Project Document: Multi-Modal Data Retrieval
Deep Learning Divas

Abby Veiman
Grace Hecke
Sabrina Fowler
Derek DeBlieck

Contents

Abstract

This project involves multi-modal data retrieval across images and text captions. After initially planning to use the NUS-WIDE dataset, we ended up pivoting to the Flickr8k dataset. The NUS-WIDE dataset is a bit outdated and we could not find a reliable source to download both the images and text tags from. The Flickr8k dataset is very similar, the main difference is that the text data consists of 5 potential captions for each image instead of a list of descriptive tags. At this stage, we are exploring the best structure to use for our autoencoder architecture. We ran experiments using a correlational autoencoder that trains on the images and captions separately, but with a shared latent space. We tested on two types of alignment loss functions to make sure the latent representations of image-caption set pairs are similar. We also built a cross-modal architecture where during training, image latents decode into the text feature space and text latents decode into the image feature space. So far it seems that the cross-modal architecture works better, but we plan to fine-tune the hyperparameters for all of our options to make sure we get the best performance possible.

Chapter 1

Milestone 1: Project Ideas

1.1 Introduction

Broadly, we are interested in doing a project involving an autoencoder. One of our teammates, Sabrina, is interested in applying ideas from mathematical neuroscience to neural networks. This process involves looking at a set of binary vectors produced during training[11] and using those to construct a mathematical object called a *neural ideal*.[14] The hope for this research is that mathematical properties of this neural ideal will tell us about trustworthiness and/or robustness of the model. This object is easier to handle with a smaller number of neurons, so an autoencoder seems like a good place to start because of the relatively small number of neurons in the inner layer that somehow contain vital information about the training data.

We started looking for project ideas involving anomaly detection, a common application of autoencoders, before expanding our search to the uses of autoencoders in multimodal data. We are interested in finding a real-world problem to solve, which is why the idea of image colorization appealed to us. Additionally, we considered potential datasets which lend themselves to anomaly detection, which turned out to be more complicated than we anticipated.

1.2 Project Idea 1: Crime Anomaly Detection with Autoencoder

Several challenges exist in the space of crime data reporting and analysis that could be mitigated with a deep learning solution. Oftentimes, unusual patterns or spikes in crime activity – whether on a local or national level – may go unrecognized for long periods of time, at victims' expense. A solution with the capacity for anomaly detection on a more micro- scale may allow for officials to prevent spikes in crime before they become more severe. Police and government officials may allocate resources according to real-time shifts in crime patterns, as opposed to after-the-fact.

Furthermore, changes in local legislation can impact rates of crime, as well as policing resources and practices. Detecting anomalies in crime data may be valuable in assessing not simply whether legislation is “good” or “bad,” but perhaps if any new patterns are present in a community after new policy takes affect. Additionally, defining what is or is not “normal” activity may not be clearly defined by humans. Even using common statistical methods, it may be difficult to compress “normal” patterns using highly dimensional, large datasets. In contrast, a deep learning model can learn complex, nonlinear relationships, creating a more flexible and less assumption-driven representation of what is actually anomalous.

The learning problem for this project is unsupervised anomaly detection – likely using an autoencoder – to identify crime incidents or patterns that deviate from typical behavior. Anomaly detection does not require labeled anomalies, so the model can be trained on historical, unlabeled crime data to represent “normal” activity. The inputs to the model would include features like crime type, time, date, location, and other demographic variables, likely from a particular city or metropolitan area. These inputs would be compressed into a “bottleneck” within an autoencoder. The output would include the reconstructed input, as well as the reconstruction errors used to detect anomalies and determine the features contributing to an anomaly [6]. Various sources of times series crime data exist publicly. One such example is FBI data from the city of Los Angeles [12].

An example of this process – identifying abnormal crime activity using an autoencoder – can be found in a study from Payame Noor University [3]. The study outlines a more specific framework for constructing and evaluating the anomaly-detecting autoencoder, and cites various other studies pertaining to autoencoders that may prove helpful throughout the project.

In summary, deep learning has the potential to answer the following questions, and beyond, while allowing officials in law enforcement and government to make the appropriate investigative and policy-making decisions:

- Are there any unusual crime patterns within a specific area (ie. sudden low crime in higher-crime areas, high crime in lower-crime areas?)
- Has their been spikes in any particular types of crimes?
- Could policing practices have changed unexpectedly?
- How has a certain policy change impacted crime in an area?

1.3 Project Idea 2: Image Colorization with Autoencoder

Image colorization presents several challenges that can be addressed effectively with deep learning, particularly using autoencoder architectures. Black-and-white or grayscale images from historical archives, medical datasets, or artistic projects often lack the interpretability and visual richness that color provides.

Automatically predicting and restoring realistic colors can make these images more informative for researchers, more engaging for audiences, and more practical in technical domains such as medical imaging. Compared to manual colorization, which is time intensive and prone to subjective variation, autoencoder based solutions offer scalability, efficiency, and consistent output quality [9, 15].

One major difficulty lies in defining what constitutes an "accurate" colorization. Many historical or archival photos have no real or truthful color references, and even when reference images exist, the relationship between grayscale intensity and true chromatic values is highly nonlinear and context dependent. Traditional algorithms struggle to handle these complex, high-dimensional patterns. Autoencoders, however, can learn compact latent representations of grayscale images and reconstruct the missing chrominance channels, capturing the complex, nonlinear structures necessary to produce context-aware and visually realistic results [9, 15].

