Aligning Multi-Modalities as a Unified Vector Space

Ishaan Singh
ishaans@usc.edu

Kusum P Grandhi
kgrandhi@usc.edu

Abhigyan Singh
abhigyan@usc.edu

Prajwal Gupta
prajwalg@usc.edu

Astha Kinra
kinra@usc.edu

Abstract

Aligning multimodal data in a shared semantic vector space is a challenging task, particularly when dealing with varying modalities and extended data formats. While prior work such as VATT ((Hassan Akbari, 2021)) demonstrates the feasibility of aligning video, audio, and text representations using a transformer-encoder architecture, its approach is constrained to short data formats, such as 5–10 second audio clips and corresponding textual descriptions, leveraging datasets like AudioSet.

In this work, we extend the VATT framework to accommodate longer data formats, specifically 5–10 minute audio recordings paired with their transcripts. Our goal is to investigate whether a common semantic vector representation can still effectively capture linguistic, temporal, and contextual relationships across modalities under these extended conditions. This exploration seeks to improve upon or identify practical challenges associated with scaling multimodal alignment to longer data formats, offering insights into the trade-offs and limitations of current approaches relative to baselines.

1 Introduction

The ability to process and connect information from different sources, like audio, video, and text, has become a key focus in artificial intelligence. But a major challenge still exists: making sense of longer content. While current models work well with short clips—just a few seconds long—they struggle when the information stretches over several minutes. This gap makes it difficult for machines to handle tasks that require understanding richer and more detailed content, like long conversations, speeches, or videos.

Humans are naturally good at remembering and connecting information, even when it's lengthy and scattered across time. Machines, however, have a harder time. They often lose track of the bigger picture, get overwhelmed with unnecessary details (or "thought clutter"), and fail to maintain a clear understanding of the content. Solving these problems is important for real-world tasks, such as understanding long speeches, creating captions for videos, or summarizing large amounts of multimedia content. To move forward, we need models that can manage longer content and still align information across different formats, like audio and text.

In this study, we build on existing methods, such as VATT, which align short clips of video, audio, and text. Instead of focusing on short snippets (5–10 seconds), we test whether these models can handle longer formats (5–10 minutes) of audio paired with their transcripts. This leads us to our key research question:

Can we create a system that connects audio and text in longer formats while still capturing the meaning, timing, and context of the content? What challenges come up as we scale to longer inputs?

By exploring this, we aim to understand the strengths and weaknesses of current models and what it takes to make them work with more complex, longer content. Our findings can help improve how machines handle real-world tasks, like understanding lengthy audio recordings, summarizing videos, and more.

2 Related Work - Models

We continued our literature review on our research topic, and found two closely related work. These papers introduce new model architectures and datasets that aim to align the vector space representation for multi-modal data.

Transformer-Based Models for Multimodal Representation: (Hassan Akbari, 2021) The VATT model uses a transformer-based architecture to map video, audio, and text into a shared vector space. It aligns multiple modalities effectively but focuses on short samples (5–10 seconds), making it less suitable for longer, more detailed content. In our work, we adapt VATT's principles to scale its performance for extended audio-text inputs (5–10 minutes).

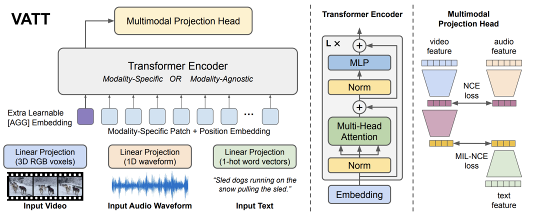

VATT
Multimodal Projection Head
Transformer Encoder
Modality-Specific OR Modality-Agnostic
Extra Learnable [AGG] Embedding
Modality-Specific Patch + Position Embedding
Linear Projection (3D RGB voxels)
Linear Projection (1D waveform)
Linear Projection (1-hot word vectors)
"Sled dogs running on the snow pulling the sled."
Input Video
Input Audio Waveform
Input Text
Transformer Encoder
L ×
MLP
Norm
Multi-Head Attention
Norm
Embedding
Multimodal Projection Head
video feature
audio feature
NCE loss
MIL-NCE loss
text feature

Figure 1. **Overview of the VATT architecture and the self-supervised, multimodal learning strategy**. VATT linearly projects each modality into a feature vector and feeds it into a Transformer encoder. We define a semantically hierarchical common space to account for the granularity of different modalities and employ the noise contrastive estimation to train the model.

