CS 294-158 Final Project Report: Learning Multimodal Representations in Token Space

Phudish Prateepamornkul, Matthew Dworkin, Elden Ren

Abstract

Learning representations between multiple modalities of data is an active area of machine learning research. Prior work has demonstrated that training a multimodal Transformer can enable translation, reconstruction, and generation between modalities. We extend the prior work of M3AE from text-image pairs to text-audio-video triplets using Meta's MUGEN dataset. We experiment with an approach that operates on a token space instead of raw data. We confirm our hypothesis that lower loss can be achieved for each modality by incorporating information from the other two. We experiment with various masking ratios from 50-95% and find that the advantage of the multimodal model varies by modality being reconstructed and diminishes with higher masking ratios.

1. Link to Video

Here is a link to a pre-recorded video where we present the work outlined in this paper.

2. Introduction

With rapid advances in Transformer-based architectures for generating and encoding text, image, and audio data, there has been growing interest in training models on multimodal datasets to enable multimodal media generation and compression. There has been active research building upon effective techniques for tokenizing data, such as VQ-VAE (van den Oord et al., 2017), to learn representations between multiple modalities of data. In this setting, Transformers are trained to perform masked token prediction on multiple modalities: given some partially observed portion of tokenized multimodal data, the model learns to reconstruct the masked tokens in the data. It has been shown that a unified encoder for vision and language data trained via masked token prediction learns generalizable representations of multimodal data that transfer well to downstream tasks (Geng et al., 2022).

The problem of encoding unimodal data is well-studied (Kingma & Welling, 2014) (van den Oord et al., 2017), and we use tokenized representations of data from existing unimodal autoencoders when training a model that extracts mutual information from multiple modalities. Unlike the model proposed by M3AE (Geng et al., 2022), our model works on the token space of the data rather than the raw data. The motivation for this is twofold: for one, existing open-source models are very effective at encoding the modalities of interest, so training on tokenized representations saves compute; also, having a unified encoder that uses tokenized representations of multiple modalities standardizes the model architecture and makes it extensible to other unimodal encoders and modalities.

In this paper, we use a Transformer-based autoencoder to learn mutual information between text, video, and audio data. With existing tokenizers, we independently encode each modality, and the model learns masked token prediction on the tokens across the three modalities. We train our model on Meta's MUGEN dataset, and we perform an empirical analysis of the multimodal Transformer. We find that incorporating information from multiple modalities enables more accurate reconstruction of the targeted modality. We also show that this multimodal training scheme does not reduce the model's masked token prediction capabilities for any given modality. Our results illustrate that mutual information can be extracted from the discrete token spaces of multiple modalities.

3. Related Work

Methods for handling multimodal data often rely on contrastive learning, such as CLIP (Radford et al., 2021) and ALIGN (Conneau et al., 2017). These methods have been effective in capturing shared information between modalities and and performing tasks like image-to-text. However, these models are limited in that they jointly train separate encoders for image and text, making it challenging for the models to encode distinct information from multiple modalities at once.

Multimodal masked autoencoders (M3AE) were proposed (Geng et al., 2022) as a novel approach for training a model

on multimodal data in masked token prediction to learn a unified encoder for both vision and language data. M3AE treats the image and text data as a single sequence and trains by masking parts of the sequence and then reconstructing them. This approach allows M3AE to learn a unified representation for both modalities without the limitations of contrastive learning. The paper demonstrates that M3AE performs well on downstream tasks and can be flexibly trained on both paired and unpaired image-text data.

4. The Data

In this project, we use Meta's Multimodal Understanding and Generation (MUGEN) dataset (Hayes et al., 2022). The dataset provides 375K samples of text-audio-video triplets. The video 3.2 second clips of a platform videogame called CoinRun where the character, called "Mugen", walks and jumps around to collect coins and kill monsters. Each is accompanied by text and audio sound effects describing what happens in the video. We use a subset of 8k samples for training and 2k samples for validation due to computational constraints.

5. Approach

Inspired by M3AE (Geng et al., 2022), we train a multimodal BERT-style (Devlin et al., 2018) Transformer (Vaswani et al., 2017) with a high mask ratio for the task of masked token prediction. We begin by tokenizing the data. For text and video, we use existing pretrained models. For audio, we learn a VQ-VAE (van den Oord et al., 2017). This effectively compresses the data and gives a set of learned token embeddings that we can later decode back into realized text, audio, and video samples. We then concatenate tokenized versions of each of the three modalities into a single sequence. We apply different mask ratios to each modality according to the masking schemes described in section 6 and train using cross entropy loss. Losses for each modality are recorded separately.