The learning problem for this project is supervised image reconstruction using an autoencoder. Training requires pairs of grayscale and color images, such as those from datasets like ImageNet or CIFAR-10 [15, 10]. Grayscale images are input to the encoder, which compresses them into latent feature representations. The decoder then predicts the chrominance values, which, when combined with the input luminance, reconstruct a full color image. Model evaluation can be performed using quantitative metrics like mean squared error, as well as qualitative assessment based on visual realism [15].

An example of this approach can be found in work by Iizuka et al., who implemented an end-to-end autoencoder based colorization model incorporating both global and local image priors to improve realism [9]. Similarly, Zhang et al. demonstrated a framework where a classification based loss helps balance realistic color diversity with context consistency, highlighting the flexibility of autoencoders in learning from large, diverse image datasets [15]. Publicly available datasets like CIFAR-10 provide sufficient training data to capture a broad range of color distributions for experimental purposes [10].

In summary, autoencoders provide a powerful framework for automated image colorization, enabling researchers and practitioners to explore questions such as:

- How effectively can grayscale images be restored in a way that appears natural and realistic to human observers?

- Can autoencoders generalize across different types of images, including portraits, landscapes, and technical imagery?

- Which autoencoder architectures and loss functions produce the most visually convincing results?

- How can automated approaches reduce the time and subjectivity involved in manual colorization?

1.4 Project Idea 3: Modernizing Correspondence Autoencoders for Cross-Modal Retrieval

Cross-modal retrieval is the task of retrieving data in one modality, such as images, given a query in another modality, such as text. This problem underlies applications such as image search engines and caption generation. Feng et al. [4] introduced the Correspondence Autoencoder (Corr-AE), where separate autoencoders are trained for each modality (image and text), but share a latent space. Queries in one modality are mapped into this shared space and then decoded into the other modality. So for instance, a query of text is encoded into the shared latent space and then decoded as an image. In their work, Restricted Boltzmann Machines were used to first normalize or eliminate modality-specific statistical properties before training Corr-AE. We propose to modernize this framework by replacing RBMs with more modern pretrained feature extractors. For instance, using Resnet18 or Resnet50 to extract features from images and using BERT to extract features from text. We will then train a correspondence autoencoder that learns to align these features into a shared latent space, where similarity can be measured directly. Our objectives are twofold: to compare our approach against the original Corr-AE as described by [4] and to compare against modern cross-modal models such as CLIP, which use contrastive learning instead of autoencoders.

This will be a supervised learning project, where the objective is to learn a shared latent space where text/image pairs are close while unassociated text/images are separated. Our input for model training will be text/image pairs. The training output will be reconstructed pairs of text/images, which are used in the training of the Corr-AE. Once the model is trained, a single modality can be input and the vectors of opposite modality closest to the input's latent vector are returned.

We propose to use the Flickr8k dataset of eight thousand images, each with multiple captions. The dataset was introduced in [8].

1.5 Conclusions

Of our three problems, we are most excited about problems 2 and 3. We were initially excited about problem 1, but after reading [3] in more detail less enthusiastic to try and replicate these results with a new dataset (for example, it is unclear how this paper defines what an anomalous crime is, and therefore it is unclear what the authors are even classifying). While we would be excited to work on something with a tangible or real-world connection, we currently don't have a clear research statement for problem 1. Currently, our ideas are too vague to be actionable.

Problem 2 is also quite tangible, which is appealing to us. However, the main paper we are using as a reference for this paper, [15], does not use an actual auto-encoder, although it contains a somewhat related idea. If we are intent on using autoencoders, we may need to pick a different project.

Finally, project 3 feels interesting, and perhaps is relatively novel research (as in, we wouldn't just be trying to replicate another paper's results). The idea of using an autoencoder to encode and decode across modalities is quite intriguing and seems like an innovative use of the architecture. While this is exciting, it is also a little daunting. This topic feels like the most complicated of the three.

We as a group need to align on our goals for this project (if we are looking to get a good grade in the course or looking for something that we could put on our GitHubs for potential employers). Something that we learned in this first phase of the project is that starting with an architecture in mind and trying to find a problem to fit it is much harder than thinking of a problem and deciding which architecture fits it best.

Below we have listed some questions that we still have:

- Besides fraud detection, what is the biggest use case that you know of for autoencoders? How hot of a topic are autoencoders in the ML/AI community right now?
- Have Corr-AE models been deployed in any actual business context, or are they purely academic? Is there even any interest in academia in them anymore?
- Do you have any other recommendations for projects involving autoencoders?
- What questions besides trustworthiness & robustness do you think might be answerable about a model by looking at the training process with a mathematical lens?

Table 1.1: Contributions by team member for Milestone 1.

Team Member	**Contribution**
Grace	Project Idea 1
Abby	Project Idea 2
Sabrina	Introduction & Conclusion
Derek	Abstract & Project Idea 3 & Conclusion

Chapter 2

Milestone 2: Project Selection

2.1 Introduction

Cross-modal retrieval is the task of retrieving data in one modality, such as images, given a query in another, such as text. It can be used for applications like image search and caption generation. Feng et al. [4] introduced the Correspondence Autoencoder (Corr-AE), which trains separate autoencoders for image and text data that share a latent space, allowing information to be translated between modalities. Their work relied on Restricted Boltzmann Machines (RBMs) to normalize modality specific features before training. Although effective at the time, RBMs have since been replaced by deep neural networks that can extract more powerful and semantically meaningful features.

