DECODING COGNITIVE STATES USING THE BAG OF WORDS MODEL ON FMRI TIME SERIES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜNEŞ SUCU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

SEPTEMBER 2017

Approval of the thesis:

DECODING COGNITIVE STATES USING THE BAG OF WORDS MODEL ON FMRI TIME SERIES

submitted by **GÜNEŞ SUCU** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering** ________________

Assist. Prof. Dr. Emre Akbaş
Supervisor, **Computer Engineering Department, METU** ________________

Examining Committee Members:

Prof. Dr. Fatoş Yarman Vural
Computer Engineering Department, METU ________________

Assist. Prof. Dr. Emre Akbaş
Computer Engineering Department, METU ________________

Assoc. Prof. Dr. Pınar Karagöz
Computer Engineering Department, METU ________________

Assoc. Prof. Dr. Sinan Kalkan
Computer Engineering Department, METU ________________

Assist. Prof. Dr. Tolga Çukur
Electrical and Electronics Engineering Dept., Bilkent University ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name:	GÜNEŞ SUCU
Signature :	

ABSTRACT

DECODING COGNITIVE STATES USING THE BAG OF WORDS MODEL ON FMRI TIME SERIES

Sucu, Güneş
M.S., Department of Computer Engineering
Supervisor : Assist. Prof. Dr. Emre Akbaş

September 2017, 64 pages

Bag-of-words (BoW) modeling has yielded successful results in document and image classification tasks. In this study, we explore the use of BoW for cognitive state classification. We estimate a set of common patterns embedded in the Functional Magnetic Resonance Imaging (fMRI) time series recorded in three dimensional voxel coordinates by clustering the Blood Oxygen Level Dependent (BOLD) responses. We use these common patterns, called the code-words, to encode activities of both individual voxels and group of voxels, and obtain BoW representations on which we train linear classifiers. We experimented with a number of different BoW representations such as encoding spatial and functional neighbors, spatial pooling, and soft and hard encoding. Our experimental results show that, on a multiclass fMRI dataset, the hard BoW encoding, when applied to individual voxels, significantly improves the classification accuracy (an average 7.22% increase when applied on average intensity per voxel and an average 15.52% increase when applied to raw intensity time series per voxel) compared to a classical multi voxel pattern analysis (MVPA) method. This preliminary result gives us a clue to generate a code-book for fMRI data which may be used to represent a variety of cognitive states to study the human brain.

Keywords: fMRI, Bag of Words, Brain Decoding, Code-word

ÖZ

FMRG ZAMAN SERİLERİ ÜZERİNDE KELİME TORBASI MODELİ KULLANILARAK BİLİŞSEL DURUMLARIN KODUNUN ÇÖZÜLMESİ

Sucu, Güneş
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Yrd. Doç. Dr. Emre Akbaş

Eylül 2017, 64 sayfa

Kelime torbası modellemesi döküman ve imge sınıflandırması çalışmalarında başarılı sonuçlar ortaya koymuştur. Bu çalışmada, algısal durumların sınıflandırması için Kelime Torbası yönteminin kullanımı incelenmektedir. Üç boyutlu voksel koordinatlarında kaydedilen Fonksiyonel Manyetik Rezonans Görüntüleme (fMRG) zaman serisinde gömülü yaygın bir dizi küme, Kan Oksijen Seviyesine Bağımlı (KOSB) yanıtlar kümelenerek hesaplanmaktadır. Hem bireysel voksellerin hem de voksel gruplarının faaliyetlerini kodlamak için kod kelimeleri adı verilen bu yaygın kalıplar kullanılmaktadır ve üzerinde doğrusal sınıflandırıcılar eğittiğimiz Kelime Torbası gösterimleri elde edilmektedir. Çalışmamızda, uzaysal ve fonsiyonel komşuları kodlama, konumsal havuzlama ve hafif ve ağır kodlama gibi farklı Kelime Torbası gösterimleri denenmiştir. Deneysel sonuçlarımız, bireysel voksellere uygulandığında ağır Kelime Torbası kodlamasının klasik bir çoklu voksel desen analizi (ÇVDA) yöntemine göre çok sınıflı bir veri setinde sınıflandırma doğruluğunu önemli ölçüde geliştirdiğini göstermektedir (voksel başına ortalama yoğunlukta uygulandığında ortalama $\%\mathbf{7.22}$ artış ve voksel başına ham zaman serisi yoğunluğuna uygulandığında ortalama $\%\mathbf{15.52}$ artış). Bu ön sonuç, fMRG verisiyle insan beynini incelemek için çeşitli algısal durumları gösterecek bir kod kitabı üretmek adına bir ipucu vermektedir.

Anahtar Kelimeler: fMRG, Kelime Torbası, Beyin Şifresini Çözme, Kod Kelimesi

To all my beloved ones

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor Assist. Prof. Dr. Emre Akbaş. He has always encouraged me to complete this study and given me unfailing support. His advices have shaped my future. He was the architect of this work and source of inspiration for me with his extensive knowledge, creative thinking and vision.

I also owe special thanks to Prof. Dr. Fatoş Tünay Yarman Vural. She has always been like a mother to me and guided me to reach success in every aspect of my life. She has also provided rich suggestions to construct this work.

I wish to thank my colleagues Itır Önal Ertuğrul, Arman Afrasiyabi, Sarper Alkan, Barış Nasır, Burak Velioğlu, Ozan Yıldız, Emre Aksan, Hazal Moğultay, Baran Barış Kıvılcım, Batuhan Karagöz, Abdullah Alchihabi and Muhammed Kocabaş. They have always supported and encouraged me to finish this study. They have also helped me to solve the technical issues I encountered during this stressful thesis period.

I also wish to thank my flatmate Okan Harput for his never-ending patience. He has been there in all the ups and downs that I have faced through this study. But mostly, I am thankful to him for accepting me as a brother. It means a lot to me.

I would like to extend my deepest gratitude to my family, without whom I couldn't accomplish this far. I am grateful to my dearest mom, Nedret and my dearest dad, Sadi for their abiding love, never-ending encouragement and confidence. I also wish to thank my sister Gizem since I have always felt her support and motivation with me.

Last but surely not the least, my heartfelt appreciation goes to my dearly beloved Ceren for her presence beside me each and every single day of this thesis period. Without her endless encouragement, unwavering motivation, joyful and helpful accompany and warm love, this thesis would not have been completed.