For comparison purposes, we also train unimodal BERT-style Transformers, which give baseline losses for our experiments. Note that throughout the project, we report only the cross-entropy losses on the predicted masked tokens, as opposed to, say, L2 reconstruction loss for audio and video. We do this because we freeze our tokenizer encoders and decoders, so lower cross-entropy loss on predicted tokens corresponds to lower reconstruction loss.

All Transformers trained in this project use a hidden size of 3072 with 12 hidden layers and 12 heads.

5.1. Tokenizing each Modality

In order to process the three modalities together in a multimodal Transformer, we first tokenize each modality with a separate model, as described below.

For text, we use a pre-trained BERT tokenizer. This enables conversion from varied-length strings into padded, fixed-length sequences of tokens. After this step, each text sample is a sequence of tokens of length 128.

For audio, we train a custom VQVAE with 1d convolutions to encode and decode the audio sequences to/from a token space. This VQVAE maps each audio sequence from shape (70560, 2) into a sequence of tokens of length 441. We chose to train this model from scratch becasue we could not find a suitable open-source pretrained model. We trained the VQVAE for 30 epochs and achieved 0.0017372 for the L2 reconstruction test loss.

For video, we use VideoGPT (Yan et al., 2021), a pre-trained model that compresses the videos from shape (3, 32, 256, 256) into spatiotemporal patches of shape (4, 16, 16). We flatten the tokenized video output from VideoGPT into a sequence of length 1024. Note: we used VideoGPT as the video tokenizer, only to realize later upon deeper inspection that VideoGPT slices the first 16 frames of each video as opposed to downsampling to 16 frames (i.e. by taking every other frame). This may help to explain some abnormalities in our results. The pre-trained model achieved 0.0449167 for the L2 reconstruction test loss.

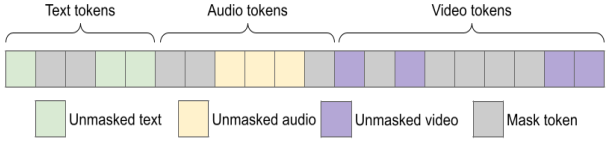
The VQ codebook embedding dimensions for text, audio, and video are 768, 256, and 256, with vocabulary sizes of 30522, 1024, and 2048 respectively.

5.2. Multimodal Transformer

After tokenization, the three data modalities are ready to be combined. We concatenate the sequences for each modality into a combined sequence such that the first 128 positions are text tokens, the next 441 are for audio tokens, and the last 1024 are for video tokens. This combined sequence is of length 1593.

Our architecture is flexible in that it allows the user to specify a separate masking ratio for each modality. Figure 1 depicts an example masked concatenated sequence for mask ratios of 50% for all modalities.

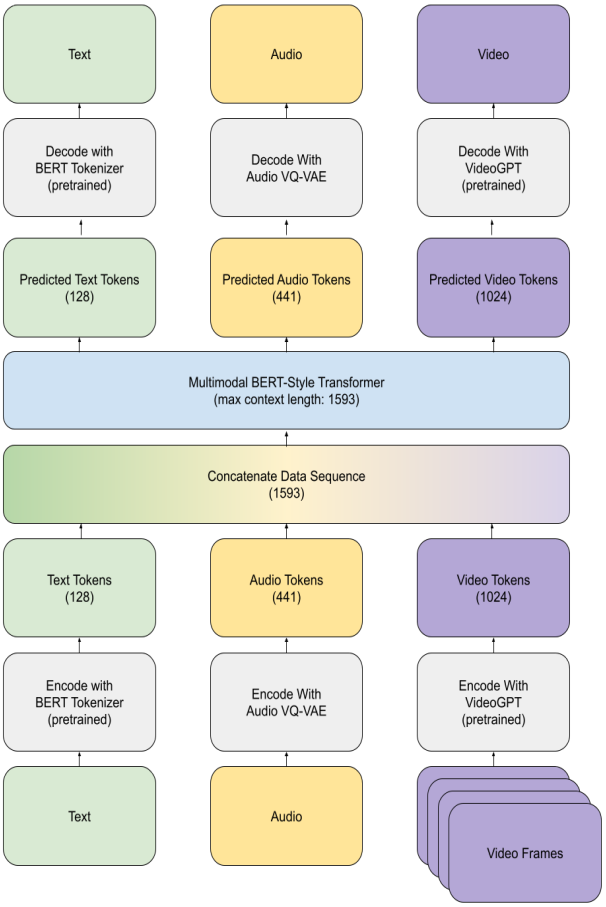
Figure 1. Example Masked Multimodal Sequence