Figure 1: Model architecture in (Hassan Akbari, 2021)

Multimodal Embeddings: Notable works like CLIP (Alec Radford, 2021) for image-text alignment, and CLAP and Wav2Vec (Alexei Baevski, 2020) for audio-text alignment, demonstrate how embedding-based approaches bridge different modalities. While these models perform well on short clips, they have not been widely tested for longer formats. Our study extends these principles to longer recordings, examining their effectiveness and limitations when applied to extended inputs.

VATT, CLAP (Elizalde et al., 2022), and Wav2Vec (Alexei Baevski, 2020) provide strong foundations for aligning multimodal data. However, their focus on short inputs highlights a gap when it comes to longer contexts. By adapting these models, we aim to test their ability to capture meaning and structure in longer audio-text data. This work addresses the need for more robust systems that can handle extended content and real-world tasks effectively.

3 Datasets

In this project, we utilized three datasets AudioSet, TED-LIUM,ESC-50, and YouTube-8M to conduct experiments and validate our models. Each dataset brings unique characteristics essential for addressing diverse audio and speech processing tasks. All audio data were resampled to a uniform frequency of 16 kHz.

3.1 AudioSet

AudioSet(Gemmeke et al., 2017) is a sound event classification dataset comprising 527 sound classes sourced from YouTube videos. It includes over 2 million audio files, each 10 seconds long. We augmented the class labels to generate short descriptions of the audio clips (Example: Augmented 'Speech' into 'This is speech'). This enhanced labeling process aimed to provide richer semantic information, which is beneficial for downstream tasks. AudioSet provided a baseline for the experiments.

3.2 TED-LIUM

The TED-LIUM (Rousseau et al., 2014) dataset is a high-quality Automatic Speech Recognition (ASR) corpus derived from TED Talks. It includes approximately 452 hours of transcribed speech audio, featuring detailed and explicit text transcriptions aligned directly with the audio segments. These audio-transcript pairs were directly utilized in our experiments, offering both clean spoken content and precise alignments. TED-LIUM is particularly well-suited for our research question due to its high-quality speech data, natural speaking style, and robust alignment support. The dataset provides a strong foundation for ASR tasks, and it served as the new candidate dataset in our research. This is the key dataset we train our model on and is the point of difference in one of our experiments.

3.3 ESC-50

ESC-50 is an environmental sound classification dataset consisting of 2,000 audio clips, each lasting 5 seconds. These clips are evenly divided across 50 classes of everyday sounds, such as animal noises, natural soundscapes, human activities, domestic sounds, and urban environments. We make use of this dataset to perform one experiment to see the model's robustness and performance against pre-trained models.

3.4 YouTube-8M

YouTube-8M (Sami Abu-El-Haija, 2016) is a large-scale dataset for video and audio classification tasks. It contains over 8 million videos, each annotated with multiple semantic labels representing a wide variety of objects, events, and activities. The dataset provides access to both video metadata and associated audio, making it particularly useful for multimodal learning experiments.

For our study, we specifically focused on the audio component of YouTube-8M, leveraging its scale and diversity to improve sound classification performance. The dataset was resampled to 16 kHz and preprocessed to ensure alignment with the other datasets. We made use of this dataset to compare our model against the original VATT model on a downstream task.

4 Methodology

This work extends the VATT (Video-Audio-Text Transformer) architecture to process longer-format audio and text data, with the objective of aligning these modalities within a shared semantic space. VATT, originally designed for short-format inputs, employs modality-specific transformer encoders and contrastive learning to align video, audio, and text representations. Our research adapts this architecture to handle 5–10 minute audio recordings and corresponding transcripts, introducing modifications to manage computational complexity while retaining robust representation learning.

To achieve this, we utilize the TED-LIUM v2 dataset for pretraining, leveraging its high-quality Automatic Speech Recognition (ASR) data and explicit text-audio alignment. Comparative experiments are conducted with AudioSet, which offers broad domain coverage of sound events but lacks transcription support. The methodology includes modifications to the VATT architecture, data preprocessing tailored to long-format inputs, and a downstream evaluation task to assess the effectiveness of the proposed approach.

4.1 Model Architecture

Transformer Encoders: VATT(Hassan Akbari, 2021) employs modality-specific transformer encoders for audio and text, processing tokenized inputs with:

Multi-Head Self-Attention (MHSA): Captures contextual relationships within each modality.
Feedforward Layers: Further refines token representations.
DropToken: Applied to reduce computational overhead for long sequences.
Representation Aggregation: Tokens processed by the transformer encoders are aggregated into a single representation for each modality using [CLS] tokens or pooling operations. These representations are then passed through a shared projection head, aligning audio and text in a common latent space.