We chose to refine the Corr-AE framework by replacing RBMs with modern pretrained architectures—ResNet for images and BERT for text—and comparing this to a version trained end to end without pretrained extractors. Using the Flickr8k dataset of 8,000 images with five captions each [6], our project aims to learn a shared latent representation that enables accurate retrieval between modalities. We will evaluate our results against the original Corr-AE [4] and more recent models such as DeViSE [5], assessing how updated feature extraction techniques impact cross-modal alignment and retrieval performance.

2.2 Problem Specification

Our project will refine a method proposed by [4] for cross-modal data retrieval. We use two autoencoders that share a latent space in order to return an image based upon a text query (or vice-versa). We will use the Flickr8k dataset, which is composed of images, each with 5 corresponding text descriptions. The model will take in a text (resp. image) query and return a image (resp. text description).

This project addresses the problem of learning a joint latent representation for multimodal data. In particular, we learn a joint latent representation for images and their associated (text) captions. Each modality also includes a decoder to reconstruct the original input from the shared latent code, forming a correlation autoencoder architecture. During training, the model jointly minimizes the within-modality reconstruction losses and a cross-modal alignment loss in the latent space. At inference time, the model supports cross-modal retrieval: given a query in one modality (e.g., a text caption), it retrieves the most similar items from the other modality (e.g., images).

Users often use search engines to find images, audio files, and various other data types using only text queries. Previous work by [4] demonstrated the capacity for correspondence auto-encoders to assist in this multi-modal data retreival. Our work aims to modernize these methods using modern feature extractors rather than out-of-date Random Boltzmann Machines. By replacing RBMs with contemporary feature extractors (ResNet for images and BERT for text), and training correlation autoencoders that share a latent space, we expect to achieve more robust alignment and improved retrieval accuracy on standard multimodal benchmarks.

We will use the Flickr8k dataset, which contains 8,000 natural images with five captions each. All code will be implemented in Python using PyTorch and standard libraries. Additionally, we may also use the NUS-Wide-10k dataset for an even closer comparison to [4]. Training will be performed on `swan` using GPU acceleration, and requires installation of `torch`, `transformers`, and `datasets` packages.

This work builds upon several key contributions in the literature. The concept of correlation autoencoders for cross-modal representation learning was first introduced by Feng et al. [4]. Devlin et al. introduced BERT, a transformer-based model for natural language understanding, which we leverage for high-quality text embeddings [2]. Together, these works motivate our investigation into modernized Corr-AE architectures for cross-modal retrieval.

2.3 Related Work

Traditional methods of cross-modal retrieval involve first applying representation learning techniques to each data modality independently, followed by correlation learning to align their features. In Feng et al. [4], the authors proposed a correlation autoencoder that performs both representation and correlation learning simultaneously. The model trains two autoencoders, one for each modality, that share a latent space, encouraging alignment between image and text features. Performance is evaluated based on reconstruction losses for each modality and a correlation loss that enforces cross-modal similarity.

In Frome et al. [5], one of the earliest deep learning approaches to cross-modal alignment, image features from a pretrained convolutional neural network are projected into the same semantic space as word embeddings. Unlike Corr-

AE, which reconstructs data in each modality, DeViSE focuses on learning a ranking loss that encourages semantically similar image-text pairs to be close in the shared space. This design allows the model to generalize to unseen classes, demonstrating the potential of leveraging pretrained unimodal representations for cross-modal tasks.

More recent work benefits from powerful backbone architectures like BERT [2] and ResNet-18 [7] to extract semantically rich features from text and image data, respectively. BERT, a transformer-based language model pretrained on a large dataset using masked language modeling and next-sentence prediction, provides contextualized embeddings that significantly improve downstream language understanding tasks. In cross-modal retrieval, BERT's token-level embeddings are often pooled to produce sentence-level representations that align well with visual features in a joint embedding space.

On the visual side, ResNet-18 offers a compact yet expressive convolutional architecture that introduces residual connections to improve training stability and performance. It is commonly used to extract fixed-length image embeddings that serve as inputs to cross-modal models. When combined, BERT and ResNet-based encoders provide strong unimodal representations that can be fine-tuned or projected into a shared embedding space using contrastive, triplet, or ranking-based losses. This modern approach has largely replaced earlier architectures like RBMs, offering both higher accuracy and better generalization in retrieval tasks.

2.4 Proposed Method 1: Corr-AE with Updated Feature Extraction

We will use the Flickr8k dataset, which contains 8,000 images with five captions each. Images will be split into groups of 6,000 for training, 1,000 for validation, and 1,000 for testing. Images will be resized to 128×128 pixels, and we will normalize each color channel individually. Next, we will use a ResNet18 architecture for image feature extraction and BERT for caption feature extraction. We will use the pre-trained weights for ResNet18 and BERT and fine-tuning with the data available in Flickr8k, but we will not train these model weights from scratch [2].

