CROSS-MODAL PREDICTION USING CANONICAL CORRELATION ANALYSIS WITH PRIVACY PRESERVATION

by

Hoa T. Nguyen

A thesis presented to the Department of Computer Science
and the Graduate School of the University of Central Arkansas
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science

Conway, Arkansas
May 2021

TO THE OFFICE OF GRADUATE STUDIES:

The members of the Committee approve the thesis of Hoa T. Nguyen presented on April 9, 2021.

O.Kursun

Olcay Kursun, Ph.D., Committee Chairperson

Ahmad Alsharif

Ahmad Alsharif, Ph.D., Committee Co-Chairperson

Bernard Chen, Ph.D.

Tansel Halic, Ph.D.

PERMISSION

Title	Cross-Modal Prediction Using Canonical Correlation Analysis With Privacy Preservation
Department	Computer Science
Degree	Master of Science, Computer Science

In presenting this thesis/dissertation in partial fulfillment of the requirements for a graduate degree from the University of Central Arkansas, I agree that the Library of this University shall make it freely available for inspections. I further agree that permission for extensive copying for scholarly purposes may be granted by the professor who supervised my thesis/dissertation work, or, in the professor's absence, by the Chair of the Department or the Dean of the Graduate School. It is understood that due recognition shall be given to me and to the University of Central Arkansas in any scholarly use which may be made of any material in my thesis/dissertation.

Hoa T. Nguyen

April 16, 2021

ACKNOWLEDGEMENT

I wish to express my appreciation to my thesis advisor, Dr. Olcay Kursun, professor of the Department of Computer Science at the University of Central Arkansas. He was always supportive and encouraging during our time working on the thesis. He taught me much about machine learning, deep learning, and other materials needed for the thesis study. Dr. Kursun also provided me with some of the deep learning programs he developed under the DART project. This work was supported, in part, by the National Science Foundation under Award No. OIA-1946391.

I would also like to extend my gratitude to the faculty members of the Department of Computer Science at the University of Central Arkansas. With their guidance, I was able to complete my undergraduate and graduate studies. The lessons that I learned from them have enabled me to discover, continuously pursue, and turn my passion for Computer Science into a career after graduation.

Finally, I would like to thank my family and friends for being by my side and supporting me through the years of my studies. They have been most encouraging and always had faith in the academic and career path that I have chosen. Everything that I have achieved would not have been possible without them.

VITA

Hoa T. Nguyen studied towards her Bachelor's degree first at Arkansas State University – Beebe and then at the University of Central Arkansas (UCA), Conway, AR. In May 2020, she graduated from UCA, Magna Cum Laude, with a Bachelor of Science in two majors, Mathematics and Computer Science. She was enrolled in the 4+1 program at UCA and managed to complete her Master's studies with the thesis option in May 2021.

ABSTRACT

Canonical Correlation Analysis (CCA) is a multi-view feature extraction method that aims at finding correlated features (similarities) across multiple datasets (also called *views* or *modalities*). CCA characterizes these similarities by learning linear transformations of each view such that their extracted features have a maximal mutual correlation. As CCA is a linear method, the features are computed by a weighted sum of each view's variables. With the learned weights, CCA can be applied to test examples and serve in cross-modal prediction by inferring the target-view variables of an example from its given variables in a source (query) view. Being a linear method, CCA's applicability on unstructured datasets in cross-modal prediction is limited. Although kernel extensions of CCA theoretically generalize it to learn nonlinear transformations, combining CCA with deep learning is the state-of-the-art method for mining cross-modal correlations among unstructured data, such as images, audio, and text/tags. Moreover, as the traditional cryptographic tools block the ability of CCA to explore the similarities among encrypted views, with multiple correlated datasets involving multiple organizations, privacy-preserving extensions of CCA have been recently proposed. This thesis proposes a CCA-based method for cross-modal prediction on two datasets: Multi-View Digits, a structured dataset used as a proof-of-concept, and CIFAR-100, an unstructured dataset used to demonstrate the mining of image-tag correlations. The proposed method is designed to take advantage of deep learning solutions for feature extraction to deal with unstructured data. This thesis also describes a procedure that incorporates the solutions to Yao's Millionaires' problem, which is studied in the cybersecurity field to support privacy preservation, into the proposed CCA-based cross-modal prediction method.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND/OR ABBREVIATIONS

Symbol	Meaning	Page
$\boldsymbol{X}$	p-dimensional dataset	4
$\boldsymbol{Y}$	q-dimensional dataset	4
$\boldsymbol{W}_x$	Canonical weight matrix of dataset $\boldsymbol{X}$	4
$\boldsymbol{W}_y$	Canonical weight matrix of dataset $\boldsymbol{Y}$	4
$\boldsymbol{w}_x$	A single weight vector, a column of the canonical weight matrix of dataset $\boldsymbol{X}$	5
$\boldsymbol{w}_y$	A single weight vector, a column of the canonical weight matrix of dataset $\boldsymbol{Y}$	5
$\boldsymbol{u}$	Projection scores/Canonical variates of a CCA component for dataset $\boldsymbol{X}$	5
$\boldsymbol{v}$	Projection scores/Canonical variates of a CCA component for dataset $\boldsymbol{Y}$	5
K	Number of CCA components to extract	5
r_K	Correlation coefficient of the Kth CCA component pair	5
$\boldsymbol{W}_y{}^+$	The pseudoinverse of the canonical weights of dataset $\boldsymbol{Y}$	5

Abbreviation	Meaning	Page
CCA	Canonical Correlation Analysis	2
fou	Fourier coefficients of digit shapes	7
fac	Profile correlations	7
kar	Karhunen-Love coefficients	7
pix	Pixel averages in 2 x 3 windows	7
zer	Zernike moments	7
mor	Morphological features	7
CNNs	Convolutional Neural Networks	10
AlexNet	AlexNet pre-trained deep learning model	12
ResNet	ResNet101 pre-trained deep learning model	12
VGG	VGG11_nb pre-trained deep learning model	12
RGB	Red-Green-Blue	12
AR	Alternating Regression for CCA training	17
PCA	Principal Component Analysis	29

CHAPTER 1: INTRODUCTION

Cross-modal learning refers to the synthesis/prediction of information in one view/modality from the information in another. Based on emphasizing mutual information among modalities, cross-modal prediction has straightforward applications in machine learning and recommendation systems (Chen et al., 2017; Chen et al., 2010; Hardoon et al., 2004; Sakar & Kursun, 2017; Zhou et al., 2020) as well as more fundamental/theoretical importance in deep learning at the intersection of AI and neuroscience (Becker & Hinton, 1992; Favorov & Ryder, 2004; Körding & König, 2000; Kursun et al., 2021; Kursun & Favorov, 2019; Phillips & Singer, 1997). Cross-modal learning is an extremely crucial component in daily living. Humans are very good at integrating various sources of information, and the cortex is a powerful discoverer of regularities reflected in multiple modalities (Favorov & Ryder, 2004; Hawkins & Blakeslee, 2004). Actions such as learning to grasp and manipulate objects, to speak and understand a language all require the integration of visual, auditory, tactile, and other modalities. In many real-world problems, where data is observed/measured in multiple modalities, cross-modal learning plays an important role in feature extraction and prediction. Throughout the thesis, the terms *modality* and *view* are used interchangeably: A modality/view is a specific representation of a phenomenon.

In the context of cross-modal prediction, the thrust of machine learning is *multi-view feature extraction*. Cross-modal learning involves the extraction of descriptive and discriminative *features* from multiple modalities. These features can be used for subsequent learning (e.g. classification and clustering), better visualization, and interpretation. These features can be powerful as part of predictive models because they can help eliminate view-specific noise (Sakar & Kursun, 2017). Moreover, as the instances that belong to the same class can reflect their class-specific features in all modalities, extraction of mutual information between the modalities helps feature extraction methods to tune to class-specific features. Simply merging the features of all views and then performing a single-view feature extraction is not a promising direction. Although extracting/preserving complementary information from different sources/modalities is also important in machine learning, extracting a common space of features for all modalities/views allows one view's samples to be predicted by projecting other views' samples onto the space, which would be heavily related to the underlying sources of correlations and class-related features (Bilenko & Gallant, 2016; Hardoon et al., 2004; Kursun et al., 2011). Unlike the single-view feature extraction, the theory and application areas of multi-view feature extraction are more complex and less understood.

One approach to model the correlation between different modalities is *Canonical Correlation Analysis* (CCA). CCA aims at maximizing the correlation between modalities (Gong et al., 2014), which enables the ability to represent different modalities using a common feature subspace that can be mutually computed. The original formulation of CCA handles two views and maximizes linear correlations between them. Many novel CCA approaches have been introduced to tackle more than two views and complex nonlinear relationships, for example, multi-view CCA (Kettenring, 1971), tensor CCA (Luo et al., 2015), kernelized CCA (Hardoon, 2004), discriminative CCA (Sakar & Kursun, 2017), ensemble CCA (Sakar et al., 2014), deep neural networks based generalized CCA (Benton et al., 2017; Guo & Wu, 2019; Sun et al., 2008).