I gratefully acknowledge that this study has been supported by TÜBİTAK (The Scientific and Technological Research Council of Turkey) project with grant number 112E315.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

BoW	Bag of Words
MVPA	Multi-voxel Pattern Analysis
MRI	Magnetic Resonance Imaging
fMRI	functional Magnetic Resonance Imaging
SPM	Statistical Parametric Mapping
HRF	Hemodynamic Response Function
BOLD	Blood Oxygen Level Dependent
SVM	Support Vector Machine
MLP	Multi-layer Perceptron
CNN	Convolutional Neural Network
ASD	Autism Spectrum Disorder
ADHD	Attention Deficit Hyperactivity Disorder
ReLU	Rectified Linear Unit

CHAPTER 1

INTRODUCTION

How does the human brain function? It is one of the most important questions the mankind has been trying to answer throughout the history of humanity. As this mysterious organ is the most complicated structure in the human body, a complete answer to this question has been elusive for scientists.

Thanks to the technological advances, with the use of functional Magnetic Resonance Imaging (fMRI), activations in the brain can be pictured in the decade we are in. fMRI is a non-invasive method and it is widely used in the experiments conducted for analyzing how human brain works. Now, fMRI is the dominant technique used to understand how human brain functions.

fMRI images reveal the brain functioning according to the changes in blood oxygen level (details can be found in Section 2.1). The changes in blood oxygen level create a response named Hemodynamic Response (HR). This response is the result of the neuronal activation and characterized by a Hemodynamic Response Function (HRF) [8] (Figure 2.2) which is represented by a time series. This time series contains intensity values on which models are built. However, these intensity values cannot be obtained at the level of a neuron. fMRI images are constructed in units of voxels. A voxel corresponds to a three-dimensional cubic space in a specific location of the brain tissue. A voxel in an fMRI image is analogous to a pixel in a conventional RGB image. Depending on the resolution used, they might contain thousands or millions of neurons.

1.1 The Problem: Cognitive State Classification

The problem that we address in this thesis is known as cognitive state classification or decoding. In this problem, the goal is to predict the cognitive state of a human at a given moment by analyzing the (fMRI) image of her/his brain around that moment.

Although the brain has a diverse variety of patterns for different cognitive states, it also shares some structural and functional properties in common. One major challenge is to decipher these common patterns from the functional magnetic resonance imaging (fMRI) data. The patterns can then be used to represent different cognitive tasks. The main goal of this study is to develop a model that uses a set of code-words which can be used to represent and classify the cognitive states.

The smallest unit of fMRI data is called a “voxel”. In brain decoding experiments using fMRI, during the memory encoding and retrieval process voxel intensity values are recorded. During the experiments, samples from different classes are viewed by the subjects. Here, the purpose is to classify these samples using the recorded voxel intensity values. So that, first features are extracted from the fMRI data using a training and test set. Then, these features are used to train and test a selected classifier.

A popular group of methods to recognize the brain activity patterns from BOLD responses is the multi-voxel pattern analysis (MVPA) methods [31, 35], where a cognitive state is represented by a feature vector extracted from the voxel intensity values measured in fMRI data [52]. In MVPA approaches, the full spatial pattern of the brain is taken into consideration so that they try to find how (rather than where) the information is encoded. Patterns are considered as the vectors of voxel intensity values. MVPA methods employ voxel intensity values as features to a classification algorithm. The representational power of the extracted features is measured by the performance of a classifier, which is trained by a set of labeled features. There are a variety of methods available in the literature to extract features from the fMRI recordings. Among them, a simple method is to take the average value of the BOLD responses measured at each voxel during a cognitive stimulus [12, 22], or to define each brain volume as a sample [34, 36].

1.2 Proposed Method

The proposed methods in this thesis aim to contribute to the innovations in decoding cognitive states of human brain using fMRI. We draw inspiration from numerous successful applications [14, 17, 27, 48] of the bag-of-words model in the computer vision literature.

There are three main steps in using the bag-of-words model for a classification problem:

1. Constructing a dictionary of code-words in an unsupervised manner,

2. Encoding the images using the dictionary and optionally applying spatial pooling,

3. Training a classifier on the encoding vectors.

Each of these steps involves many choices. For example, in the first step, one should decide what kind of pre-processing to apply to voxel intensities and which metric to use while forming the dictionary. Similarly, in the second step, one can use either a hard encoding or a soft encoding method. We elaborate on all of these choices in Chapter 3.

We propose to use the bag-of-words model to generate a code-book and to decode the cognitive states by using fMRI data. In our method, we create code-books of various sizes from the raw fMRI data and then match every time series in the raw data to a code-word from the code-book. By doing so we represent each voxel, or a group of voxels by a histogram of k bins where k is the size of the code-book. As the ultimate goal, we aim to create a *meaningful* brain alphabet where the code-words have functional or anatomical correspondences. This work is a small step towards our goal.

Applying the bag-of-words model to an fMRI dataset is not a straightforward practice. One should make decisions over many questions such as:

- Should we pre-process voxel intensities or just use them in raw form?
- How should we pre-process voxel intensities?
- What metric should we use while forming the code-words?
- Should we use hard-encoding or soft-encoding?
- Should we do voxel selection first?
- Should we use spatial pooling or just apply the model to individual voxels?
- Should we include voxel neighborhoods in the model? Should we use functional or spatial neighborhood?

We aimed to do an extensive experimental study where we employed a wide range of BoW models on fMRI data. We applied it on average of time series and raw time series and obtained a significant classification accuracy improvement compared to the baseline MVPA models. We also applied the BoW model on zero-mean time series (trends) and average of time series combined with zero-mean time series. Additionally, we tried to utilize spatial pooling, voxel selection and neighborhood information and analyzed their results. Lastly, in order to further validate our model we applied it on a different dataset. Details of our experimental work can be found in Chapter 4.

1.3 Novelties and Contributions

To the best of our knowledge, there are only two studies [16, 49] where the bag-of-words model was applied in the analysis of fMRI data. We differ from these two works in the following ways.

- In previous work, the focus was not on encoding the voxels individually. Solmaz et al. [49] encoded the whole brain as a single entity. This is the extreme point of spatial pooling (more on this in Chapter 3). In another study, Ertugrul et al. [16] encoded the arc-weights of the meshes constructed on anatomical regions level.

- Another main focus, which is not presented in previous work, is that the current work analyzes both functional and spatial neighborhood among the voxels. In [16], the neighborhood information was only used on anatomical regions level.

- In this work, we employed a wide variety of BoW techniques to decode the cognitive states. Such a comprehensive application of BoW (details of which are explained in Chapter 4) does not exist in previous work.

- The major conclusion of this work is that, when applied in an appropriate way the bag-of-words model yields a par or better performance compared to the baseline MVPA models presented in the literature. Therefore, the proposed methods appear to be a promising alternative to the classical MVPA methods in the classification of various cognitive states.