Next, we take the features extracted from our images and captions and create two auto-encoders: the first for text and the second for images. We will train these auto-encoders so that they share a latent space. Intuitively, the loss function that we are trying to minimize during the training of these auto-encoders is a weighted sum of the loss of each single-modality auto-encoder added to a distance metric between the encoded representations. See [4] for a more formal definition of our loss function. We plan to test several different autoencoder architectures. As we have already extracted features from the raw image/text, we believe that these autoencoders can be relatively shallow neural networks. We will test the effect of differing the amount of hidden layers as well as the amount of nodes in the shared latent space, but believe that 256 nodes in the latent space with only three or fewer hidden layers will be sufficient.

We hope that replacing the Random Boltzmann Machines with more modern feature extraction will allow us to improve on the results of [4]. However, their paper used different datasets. We believe that our dataset to be slightly more interesting, as our dataset contains multiple captions per image. However, to be as comparable as possible to the original work done in [4], we could instead use the Nus-WIDE-10k dataset, one of the three datasets on which [4] tested their autoencoder. We propose to judge performance is based upon two metrics: Recall@50 (the percentage of queries where the top result is returned in the top 50 results) and Median Rank (the median rank of the correct result). See [13] for an example of the use of these metrics on the Flickr8k dataset.

However, if we wish to compare our methods directly with [4], we will instead need to use the NUS-Wide-10k dataset. This should not change our preprocessing or architecture significantly. However, this dataset contains only one caption per image, and these images are grouped into 10 categories, and thus we will need to use the m-Average Precision and Top 20% metrics to evaluate performance.

Our next step is implement the end-to-end Corr-AE and preprocess the Flickr8k dataset, followed by initial training runs and validation evaluations. In the subsequent milestone, we will refine our model architecture, adjust hyperparameters (such as the weights in the loss function that balances the single-modality loss and the latent space distance, the number of hidden layers, and the number of nodes in the latent space), and compare to [4].

2.5 Proposed Method 2: Corr-AE Without Pretrained Features

If implementing an alternative method, we will create an end-to-end Corr-AE trained directly on raw image and text inputs, without using any pretrained feature extractors and learning both modalities from scratch. Like in method 1, we will use the Flickr8k dataset, containing 8,000 images with five captions each. We will split the data into 6,000 training, 1,000 validation, and 1,000 testing examples. The image data will be resized to 128×128 pixels and normalized. Text data will be tokenized, and the resulting text sequences will be padded to the same length for batch processing.

The model will consist of two autoencoders sharing a latent space: one for images and one for text. The image encoder will use a small convolutional neural network (CNN). The text encoder will use a bidirectional long short-term memory network (LSTM) for sequential learning of caption text, as demonstrated in Frome et al. for semantic text embeddings [5]. Both encoders will project to a shared 256-dimensional latent space, trained using reconstruction and contrastive alignment losses [4]. This will ensure not only proper reconstruction for each modality, but also alignment of both modalities. Recall@K and median rank (MedR) will be used to evaluate performance and examine both top-K retrieval accuracy and overall ranking quality, as utilized in the aforementioned reference literature [4, 5]. We will compare results against the original Corr-AE by Feng et al., which utilizes RBMs and no pretrained feature extractors [4]. We may also compare results against Frome et al., which implements contrastive alignment loss and pretrained visual features with a text embedding space [5].

Before the next milestone, we will implement the end-to-end Corr-AE and preprocess the Flickr8k dataset, followed by initial training runs and validation evaluations. In the subsequent milestone, we will refine our model architecture, adjust hyperparameters, and compare early results, making final adjustments as necessary.

2.6 Conclusions

We will be closely following [4], attempting to modernize their feature extraction in hopes of improving performance of correspondence autoencoders on image/text datasets. We slightly prefer the first method, while the second method feels slightly simpler and a good backup plan if we struggle to get the feature extraction to work properly in method 1. We still have one, rather large question: Flickr8k consists of 8,000 ungrouped/unlabelled text/description pairs (5 descriptions for each image). Because these images aren't grouped, we are not sure how performant our model will be. Perhaps we could train with only 2/5 captions and test with the other 3/5? Or maybe we need to pick another dataset. Some research (such as [5]) has shown that text can be used fairly well for on-shot or no-shot cross-modal retrieval, but we have not read their paper in great detail. Their methods may go quite beyond the scope of this project/class (and thus it wouldn't be wise for us to try something more aligned with their approach), or perhaps our method really can generalize to previously unseen images/captions. We can always use the Nus-WIDE-10k dataset if our results are terrible on Flickr8k, or we could train our model to only see a portion of the text captions and use the others for testing image retrieval.

Table 2.1: Contributions by team member for Milestone 2.

Team Member	**Contribution**
Derek DeBlieck	Problem Specification, Proposed Method 1, and Conclusions
Grace Hecke	Proposed Method 2
Abby Veiman	Update Abstract and Introduction
Sabrina Fowler	Related Work

Chapter 3

Milestone 3: Progress Report 1

3.1 Introduction

For our project, we are using a Correlated Auto-Encoder (Corr-AE) for multimodal data retrieval. For example, given a text input, our model will return an image from the test data that most closely represents this text. We chose to follow our first proposed method. Specifically, we will use BERT and a ResNet model for feature extraction before training our coder/decoder. We chose this method because we believe that we can improve the precision of Corr-AEs with more modern feature extraction; ignoring feature extraction entirely (Proposed Method 2), while simpler, seems likely to degrade performance significantly.