Although, as reviewed by Bilenko and Gallant (2016), the (pseudo) inverse of the CCA's transformation can be used for cross-modal prediction, in this thesis, a novel framework is proposed and tested for using the features learned by CCA to perform a search for highly matching examples in the target view. Two experimental datasets are used in this thesis: A multi-view digits dataset with modalities corresponding to different sets of features extracted from digit images; and a labeled image dataset, where one modality was image features extracted by a deep learning method, and the other was the set of noisy tags describing the class/superclass of the images. The proposed method is designed to achieve cross-modal recommendation without having to reconstruct the target view (as explained in Section 2.1). Taking advantage of the mutually correlated features that CCA learns to extract, the proposed cross-modal method performs its nearest neighbors search in the canonical subspace common to both the query and the target views. The thesis also looked into feature extraction with deep learning for preprocessing image features and touched on the issue of privacy preservation during the CCA training.

The thesis is organized as follows. The materials and methods used in the thesis are reviewed in Chapter 2. The proposed CCA-based method for cross-modal prediction is described in Chapter 3. The experimental results on some exemplary benchmark datasets are presented in Chapter 4. Finally, the conclusions and potential future work are discussed in Chapter 5.

CHAPTER 2: MATERIALS & METHODS

This chapter provides brief descriptions of the materials and methods used in the thesis: The explanation of the role CCA plays in the cross-modal prediction method, the definition of a multi-view dataset, an introduction of the experimental datasets, and an overview of the convolutional neural networks used for image feature extraction.

2.1 CCA For Cross-Modal Prediction

The standard approach of CCA is a linear dimensionality reduction method that requires two views as inputs, which are used to guide each other in the feature extraction process (Hotelling, 1992; Sakar & Kursun, 2017; Sakar et al., 2014; Yuan & Sun, 2013), which is illustrated in Figure 2.1.1. The two input views have different sets of features and *N* examples, $\boldsymbol{X} \in \mathbb{R}^{p \times N}$ and $\boldsymbol{Y} \in \mathbb{R}^{q \times N}$. They can be of different dimensionality (View-1 can have p features, and View-2 can have q features). CCA aims at finding component pairs to link these views.

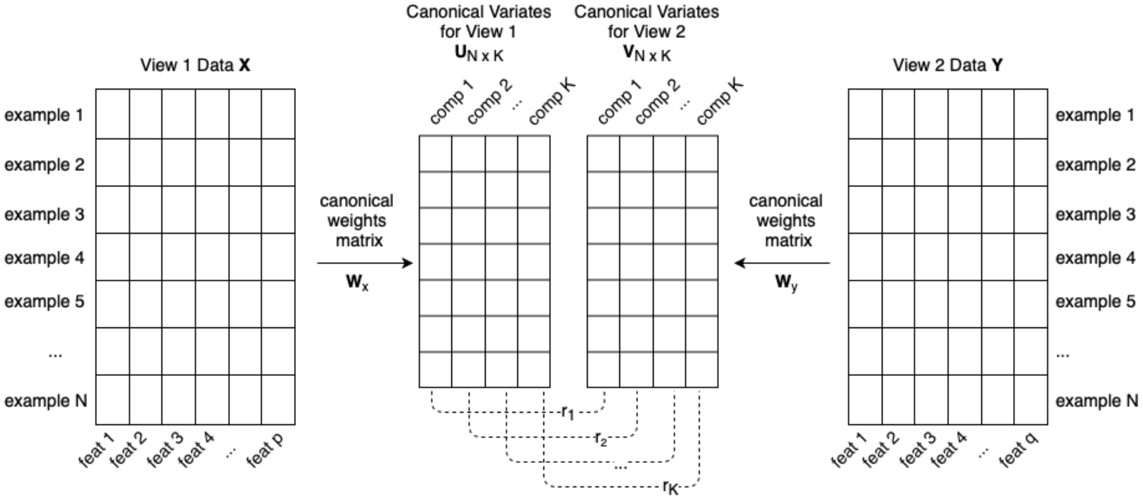

Figure 2.1.1. A general schematic for Canonical Correlation Analysis (CCA) learning of two-view data.

As CCA is linear, extracting maximally correlated features of the views can be expressed as finding *K* pairs of *canonical weights* $\boldsymbol{W}_x \in \mathbb{R}^{p \times K}$ and $\boldsymbol{W}_y \in \mathbb{R}^{q \times K}$, such that when the two sets of feature vectors are projected onto the canonical space, their K component pairs, or *canonical variates/projections scores*, are maximally correlated. For example, take $\boldsymbol{w}_x \in \mathbb{R}^{p \times 1}$ and $\boldsymbol{w}_y \in \mathbb{R}^{q \times 1}$ as the canonical weight vectors of View-1 and View-2, respectively. CCA maximizes the correlation between the canonical variates $\boldsymbol{u} = {\boldsymbol{w}_x}^T\boldsymbol{X}$ and $\boldsymbol{v} = {\boldsymbol{w}_y}^T\boldsymbol{Y}$, where the superscript *T* denotes transpose. The pair of canonical variates with the maximal correlation is the principal component, and if multiple components are sought (*K* > 1), within each view the canonical weight vectors must be orthogonal. Theoretically, the maximum number of components is equal to the minimum of the ranks of the two views; however, *K* components can be extracted such that the correlation coefficient r_K is greater than a given correlation threshold T and r_{K+1}, or such that r_{K+1} is sufficiently low (refer to Eq. 1).

$$\begin{aligned} &r_i = max\ corr\left({\boldsymbol{w}_{x_i}}^T\boldsymbol{X}, {\boldsymbol{w}_{y_i}}^T\boldsymbol{Y}\right), \quad for\ 1 \leq i \leq K \\ &{\boldsymbol{w}_{x_i}}^T\boldsymbol{w}_{x_j} = {\boldsymbol{w}_{y_i}}^T\boldsymbol{w}_{y_j} = 0, \qquad for\ 1 \leq i,j \leq K, i \neq j \\ &r_K \geq T > r_{K+1} \end{aligned} \tag{1}$$

After using CCA to transform two or more datasets, Bilenko and Gallant (2016) suggested that a dataset's samples could be predicted as the dot product between the inverse of its canonical weights and the projected samples of the remaining dataset's samples onto the canonical space. This thesis refers to this prediction method as the *pseudoinverse method*, because:

$$\boldsymbol{Y}^{pred} = {\boldsymbol{W}_y}^{+} \cdot ({\boldsymbol{W}_x}^T\boldsymbol{X}) \tag{2}$$

Where $\boldsymbol{Y}^{pred}$ Where is View-2's feature vectors to be predicted; ${\boldsymbol{W}_y}^{+}$ is the pseudoinverse of canonical weights of View-2; $\boldsymbol{X}$ is View-1's query feature vectors; and $\boldsymbol{W}_x$ is the canonical weights of View-1.

Bilenko and Gallant (2016) assessed the efficiency of this method by the total correlation coefficient of the predicted and the actual samples (sum of q correlation coefficients, as View-2 is q-dimensional).

$$total\ correlation\ coefficient\ = \sum_i^q corr\left(\boldsymbol{Y}_i{}^{pred}, \boldsymbol{Y}_i{}^{actual}\right) \qquad (3)$$

2.2 Multi-View Datasets

A *dataset* is commonly defined as a tabular collection of data used for an analysis question. In real-world applications, however, it is not possible to include every aspect of interest in a single table simply because of the vast number of features that might involve. Instead, there can be multiple datasets that provide diverse and complementary information from distinct perspectives of the same phenomenon. A collection of such individual datasets makes a multi-view dataset. Depending on the subject of analysis, publicly available datasets may come with an intentional separation of views, while others only present one table of data, which can then be split into views based on the variables and data information given.

The multi-view datasets used in this thesis were Multi-View Digits and CIFAR-100. These choices were motivated by two considerations. First, the datasets were publicly available and they included rich data. The former had six different views and the latter was a well-known dataset for deep learning image classification.

Multi-View Digits dataset: Originally named *Multiple Feature* in the UC Irvine Machine Learning Repository, this dataset is referred to as *Multi-View Digits* here to better represent the information that it provides. The dataset includes features of handwritten numerals (*0-9*), which are extracted from a collection of Dutch utility maps and digitized into binary images. For each digit, 200 patterns are being considered, totaled to 2,000 patterns. The digits are represented by six feature sets: 76 Fourier coefficients of the character shapes (fou), 216 profile correlations (fac), 64 Karhunen-Love coefficients (kar), 240 pixel averages in 2 x 3 windows (pix), 47 Zernike moments (zer), and 6 morphological features (mor) (Dua & Graff, 2019).

The Multi-View Digits dataset was used for the demonstration of querying one modality to acquire similar examples from another. Among the feature sets, some display a much higher correlation than others. For example, the relationship between Karhunen-Love coefficients and the pixel averages is prominent because both of them perform linear weighted sums of the pixels and thus are easily convertible to each other (the former keeps the high variance components and the latter simply takes averages of local groups of pixels). Therefore, CCA could easily find a linear transformation of the views that highly correlate. For this thesis, trials were performed for all 15 pairwise feature sets.

CIFAR-100 dataset (Krizhevsky & Hinton, 2009): This dataset contains labeled images collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton (Figure 2.2.1). The images are categorized into 100 classes with 6,000 images each, and the classes are further grouped into 20 superclasses (Table 2.2.1).