Our multimodal Transformer is trained BERT-style on masked token prediction using cross-entropy loss over logits for the respective modality. The total loss is taken as a weighted sum of the losses from each modality, where the relative weights are a hyperparameter choice which is $\mathcal{L}_{total} = 0.26\mathcal{L}_{text} + 0.14\mathcal{L}_{audio} + 0.60\mathcal{L}_{video}$. Losses for each modality are recorded separately and later compared to their unimodal baseline. We train multiple versions of this model based on the masking scheme described in figure 3 (i.e. all modalities have the same masking ratio during training). We train one model per masking ratio in {50%, 70%, 90%, 95%}.

See figure 5.2 for our multimodal Transformer architecture.

Figure 2. Multimodal Transformer Architecture



5.3. Unimodal Transformers

To enable a comparison for the multimodal Transformer, we train separate Transformers on each modality for each masking ratio in {50%, 70%, 90%, 95%}. These unimodal Transformers are each a simple BERT-style Transformer trained on masked token prediction with cross-entropy loss. Their input is the output of the corresponding unimodal encoder shown in the bottom of figure 5.2.

6. Experiments

In this project we are primarily interested in investigating the extent to which we can learn the mutual information between modalities to aid in the task of reconstruction.

For each of the following three experiments, we draw a quantitative comparison between the test losses of the unimodal and multimodal Transformers. We also provide a qualitative comparison by showing the corresponding reconstructions.

Results from each experiment are shown below in section 7. Analysis follows in section 8.

6.1. Effect of More Modalities

We hypothesize that the multimodal Transformer can achieve lower cross-entropy loss for each modality, as compared to its corresponding unimodal Transformer. This is because the multimodal Transformer has access to the corresponding information from the other modalities during training and can develop an understanding of the mutual information between the data types.

We begin by training a multimodal Transformer and three corresponding unimodal Transformers for a variety of masking ratios ranging from 50-95%. For each modality, we compare the test loss achieved by the multimodal Transformer to that achieved by the corresponding unimodal Transformer. Losses for each masking ratio are shown in section 7 below.

Figure 3. Example Masking Scheme for Experiment 6.1

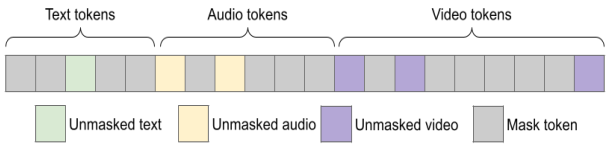


Figure 3 shows an example masked multimodal sequence for this experiment, where each modality is masked at 70%.

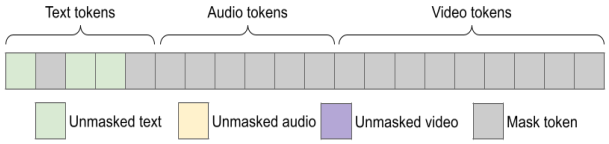
6.2. Using the Multimodal Transformer Unimodally

After training on a particular masking ratio, we are interested to see how the multimodal Transformer performs when used as a unimodal Transformer. We are curious to investigate whether the performance multimodal Transformer degrades as a result of being trained with the other data modalities.

We test this by applying the usual masking ratio for one modality, but masking the other two modalities at 100%. We run this on the model trained in section 6.1 for the corresponding mask ratio.

Figure 4 shows an example masked multimodal sequence

Figure 4. Example Masking Scheme for Experiment 6.2



for this experiment, where the text modality is masked at 50% and audio and video are masked at 100%.

6.3. Reconstructing One Modality from the Other Two

Finally, we evaluate the performance of the multimodal Transformer when given access to the information from the other modalities. This is done by applying the usual masking ratio (ranging from 50-95%) to one modality, and masking the other two at 0%. Again, this is run with the model trained in section 6.1 for the corresponding mask ratio.

We hypothesize that the multimodal Transformer will achieve qualitatively better reconstructions as a result of having access to more data from the other modalities at test time.