Parts of the work presented in this thesis appeared in the following publication:

- G. Sucu, E. Akbas, I. Oztekin, E. Mizrak, F. T. Yarman Vural, "Decoding cognitive states using the bag of words model on fMRI time series." 1st International Workshop on Machine Learning for Understanding the Brain (MLUB), Signal Processing and Communication Application Conference (SIU), 2016 24th. IEEE, 2016

1.4 Outline of the Thesis

In Chapter 2, a brief summary of fMRI is presented. In addition, a general overview of MVPA techniques used to classify cognitive states of brain is provided. Finally, the BoW model, on which this thesis is built, is introduced.

Chapter 3 introduces the proposed methods to decode cognitive states. Here, the motivation behind this study is discussed first. Then the BoW encoding models improving the classification performance are explained in detail.

Chapter 4 presents the analysis of the proposed methods in Chapter 3. Here, the results of the experiments and the classification performance comparisons of the proposed methods and the MVPA techniques are provided.

In Chapter 5, the overall outcomes of this study are discussed and the possible directions of this work are pointed out.

CHAPTER 2

AN OVERVIEW OF FUNCTIONAL MAGNETIC RESONANCE IMAGING (FMRI), MULTI-VOXEL PATTERN ANALYSIS (MVPA) AND BAG OF WORDS (BOW)

In this chapter, in order to provide information to the reader, one of the currently popular neuroimaging techniques, fMRI, is presented. Also current multi voxel pattern analysis methods (MVPA) are overviewed to show the reader how cognitive states are decoded. Finally, the backbone of this study, the bag of words (BoW) model, is introduced to the reader and details of how this model is used in the literature are explained.

2.1 Functional Magnetic Resonance Imaging (fMRI)

As the brain images help to understand the complicated structures of the brain, visualization of brain functioning is one of the breakthroughs of this century. With the help of the innovations in neuroimaging, functional neuroimaging methods have become popular in detecting activation of brain regions while the subject performs a cognitive task. During the cognitive task to which the subject is exposed, the neuroimaging technique helps reveal the active parts of the brain. Therefore, the functional methods in neuroimaging are now the most powerful tools to understand how brain works.

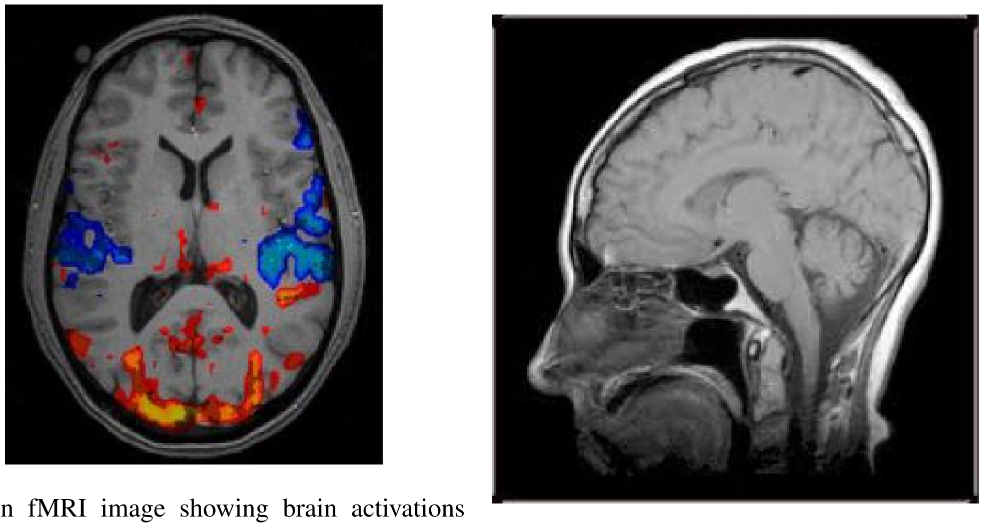
One of the most popular neuroimaging techniques used in visualizing the brain activations is Functional Magnetic Resonance Imaging (fMRI). As the name implies, the scanner has three main components, "magnetic", "resonance" and "imaging" [20], respectively.

- **Magnetic:** The scanner creates a static magnetic field while aligning nuclei of atoms in the human body. This gives the name of the first component. Nuclei of hydrogen atoms(protons) are used by the MRI machine as the body contains a lot of water. With the help of a powerful electro-magnet, MRI scanner creates a static magnetic field as the first step. This field is about 3 - 4 teslas(T) and 50000 times greater than the field the earth creates. Therefore, it has the capacity to affect the magnetic nuclei of hydrogen. Without this huge magnetic field, the protons would head to random directions. With this strong magnetic field, they are aligned in the same direction in the scanner. This is called equilibrium state.

- **Resonance:** After the alignment of the nuclei, the radiofrequency coils of MRI scanner emit electromagnetic waves that resonate at a particular frequency to perturb the equilibrium by disturbing the nuclei [20]. In this phase named "resonance", the atoms are excited and absorb the energy the radiofrequency pulse emits. After that, the hydrogen atoms return to the equilibrium point and release the energy as the radiofrequency pulse is stopped. Although there is a continuous static field in the MRI scanner, the radiofrequency fields are created for short periods and then disappear. The radiofrequency coils detect the released energy which is defined as MR signals. But it cannot be directly employed for imaging because it does not contain any spatial information.

- **Imaging:** In this phase, MR signals are converted into the brain images. Most of the current scanners use the technique mentioned in the work done by Lauterbur et. al. [26]. In order to generate 2D and 3D MRI images, three orthogonal gradients are used in this work. Nuclei of atoms at different locations shake in different speeds as the gradient coils create additional magnetic fields. The spatial information is extracted using Fourier analysis.

Functional Magnetic Resonance Imaging (fMRI) is a technique which measures the changes in brain functioning. The difference between fMRI and MRI is that fMRI



(a) An fMRI image showing brain activations (Source: [1]).

(b) An MRI image of head (Source: [2]).

Figure 2.1: fMRI vs MRI

visualizes the activity of the brain (Figure 2.1a) while MRI shows only the anatomical structure (Figure 2.1b).

Therefore, images of fMRI scans show the activity in the anatomical structure of the brain [3] whereas images of MRI scans show the anatomical structure of the brain. However, the neuronal activity is not directly imaged by fMRI. It visualizes physiological changes correlated with neuronal activity of the brain [20].