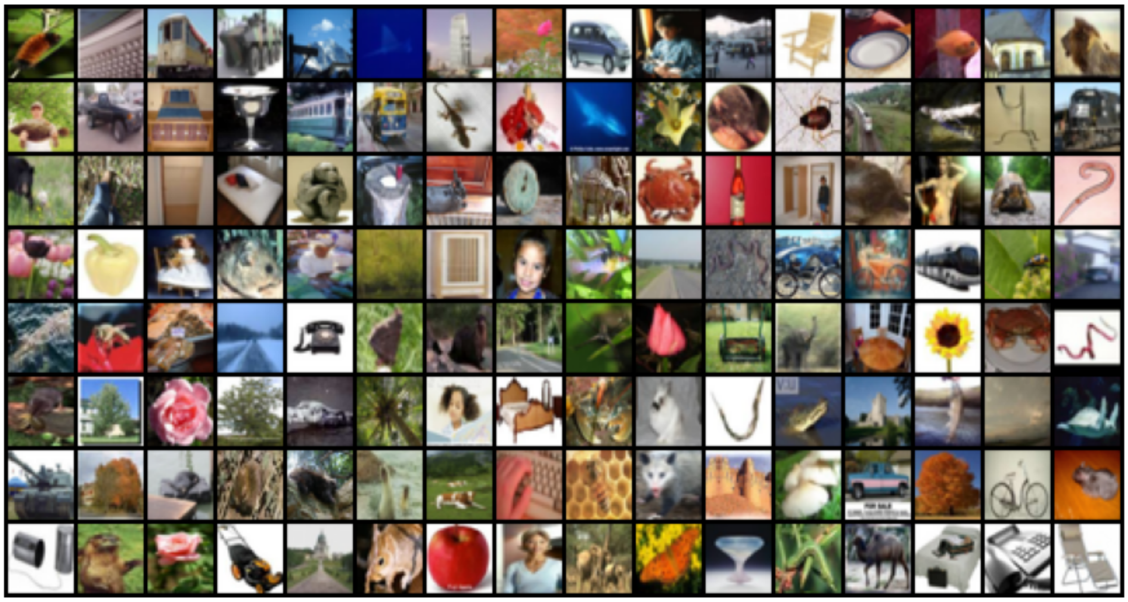

Figure 2.2.1. Some exemplary images of the CIFAR-100 dataset.

Superclass	**Class**
aquatic mammals	beaver, dolphin, otter, seal, whale
fish	aquarium fish, flatfish, ray, shark, trout
flowers	orchids, poppies, roses, sunflowers, tulips
food containers	bottles, bowls, cans, cups, plates
vehicles 2	lawn-mower, rocket, streetcar, tank, tractor

Table 2.2.1. Five exemplary superclasses and their corresponding classes in the CIFAR-100 dataset.

Theoretically, CIFAR-100 does come with two views, where one view is the image-view and the second view is the class label of that image. Typically, such datasets are used for supervised classification/discrimination. CCA and *linear discriminant analysis* (a form of linear dimensionality reduction performable by a linear classifier such as perceptrons and logistic regression) are equivalent when the class label is used as the second view (Bartlett, 1938; Kursun et al., 2011). With that background and the existence of classes and superclasses in the CIFAR dataset, a more sophisticated version of the class-label information was created. Each image belongs to a superclass, which will not be directly serving as its *tag*. With the class and superclass information, a binary 600x100 dataset was constructed, where a 0 value in the dataset means that an image does not belong to a particular class and 1 otherwise. The encoding scheme was not simply a one-hot encoding operation – A probabilistic approach was used to ensure the uncertainty of an image's true tag and enforce the superclass information. In other words, for a particular image, the highest probability of getting a 1 was applied for its true class, a fairly high probability for classes in the same superclass, and a low probability for the rest of the classes. With these specifications, for instance, if an image was known to be labeled *beaver* (*aquatic mammals* superclass) a possible set of entries for it in the binary dataset would be:

Beaver	**Dolphin**	**Otter**	**Seal**	**Whale**	**…**	**Bottle**	**…**	**Rabbit**	**…**	**Tractor**
1	0	1	1	0	…	0	…	1	…	0

Although ideally, such tagging task should be performed as part of a survey with human subjects, for simplicity, the described noisy encoding scheme was used. If the *beaver* entry had a 1, and everything else was 0, it would be more suitable for a supervised classification problem, not for multi-view feature extraction and/or cross-modal prediction problem. It has been shown that when the class label is used as the second view, CCA is equivalent to linear discriminant analysis (Bartlett, 1938; Kursun et al., 2011). When using the proposed noisy encoding scheme that favors but not exactly identifies the class and the superclass, the demand on CCA would be to find the most matching linear combinations of image features with linear combinations of these noisy tags. As multiple 1's are associated with an image (with more probability of being 1 for the classes that belong to the superclass of the image), CCA captures superclass information, and the proposed CCA-based cross-modal prediction learns to suggest which superclass the given query image belongs to. Note that this cross-modal prediction is achieved without explicitly training a supervised classifier for predicting the superclasses. Furthermore, this dataset offers a controlled data set that can be used for sensitivity analysis, for example, to explore how much noise can be tolerated, or to test the robustness of various deep learning methods against factors such as noise and the number of CCA components.

In this thesis, the tag-related dataset is referred to as CIFAR-100-tag and the image dataset as CIFAR-100-image. Deep learning models were used as feature extractors for images in the CIFAR-100-image dataset. The CIFAR-100-tag data and the deep-learning features were the two modalities fed to the cross-modal prediction model.

2.3 Convolutional Neural Networks

Part of the analysis using CIFAR-100-image is feature extraction with deep learning models based on the *convolutional neural networks* (CNNs). Similar to *neural networks*, CNNs consist of several neurons with learnable weights and biases. A neuron is connected to and receives inputs from other neurons in the network. It then takes a weighted sum over the inputs, passes the result through an activation function, and outputs its response.

O'Shea and Nash (2015) provided a thorough explanation of how CNNs differ from neural networks. A CNN processes an input image with a set of convolutional layers, each of which comprises independent filters. Convolving the whole image with a filter results in a feature map. Following this procedure, feature maps created from convolving the image with all filters combine into a convolutional layer. The initialization of filters is random, and the CNNs learn them subsequently. CNNs involve pooling layers, whose function continuously reduces the spatial size of an image's representation to decrease the number of parameters and the amount of computation in the network. CNNs use the rectified linear activation function (ReLU), which returns the input directly if it is positive and zero otherwise. Figure 2.3.1 shows the typical architecture of a CNN: An input image goes into the network where its features are extracted at each convolutional layer, and the outputs are inputs to the next layer. The fully-connected (FC) layer has connections to all activations from the previous layer, and its outputs are fed into a softmax activation function to calculate the probability of the input image being labeled as a specific tag, hence the class classification for the image.

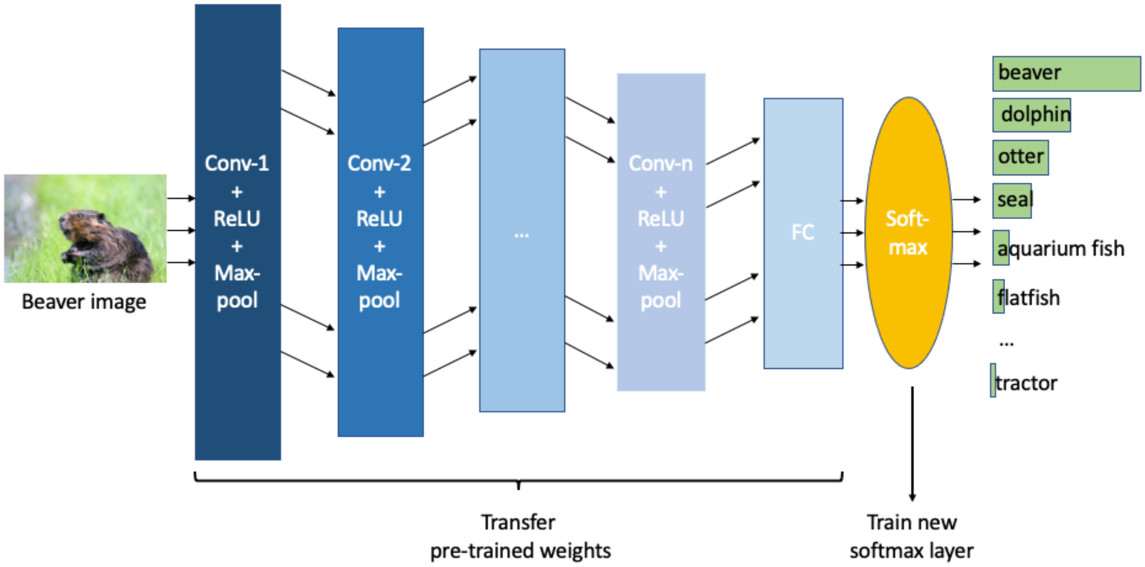

Figure 2.3.1. An example of CNN architecture.

