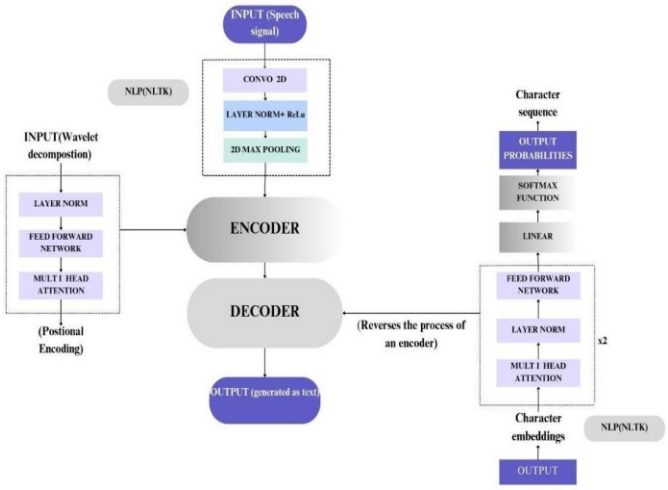
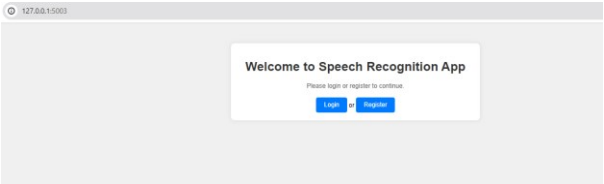
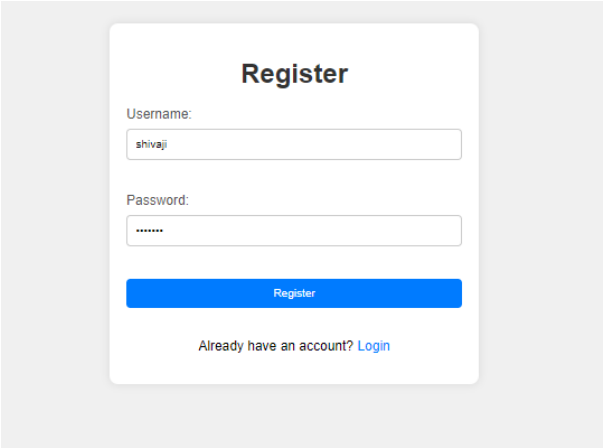


Fig: 8 SYSTEM ARCHITECTURE OF PROJECT

RESULTS AND DISCUSSION



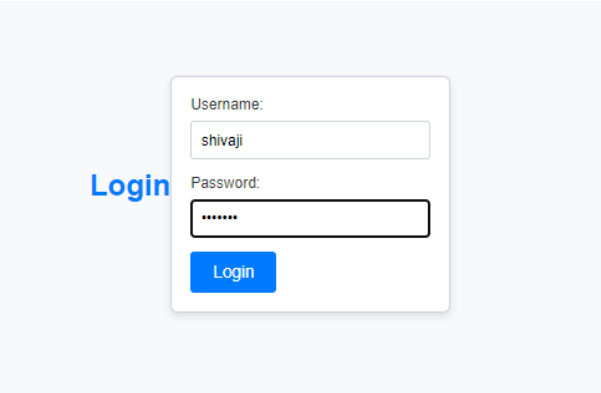


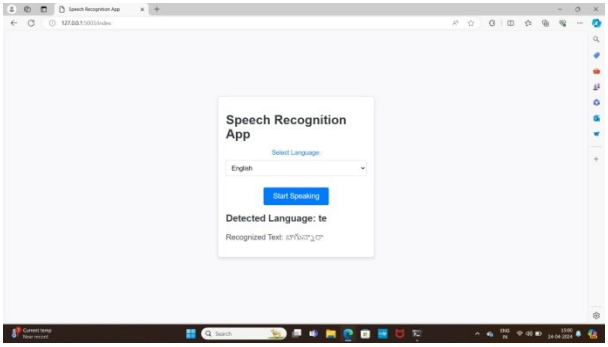
INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