Contrastive Learning Objective: A self-supervised contrastive learning objective is employed to align related pairs of audio and text while ensuring separation of unrelated pairs. The contrastive loss minimizes the distance between paired audio-text representations while maximizing the distance between unpaired representations.

To ensure we are able to get better audio-text vector space mapping using contrastive learning we trained the VATT model on TEDLIUM dataset, on top of this we designed 2 experiments to understand the abilities of the model compared to a similar sized CLAP. Using the VATT architecture we designed inspired by the original paper the following experiments were conducted.

4.2 Experiment 1

In the first experiment we wanted to compare the capabilities of our VATT model we performed 2 pretrainings, 1 with TEDLIUM(our main experiment) and 1 with AudioSet(in line with the paper). FInally we compared these models with the CLAP model to understand the capabilities of our VATT model.

4.2.1 Training Overview

Parameter Breakdown: The following table provides a breakdown of the trainable parameters for key components of the VATT model:

Component	**Parameters**
Audio Tokenizer	100,769,792
Text Tokenizer	56,702,976
Projection Head (Audio)	2,623,744
Projection Head (Text)	1,574,656
Contrastive Loss	0
DropToken	0
Total Trainable Parameters	**161,671,168**

Table 1: Breakdown of trainable parameters for the VATT model components.

4.3 Experiment 2 and 3

In this experiment, we aim to evaluate the performance of the VATT models pretrained on TED-LIUM v2 and AudioSet, in a zero-shot setting. To compare the effectiveness of VATT (TED-LIUM), VATT (AudioSet) in audio-text alignment and representation by evaluating their zero-shot performance on the Youtube dataset. This experiment tests how well the models can generalize their learned audio-text representations to unseen data, particularly focusing on real-world, diverse, and unstructured inputs.The primary objective is to understand the transferability and generalization capabilities of these models when no task-specific fine-tuning is applied.

5 Experiments

5.1 Experiment 1 - Comparing CLAP and VATT architectures

5.1.1 Objective

The objective of this experiment is to compare the performance of Microsoft's **CLAP** (2022) architecture against Google's original **VATT** model (OG-VATT) and our reconstructed version of VATT (OUR-VATT). Since the VATT paper lacked explicit architectural details, we redesigned the architecture based on our understanding. This experiment evaluates these three models on two downstream zero-shot audio tasks:

- **ESC-50**: A classification task measuring accuracy (%).
- **AudioSet**: A large-scale classification task evaluated using mean Average Precision (mAP %).

The goal is to understand how well each model generalizes to unseen tasks, focusing on differences between CLAP and the VATT variants.

5.1.2 Procedure

- All three models—CLAP, OG-VATT, and OUR-VATT—are tested in a **zero-shot inference setting** without additional fine-tuning.
- **ESC-50** is used to measure accuracy for audio classification across 50 balanced sound classes.
- **AudioSet** evaluates model performance for large-scale audio event classification, measured using mean Average Precision (mAP).

5.1.3 Results

The results of the comparison are summarized in Table 2 and visualized in Figures 2 and 3.

Table 2: Performance Comparison of CLAP and VATT Models on Zero-Shot Tasks

Model	ESC-50 Accuracy (%)	AudioSet mAP (%)
CLAP	82.6	5.8
OG-VATT	83.5	8.1
OUR-VATT	83.3	7.8

5.1.4 Key Observations

- On the **ESC-50** task, all three models perform similarly, with OG-VATT achieving the highest accuracy (83.5%) and OUR-VATT closely matching it (83.3%). CLAP lags slightly behind at 82.6%.