Figure 5. Example Masking Scheme for Experiment 6.3

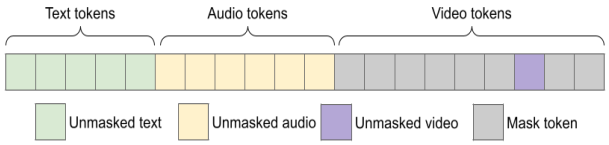


Figure 5 shows an example masked multimodal sequence for this experiment, where the video modality is masked at 90% and audio and video are masked at 0%.

7. Results

Figures 6-8 give quantitative results. Shown in blue and orange are test losses for experiment 6.1 for each modality. Depicted in green are the results from experiment 6.2 where the multimodal Transformer loses access to the other data modalities at test time. In red are the losses from experiment 6.3 where the multimodal Transformer is granted full access to the other data modalities at test time.

Figures 9-13 give qualitative results from our experiments. We show each of the multimodal reconstruction from the three experiments, the unimodal reconstruction, as well as the original sample.

Figure 6. Plot of text loss

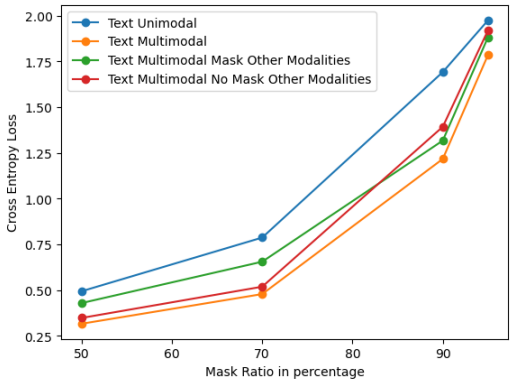


Figure 7. Plot of audio loss

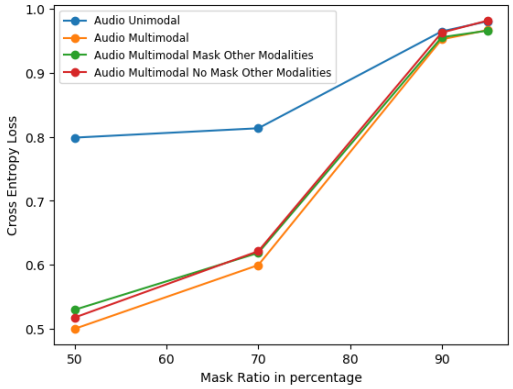
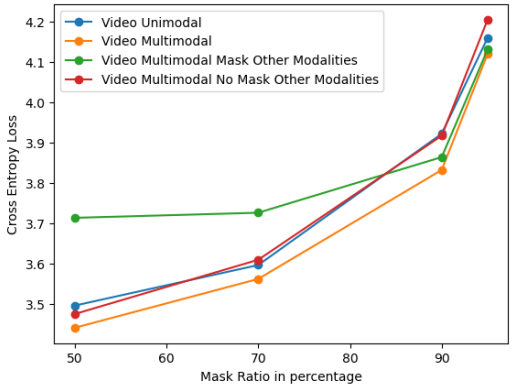


Figure 8. Plot of video loss



8. Analysis

8.1. Quantitative Analysis

The qunatitative results confirm our hypothesis from experiment 6.1 that the multimodal Transformer can achieve lower loss than the corresponding unimodal Transformer. This was true across all data modalities and masking ratios. However, we found that the performance boost of the multimodal Transformer was smaller for higher masking ratios.

Figure 9. Original Sample

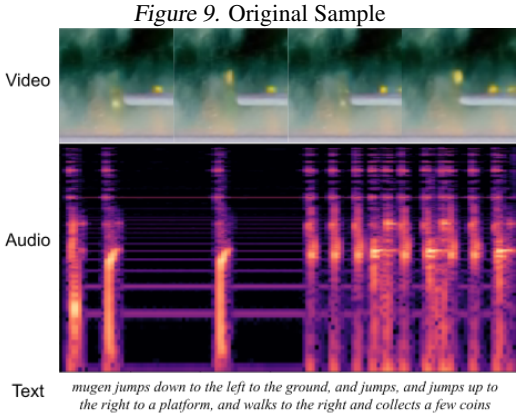


Figure 10. Unimodal Reconstruction

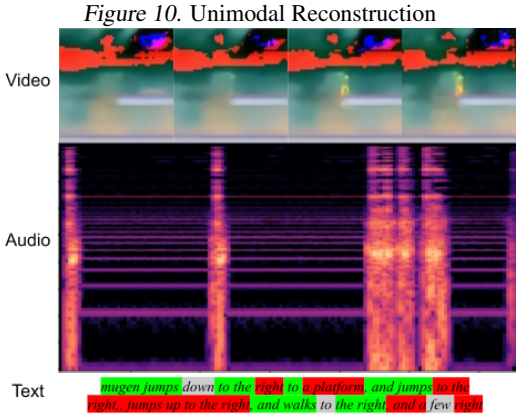


Figure 11. Multimodal Reconstruction (Exper. 6.1)