In this thesis, the image feature extractors used were pre-trained CNN models. Their abilities in discovering distinguishable patterns in images were compared. The application of pre-trained CNNs is part of transfer learning, a subfield of machine learning and artificial intelligence that exercises the knowledge gained from a source task to a different but similar target task, which is one of the benefits of deep learning systems (Goodfellow, 2016; Kursun et al., 2021; Yosinski et al., 2014). Pre-trained CNNs are models that are already built on very large datasets for image classification. These models are then made public to be repurposed and fine-tuned with regards to the learned layers, features, weights, and biases, thus achieve higher accuracy and generate the intended output format (Shin et al., 2016). Pre-trained CNN models provide a shortcut to training a CNN from scratch, which may take up time and resources depending on the size of the dataset.

Three pre-trained CNNs were investigated: AlexNet, ResNet, and VGG (Paszke et al., 2019). All three models were trained on the ImageNet database, which has more than 14 million images grouped into about 22 thousand classes (according to statistics recorded on ImageNet's homepage). AlexNet architecture includes eight layers, five of which are convolutional layers, and three are fully connected layers. The input to AlexNet must be RGB images of size 256×256 (Krizhevsky et al., 2012). ResNet (short for Residual Network) was built and trained on one million 224x224 colored images from ImageNet (He et al., 2016). There have been multiple versions of ResNet depending on the number of layers. For this thesis, ResNet101 was used (101 layers). VGG requires RGB images of dimensions 224x224 (Simonyan & Zisserman, 2014). Similar to ResNet, there are multiple versions of VGG. This thesis used VGG11_bn, which is a batch-norm version.

CHAPTER 3: CCA-BASED CROSS-MODAL PREDICTION PROCEDURE

Section 2.1 reviewed a cross-modal prediction approach that worked by inverting the canonical weight matrix of the target view. This essentially means transforming the query view to the canonical subspace and then, using the inverted weights, converting the canonical subspace to obtain the most fitting (but artificial) representations in the target view. This thesis proposes that for cross-modal prediction, performing the search within the canonical subspace is better than generating artificial representations. Even if the sample reconstruction generates a realistic target-view example, it is not a real example in the target-view dataset. To find the real examples, a nearest neighbors search would be performed for the retrieval task of relevant items from the target view. Therefore, instead of inverting the CCA features of the query to obtain a reasonable representation in the target-view space, the proposed approach applies a nearest neighbors search within the canonical subspace. As the CCA features are generally much fewer than the original dimensionality, the proposed approach offers higher performance in the accuracy rate, which will be reported in Chapter 4.

The proposed cross-modal prediction process involved transforming the data with CCA and applied an unsupervised nearest neighbors model to acquire similar examples from one view given the features from another view. The performance of the proposed prediction process was addressed based on three performance metrics: cross-modal classification error, cross-modal-top-3 classification error, and within-/cross-modal intersection error. These errors also provided a basis to assess the benefits of using canonical variates in the nearest neighbors model, as opposed to reconstructing them back to their original space (the pseudoinverse method). This chapter explains the procedure to demonstrate the proposed method of cross-modal prediction with CCA.

3.1 Application of CCA & Nearest Neighbors Model

Cross-modal learning was performed with an *n*-component CCA model: CCA found the weights that maximize the correlation between the two views while limiting to *n* components in the final model equation. The steps to train a CCA model and obtain the canonical space for two views appear in Table 3.1.1.

Step	Description
1	Subsequently split the two views' data and their labels into train and test sets, respectively.
2	Trained a CCA model using the training data of the two views.
3	Transformed the training and testing data of both views into canonical variates using the trained CCA model.

Table 3.1.1. Steps to train a CCA model and obtain the canonical space for two views.

With the training canonical variates from the query view, an *unsupervised nearest neighbors* model (Ball Tree algorithm) was trained and used to query the test canonical variates from the target view. The model returned target view representations (neighbors) whose features correlated most to the query's features. A summary of this cross-modal prediction process is illustrated in Figure 3.1.1.

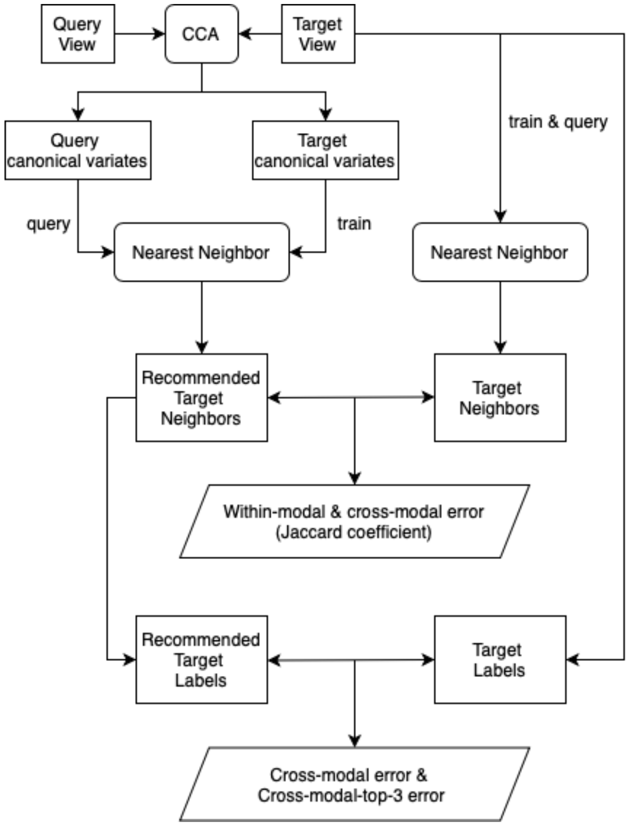
Query View
CCA
Target View
Query canonical variates
Target canonical variates
train & query
query
train
Nearest Neighbor
Nearest Neighbor
Recommended Target Neighbors
Target Neighbors
Within-modal & cross-modal error (Jaccard coefficient)
Recommended Target Labels
Target Labels
Cross-modal error & Cross-modal-top-3 error

Figure 3.1.1. Proposed cross-modal prediction process with CCA and evaluation method using within-/cross-modal intersection error, cross-modal classification error, and cross-modal-top-3 classification error.

The within-/cross-modal intersection error evaluated how much CCA helped in the cross-modal learning. This is a value adapted from the Jaccard similarity coefficient - a similarity measure between finite sample sets (Jaccard, 1912):

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

As used in this thesis, *A* was the set of target-view examples recommended by CCA, and *B* represented the target-view examples that would be returned by a search that queried the target-view directly if the true target-domain examples were available. In other words, *B* represented the ideal neighbors of a test example found by a standard single-view nearest neighbors search if the target-view of the test example was actually known. With both sets defined, the within-/cross-modal intersection error returns the percentage of the test example with *A* and *B* sets having a non-empty intersection. For example, if most test examples had empty intersection between their *A* and *B* sets, then the recommendations and the ideal neighbors were not matching at all, thus leading to a high within-/cross-modal intersection error.

3.2 Classification Accuracy Evaluation Method

Good recommended neighbors would share similar features to the query examples, but their class labels might not match the expected recommendations due to biases in the data. For example, the extracted features of handwritten numerals in the Multi-View Digits dataset might be affected by noises such as handwriting styles (some digits might be written by the same person, so they have the same stroke strength, straightness, etc.). As CCA was projecting the features onto a common latent space, its results might be biased by these noises. The nearest neighbors model would then return neighbors with the same biased features but are not of the same digits.

Figure 3.1.1 also illustrates the proposed method to verify CCA's ability to retain enough classification-related information (i.e., features or dimensions). The cross-modal classification error was computed to report the mismatched recommended labels from querying examples in a different view. After the search within the canonical space returned recommended target-view representations, their labels were then compared to the target view's actual test labels to get the cross-modal classification error. Additionally, the top-3 most recommended labels (based on their majorities among the found neighbors in the target view) gave the cross-modal-top-3 classification error, which offers a softer measure of performance to evaluate the accuracy of the cross-modal prediction.

3.3 Privacy Preservation Based On Millionaires' Problem & Alternating Regression

The training process of the CCA model requires data from all modalities involved. However, this might not be practical in real-world use cases because the availability of data depends on privacy constraints. The progress of e-commerce platforms and data mining comes with the continuously increasing volume of sensitive information. Inevitably, means for secure and private information transactions and computation come in great demand. This is where the Yao's Millionaires' problem finds its application. Introduced in 1982 by Andrew Yao, a computer scientist and computational theorist, the problem discusses *secure multi-party computation* (SMC) by presenting an example of two millionaires who want to know which of them is richer without revealing their actual wealth. In a larger sense, SMC asks for protocols to enable a computational function that can be used by several parties without the need to expose their input (Yao, 1982). The Millionaires' problem is an important factor to consider in commercial applications, where a comparison between confidential parties occurs.