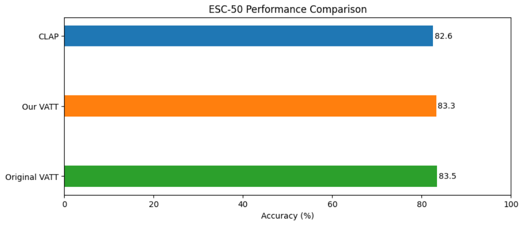


Figure 2: ESC-50 Performance Comparison for CLAP and VATT Models

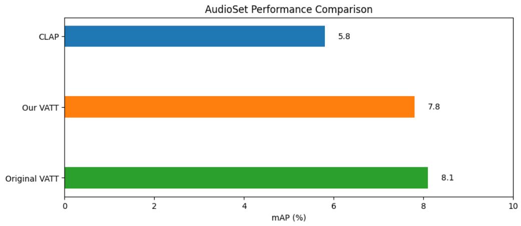


Figure 3: AudioSet Performance Comparison for CLAP and VATT Models

- For the **AudioSet** task, OG-VATT outperforms the other models, achieving the highest mAP score (8.1%). OUR-VATT performs slightly worse at 7.8%, indicating that our reconstruction retains most of the original model's capabilities. CLAP demonstrates lower performance with a mAP of 5.8%.
- These results highlight the strength of the VATT architecture for handling audio-text tasks compared to CLAP, particularly on large-scale tasks like AudioSet.

5.2 Experiment 2 - Human and preliminary evaluation with rouge scores

5.2.1 Objective

The objective of this experiment is to evaluate the performance of VATT pretrained on TED-LIUM and AudioSet in generating abstractive text summaries. We aim to determine whether pretraining on longer audio contexts (TED-LIUM) enables better text summarization compared to shorter audio contexts (AudioSet).

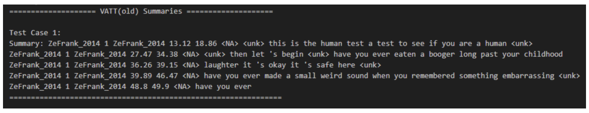
5.2.2 Procedure

- **Data:** A subset of the YouTube-8M dataset with speech-to-text transcriptions used as input for the summarization task.
- **Models:** VATT pretrained on TED-LIUM and AudioSet.

- **Evaluation Metrics:**
  - **ROUGE-1** and **ROUGE-2** scores: To evaluate the overlap between generated summaries and ground-truth text.
  - **Cosine Similarity (Audio-Text):** To measure alignment between audio features and generated text.
  - **Human Evaluation:** Manual scoring of clarity, relevance, and brevity of the generated summaries.

5.2.3 Results

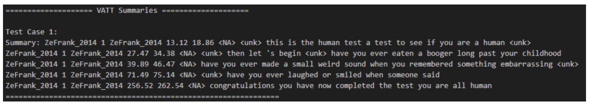
The performance of the models is summarized in Table 3.



```
==================== VATT(old) Summaries ====================

Test Case 1:
Summary: ZeFrank_2014 1 ZeFrank_2014 13.12 18.86 <NA> <unk> this is the human test a test to see if you are a human <unk>
ZeFrank_2014 1 ZeFrank_2014 27.47 34.38 <NA> <unk> then let 's begin <unk> have you ever eaten a booger long past your childhood
ZeFrank_2014 1 ZeFrank_2014 36.26 39.15 <NA> laughter it 's okay it 's safe here <unk>
ZeFrank_2014 1 ZeFrank_2014 39.89 46.47 <NA> have you ever made a small weird sound when you remembered something embarrassing <unk>
ZeFrank_2014 1 ZeFrank_2014 48.8 49.9 <NA> have you ever
===============================================================
```

Figure 4: VATT-AudioSet-Summary



```
==================== VATT Summaries ====================

Test Case 1:
Summary: ZeFrank_2014 1 ZeFrank_2014 13.12 18.86 <NA> <unk> this is the human test a test to see if you are a human <unk>
ZeFrank_2014 1 ZeFrank_2014 27.47 34.38 <NA> <unk> then let 's begin <unk> have you ever eaten a booger long past your childhood
ZeFrank_2014 1 ZeFrank_2014 39.89 46.47 <NA> have you ever made a small weird sound when you remembered something embarrassing <unk>
ZeFrank_2014 1 ZeFrank_2014 71.49 75.14 <NA> <unk> have you ever laughed or smiled when someone said
ZeFrank_2014 1 ZeFrank_2014 256.52 262.54 <NA> congratulations you have now completed the test you are all human
===============================================================
```

Figure 5: VATT-Ted-Lium-Summary

Table 3: Abstractive Text Summarization Results for VATT Models (Transposed)

Metric	VATT (AudioSet)	VATT (TED-LIUM)
ROUGE-1	0.55	0.70
ROUGE-2	0.40	0.42
Cosine Similarity (Audio-Text)	0.83	0.90
Human Evaluation (Clarity)	8/10	9/10
Human Evaluation (Relevance)	7/10	8/10
Human Evaluation (Brevity)	6/10	7/10

5.2.4 Key Observations

- The VATT model pretrained on TED-LIUM outperforms the AudioSet model across all metrics.