In the study conducted by Pauling and Coryell [43], it was found that magnetic behaviour of hemoglobin differs depending on whether it is bound to oxygen or not. The magnetic moment of oxygenated hemoglobin is zero as it does not have any unpaired electrons(diamagnetic). On the contrary, deoxygenated hemoglobin (dHb) has unpaired electrons so it is paramagnetic and has strong magnetic moment. Since arterials having oxygenated blood does not distort the magnetic field around the tissue, higher level of MR signal intensity is obtained where blood is oxygenated [51], [39]. The active brain regions can be found with MRI by detecting the changes in blood oxygenation [38], [40]. Measuring these changes also determines the blood-oxygenation-level-dependent (BOLD) contrast.

An active neuron needs some energy to return to its original state. This energy comes

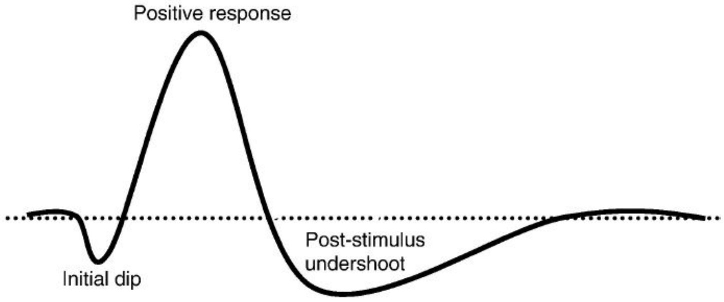


Figure 2.2: Hemodynamic Response Function (HRF), from [19].

with blood flow to the brain. With this blood flow, glucose and oxygen are transported to the active area. Therefore, neurons can return to their original state. Oxygen bounds to deoxygenated hemoglobin. So with the decrease in dHB, MR signal increases in the activated regions. As a result, fMRI based on BOLD contrast gives information about the activity in the brain by measuring the oxygenation level of blood. Some of the early studies using BOLD based fMRI can be found in the literature [7], [25] [41]. fMRI based on BOLD is still the most powerful tool for visualizing the brain activity.

Hemodynamic Response (HR) is the increase in MR signal and it is the result of the neuronal activation. This response is characterized by a Hemodynamic Response Function (HRF) [8] (Figure 2.2). As it is seen in Figure 2.2, there is an initial dip following the start of a neuronal activity. Cause of this can be the initial oxygen extraction. The peak value is the maximum value in BOLD HR signal and achieved nearly 4 to 6s after the stimulus. The undershoot point comes after the activity stops in the neuron and the BOLD HR signal decreases below the original level. In time, the signal again recovers itself to the original level. In our fMRI experiments, the average of time-series and the whole time-series of the given stimulus are used.

During each fMRI scan, 2D slices of brain are formed with the recorded BOLD signal measurements. After that, these slices are accumulated and 3D brain images are formed. As these images are three dimensional, they are partitioned into voxels (volumetric pixels) [8]. Voxels are the smallest spatial units of fMRI data, and depending on the resolution used, they might contain thousands of neurons.

2.2 Multi-voxel Pattern Analysis (MVPA)

The non-invasive nature of the functional magnetic resonance imaging (fMRI) has increased the power of researchers to analyze the human brain. At the first studies conducted using fMRI, researchers concentrated on detecting the active brain regions under a specific cognitive task [30]. In the experiments, subjects are exposed to a stimuli of a particular cognitive task. For each trial fMRI measurements are recorded. After that, by averaging the fMRI recordings, the active parts of the brain can be detected. In this approach the focus is on the individual voxels so that it is called univariate approach.

On the other hand, recent studies focus on the full spatial pattern of the brain instead of the univariate approach and try to recognize this full pattern [37]. This approach is called multi-voxel pattern analysis (MVPA). MVPA approaches seek to find the answer of *how* the information is encoded instead of trying to find *where* the information is encoded. As a result, MVPA methods enable the identification of non-local relationships between the brain and the specific cognitive task [45]. In the MVPA based fMRI studies, patterns are considered as the vectors of voxel intensity values.

MVPA methods have four main steps [37]:

- **Feature Selection:** In this phase, the noisy voxels, which may reduce the classification performance, are eliminated with the help of the feature selection algorithms. Therefore, the voxels carrying information are kept.

- **Pattern assembly:** In this phase, brain patterns are extracted from the data. A label is given to each pattern according to the experimental condition generated by the pattern.

- **Classifier training:** In this phase, the training set is constructed by selecting a subset of samples that belongs to different cognitive tasks. A classifier is trained with this training set. Therefore, a function mapping between the experimental condition and the brain pattern is obtained.

- **Generalization:** Here, the model generated in the previous phase is used to predict the labels of the test samples [53]. In this way the model is tested with the test samples. The generalization performance of the classifier is measured with the accuracy of the classifier.

There are some advantages of MVPA methods compared to the univariate methods. First of all, there may be knowledge loss in the univariate methods since the voxels with insignificant response to a cognitive task are eliminated. But these voxels can give information about the cognitive task in terms of presence and absence of it [37]. On the other hand, in MVPA methods the weak information can be collected from different spatial points in an effective way. Secondly, spatial patterns that may carry significant information can disappear as a result of the spatial smoothing used in the most of the univariate methods. Lastly, MVPA methods take the combination of different brain regions into account as it can be informative about the cognitive task although they are not so individually. Univariate methods do not consider this information.

In the last decade, several studies using MVPA approaches to decode cognitive states have been carried out. Haxby et al. [18] showed that multi-voxel patterns drawn from brain activity in ventral temporal cortex can be used to distinguish different cognitive states. In this study, objects from different categories (faces, cats, non-sense objects and man-made objects) were viewed by the subjects. This study showed that faces and objects representations have an overlap and they are distributed in ventral temporal cortex. However, with the use of MVPA methods, they could not find the different patterns for these categories coming from distinct responses in different regions. Additionally, Kamitani et al. [23] used MVPA methods in the orientation detection problem. Davatzikos et al. [15] also used MVPA approaches to recognize the discriminative patterns of brain activities with fMRI data recorded during a truth-telling or lying experiment. In the study conducted by Mitchell et al. [32], MVPA methods were used to decode the cognitive states in the experiment in which the subjects are presented a sentence or a picture. There are also subclasses (an ambiguous or non-ambiguous sentence and twelve categories of pictures) in the experiment.

MVPA approaches were also used for memory retrieval tasks. Polyn et al. [44] used these methods to detect the similarity of the patterns of brain activity in the encoding and the retrieval phases. In this experiment, pictures from three different classes were viewed by the subjects in the encoding phase. In the retrieval phase, they were expected to remember these pictures.