Any solutions to the Millionaires' problem can be used in the training of the canonical correlation algorithm when considering the concept of *alternating regression* (AR) for CCA's implementation as in (Lai & Fyfe, 1998; Sakar & Kursun, 2017; Wold, 1966). The AR method for CCA training randomly initializes $\boldsymbol{w}_x$ and $\boldsymbol{w}_y$ of Eq. 1. It then uses an iterative approach based on backpropagation to update $\boldsymbol{w}_x$ and $\boldsymbol{w}_y$ as opposed to an analytic solution based on cross-covariance matrix computation. At every iteration of alternating regression, the neural networks of each view try to produce an output that maximally agrees with the outputs of the other views. The weights, $\boldsymbol{w}_x$ and $\boldsymbol{w}_y$, are updated via backpropagation (Alpaydin, 2014; Favorov & Ryder, 2004; Sakar & Kursun, 2017). Although the error to be minimized is based on the differences of the outputs, a privacy-preserving algorithm can be obtained by making small updates to increase the agreement without loss of generality, where the extraction of one CCA covariate is outlined, and the other covariates can be found by the deflation or symmetric lateral inhibition (Alpaydin, 2014; Girolami & Fyfe, 1997). The proposed procedure for the privacy-preserving algorithm appears in Table 3.3.1.

```
Randomly initialize $\boldsymbol{w}_x$ and $\boldsymbol{w}_y$
Repeat
    Compute the projection scores for each training example as:
    $\boldsymbol{u} = \boldsymbol{w}_x{}^T\boldsymbol{X}$ and $\boldsymbol{v} = \boldsymbol{w}_y{}^T\boldsymbol{Y}$ (as in Eq. 1)
    Feed the projection scores to the privacy preserving solution of the Millionaire problem
    If $|\boldsymbol{u}| > |\boldsymbol{v}|$
        Apply the backpropagation algorithm to update $\boldsymbol{w}_y$ such that a similar example would produce a $\boldsymbol{v}$ with a higher magnitude
    Else
        Apply the backpropagation algorithm to update $\boldsymbol{w}_x$ such that a similar example would produce an $\boldsymbol{u}$ with a higher magnitude
Until convergence
```

Table 3.3.1. Proposed CCA-privacy-preserving algorithm in cross-modal prediction.

Some solutions to the Millionaire's problem are based on homomorphic encryption (Kumar & Gupta, 2013; Lin & Tzeng, 2005). Homomorphic encryption (Gentry, 2009) is a form of public-key encryption that allows computations to be carried out over encrypted data, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the original data. Homomorphic encryption can be expressed as follows:

$$E(m_1) \times E(m_2) = E(m_1 \otimes m_2) \qquad (5)$$

Where $E(m)$ represents the encryption of a message m; operation × is the multiplication operation of two encrypted messages; and $\otimes$ is an equivalent operation in plaintext domain, which could be addition (Paillier, 1999) or multiplication (ElGamal, 1985).

CHAPTER 4: EXPERIMENTAL RESULTS

This chapter reports the results for the cross-modal prediction procedure as described in Chapter 3 on the Multi-View Digits and CIFAR-100 datasets. The within-/cross-modal intersection error, cross-modal classification error, and cross-modal-top-3 classification error were interpreted to reflect on the performance of the prediction process. For both experiments with the datasets, the errors were computed for the two cross-modal prediction methods explained in this thesis: The pseudoinverse method, where a nearest neighbors search was performed after the reconstruction of the canonical space, and the proposed method, where the search for most fitting representations of the target view was within the canonical space. With the Multi-View Digits dataset, all 15 modality pairs were investigated, and trials were performed for different correlation thresholds that were used to select the number of CCA components to extract. With the CIFAR-100 dataset, the efficiency of three pre-trained CNNs (AlexNet, ResNet, and VGG) in extracting image features to be fed into CCA was addressed. Additionally, different numbers of CCA components were also used to reflect on the change in the prediction accuracy.

4.1 Multi-View Digits Dataset

The Multi-View Digits dataset was used for the demonstration of cross-modal prediction by a nearest neighbors search in the canonical subspace of two modalities. This thesis worked with all 15 pairs among the six feature sets provided by the dataset. For each modality pair, three test errors were calculated: within-/cross-modal intersection error, cross-modal classification error, and cross-modal-top-3 classification error. Each modality pair had two options for the target and query view. As a result, there were 30 values found for each of the three errors. Furthermore, the three errors were also computed for the pseudoinverse cross-modal prediction method.

As each modality has a different number of features, the initial number of components used for the CCA model was chosen to be the minimum rank of a modality pair. Then, the first *K* learned components were selected such that $r_K \geq 0.5 > r_{K+1}$, to create the CCA model that would be used for the cross-modal prediction process. All errors computed are reported in Table 4.1.1.

Query	Target	Within-/cross-modal intersection error		Cross-modal classification error		Cross-modal-top-3 classification error	
		(a)	(b)	(a)	(b)	(a)	(b)
fac	fou	0.68125	0.69000	0.90000	0.74000	0.67000	0.61125
	kar	0.08000	0.04500	0.57125	0.29750	0.20500	0.08375
	mor	0.77250	0.96125	0.78375	0.92500	0.55250	0.69500
	pix	0.22875	0.21500	0.49375	0.43875	0.29000	0.25125
	zer	0.01125	0.03375	0.22500	0.32750	0.03125	0.09625
fou	fac	0.09250	0.70375	0.19625	0.62750	0.04125	0.28375
	kar	0.11375	0.26250	0.21375	0.27500	0.05625	0.15375
	mor	0.47125	0.92375	0.33875	0.70375	0.10500	0.38375
	pix	0.07500	0.27000	0.17625	0.29750	0.04250	0.15250
	zer	0.21375	0.39625	0.26875	0.38375	0.07000	0.12875
kar	fac	0.00000	0.14500	0.06125	0.31000	0.00500	0.06250
	fou	0.11375	0.16750	0.20250	0.25750	0.03500	0.06000
	mor	0.42625	0.92875	0.35750	0.77750	0.15625	0.43375
	pix	0.00000	0.00000	0.06250	0.06875	0.00375	0.00375
	zer	0.05875	0.39375	0.21875	0.55000	0.06500	0.23125
mor	fac	0.37250	0.60375	0.48375	0.57375	0.14250	0.22250
	fou	0.34500	0.54375	0.37625	0.70375	0.06500	0.20500
	kar	0.30750	0.47500	0.46500	0.47000	0.11125	0.30000
	pix	0.22500	0.48000	0.37750	0.49750	0.11625	0.28250
	zer	0.29250	0.82125	0.38125	0.82500	0.09375	0.51250
pix	fac	0.00125	0.20500	0.08125	0.40875	0.00875	0.09125
	fou	0.13750	0.17625	0.32000	0.26125	0.10875	0.05750
	kar	0.00000	0.00000	0.05500	0.12125	0.00250	0.00875
	mor	0.41250	0.95375	0.38500	0.82875	0.14750	0.54625
	zer	0.10250	0.56000	0.43875	0.62750	0.13875	0.30875
zer	fac	0.00500	0.30000	0.18500	0.47250	0.01375	0.20625
	fou	0.76500	0.70875	0.89500	0.75125	0.64500	0.61500
	kar	0.64750	0.58750	0.90000	0.78250	0.43250	0.49375
	mor	0.79250	0.96125	0.89125	0.91000	0.67000	0.60625
	pix	0.66625	0.56125	0.90000	0.68000	0.60750	0.51500

Table 4.1.1. Test errors computed for Multi-View Digits modality pairs when performing nearest neighbors search (a) within the canonical space and (b) after reconstructing the space.

Performing the search for 20 nearest neighbors within the canonical space yielded small errors for most target-query pairs. For several modality pairs, the choice of which view was the query/target view had a significant impact to increase or decrease the errors. For example, when Zernike moments were used to query for the most fitting representations of Karhunen-Love coefficients, all three errors found were relatively high. With a within-/cross-modal intersection error of 0.6475, only 35.25% of the time was there an agreement between the true neighbors and the recommendations by cross-modal learning. The class labels of the recommended had only a 10% match rate with the true ones when using a cross-modal query, while the top-3 match rate was 56.75%. However, if the Karhunen-Love coefficients were the query view and Zernike moments were the target, all three rates improved significantly to 94.13%, 78.13%, and 93.5%, respectively.

It was also worth noting a few modality pairs with near-zero errors. One such pair was the Karhunen-Love coefficients and pixel averages. No matter which of the two views was the query/target view, the errors were close to or even zero. As explained in Section 2.2, this was because Karhunen-Love coefficients are obtained by a linear method that takes the weighted averages of pixels. In other words, the relationship between these two modalities was so prominent that CCA could linearly transform their examples easily and returned the correct representations between them.