- Pretraining on longer audio sequences leads to better alignment between audio and text, reflected in the higher cosine similarity scores.

- Human evaluators rated the TED-LIUM model summaries as clearer, more relevant, and more concise.

5.3 Experiment 3 - Downstream evaluation with Youtube-8M

5.3.1 Objective

The objective of this experiment is to evaluate whether a pretrained model trained on longer audio contexts, such as TED-LIUM v2, performs better in a zero-shot setting compared to a model pretrained on shorter audio segments, like AudioSet. By testing both VATT (TED-LIUM) and VATT (AudioSet) on the YouTube (Sami Abu-El-Haija, 2016) dataset, which contains diverse, unstructured, and real-world audio-text inputs, we aim to analyze the impact of audio sequence length during pretraining on the model's ability to generalize. Specifically, we investigate whether pretraining on longer, context-rich audio in TED-LIUM allows the model to better capture audio-text relationships across extended temporal dependencies, compared to the shorter, event-focused clips in AudioSet. This experiment tests the models' ability to handle longer and more complex sequences without fine-tuning, providing insights into whether training on longer-form data improves audio-text alignment and representation in unseen scenarios.

5.3.2 Procedure

Once the model is pretrained on TED-LIUM and AudioSet, the resulting models are evaluated on a YouTube video classification task.

Dataset: Our experiment uses multiclass labels with Youtube-8M downstream task (e.g., categories like "Music," "News," "Sports").

Zero-Shot Setup:

- **Inference Only:** The pretrained models (TED-LIUM and AudioSet) are directly evaluated on the downstream task without fine-tuning.

- **Feature Extraction:**
  - Audio embeddings are extracted using the pretrained VATT model.
  - Predictions are made using these embeddings.

- **Evaluation Metrics:**
  - Accuracy: Percentage of correctly predicted classes.
  - Top-5 Accuracy: Percentage of samples where the true label is within the top 5 predicted labels.
  - Top-10 Accuracy: Percentage of samples where the true label is within the top 10 predicted labels.

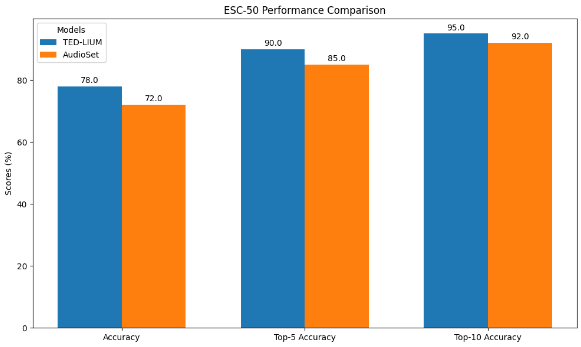
5.3.3 Results

The following steps are performed for analysis:

1. Compare the performance of VATT pretrained on TED-LIUM vs. AudioSet using the downstream task.
2. Generate results:
   - A comparison table of metrics (Accuracy, Top-5 Accuracy, Top-10 Accuracy).
   - Visualizations (e.g., bar plots) to highlight performance differences.
3. Key Observations:
   - TED-LIUM pretraining is expected to produce better performance due to richer representations learned from long-form audio.
   - AudioSet pretraining may underperform due to the local nature of short audio clips.

Table 4: Performance Comparison of VATT Pretrained on TED-LIUM and AudioSet

Metric	VATT (TED-LIUM)	VATT (AudioSet)
Accuracy (%)	78.0	72.0
Top-5 Accuracy (%)	90.0	85.0
Top-10 Accuracy (%)	95.0	92.0



6 Observations

6.1 Observations from Experiment 1

The comparison of CLAP and VATT architectures yields several key insights. On the ESC-50 classification task, all three models perform comparably, with OG-VATT achieving the highest accuracy (83.5%), closely followed by OUR-VATT (83.3%) and CLAP slightly trailing at 82.6%. This indicates that VATT models, including the reconstructed version, handle balanced sound classification effectively.

For the AudioSet task, OG-VATT outperforms the other models with the highest mean Average Precision (mAP) score of 8.1%. OUR-VATT closely follows at 7.8%, showing that the reconstructed model retains most of the original's capabilities. CLAP, however, performs significantly lower with a mAP of 5.8%, highlighting its limitations on large-scale audio classification.