In some of the studies, MVPA approaches were used for illness diagnosis. In the work conducted by Craddock et al. [13], resting state functional connectivity patterns obtained from healthy participants and patients with major depression were provided as features to Support Vector Machines (SVM). Therefore, they distinguished the activations of healthy participants and patients using MVPA techniques. In addition, Shen et al. [47] discriminated between schizophrenic patients and healthy participants using the resting state functional connectivity patterns in machine learning tools. Moreover, MVPA methods were used for discriminating the brain activations of healthy people and patients with Autism Spectrum Disorder (ASD) [11].

In all of the studies mentioned above, brain activation patterns were recorded as fMRI data and used for decoding cognitive states with the help of MVPA methods. In these studies, the purpose is to discriminate between the brain activity patterns of different cognitive tasks.

fMRI has the power of measuring and imaging the individual voxel intensities and classical MVPA methods use these intensity values as features to the classifiers. However, this is not adequate to fully decode the complex structure of the brain. As a result, more complicated models are necessary to analyze the brain functioning. With this motivation we employed the bag-of-words model to decode the cognitive states. In the next section, we give a summary of this approach with the examples from the literature.

2.3 Bag of Words (BoW)

The BoW model was first used in the context of text categorization [21, 50]. Later it has been adopted to computer vision problems such as image classification [14] and image retrieval [48]. Csurka et al. [14] extracted affine invariant descriptors of image patches and then used vector quantization methods to form the visual words from these descriptors. In another work, Sivic and Zisserman [48] used a similar approach to form their visual words and applied it to object and scene retrieval in videos. After its introduction to computer vision community, the BoW model has rapidly become a preferred baseline model in object and image classification tasks, and several variations [17, 27] have been proposed.

Despite its widespread use in the computer vision community, the BoW model has not been frequently used in neuroimage related machine learning problems. To the best of our knowledge, the work done by Solmaz et al. [49] is one of the studies that employed the BoW model on fMRI data. They proposed a BoW method to classify Attention Deficit Hyperactivity Disorder (ADHD) conditioned subjects and control subjects using fMRI data of resting state brains. In their method, they have constructed a network of voxels based on the correlation between any two voxels and used a BoW model to capture the features of the network. Practically, they represented a whole fMRI image by a histogram of words where each word represented a certain level of activity in a network of voxels. They also applied the same model to raw intensity time-series of the voxels and represented each image by a histogram. Finally, they concatenated these two histograms and trained a Support Vector Machine (SVM) for classification. Our method differs from the work of Solmaz et al. [49] in two ways. First, we do not use the BoW model to encode the whole brain as a single entity, instead we apply the BoW model to individual voxels and to anatomical regions. And second, we pre-process (e.g. demeaning and averaging) the BOLD intensities before feeding them to the BoW model. In another work done by Ertugrul et al. [16] the BoW model has been used on fMRI data. In their method, they have constructed networks(meshes) of anatomical regions. After that, they have calculated the arc-weights of these meshes. Finally, they have encoded these arc-weights with the BoW techniques to classify the fMRI images. Our method differs from the work of Ertugrul et al. [16] in two ways. First, we do not use the BoW model to encode the whole brain in anatomical regions level, instead we apply the BoW model to individual voxels. And second, they encode the arc-weights of the meshes constructed of anatomical regions, but we encode directly the voxel intensities.

CHAPTER 3

THE BAG-OF-WORDS MODEL

In this chapter, we describe the bag-of-words model which is the representation model that we use to encode fMRI images. Originally used in the context of information retrieval and text document classification [21, 50], it is a simplifying representation model which emphasizes the frequencies of discrete words and ignores their spatial location. In the following, we first give a motivational account of how a simple bag-of-words encoding could solve the non-linear XOR problem (which cannot be directly solved by linear classifiers) and then describe how we used it to encode fMRI images.

3.1 Motivation

Consider the logical exclusive-OR (XOR) problem which is a good example for linearly inseparable patterns. In this problem the classes cannot be separated with a single line.

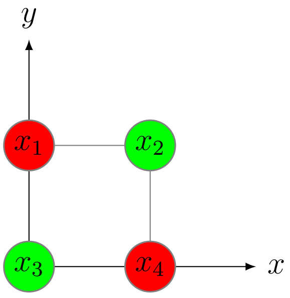


Figure 3.1: The exclusive-OR (XOR) problem.

In Figure 3.1 the red data points x_1 and x_4 represent the negative class and the green data points x_2 and x_3 represent the positive class. A linear classifier cannot produce a perfect classification for this problem. However, if we utilize a bag-of-words encoding, then the resulting representation would be perfectly solvable by linear classifiers.

To see this, let us first cluster these data points using k-means with $k = 4$. x_1, x_2, x_3, and x_4 will become the means of the clusters c_1, c_2, c_3, and c_4, respectively. Now, let us represent each datapoint with a vector of length k where we set the i^{th} entry of this vector to 1 if the datapoint belongs to the i^{th} cluster (otherwise, the entry is set to 0). This is known as the hard-encoding method which we describe in Section 3.3.2.1. Here are the resulting representation vectors:

- $r_1 = [1\ 0\ 0\ 0]^T$ for x_1,
- $r_2 = [0\ 1\ 0\ 0]^T$ for x_2,
- $r_3 = [0\ 0\ 1\ 0]^T$ for x_3,
- $r_4 = [0\ 0\ 0\ 1]^T$ for x_4.

Let us now consider the linear decision boundary given by $w = [0\ 1\ 1\ 0]^T$. Considering that $w^T r_1 = 0$, $w^T r_4 = 0$ and $w^T r_2 = 1$, $w^T r_3 = 1$ these points are now perfectly separable by a linear decision boundary (As can be seen the data points belonging to the negative class have the value 0 and the data points belonging to the positive class have the value 1). This is a simple but convincing example where using k-means and hard-encoding could help us solve non-linear classification problems. This is the motivation behind our decision to use the bag-of-words model to tackle the fMRI classification problem.

3.2 Dataset representation and notation

An fMRI dataset D consists of N images and their class labels,

Table 3.1: The notations used in the thesis.

Symbol	Explanation
D	Dataset of fMRI images.
$\mathbf{x}_i$	i^{th} image.
y_i	Class label of the i^{th} image, a scalar.
$\mathbf{v}_{ij}$	j^{th} voxel of the i^{th} image.
M	Number of voxels in an image.
$p(\cdot)$	Pre-processing function applied on a voxel.
$\mathbf{f}_i$	Feature vector representing the i^{th} image.
$\mathbf{c}$	A code-word in a dictionary. $\mathbf{c}_\ell$ is the ℓ^{th} code-word in the dictionary.
K	Size of the dictionary, i.e. the number of code-words in the dictionary.

$$D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}, \tag{3.1}$$

where $y_i \in \{1, 2, \ldots, C\}$ and C is the number of classes. An image $\mathbf{x}_i$ consists of M voxels,

$$\mathbf{x}_i = \{\mathbf{v}_{ij}\}_{j=1}^{M}, \tag{3.2}$$

where each voxel, $\mathbf{v}_{ij}$, is a time series, i.e. a vector, of BOLD intensity values. The 3D location of each voxel is fixed and known.