Compared with the pseudoinverse method of reconstructing the canonical space and generating the most fitting (and artificial) representations in the target view, the proposed method of performing the nearest neighbors search within the canonical space was more accurate. As shown in Figure 4.1.1, most errors computed from using the proposed method were smaller than those of the pseudoinverse method. For more insight into how much improvement the search within the canonical space had on the prediction accuracy, the error difference between the two methods was investigated. For each pair of corresponding errors (based on the modality pair used), the difference was calculated. Whichever method the smaller error belonged to, the average difference (Eq. 6) was computed for that method and compared to the average difference of the other method. For the proposed method, the average error difference was 0.2579 for the within-/cross-modal intersection error. This meant that for all modality pairs that received a smaller within-/cross-modal intersection error, the average error drop compared to using the pseudoinverse method was 0.2579. The proposed method had an average difference of 0.2051 for the cross-modal classification error and 0.1557 for the cross-modal-top-3 classification error. On the other hand, with the pseudoinverse method, the average differences were 0.0540, 0.1470, and 0.0652, respectively. The fact that all three error differences of the proposed method were bigger than those of the pseudoinverse method showed that cross-modal prediction using canonical variates directly was able to decrease the errors, hench improved the accuracy of the prediction.

$$average\ error\ difference = \frac{\sum(bigger\ error - smaller\ error)}{number\ of\ smaller\ errors} \tag{6}$$

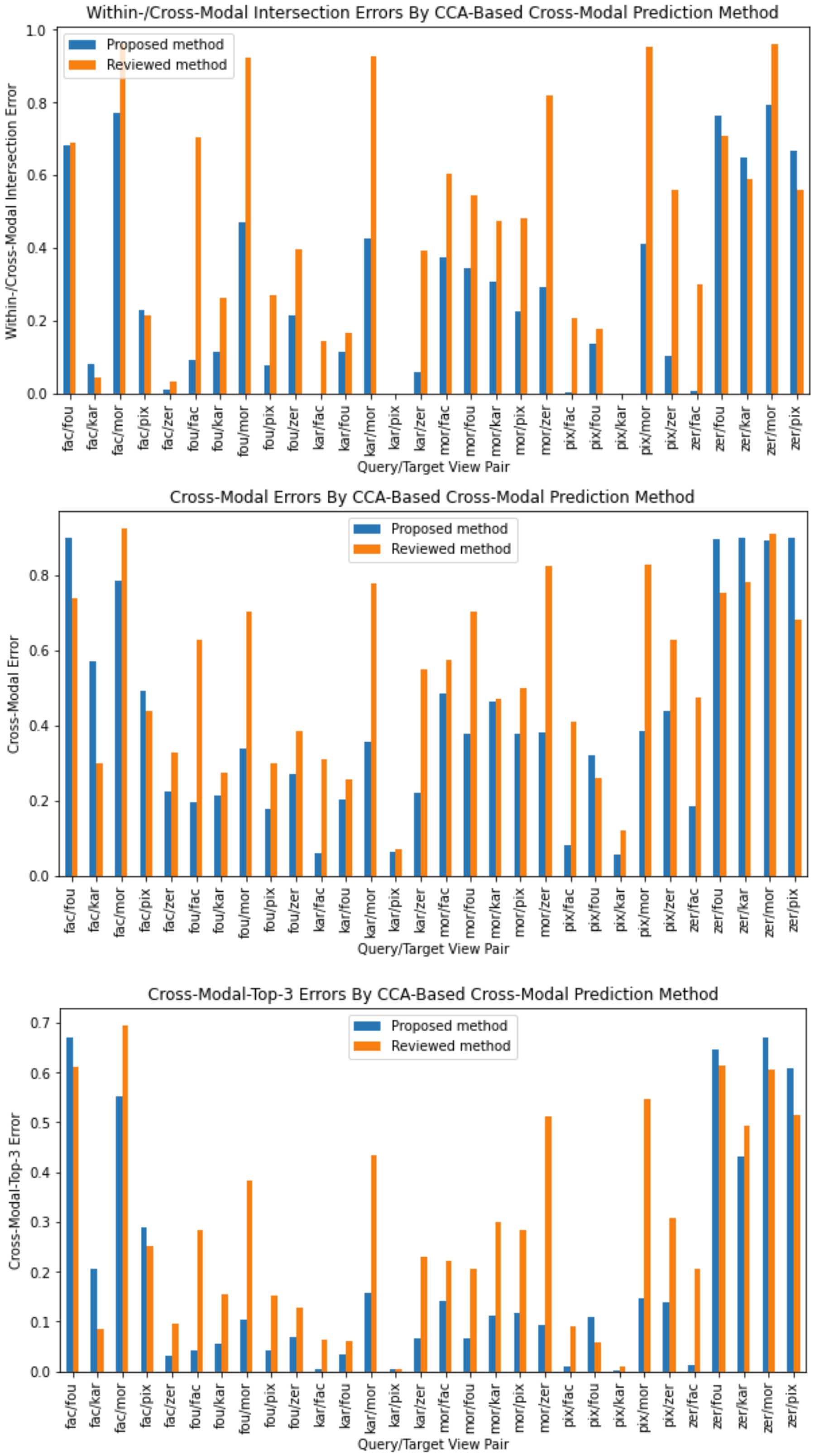

Figure 4.1.1. Test errors for 30 query/target view pairs of the Multi-View Digits dataset using the proposed and pseudoinverse cross-modal prediction methods.

An investigation into the effect of different correlation thresholds (used to choose the number of extracted CCA components) on the test errors was also conducted. This experiment was performed with pixel averages as the query modality and Fourier coefficients as the target modality. These two views were chosen because with a correlation threshold of 0.5, although the proposed cross-modal prediction method produced a smaller within-/cross-modal intersection error than that of the pseudoinverse method, the accuracy of its recommendations was not as good. Experimenting with other correlation thresholds would provide insights into a more optimized threshold value.

Figure 4.1.2 compares the test errors found using the two cross-modal prediction methods with correlation thresholds ranging from 0.1 to 0.9. The proposed method observed a downward trend for all three errors as the thresholds increased from 0.1 to approximately 0.7, at which point the accuracy error of the recommendations started to increase almost exponentially. On the other hand, the test errors produced by the pseudoinverse method gradually increased with higher correlation thresholds and significantly escalated after the correlation threshold exceeded 0.8.

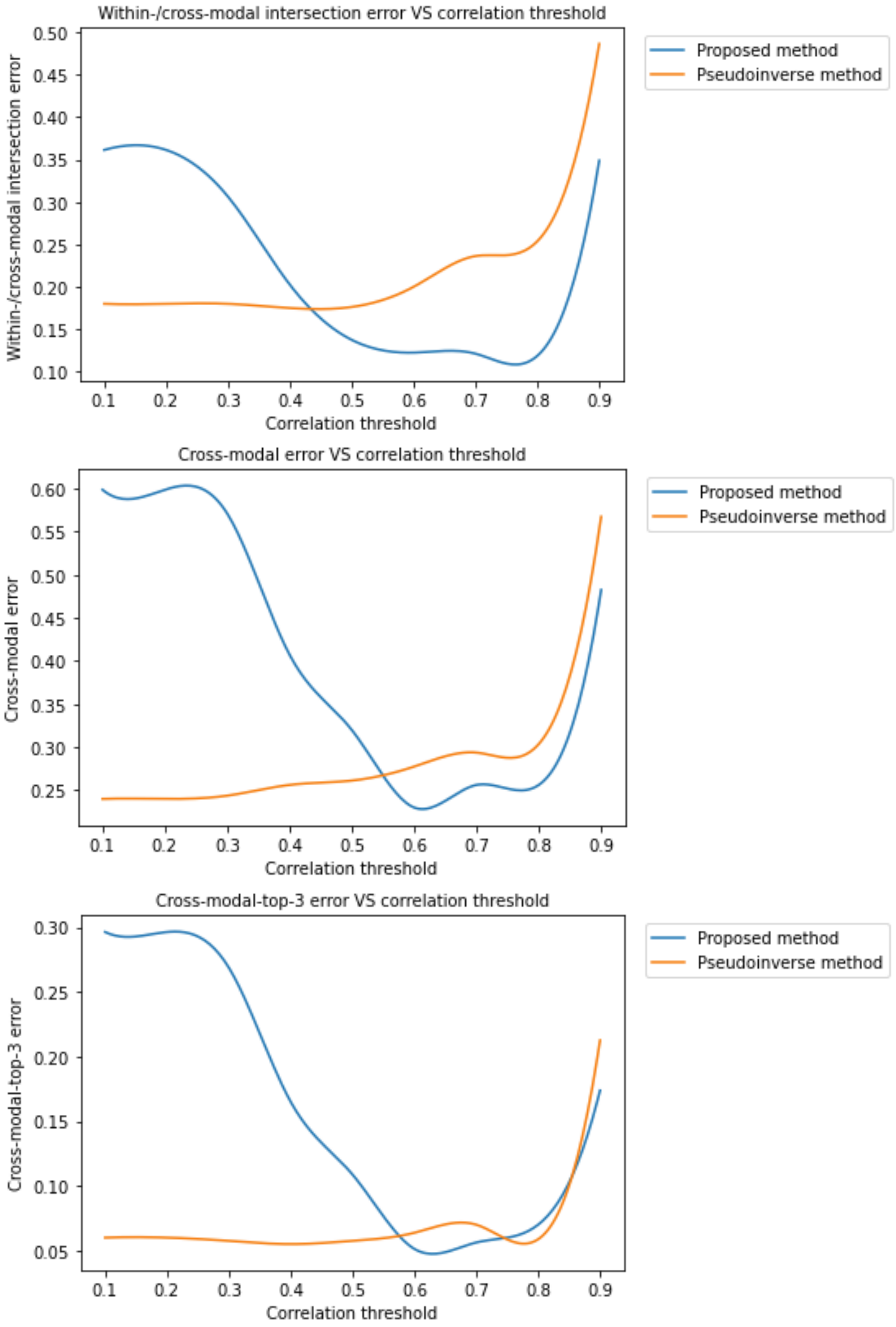

Figure 4.1.2. Test errors produced by using different correlation thresholds when applying the proposed and CCA-pseudoinverse cross-modal prediction methods with pixel averages as the query modality and Fourier coefficients as the target modality.