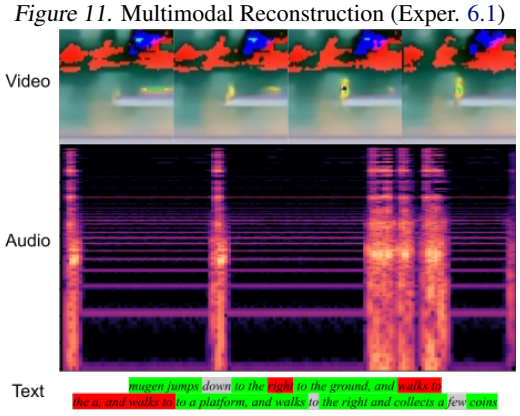


Figure 12. Multimodal Reconstruction (Exper. 6.2)

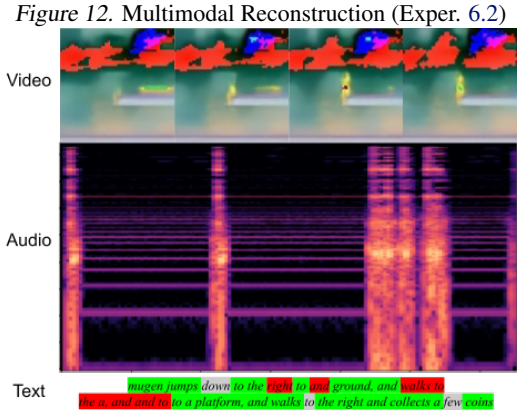
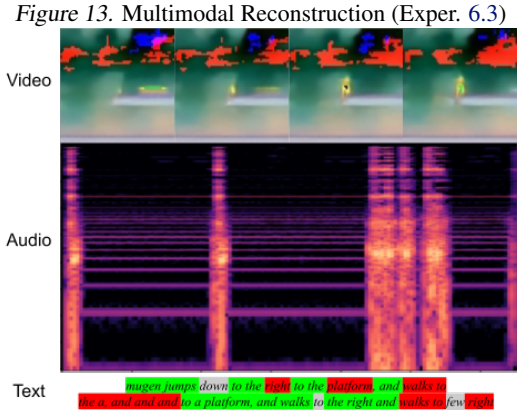


Figure 13. Multimodal Reconstruction (Exper. 6.3)



This makes sense because for higher masking ratios, there is less additional information coming from the other data modalities, which in turn diminishes the advantage of the multimodal Transformer.

Comparing the green and blue curves (experiment 6.2), we find that the multimodal Transformer outperforms its corresponding unimodal Transformer, even when used unimodally. In other words, training jointly on all data modalities led to better unimodal understanding, as is evidenced by the lower cross-entropy loss. This was true across all modalities and masking ratios, except for 50% and 70% masking on video.

Comparing the red and blue curves (experiment 6.3), we find results similar to those of 6.1. Reconstruction of each modality benefits from the information in the other modalities, and the advantage is higher at lower masking ratios.

8.2. Qualitative Analysis

We now turn to a qualitative analysis of the results, comparing figure 10 to each of figures 11-13. Each shows video, audio, and text reconstructions for models trained with a fixed 90% masking ratio on all modalities. Figures 10 and 11 show reconstructions for masking at 90% on all modalities, while figure 12 shows reconstructions from the multimodal Transformer being evaluated *unimodally* and figure 13 shows reconstructions from the multimodal Transformer evaluated with *full access* to all other modalities (masked at 0%).

Text tokens that were *not masked* are shown in gray. Masked tokens that were predicted correctly (as compared to the original sample) are shown in green and those that were predicted incorrectly are shown in red.

Across the board, we see qualitative results that are consistent with the quantitative results discussed above. Qualitative analysis of the audio and video reconstructions is hard for multiple reasons: for one, small changes in the audio and video are less obvious and also the videos are downsampled so heavily that the reconstructions all look of relatively the same quality. In other words, the performance gains we see in the quantitative results pale in comparison to the video quality lost from downsampling. Therefore, the following analysis focuses on the text reconstructions.