Thus far, we have somewhat lackluster results. While we found the GitHub of the paper that we are attempting to update (Feng [4]), the code was quite complicated and used many packages that were out-of-date. We decided it would be quicker to start from scratch. Additionally, we had to filter the NUS-WIDE dataset [1] to match their train/test data; this data preprocessing took longer and was more complicated that expected. However, after this preprocessing was completed, we have finished the first half of our feature extraction, using a Resnet model to extract features from the images. We also trained a very simple autoencoder; while we need to make significant changes to this autoencoder (for example, make it multimodal), we believe that this toy example helped us learn how to code the true Corr-AE much faster. It remains to extract features from the text data, build and train our autoencoder, and then find suitable hyperparameters (perhaps through a grid search).

3.2 Related Work

Currently, our model does not produce any particular result metrics for comparison with other related work referenced in this paper. However, these sources have nonetheless been fundamental for our work.

Traditional methods of cross-modal retrieval involve first applying representation learning techniques to each data modality independently, followed by correlation learning to align their features. In Feng et al. [4], the authors proposed a correlation autoencoder that performs both representation and correlation learning simultaneously. The model trains two autoencoders, one for each modality, that share a latent space, encouraging alignment between image and text features. Performance is evaluated based on reconstruction losses for each modality and a correlation loss that enforces cross-modal similarity.

In Frome et al. [5], one of the earliest deep learning approaches to cross-modal alignment, image features from a pretrained convolutional neural network are projected into the same semantic space as word embeddings. Unlike Corr-AE, which reconstructs data in each modality, DeViSE focuses on learning a ranking loss that encourages semantically similar image-text pairs to be close in the shared space. This design allows the model to generalize to unseen classes, demonstrating the potential of leveraging pretrained unimodal representations for cross-modal tasks.

More recent work benefits from powerful backbone architectures like BERT [2] and ResNet-18 [7] to extract semantically rich features from text and image data, respectively. BERT, a transformer-based language model pretrained on a large dataset using masked language modeling and next-sentence prediction, provides contextualized embeddings that significantly improve downstream language understanding tasks. In cross-modal retrieval, BERT's token-level embeddings are often pooled to produce sentence-level representations that align well with visual features in a joint embedding space.

On the visual side, ResNet-18 offers a compact yet expressive convolutional architecture that introduces residual connections to improve training stability and performance. It is commonly used to extract fixed-length image embeddings that serve as inputs to cross-modal models. When combined, BERT and ResNet-based encoders provide strong unimodal representations that can be fine-tuned or projected into a shared embedding space using contrastive, triplet, or ranking-based losses. This modern approach has largely replaced earlier architectures like RBMs, offering both higher accuracy and better generalization in retrieval tasks.

3.3 Experimental Setup

Thus far, we have created and saved image embeddings using a pretrained ResNet-50 model. For preprocessing, we downloaded the NUS-WIDE dataset from Kaggle [1] and filtered this dataset to 1,000 images of each of each of the 10 most populous classes. We loaded the 10-class image list and one-hot label files, verifying alignment between image paths and labels and organizing them into a DataFrame. Each image is assigned both a categorical label and an integer index. We then performed a stratified split into training (80%), validation (10%), and test (10%) sets. All images were resized to 128×128 pixels and pixel values were normalized to the range [0, 1]. We then batched into groups of 32.

For feature extraction, we used ResNet-50 pretrained on ImageNet, keeping its convolutional layers frozen to preserve the learned visual features. A global average pooling layer was applied to the final convolutional output, followed by a fully connected layer of size 256 with ReLU activation, producing compact embeddings. These embeddings were generated for all three splits and saved alongside CSV metadata with image paths and embedding indices.

We originally planned to use ResNet-18, but found ResNet-50 to be more accessible and better supported in TensorFlow. Future experiments may explore the trade-offs between these architectures, and we may pursue alternative means to implement ResNet-18, as originally planned. Though we have not yet performed retrieval or alignment, our current code establishes a consistent preprocessing and embedding extraction pipeline.

3.4 Experimental Results

Unfortunately, at this stage we are unable to present any meaningful results. The data preprocessing took much longer than anticipated, and the code used by [4], which we were planning to update and improve, of was quite complicated and out-of-date. As we had to start from scratch rather than borrow and improve code from the authors of [4], we are somewhat behind at this checkpoint, but we believe that we are still well-positioned to display high-quality, meaningful results by the end of this course.

3.5 Discussion

While we do not have real results to discuss yet, we can discuss what we have learned from this project thus far. The first is that [4], whose results we are trying to improve, used relatively small datasets for testing/validation/training (10k images total). Their results were based on the NUS-WIDE-10k dataset, which is a very small subset of NUS-WIDE, a dataset containing nearly 300K images. They limited their data to 1,000 images from each of the largets 10 classes in NUS-WIDE, which has 81 classes total. It is not clear why they chose to limit their training/testing dataset to such a small sample, or why they chose to include only 10 of the 81 groups of images/labels in NUS-WIDE, but to us this seems rather unnecessary. We have debated increasing our dataset to include the entirety of NUS-WIDE. Though this may degrade performance as there will be many more classes (81 vs. 10), we believe that such a simple dataset is unhelpful in evaluating the performance of multi-modal data retrieval. For example, a text query in google searching for images is not limited to only

10 classes. In order for Corr-AE to be of any interest, it must perform well with many classes.