For thresholds ranging between 0.1 to 0.7, two different error trends observed for the two prediction methods could be explained by their predicting mechanisms. The higher the correlation threshold, fewer but more correlated CCA components were extracted between the modalities. This was more beneficial for the proposed method because these components better represented the correlation between the two modalities, which supported the search for the most fitting representations of the target view within the canonical subspace. Fewer CCA components negatively affected the pseudoinverse method because it was attempting to reconstruct them into their original space. Its results were artificial and also prone to be too distinct from the real data if the available number of CCA components to reconstruct dropped too low. Additionally, the fact that both methods did not perform well when using thresholds exceeding 0.8 could be because CCA was overfitted by the training data.

4.2 CIFAR-100 Dataset

In addition to the Multi-View Digits dataset, CCA-based cross-modal prediction was also applied to the CIFAR-100 datasets. Deep learning feature extraction was performed to obtain one modality, the CIFAR-100-image dataset. The second modality, CIFAR-100-tag, was created based on the class and superclass information from CIFAR-100. This dataset was created based on a probabilistic approach such that for a particular image, the entry corresponding to its true class had the highest probability of getting a value of 1, a fairly high probability for classes in the same superclass, and a low probability for the rest of the classes.

Three pre-trained CNN models – AlexNet, ResNet, and VGG – were used as image feature extractors and compared with respect to their abilities in discovering distinguishable patterns in images, addressed by the corresponding CCA-based cross-modal prediction performance. Both AlexNet and VGG extract 4,096 image features at its last layer and feeds them into the final classifier (softmax) layer, while ResNet extracts 512 features. These features are highly descriptive and are suitable to be transferred to this CIFAR image-domain classification task. For this thesis, they were transferred and used in the CCA-based cross-modal prediction. To help CCA's convergence and to reduce its training runtime, *principal component analysis* (PCA) was applied to the features extracted by all three models to reduce the dimensionality and eliminate collinearities within these features: The number of PCA features was fixed at 500 for comparisons among models. 500 PCA features covered more than 80% of the total variance for all deep learning models.

As opposed to selecting *K* learned components such that the correlation coefficient between the modalities' components was over a given threshold (as done for the Multi-View Digits dataset), the number of CCA components used in the cross-modal prediction process with the CIFAR-100 datasets was chosen to be 20. This helped to cut down the runtime that would have been required to find the minimum rank between the high-dimensional training sets of CIFAR-100-image and CIFAR-100-tag (50,000x500 and 50,000x100, respectively). The highest correlation learned between the modalities' 20 components for the testing data was 0.5279, 0.5355, and 0.5350, respectively for AlexNet, ResNet, and VGG.

The cross-modal prediction aimed at recommending 100 most fitting examples with the deep-learning image features and tags representations taking turns to be the query/target modality. Table 4.2.1 compares the test errors produced by the prediction process using canonical variates and those output by the pseudoinverse method.

Query	Target	CNN	Within-/cross-modal intersection error		Cross-modal classification error		Cross-modal-top-3 classification error	
			(a)	(b)	(a)	(b)	(a)	(b)
Image	Tag	AlexNet	0.1762	0.0186	0.3465	0.3553	0.1664	0.1793
		ResNet	0.1611	0.0094	0.2925	0.3015	0.1336	0.1469
		VGG	0.1483	0.0182	0.2989	0.3039	0.1271	0.1359
Tag	Image	AlexNet	0.5216	0.4751	0.2438	0.5045	0.1106	0.2914
		ResNet	0.449	0.3852	0.2437	0.4459	0.1027	0.244
		VGG	0.4853	0.4447	0.2495	0.4861	0.1040	0.2688

Table 4.2.1. Test errors produced by the (a) proposed and (b) pseudoinverse methods of cross-modal prediction, using tag representations and image features extracted by AlexNet, ResNet, and VGG.

With the query and target views as images and tags, respectively, the cross-modal prediction was analogous to finding the tag representations that might label a given image. With AlexNet features, the within-/cross-modal intersection errors were small. The error of the proposed method was 0.1762, which means that about 82% of the time, there was an intersection between the set of recommended target-view representations and the set that would have been returned by a single-view nearest neighbors search with a target-view query. On the other hand, the within-/cross-modal intersection error of the pseudoinverse method was smaller (0.0186). This could be due to the noise in the tag-view to be predicted. Similarly, for the prediction processes using the query data by ResNet and VGG, higher within-/cross-modal intersection errors were produced by the proposed method. Therefore, the accuracy of the recommendations should be best measured by the percentage of matching classifications (cross-modal classification error and cross-modal-top-3 classification error reported in Table 4.2.1).

Although the proposed method produced a higher within-/cross-modal intersection error regardless of pre-trained CNN used, it was more accurate in the superclasses of the recommended tag representations. With AlexNet image features as the query, the proposed method output a cross-modal classification error of 0.3465, which means that about 65 out of 100 tag representations recommended were correct. This error was comparable to the pseudoinverse method. However, the proposed method was more beneficial in the way that it performed its search in a low dimensional space: It eliminated the additional computation to reconstruct the 500-dimensional image features. Moreover, the search in a low dimensional space was slightly more accurate than the pseudoinverse method. When considering the top-3 classification error, the proposed method had a smaller error. Furthermore, for both the proposed and pseudoinverse methods, cross-modal prediction using image features extracted by ResNet and VGG produced better results than those by AlexNet. With ResNet and VGG, the top-3 tag representations contained the true superclass approximately 87% of the time (the top-3 error of VGG was slightly lower). Overall, when comparing the pre-trained CNNs based on the corresponding CCA-based cross-modal prediction performance, ResNet and VGG were comparable in their ability to extract discriminative image features, and they were better than AlexNet.

If the tag representations were used as the query view, the cross-modal recommendation aims at retrieving the most fitting images associated with the given noisy tag vector. For this cross-modal prediction process, the improvement in accuracy using the proposed method was more apparent. The within-/cross-modal intersection errors of both methods were moderately high regardless of which pre-trained CNN was used. However, the proposed method performed significatly better than the pseudoinverse method, cutting the classification errors down by almost half. Furthermore, the image features extracted by the three pre-trained CNNs were all beneficial to either cross-modal prediction method, with ResNet having the best performance in terms of all three test errors.

To investigate the effect of the number of CCA components on the test errors of the cross-modal prediction, an analysis was performed using the ResNet image features in the query modality and tag representations as the target modality. ResNet model was selected as a representative because it was the most accurate of the deep learning models. Figure 4.2.1 reports the test errors versus the number of CCA components used. The cross-modal classification errors showed that the predictions made by the proposed method were more accurate regardless of the number of components. There was a similar downward trend in the classification errors for both methods, and the proposed method's performance persisted to be better. There was not much improvement to the cross-modal classification errors if the number of CCA components became greater than 20, which was expected because there are 20 superclasses.

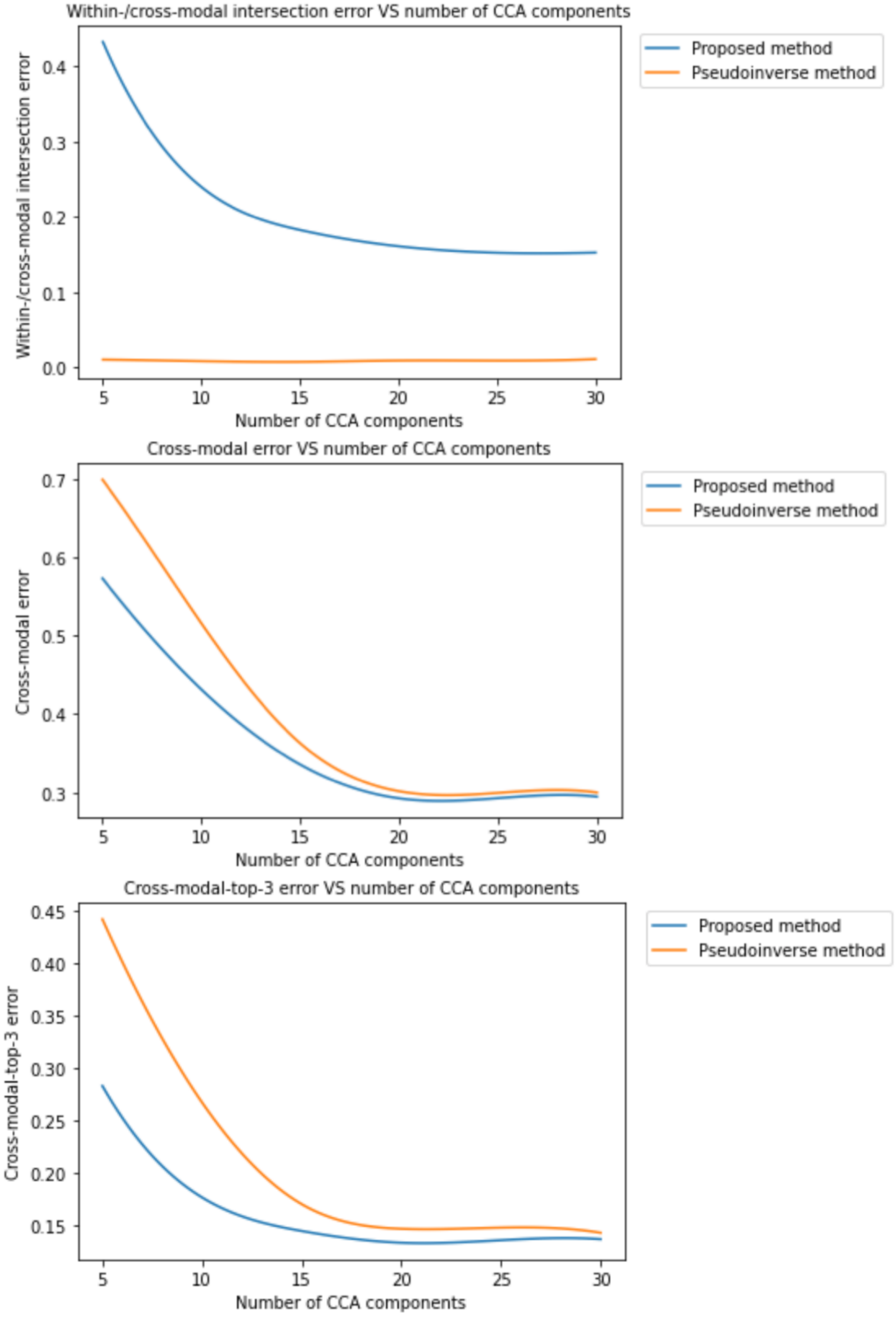

Figure 4.2.1. The number of CCA components versus the test errors on the CIFAR dataset (image features extracted by ResNet in the query modality).