In Table 3.1, we present the notations and symbols used in the current and the next chapters.

3.3 The bag-of-words model

Using the bag-of-words model for a classification problem involves three steps:

1. Constructing a dictionary of words in an unsupervised manner,
2. Encoding the examples (e.g. images or documents) using the dictionary and optionally applying spatial pooling,
3. Training a classifier on the encoding vectors.

In the following subsections, we describe each of these steps.

3.3.1 Constructing the dictionary

Before we construct the dictionary (or generate the code-words), we first pass the time series vector of each voxel through a pre-processing function, $p(\cdot)$. Using $p(\cdot)$, we either take the mean of the time series or make it a zero-mean vector.

Let S be the set of all pre-processed voxel time series in the training set, that is

$$S = \{p(\mathbf{v}_{ij}) \mid i \in \text{TrainSet}, \ \forall j\}. \tag{3.3}$$

We use the k-means clustering algorithm [29] to construct the dictionary. Here, the parameter K defines the size of the dictionary, i.e. the number of code-words. The k-means algorithm groups the similar elements in S under the same cluster. We experimented with two similarity metrics: (i) Euclidean distance (Equation 3.4) and (ii) Pearson correlation (Equation 3.5).

$$d(\mathbf{m}_1, \mathbf{m}_2) = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T. \tag{3.4}$$

Equation 3.4 represents the squared Euclidean distance. It is the distance between two row matrices of same size, namely, $\mathbf{m}_1$ and $\mathbf{m}_2$.

$$d(\mathbf{m}_1, \mathbf{m}_2) = 1 - \frac{(\mathbf{m}_1 - \vec{\overline{\mathbf{m}}}_1)(\mathbf{m}_2 - \vec{\overline{\mathbf{m}}}_2)^T}{\sqrt{(\mathbf{m}_1 - \vec{\overline{\mathbf{m}}}_1)(\mathbf{m}_1 - \vec{\overline{\mathbf{m}}}_1)^T}\sqrt{(\mathbf{m}_2 - \vec{\overline{\mathbf{m}}}_2)(\mathbf{m}_2 - \vec{\overline{\mathbf{m}}}_2)^T}}, \tag{3.5}$$

where

- $\vec{\overline{\mathbf{m}}}_1 = \frac{1}{p}(\sum_{j=1}^{p} \mathbf{m}_{1j})\vec{1}_p$,
- $\vec{\overline{\mathbf{m}}}_2 = \frac{1}{p}(\sum_{j=1}^{p} \mathbf{m}_{2j})\vec{1}_p$,
- $\vec{1}_p$ is a row vector of p ones.

Equation 3.5 represents the Pearson correlation distance. It is the distance between two row matrices of same size, namely, $\mathbf{m}_1$ and $\mathbf{m}_2$.

The clustering algorithm produces a set of k centers, $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$. Assuming that these centers are the most representative features of the patterns in each cluster, we call them the code-words.

3.3.2 Encoding the examples

In the encoding step, we map each image to a feature vector $\mathbf{f}$. To do this, there are two different options for encoding the images, namely, soft encoding and hard encoding. There is also an optional spatial pooling operation which we will elaborate in Subsection 3.3.2.2. The following two subsections describe the hard and the soft encoding methods.

3.3.2.1 Hard Encoding

In hard-encoding, a voxel $\mathbf{v}_{ij}$ is encoded by a one-hot[1] activation vector $\mathbf{a}_{ij}$ whose all but the n^{th} element are zeros. $\mathbf{a}_{ij}(n) = 1$ where

[1] Only one entry is 1, others are 0s.

$$n = \arg\min_{k} ||\mathbf{c}_k - p(\mathbf{v}_{ij})||_2^2. \tag{3.6}$$

In effect, this encoding represents each voxel by the index of the cluster center that is most similar to it.

The BoW encoding of an fMRI image, $\mathbf{x}_i$, can be computed in two different ways. (1) encoding each voxel individually, (2) encoding voxels in groups, a technique which is known as *spatial pooling*.

In the first method, we concatenate all the activation vectors of an image's voxels. To be precise, the feature vector for the i^{th} image is

$$\mathbf{f}_i = [\mathbf{a}_{i1}, \mathbf{a}_{i2}, \ldots, \mathbf{a}_{iM}]. \tag{3.7}$$

The second method, i.e. spatial pooling, is described in the next subsection.

3.3.2.2 Spatial pooling

In spatial pooling [27], voxels are first grouped in a pre-defined way and then the image encoding is computed. The pre-defined grouping can correspond to anatomical brain regions or some arbitrary grouping as well. Let the voxel groupings be denoted by $R_1, R_2, \ldots, R_G$ where G is the number of total groups, each group specifies a set of voxels and $|R_i|$ is the number of voxels in group R_i. Then, the feature vector for the i^{th} image is

$$\mathbf{f}_i = \left[\frac{1}{|R_1|} \sum_{j \in R_1} \mathbf{a}_{ij}, \frac{1}{|R_2|} \sum_{j \in R_2} \mathbf{a}_{ij}, \ldots, \frac{1}{|R_G|} \sum_{j \in R_G} \mathbf{a}_{ij} \right]. \tag{3.8}$$

Here, pooling operation is simply averaging. Although there are alternatives to averaging (e.g. taking the max), in this work we only experimented with averaging.

3.3.2.3 Soft Encoding

In the soft-encoding method, the winner-take-all mechanism of the hard-encoding method is softened. In this work, we follow Coates et al. [9], [10] and compute the BoW encoding for each voxel as follows.

First, activation of each voxel for each cluster center is computed as

$$\mathbf{a}_{ijk} = \max\{0, \mu(\mathbf{z}) - z_k\}, \tag{3.9}$$

where $z_k = ||p(\mathbf{v}_{ij}) - \mathbf{c}_k||_2$ and $\mu(\mathbf{z})$ is the mean of the elements of $\mathbf{z}$. If the distance to the centroid $\mathbf{c}_k$ is above average, this activation takes the value of 0, otherwise, it takes a value that is reversely proportional to its distance to the cluster center (hence, the activation value becomes a *similarity* value). In effect, this creates a competition between features as it causes nearly half of the features to be set to 0.

Finally, the feature vector $\mathbf{f}_i$ is computed using one of two ways: (1) encoding individual voxels as in Equation 3.7 or (2) groupings voxels spatially, i.e. spatial pooling, as in Equation 3.8.