3.6 Work Plan

We realize that we are slightly behind at this point in time, but we are confident that we will be able to deliver meaningful results by the end of this project. In the next week we are hoping to have written the code necessary to train our Corr-AE model on the smaller, NUS-WIDE-10k dataset. At this point, we will begin a grid search for tuning our hyperparameters (in the loss function). We expect this to be rather computationally expensive, but we believe that with the HCC we should not have any issues finding appropriate values for these hyperparameters. Finally, if time permits, we will extend our results to the entire NUS-WIDE dataset. One question that we have is that, rather than captions, like we anticipated, our the text modality is actually just a list of tags (i.e., one word descriptions) of the data. We are unsure how useful BERT will for feature extraction on such short text. We could change our data source instead to a list of images and full-length captions pulled from Wikipedia ([4] also tested their Corr-AE on this dataset), but as our preprocessing took so long we are rather loath to do this.

3.7 Conclusion

Thus far, we have preprocessed our data and built the feature extraction for the image modality of our data. We believe that in the next week we can complete feature extraction for the text modality and train our autoencoder. While this does not allow us to share meaningful results at this time, we are still confident that we are on-track to deliver meaningful results by the end of the term. In the next week we plan to complete the feature extraction for the text modality and have code written to train our Corr-AE. At this point, we will be ready to tune hyperparameters (the main hyperparameters are the architecture of the autoencoders and the weight in our loss function that balances reconstruction loss of the two modalities with the similarity score of the two modalities in the shared latent space). Once this is done, we plan on extending results to the entire NUS-WIDE dataset.

Our biggest question is if BERT will be able to meaningfully extract any features from multiple one-word tags of images rather than sentence-length captions. If not, would you advise we switch the dataset that we are working with?

Table 3.1: Contributions by team member for Milestone 3.

Team Member	**Contribution**
Derek DeBlieck	Code, Sections 3.1, 3.4-3.7
Sabrina Fowler	Data Preprocessing (Code)
Grace Hecke	Code, Experimental Setup
Abby Veiman	Code, Related Work

Chapter 4

Milestone 4: Progress Report 2

4.1 Introduction

Our project focuses on utilizing autoencoders with pretrained feature extractors for multi-modal data retrieval. For example, given a text input, our model will return an image from the test data that most closely represents this text. In the previous milestone, we trialed with feature extraction using ResNet-50 and developed an experimental autoencoder. This milestone, we built upon this preliminary work to develop three autoencoder-based experiments. We developed two self-reconstruction multi-modal autoencoders — one using MSE alignment loss for simplicity and stability, and the other with contrastive alignment loss for optimizing retrieval. We also developed a cross-modal autoencoder, in which image latents decode into the text feature space, and text latents decode into the image feature space.

Thus far, the cross-modal architecture has yielded the best results in both Recall@k and median rank. The self-reconstruction models yielded poorer results, but performed more comparably, though the use of contrastive alignment had a slight edge over MSE when it comes to median rank. In the upcoming milestone, we will continue model-refinement, such as by tuning hyperparameters. We will also determine our best performing model, and ultimately present and evaluate final discoveries and performance.

4.2 Related Work

To call back to our initial most influential works, Feng et al. [4] introduced the correlation autoencoder, which performs both representation and correlation learning through paired autoencoders sharing a latent space. Their design inspired our multi-modal autoencoder approach, though their experiments were limited by smaller datasets and older architectures.

Frome et al. [5] proposed DeViSE, an early deep learning model that projected image features from a CNN into the same semantic space as word embeddings, using a ranking loss to align modalities. Unlike the correlational autoencoder, DeViSE did not include reconstruction, focusing instead on contrastive-style similarity learning. This partially motivated our inclusion of a contrastive alignment loss as an alternative to mean squared error (MSE) in our own experiments.

In this milestone, we found ourselves in search of a suitable loss function for the shared latent space of our multi-modal autoencoder. We looked to more recent work by Zolfaghari et al. [16], which demonstrated the effectiveness of contrastive loss functions for cross-modal alignment, achieving strong retrieval results in video–text tasks. Their findings directly informed our use of a temperature-scaled contrastive loss, which produced modest improvements in median rank compared to MSE.

We continue to build upon various works which effectively incorporate pretrained feature extractors like BERT [2] and ResNet [7]. We implement BERT for text and ResNet-50 for images to produce semantically rich embeddings. Ultimately, these studies guided our experiments with self-reconstruction and cross-modal autoencoders.

4.3 Experimental Setup

We created three autoencoder-based experiments, building upon our image modality feature extraction from the previous Milestone 3. We implemented two self-reconstruction autoencoders (for both MSE and contrastive alignment loss) and one cross-modal autoencoder. The experiments were conducted using Flickr8k as opposed to NUS-WIDE, as its caption-structure is more conducive for our project's aim.

ResNet-50 was applied for feature extraction of the image modality, and BERT embeddings for the text modality. The 8,000-image Flickr8k dataset was split into training (70%), validation (15%), and testing (15%). Before feature extraction, normalization was applied using the standard ImageNet normalization in PyTorch. After feature extraction, normalization was applied using the statistics of the training set with z-score normalization. This normalization is performed separately for image and caption features and then applied to all splits.