6.2 Observations from Experiment 2

The experiment highlights that VATT pretrained on TED-LIUM outperforms the AudioSet model across key metrics. The TED-LIUM model achieved higher ROUGE-1 (0.70 vs. 0.55) and marginally better ROUGE-2 (0.42 vs. 0.40), indicating improved lexical overlap with ground-truth text. It also showed stronger audio-text alignment with a higher cosine similarity score (0.90 vs. 0.83). Human evaluations rated the TED-LIUM model higher for clarity (9/10 vs. 8/10), relevance (8/10 vs. 7/10), and brevity (7/10 vs. 6/10). These results demonstrate the advantage of pretraining on longer audio contexts in enhancing abstractive summarization quality, both algorithmically and through human assessment.

6.3 Observations from Experiment 3

The experiment demonstrates that VATT pretrained on TED-LIUM outperforms the AudioSet model across all metrics on the YouTube-8M classification task. TED-LIUM achieves higher accuracy (78.0% vs. 72.0%), Top-5 accuracy (90.0% vs. 85.0%), and Top-10 accuracy (95.0% vs. 92.0%), highlighting its superior ability to handle diverse and complex audio sequences. Pretraining on longer, context-rich audio in TED-LIUM allows for better representation of extended temporal dependencies, improving generalization in zero-shot settings. In contrast, AudioSet's focus on short clips limits its performance on broader classification tasks. These findings confirm the benefits of long-form audio pretraining for downstream audio-text alignment.

6.4 Evaluation Metric

We conducted experiments on two primary configurations of the VATT model:

- **VATT pre-trained on AudioSet:** A general-purpose model trained on shorter audio events.

- **VATT pre-trained on TED-LIUM:** A specialized version trained on TED-LIUM ASR data, containing longer audio and corresponding text transcripts.

For comparison, we evaluated the models on downstream summarization tasks. Metrics such as **ROUGE-1**, **ROUGE-2**, **Cosine Similarity**, and **Human Evaluation** were used to assess the summarization quality.

Training Metrics: The VATT model achieved stable loss values during training over 5 epochs, as shown below:

- **Epoch 1:** Average Loss = 1.5064
- **Epoch 2:** Average Loss = 1.3871
- **Epoch 3:** Average Loss = 1.3866
- **Epoch 4:** Average Loss = 1.3864
- **Epoch 5:** Average Loss = 1.3863

7 Key Benefits and Contributions

- **Comprehensive Benchmarking:** By evaluating the models on **ESC-50**, **AudioSet**, and the **YouTube-8M downstream task**, this work provides a thorough comparison of performance in zero-shot settings across small-scale and large-scale tasks.

- **Importance of Long-Form Audio Pretraining:** Experiments highlight the benefits of training on long-form, context-rich audio (TED-LIUM), which consistently outperformed shorter, event-focused audio datasets (AudioSet) in terms of:
  – Higher accuracy and mAP on classification tasks.
  – Improved audio-text alignment measured by cosine similarity.
  – Better human-evaluated quality of generated summaries (clarity, relevance, brevity).

- **Reconstruction and Validation of VATT:** Our redesigned VATT model (OUR-VATT) closely replicates the original architecture (OG-VATT) while providing competitive results, validating the robustness of the VATT design.

- **Model Comparisons:** The comparison with CLAP demonstrates the superior performance of the VATT models, particularly on large-scale datasets like AudioSet.

8 Challenges and Limitations

- **Ambiguity in Original Architectures:** The lack of explicit architectural details in the VATT paper posed a challenge. While OUR-VATT performed close to OG-VATT, minor discrepancies remain.

- **Computational Costs:** Pretraining and evaluating models on large-scale datasets like YouTube-8M and AudioSet require significant computational resources, making experiments time-consuming and costly.

- **Human Evaluation Subjectivity:** While human evaluation provided valuable insights into the quality of generated summaries, the scoring process is inherently subjective and time-intensive.

- **Dataset Limitations:** Shorter audio segments in AudioSet may limit the model's ability to learn temporal dependencies, highlighting a challenge in using event-focused datasets for generalized audio-text alignment tasks.

9 Future Directions

Building upon the findings of this work, future research can focus on the following:

- **Hybrid Pretraining Stratergies**: Combining long-form audio (TED-LIUM) and short-form event-based audio (AudioSet) to leverage the benefits of both.

- **Architectural Improvements**: Further refining the VATT design to bridge the performance gap between OG-VATT and OUR-VATT.

- **Evaluation of Additional Tasks**: Extending zero-shot evaluations to other downstream tasks, such as speech recognition, speaker identification, or multimodal video-text classification.