3.3.3 Training a classifier

Finally, we train linear SVMs on the BoW encodings (i.e. the feature vectors $\mathbf{f}_i$s) of the training images. Here, we optimize the C parameter in SVM using K-fold cross validation.

3.4 Chapter summary

In this chapter, we first give the motivation behind why we used the bag-of-words model in cognitive state classification with fMRI data. After that, the dataset representation is presented and the notations used in the chapter are described. Finally, the bag-of-words model, which is the representation model that we use to encode fMRI images, is described. Each of the three steps(constructing the dictionary, encoding the fMRI images and training the classifier) is explained in detail.

CHAPTER 4

EXPERIMENTS

In this chapter, we describe the experiments we conducted. First, we describe the main dataset we used in our experiments and present a basic exploratory analysis of the data. This analysis guides us in forming our experimental questions. Then, we present two baseline MVPA results with which we compare the results we obtain using the BoW modeling. Finally, we present BoW results on another dataset as a validation of our proposed method. In addition, we also report performances of two neural network models, namely multi-layer perceptron and convolutional neural network.

4.1 The dataset: Emotional Memory Retrieval Dataset

The main dataset we used in this work is a neuroimage (or image) dataset collected from 12 human subjects. For each subject, there are 210 training and 210 testing images. Each image in the dataset belongs to one of four classes.

The dataset was originally collected for an emotional memory retrieval experiment [33] where the stimuli consisted of two neutral (*kitchen utensils* and *furniture*) and two emotional (*fear* and *disgust*) categories of images. Each trial started with a 12-second fixation period followed by a 6-second encoding period. In the encoding period, participants were presented with 5 images from the same category (See Figure 4.1), each image lasting 1200 milliseconds on the screen. Following the fifth image, a 12-second delay period was presented in which participants solved three math problems consist-

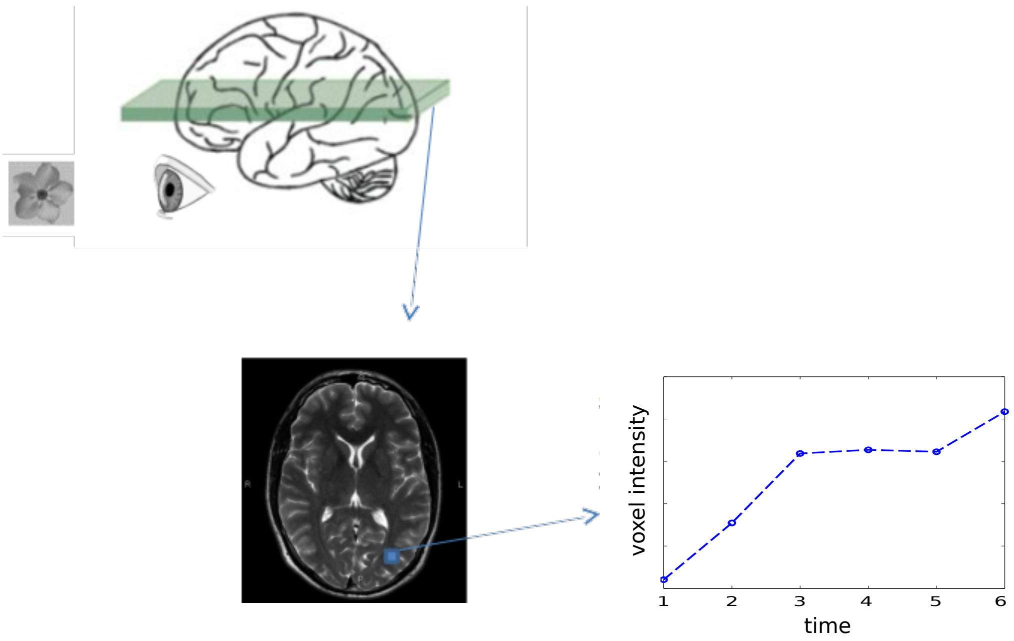


Figure 4.1: Time series intensity generation in a voxel.

ing of addition or subtraction of two randomly selected two-digit numbers. Following the third math problem, a 2-second retrieval period started in which participants were presented with a test image from the same category and indicated whether the image was a member of the current study list or not. The reader is referred to [33] for further details. For neuroimage classification, we employed measurements obtained during the encoding and retrieval phases as our training and testing data, respectively.

Figure 4.1 summarizes the entire experimental process. A randomly selected voxel is depicted in an axial slice. An fMRI time series is recorded at this voxel, for each stimulus for all the 4 classes. Also the intensity values of a sample response obtained from this voxel are visualized here.

4.1.1 Pre-processing

We applied spatial normalization [6] to the raw fMRI data using MATLAB's Statistical Parametric Mapping (SPM) toolbox [5]. We normalized the brain of each subject to a standard template [6]. The voxel size was set to $4\text{x}4\text{x}4 = 64\text{mm}^3$. The resulting number of voxels in each fMRI image was 22917.