Both self-reconstruction architectures consist of two independent autoencoders: one for images and one for text. Each autoencoder encodes its input into a shared latent space of 512 dimensions and decodes it back to the original feature space. Specifically, the image autoencoder consists of a $2048 \to 1024 \to 512$ encoder and a $512 \to 1024 \to 2048$ decoder, while the text autoencoder has a $768 \to 512 \to 512$ encoder and a $512 \to 512 \to 768$ decoder, both with ReLU activations between layers. To align the latent spaces of images and captions, we utilized an alignment loss, either mean squared error (MSE) or contrastive loss depending on the experiment. The experiment with contrastive loss utilizes a temperature hyperparameter of 0.07. The total loss is computed as the sum of the image reconstruction loss, the text reconstruction loss, and the weighted alignment loss.

The cross-modal autoencoder implements cross-modal reconstruction. In this model, the image encoder maps image features to the latent space, which is then decoded by the text decoder to reconstruct text features, and vice versa for text inputs. The encoder and decoder architectures are identical in size to those of the self-reconstruction model, with image and text encoders again producing 512-dimensional latent representations. ReLU activations are applied between layers, and a dropout rate of 0.3 is applied in the encoder layers for regularization. The training objective combines cross-modal reconstruction loss and a contrastive alignment loss. These two components are weighted in the total loss by λ_{recon} (for cross-modal reconstruction) and $\lambda_{\text{contrastive}}$ (for the contrastive alignment), both set to 1.0. A temperature of 0.07 is used for the contrastive component.

Both models were trained using the Adam optimizer with a learning rate of 1e-3, weight decay of 1e-5, and a batch size of 128 for a maximum of 40 epochs. Training was performed in mini-batches, and for each batch, features were passed through their respective encoders and decoders to compute latent representations and reconstructions, and both reconstruction and alignment losses were calculated. Gradients were backpropagated to update model parameters accordingly. After each epoch, model performance was evaluated on the validation set without gradient updates, and checkpoints were saved, including the model parameters, optimizer state, and performance metrics. The best model in a given experiment was identified based on the lowest validation loss and saved for subsequent evaluation and retrieval experiments.

4.4 Experimental Results

Table 4.1: Validation and Test Performance Metrics for Autoencoder Models

Model	Recall@1		Recall@5		Recall@10		Median Rank	
	Validation	Test	Validation	Test	Validation	Test	Validation	Test
CorrAE (MSE)	0.0507	0.0521	0.1641	0.1667	0.2521	0.2486	38.0	42.0
CorrAE (Contrastive)	0.0465	0.0504	0.1588	0.1741	0.2631	0.2768	30.0	30.0
Cross-Modal AE	0.1333	0.1336	0.3629	0.3718	0.4906	0.5033	11.0	10.0

In our experiments, we looked at retrieval performance on both the validation data and the test data. In particular, we computed Recall@K metrics for image retrieval given captions.

We also set up some visualization for the top 5 images retrieved for a given caption. Figures 4.1 and 4.2 are two examples of these visualizations for different autoencoder structures.

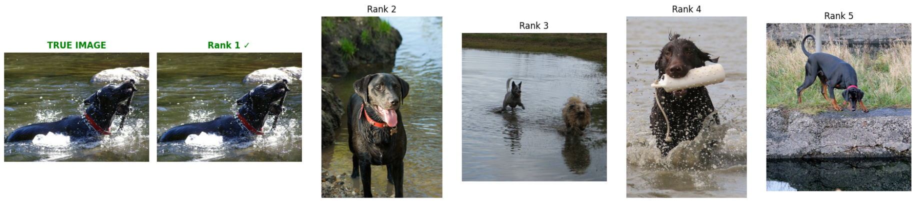

Figure 4.1: Retrieval Example for Cross-Modal Model
Caption: “a black dog wearing a red collar is dragging a rope though a river”

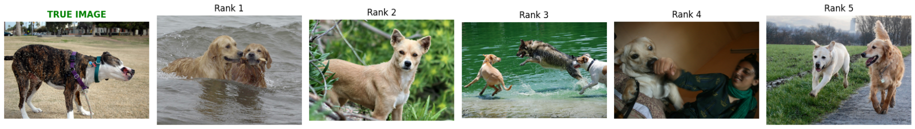

Figure 4.2: Retrieval Example for Corr-AE Model with MSE Alignment Loss
Caption: “A dog shakes water off of himself”

4.5 Discussion

The results in Table 4.1 show that the Cross-Modal Autoencoder achieved the highest retrieval performance among the three models, with a Recall@10 of 0.5033 and a median rank of 10 on the test set. Both self-reconstruction variants (MSE and contrastive alignment) yielded lower recall values and higher median ranks, with the MSE model reaching a Recall@10 of 0.2486 (median rank 42) and the contrastive model 0.2768 (median rank 30). These differences suggest that incorporating cross-modal reconstruction may lead to stronger correspondence between the image and text embeddings compared to self-reconstruction with alignment loss alone.

Within the two self-reconstruction experiments, both autoencoders achieved marginal difference in their recall metrics, though the contrastive alignment experiment yielded better median ranks. This pattern suggests that contrastive alignment may provide marginal benefits over MSE in guiding paired image–text representations toward similar latent embeddings.