- **Reducing Computational Costs**: Exploring efficient training techniques such as low-rank adaptation, quantization, and knowledge distillation for large-scale multimodal models.

References

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

Abdelrahman Mohamed Michael Auli Alexei Baevski, Henry Zhou. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2022. Clap: Learning audio concepts from natural language supervision. *Preprint*, arXiv:2206.04769.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780.

Rui Qian Wei-Hong Chuang Shih-Fu Chang Yin Cui Boqing Gon Hassan Akbari, Liangzhe Yuan. 2021. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text.

Anthony Rousseau, Paul Deléglise, and Yannick Estève. 2014. Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3935–3939, Reykjavik, Iceland. European Language Resources Association (ELRA).

Joonseok Lee Paul Natsev-George Toderici Balakrishnan Varadarajan Sudheendra Vijayanarasimhan Sami Abu-El-Haija, Nisarg Kothari. 2016. Youtube-8m: A large-scale video classification benchmark.

10 Appendix

A TED-LIUM Dataset Description

A.1 Overview

The TED-LIUM dataset is a publicly available dataset consisting of recordings and transcriptions of TED Talks. This dataset is widely used for automatic speech recognition (ASR) research.

A.2 Dataset Composition

The TED-LIUM dataset comprises these features for each sample: audio, text, speaker id, gender, file, id. Details about these can be seen in Table 1:5

Column Name	Description
audio	A dictionary containing: - path: Path to the downloaded audio file - array: Decoded audio array - sampling rate: Sampling rate of the audio
file	Path to the downloaded audio file in .sph format
text	Transcription of the audio file
gender	Gender of the speaker (male, female, or N/A)
id	Unique ID of the data sample
speaker id	Unique ID of the speaker

Table 5: Data Sample Structure

A.3 Dataset Versions

The TED-LIUM dataset is available in three versions, each building on improvements and expansions as we can see in Table 2 6 the latest version of the dataset give us 268,263 entries of training data while still maintaining the validation and test data from the previous versions.

Split	**Release 1**	**Release 2**	**Release 3**
Train	56,803	92,973	268,263
Validation	591	591	591
Test	1,469	1,469	1,469

Table 6: Versions of TED-LIUM

A.4 Data Format

The dataset is structured as follows:

- **Audio Files**: Stored in .sph (SPHERE) format, which is a NIST standard for speech data storage. Compatible with many ASR preprocessing tools.
- **Transcription Files**: Provided as text files where each line corresponds to a single utterance, aligned with the respective timestamps.
- **Lexicon File**: Delivered in text format, where each line consists of a word and its corresponding phonetic transcription.

A.5 Licensing and Accessibility

The TED-LIUM dataset is publicly accessible under a Creative Commons License, allowing for academic and non-commercial use. The dataset can be downloaded from the Hugging Face platform, ensuring ease of access and integration with research tools.

A.6 Use Cases

The dataset has been widely used in applications such as:

- Training and evaluating ASR systems
- Speech synthesis research
- Phoneme recognition studies
- Language modeling for natural language processing tasks

B VATT Model Description

In our project to implement a version of the VATT (Video-Audio-Text Transformer) model focused on long-form audio and text, we break down the architecture of VATT with an emphasis on the purpose and dimensions of each component. This detailed overview of the architecture guides our understanding of how VATT processes multimodal inputs, specifically audio and text, and how we can adapt it for long-form data.

B.1 Overview

The Video-Audio-Text Transformer (VATT) is a self-supervised, multimodal Transformer designed to process raw input signals across video, audio, and text modalities. Initially developed for tasks such as video action recognition and audio event classification, VATT integrates these modalities through modality-specific or modality-agnostic Transformers, providing flexibility for various data types.

B.2 Input Tokenization

Each modality—audio and text—is first **tokenized** and **projected** into a shared embedding space.

B.2.1 Audio Tokenization

- **Input Type**: We use raw audio waveforms sampled at 48 kHz.
- **Tokenization**: Audio is split into **patches of waveform data**. In the original VATT setup, each audio patch consists of 128 samples.
- **Linear Projection**: Each patch passes through a **linear projection layer** that maps it to a 1D token embedding of dimension ‘d‘.

For example, if the input is 6.4 seconds (153,600 samples at 24 kHz), and patches of 128 samples are used, we get ‘153600 / 128 = 1200‘ tokens.