CHAPTER 5: CONCLUSION AND FUTURE WORK

In this thesis, a novel CCA-based cross-modal prediction method was proposed and investigated. For the two-view experimental datasets, first, CCA was used as a feature extractor to find a common (canonical) subspace between the two views of the training examples. One of the views was designated to be the query view and the other one to be the target view. During the test phase, the cross-modal prediction was performed by computing the canonical variates using the given data in the query view and then applying a nearest neighbors search to retrieve the most matching examples in the target view. The thesis also investigated an alternative method referred to as the pseudoinverse method (Bilenko & Gallant, 2016), which reconstructed the representations in the target view from the canonical space and then performed a nearest neighbors search in a much higher dimensional space. Experiments were conducted on two multi-view datasets, Multi-View Digits and CIFAR-100. The experimental results showed that the proposed approach was more accurate than applying the pseudoinverse method.

The thesis also used and compared pre-trained deep learning models for preprocessing the image modality of the CIFAR-100 dataset. The models used included AlexNet, VGG, and ResNet. Among these models, ResNet had extracted the most discriminative image features. Moreover, as deep learning models extracted high-dimensional representations of the images, the proposed nearest neighbor search in the canonical space was shown to be much more effective than the pseudoinverse method. The effectiveness of the canonical space was due to its compactness and its richness in discriminative features, which was also suggested by Kursun et al. (2011).

Additionally, this thesis also proposed a procedure for privacy preservation in the training phase of CCA based on the alternating regression method. Each iteration of alternating regression applies a solution to the Millionaires' problem (Kumar & Gupta, 2013; Lin & Tzeng, 2005) and then updates the canonical weights of the views to maximize their agreement via the backpropagation algorithm. This proposed procedure was designed such that the canonical components of the modalities maximally agree while preserving the privacy of each modality's data from one another. Future work includes implementing and testing the realtimeness of the proposed cross-modal prediction method with and without privacy preservation on larger datasets.

REFERENCES

Alpaydin, E. (2014). *Introduction to Machine Learning* (3rd ed.). Cambridge, MA: MIT Press.

Bartlett, M. S. (1938). Further aspects of the theory of multiple regression. In *Mathematical Proceedings of the Cambridge Philosophical Society* (Vol. 34, No. 1, pp. 33-40). Cambridge University Press.

Becker, S., & Hinton, G. E. (1992). Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, *355*(6356), 161-163.

Benton, A., Khayrallah, H., Gujral, B., Reisinger, D. A., Zhang, S., & Arora, R. (2017). Deep generalized canonical correlation analysis. *arXiv preprint arXiv:1702.02519*.

Bilenko, N. Y., & Gallant, J. L. (2016). Pyrcca: regularized kernel canonical correlation analysis in python and its applications to neuroimaging. *Frontiers in neuroinformatics*, *10*, 49.

Chen, Z., Lu, F., Yuan, X., & Zhong, F. (2017). TCMHG: Topic-based cross-modal hypergraph learning for online service recommendations. *IEEE Access*, *6*, 24856-24865.

Chen, N., Zhu, J., & P Xing, E. (2010). Predictive subspace learning for multi-view data: a large margin approach. In *Proceedings of the Neural Information Processing Systems* (pp. 361–369).

Dua, D., & Graff, C. (2019). *UCI Machine Learning Repository*. Retrieved from http://archive.ics.uci.edu/ml

ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, *31*(4), 469-472.

Favorov, O. V., & Ryder, D. (2004). SINBAD: A neocortical mechanism for discovering environmental variables and regularities hidden in sensory input. *Biological Cybernetics*, *90*(3), 191-202.

Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing* (pp. 169-178).

Girolami, M., & Fyfe, C. (1997). Stochastic ICA contrast maximisation using Oja's nonlinear PCA algorithm. *International journal of neural systems*, *8*(05n06), 661-678.

Gong, Y., Ke, Q., Isard, M., & Lazebnik, S. (2014). A multi-view embedding space for modeling internet images, tags, and their semantics. *International journal of computer vision*, *106*(2), 210-233.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1, No. 2). Cambridge: MIT press.

Guo, C., & Wu, D. (2019). Canonical Correlation Analysis (CCA) Based Multi-View Learning: An Overview. *arXiv preprint arXiv:1907.01693*.

Hardoon, D. R., Szedmak, S., & Shawe-Taylor, J. (2004). Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, *16*(12), 2639-2664.

Hawkins, J., & Blakeslee, S. (2004). *On intelligence*. Macmillan.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Hotelling, H. (1992). Relations between two sets of variates. In *Breakthroughs in statistics* (pp. 162-190). Springer, New York, NY.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. 1. *New phytologist*, *11*(2), 37-50.

Kettenring, J.R. (1971) Canonical analysis of several sets of variables. Biometrika 58:433-451.

Körding, K. P., & König, P. (2000). Learning with two sites of synaptic integration. *Network: Computation in neural systems*, *11*(1), 25-39.

Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*, 1097-1105.

Kumar, A., & Gupta, A. (2013). Some efficient solutions to yao's millionaire problem. *arXiv preprint arXiv:1310.8063*.

Kursun, O., Alpaydin, E., & Favorov, O. V. (2011). Canonical correlation analysis using within-class coupling. *Pattern Recognition Letters*, *32*(2), 134-144.

Kursun, O., Dinc, S., & Favorov, O. V. (2021). Contextually Guided Convolutional Neural Networks for Learning Most Transferable Representations. *arXiv preprint arXiv:2103.01566.*

Kursun, O., & Favorov, O. V. (2019). Suitability of features of deep convolutional neural networks for modeling somatosensory information processing. In *Pattern Recognition and Tracking XXX* (Vol. 10995, p. 109950G). International Society for Optics and Photonics.

Lai, P. L., & Fyfe, C. (1998, April). Canonical correlation analysis using artificial neural networks. In *ESANN* (pp. 363-368).

Lin, H. Y., & Tzeng, W. G. (2005). An efficient solution to the millionaires' problem based on homomorphic encryption. In *International Conference on Applied Cryptography and Network Security* (pp. 456-466). Springer, Berlin, Heidelberg.

Luo, Y., Tao, D., Ramamohanarao, K., Xu, C., & Wen, Y. (2015). Tensor canonical correlation analysis for multi-view dimension reduction. *IEEE transactions on Knowledge and Data Engineering*, *27*(11), 3111-3124.

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In *International conference on the theory and applications of cryptographic techniques* (pp. 223-238). Springer, Berlin, Heidelberg.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*.

Phillips, W. A., & Singer, W. (1997). In search of common foundations for cortical computation. *Behavioral and brain Sciences*, *20*(4), 657-683.

Sakar, C. O., & Kursun, O. (2017). Discriminative feature extraction by a neural implementation of canonical correlation analysis. *IEEE Transactions on neural networks and learning systems*, *28*(1), 164-176.

Sakar, C. O., Kursun, O., & Gurgen, F. (2014). Ensemble canonical correlation analysis. *Applied intelligence*, *40*(2), 291-304.

Shin, H. C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., & Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, *35*(5), 1285-1298.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sun, L., Ji, S., & Ye, J. (2008). A least squares formulation for canonical correlation analysis. In *Proceedings of the 25th international conference on Machine learning* (pp. 1024-1031).

Wold, H. (1966). Nonlinear Estimation by Iterative Least Squares Procedures in: David, FN (Hrsg.), Festschrift for J. *Neyman: Research Papers in Statistics, London*.

Yao, A. C. (1982). Protocols for secure computations. In *23rd annual symposium on foundations of computer science (sfcs 1982)* (pp. 160-164). IEEE.

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. *arXiv preprint arXiv:1411.1792*.

Yuan, Y. H., & Sun, Q. S. (2013). Multiset canonical correlations using globality preserving projections with applications to feature extraction and recognition. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(6), 1131-1146.

Zhou, Y., Mishra, S., Verma, M., Bhamidipati, N., & Wang, W. (2020, April). Recommending themes for ad creative design via visual-linguistic representations. In *Proceedings of The Web Conference 2020* (pp. 2521-2527).