The cross-modal autoencoder’s higher recall and lower median rank indicate that it produced embeddings more effective for retrieval in this particular experimental setting with Flickr8k. Overall, these findings provide empirical evidence that the cross-modal autoencoder yielded stronger retrieval performance under our tested conditions, while both self-reconstruction autoencoders achieved more modest alignment between modalities.

Figure 4.1 shows a very promising example of image retrieval with the cross-modal architecture. It shows the top 5 ranked images for the caption “a black dog wearing a red collar is dragging a rope through a river”. Here the correct image was ranked first and the rest of the top 5 images closely match the caption description, all showing a black dog in or near water, and two of which prominently feature a red collar.

In Figure 4.2 we see an example where the correct image was not ranked in the top 5, but the images that were chosen seem somewhat reasonable given the caption. All contain a dog or dogs and two of the top three contain water. This indicates that the feature extraction was successful and that the cosine similarity computations accurately reflect the data.

4.6 Work Plan

In the coming weeks, we aim to efficiently test different hyperparameters to optimize model performance, particularly for our most-promising model, the cross-modal autoencoder. This could be done using grid search on values like learning rate, batch size, contrastive temperature, etc. It may be best to optimize particularly for Recall@5, so as to not set too harsh or too loose performance expectations, though we are open to suggestions regarding the hyperparameter-tuning process. We will also need to present the final performance and discoveries of our project.

We also want to explore the efficacy of our model to perform retrieval in the opposite direction. So far we have been focusing on image retrieval given a caption, but we would also like to have good metrics with caption retrieval given an image. We may have to modify how we gauge success with this given that each image has 5 “correct” captions.

4.7 Conclusion

This milestone, we built upon our earlier work by developing and comparing three multi-modal autoencoder models: two self-reconstruction variants (using MSE and contrastive alignment) and one cross-modal autoencoder. Across our experiments on the Flickr8k dataset, the cross-modal architecture achieved the strongest retrieval performance, with noticeably higher recall and lower median ranks compared to the self-reconstruction models. Between the two self-reconstruction models, the contrastive alignment version performed slightly better than MSE, suggesting that contrastive loss may better align paired embeddings under our experimental conditions.

Moving forward, we plan to focus on refining the cross-modal model through hyperparameter tuning to improve retrieval results. Our next milestone will also center on presenting these final findings. As we begin the final stage, our remaining questions include: Are there any aspects of our model’s architecture that we should continue to experiment with? Are there any suggestions on hyperparameter tuning, such as which particular hyperparameters to prioritize? Given results thus far, what practical goals might we aim for in the next milestone? Is it worth doing hyperparameter tuning on the self-reconstruction autoencoder or should we focus exclusively on the cross-modal architecture?

Table 4.2: Contributions by team member for Milestone 4.

Team Member	**Contribution**
Derek DeBlieck	Feature Extraction & Architecture Design
Sabrina Fowler	Architecture Design & Experiment Running
Grace Hecke	Report Writeup & Code Review
Abby Veiman	Report Writeup & Code Review

Bibliography

[1] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pages 1–9, 2009.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. URL: `https://api.semanticscholar.org/CorpusID:52967399`.

[3] Z. Dorrani. Anomaly detection in emerging crimes with deep autoencoder architecture. *Contributions of Science and Technology for Engineering*, 2(3):45–56, 2025. `doi:10.22080/cste.2025.28900.1023`.

[4] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. *Proceedings of the 22nd ACM international conference on Multimedia*, 2014. URL: `https://api.semanticscholar.org/CorpusID:207216960`.

[5] Andrea Frome, Gregory S Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. 26:2121–2129, 2013.

[6] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2nd edition, 2019.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL: `http://arxiv.org/abs/1512.03385`, `arXiv:1512.03385`.

[8] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013.

[9] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color!: Joint end-to-end learning of global and local image priors for automatic

image colorization with simultaneous classification. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016. doi:10.1145/2897824.2925974.

[10] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL: http://www.cs.toronto.edu/~kriz/cifar.html.

[11] Yajing Liu, Christina M Cole, Chris Peterson, and Michael Kirby. Relu neural networks, polyhedral decompositions, and persistent homolog, 2023. URL: https://arxiv.org/abs/2306.17418, arXiv:2306.17418.

[12] Los Angeles Police Department, LAPD OpenData. Los Angeles Police Department Crime Data. https://data.lacity.org/Public-Safety/Crime-Data-from-2020-to-Present/2nrs-mtv8, 2025. Updated September 3, 2025; Latitude/Longitude location data included; Bi-monthly refresh rate.

[13] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, and Alan L. Yuille. Explain images with multimodal recurrent neural networks, 2014. URL: https://arxiv.org/abs/1410.1090, arXiv:1410.1090.

[14] Nora Youngs. The neural ring: using algebraic geometry to analyze neural codes, 2014. URL: https://arxiv.org/abs/1409.2544, arXiv:1409.2544.

[15] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *Computer Vision – ECCV 2016*, pages 649–666. Springer, 2016.

[16] Mohammadreza Zolfaghari, Yi Zhu, Peter Gehler, and Thomas Brox. Crossclr: Cross-modal contrastive learning for multi-modal video representations, 2021. URL: https://arxiv.org/abs/2109.14910, arXiv:2109.14910.