First, comparing figure 11 to figure 10, we see that the multimodal Transformer makes fewer mistakes in recovering the text, presumably drawing clues from the audio and video to help with reconstruction.

Next comparing figure 12 to figure 10, we observe that even when losing access to the other modalities, the multimodal Transformer is able to make fewer mistakes than its unimodal counterpart.

Finally, comparing figure 13 to figure 10, we see that both models make similar mistakes in text recovery, which is consistent with the quantitative results where their losses were relatively close.

None of the models were able to reconstruct audio or video very well, and their mistakes were all very similar for these modalities. For audio, we believe that this is because most of the text and video (common words like ”to”, ”and”, ”a” and common video chunks like background and platforms) don't provide much information for recovering what should be in the audio sequence. Essentially, the model would need to get lucky and be given a word like ”coin” or ”jumps” that has a corresponding sound effect in order to gain an edge in reconstruction. For video, we believe the poor reconstructions are due to the small codebook of VideoGPT (just 2048), as well as the fact that VideoGPT was pretrained on a different distribution of video data (one that includes far more red than is present in the MUGEN dataset).

Interestingly, in figures 11-13, the multimodal Transformer was able to correctly identify and place the coin (fuzzy yellow patch to the right of Mugen) that is seen in the original video.

9. Ablations

We trained with and without additive modality type encodings and found that the results were comparable. We also tried adding the modality type encodings before vs. after the layer norm and dropout and found that the results were comparable too.

We tried a few different relative weightings for the loss terms in the multimodal Transformer. We found that weighing each term by a factor proportional to the magnitude of the corresponding unimodal losses achieved by the best results.

10. Limitations and Future Work

One of the biggest limitations for this project was compute. We were fortunate to be given access to some A100 GPUs on a preemptive basis that made this project feasible to begin with. However, the sheer size of the MUGEN dataset and the constraints of completing this project over a span of a few months meant that we had to make a few compromises. Namely, we had to limit ourselves to a very small subset ($<3\%$) of the data, we could train for a limited number of epochs (around ten per model), we could not fine-tune the tokenizers, we had to settle for relatively small Transformer models, and we had to compress each modality heavily (i.e. just 2048 tokens for 3.2 second RGB videos with square frames of size 256).

We believe that given more computational resources, this project would benefit from the following:

1. Utilizing the entire dataset
2. Training for more epochs
3. Increased model dimensions and codebook sizes
4. Fine-tuning the VideoGPT tokenizer

Another limitation of this project is that it investigates *paired data* only. The results may not generalize the same to unpaired data.

Future work should start by addressing each of the limitations listed above. We also suggest a few other directions to take this project, including extending to even more modalities, analyzing the attention between tokens of each modality, and applying the approach to *unpaired* data.

Investigating these can shed further light on how multimodal Transformers are able to learn mutual information between multiple modalities and yield interesting insights into how well this architecture can generalize.

Acknowledgements

Thanks to the Spring 2024 course staff of CS294-158 Deep Unsupervised Learning for multiple discussions in office hours and for general guidance on this project. This includes TAs Philipp Wu, Wilson Yan, Kevin Frans, and Professor Pieter Abbeel.

Thanks to classmate Konpat Preechakul for helpful discussions about this project.

Thanks to the staff at the Statistical Computing Facility (SCF) in Berkeley's Statistics department for providing us

access to computational resources for this project. It would not have been feasible otherwise.

References

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. Word translation without parallel data. *CoRR*, abs/1710.04087, 2017. URL http://arxiv.org/abs/1710.04087.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Geng, X., Liu, H., Lee, L., Schuurmans, D., Levine, S., and Abbeel, P. Multimodal masked autoencoders learn transferable representations, 2022.

Hayes, T., Zhang, S., Yin, X., Pang, G., Sheng, S., Yang, H., Ge, S., Hu, Q., and Parikh, D. Mugen: A playground for video-audio-text multimodal understanding and generation, 2022.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL https://arxiv.org/abs/2103.00020.

van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *CoRR*, abs/1711.00937, 2017. URL http://arxiv.org/abs/1711.00937.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

Yan, W., Zhang, Y., Abbeel, P., and Srinivas, A. Videogpt: Video generation using VQ-VAE and transformers. *CoRR*, abs/2104.10157, 2021. URL https://arxiv.org/abs/2104.10157.