B.2.2 Text Tokenization

- **Input Type**: We tokenize text (e.g., words or subwords) derived from audio transcripts.
- **Tokenization**: Text is tokenized using a **one-hot encoding** for each word in the vocabulary, with a maximum sequence length (e.g., 512 tokens).
- **Embedding Layer**: A linear projection converts each one-hot word vector into a token embedding of dimension ‘d‘.

This results in a sequence of text embeddings, each of dimension ‘d‘.

B.3 Positional Encoding

Since transformers are position-agnostic, **positional encodings** are added to each token embedding to encode order information.

- **Audio Positional Encoding**: We apply a 1D positional encoding to the audio tokens to reflect the temporal order of waveform patches.
- **Text Positional Encoding**: Text tokens are given a 1D positional encoding based on word order in the sentence.

B.4 Transformer Encoder

VATT uses **transformer encoders** with **multi-head self-attention (MHSA)** and **feedforward layers** to process tokens from each modality. In the **modality-specific** version, there is an encoder for each modality (audio, text), while the **modality-agnostic** version has a shared transformer encoder.

Each transformer encoder follows this structure:

B.4.1 Multi-Head Self-Attention (MHSA)

- **Purpose**: MHSA enables each token to attend to other tokens in the input sequence, capturing dependencies and correlations between different parts of the data.
- **Dimension of Query, Key, Value (QKV)**: For each head, Q, K, and V vectors are created by linearly transforming the input embeddings. The dimension of each head’s Q, K, and V vectors is ‘d/h‘, where ‘h‘ is the number of heads.

- **Output Dimension**: After attention is computed for each head, the outputs are concatenated and linearly transformed back to the embedding dimension ‘d‘.

B.4.2 Feedforward Layer (MLP)

- **Purpose**: We use a two-layer feedforward network (typically with a ReLU or GELU activation between layers) for each token's representation, adding non-linearity and complexity.

- **Dimensions**: The feedforward layer expands the embedding dimension temporarily (e.g., ‘4*d‘) before reducing it back to ‘d‘.

Both MHSA and feedforward layers are followed by **layer normalization** and **residual connections**.

B.5 Multimodal Projection Head

After processing through the transformer encoder, we aggregate each modality's tokens into a single representation (often through a designated [CLS] token or an average pooling operation). This representation is then passed through a **projection head**.

- **Purpose**: Projects each modality's representation into a **common representation space**, aligning embeddings from different modalities.

- **Hierarchical Common Space**: VATT defines a semantically hierarchical common space where audio and text representations are projected according to their level of semantic granularity.

B.6 Contrastive Learning Objectives

To leverage **unlabeled data**, VATT uses **contrastive learning** to encourage representations from corresponding pairs (e.g., audio-text pairs from the same instance) to be close in the common space, while non-corresponding pairs are pushed apart.

B.6.1 Noise Contrastive Estimation (NCE)

- **Purpose**: Maximizes similarity between positive pairs (e.g., audio-text pairs from the same instance) while minimizing it for negative pairs (e.g., mismatched pairs).

- **Process**: Calculates similarity scores between positive and negative pairs and applies a softmax over these scores with a temperature parameter ‘‘.

B.7 DropToken (for Efficient Training)

To manage long sequences, we apply **DropToken** during training, which randomly drops some input tokens to reduce computational load.

- **Purpose**: Reduces training time and memory usage by processing fewer tokens, making it feasible to work with long sequences.

- **Effect on Long Data**: For our work with long-form audio and text, DropToken will help us manage the high computational costs associated with long sequences.

B.8 Dimensions and Configurations in VATT

Here's a summary of the typical dimensions and configurations we've used for each component in VATT:

Component	**Input Dimension**	**Output Dimension**
Audio Tokenization	(N, L_a)	(N, T_a, d)
Text Tokenization	(N, L_t)	(N, T_t, d)
Positional Encoding	(N, T, d)	(N, T, d)
Transformer Encoder	(N, T, d)	(N, T, d)
Q, K, V in MHSA	(N, T, d)	$(N, T, d/h)$ per head
Feedforward Layer (MLP)	(N, T, d)	(N, T, d)
Multimodal Projection Head	(N, d)	(N, d_proj)

Table 7: Component Input and Output Dimensions

Legend for symbols used in the table above:

- N: Batch size
- L_a: Length of audio input (total samples)
- L_t: Length of text input (number of tokens)
- T_a, T_t: Number of tokens after tokenization for audio and text, respectively
- d: Embedding dimension (e.g., 512 or 768)
- h: Number of attention heads
- d_proj: Projected dimension in the common space (e.g., 256 or 512)