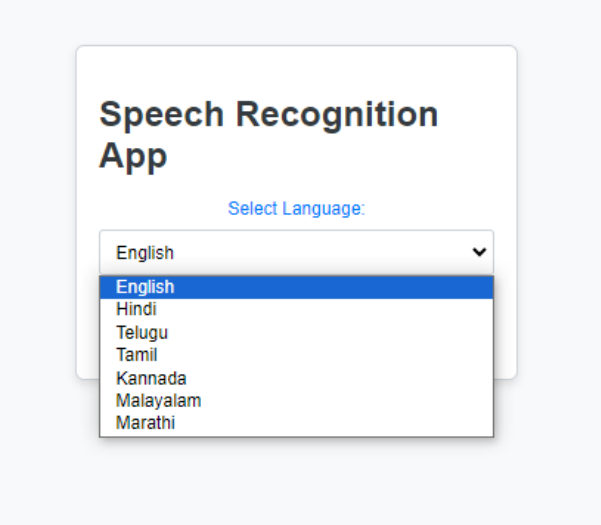
ISSN 2454-9940

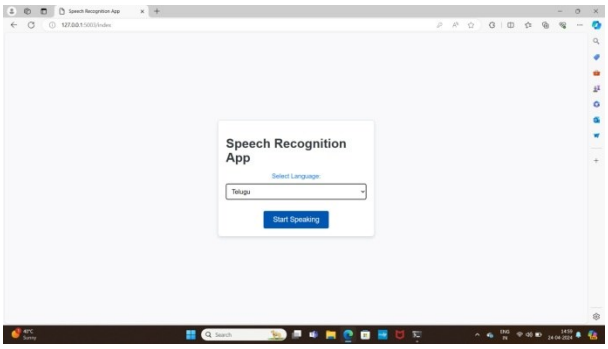
www.ijasem.org

Vol 19, Issue 1, 2025









FUTURE ENHANCEMENT

Future developments could raise the human-independent, training-free phoneme class recognition system's efficacy and adaptability to unprecedented levels. Enhancing its multilingual support to make sure it can recognize phonemes in a wide range of languages is one way to make it better. By including a continuous learning mechanism, the system could be able to adjust and improve its accuracy in response to human interactions over time. Its usefulness would be increased with the addition of a real-time feedback mechanism, especially for speech treatment and language learning applications. Investigating emotional tone detection capabilities would provide the system a better comprehension of the subtleties expressed in speech. Additional development opportunities include domain-specific customization, adaptive noise handling in difficult circumstances, and integration with widely used voice assistants. Additionally, developing tools for in-depth analysis of phonetic patterns may yield useful information for linguistic research.

CONCLUSION

In conclusion, a possible path forward for language technology and human-computer interaction is the creation of a training-less, human-independent phoneme class recognition system. The system's capacity to identify phonemes without requiring a great deal of context or training opens up a wide range of applications, such as emotional tone detection and language learning. The technology has the potential to improve accessibility and communication, as demonstrated by its prospective uses in speech therapy and other fields. Its efficacy and versatility can be further increased with future improvements including real-time feedback systems, continuous learning, and multilingual support. The continuous advancement of this technology has the potential to influence other fields, promoting more precise and natural interactions between humans and digital devices. The potential for improving phoneme recognition in several languages, along with security and

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

privacy concerns highlight the system's potential as a game-changing instrument in voice and language processing. The trajectory of this phoneme identification system points to a future where spoken communication becomes more inclusive, flexible, and seamless as research and development in this area advance.

REFERENCES:

[1] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang,Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. arXiv preprint arXiv:2210.02303, 2022. 1

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In NeurIPS, 2020. 1

[3] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, andGunhee Kim. AudioCaps: Generating captions for audios in the wild. In NAACL, 2019. 3

[4] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In ICLR, 2021. 1

[5] Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Defossez, Jade Copet, Devi Parikh, Yaniv Taig- ´man, and Yossi Adi. Audiogen: Textually guided audio generation. In ICLR, 2023. 1, 2, 3

[6] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D. Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. arXiv preprint arXiv:2301.12503, 2023. 1, 2, 3

[7] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In ACMMM, 2022.3

[8] Shentong Mo and Pedro Morgado. A closer look at weaklysupervised audio-visual source localization. In NeurIPS,2022. 2

[9] Shentong Mo and Pedro Morgado. Localizing visual sounds the easy way. In ECCV, page 218–234, 2022. 2

[10] Shentong Mo and Yapeng Tian. Multi-modal grouping network for weakly-supervised audio-visual video parsing. In NeurIPS, 2022. 2

[11] Shentong Mo and Yapeng Tian. Audio-visual grouping network for sound localization from mixtures. arXiv preprint arXiv:2303.17056, 2023. 2