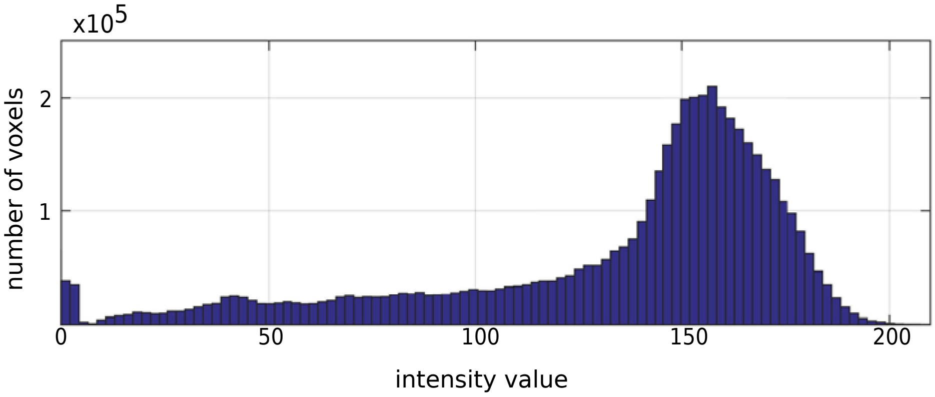


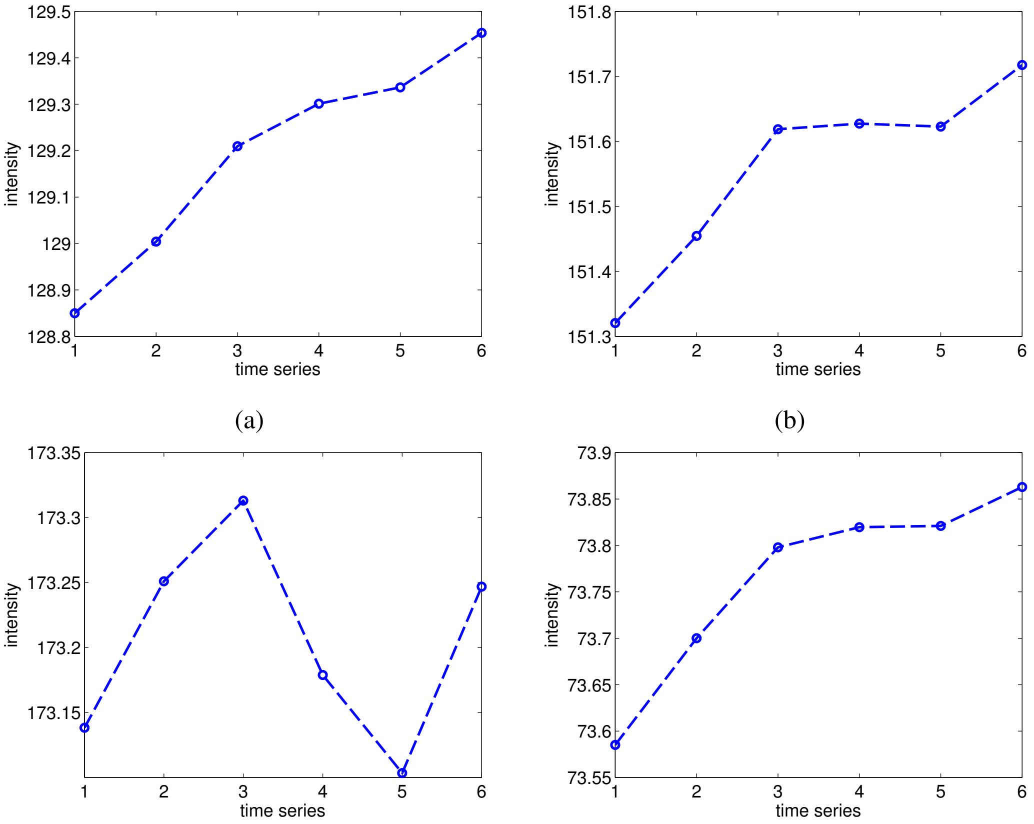
Figure 4.2: Histogram of voxel intensity values.

4.2 Basic Exploratory Data Analysis

In order to plan our classification experiments, first we need to have a basic understanding of the data. In this section, we present a very basic exploratory analysis of the dataset. This analysis gave us intuition about how should we plan our experiments.

Each voxel $\boldsymbol{v}_{ij}$ in the dataset has a time series consisting of 6 intensity measurements. We first explored how these intensity values are distributed. Figure 4.2 gives the histogram of voxel intensity values. The histogram gets dense around value 157 and has a heavy tail towards the smaller values. We have N images and for each image M voxels in the dataset. So there exist $N \times M$ many voxels in total. We take all these voxels and put them in an order as each voxel is located on top of another voxel. So we get a $N \times M$ by 6 matrix. Next, we ran the k-means clustering method [29] on this matrix, each row of which is the 6-dimensional time series, i.e. vectors representing voxel intensity over time, for various k values and visualized the resulting cluster centers. For relatively small k values, we observed that the cluster centers had similar shapes, i.e. how the time series changed (increased or decreased) over time, but had different average intensities (See Figure 4.3). For relatively large k values, we observed some variations in shape.

In order to capture the time series change patterns, i.e. the shapes, more effectively, we made each voxel's time series a zero-mean vector by subtracting the average intensity of each voxel from its time series. Then, we ran k-means again on these zero-



(a) (b)

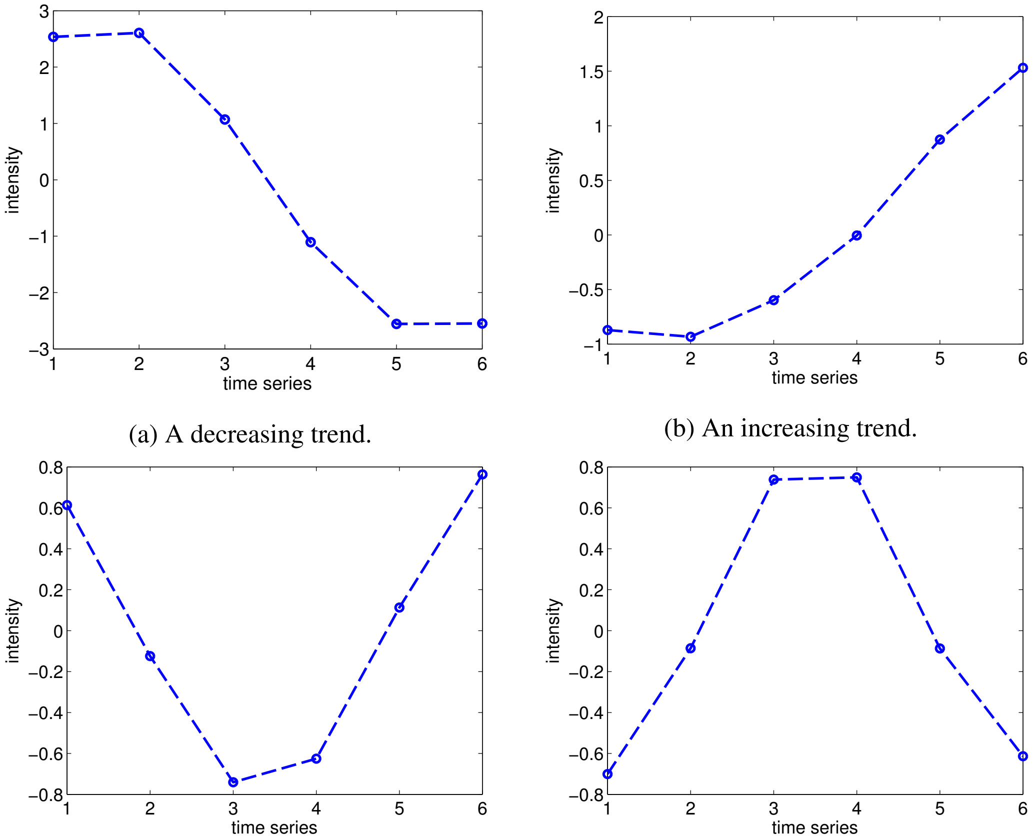
(c) (d)

Figure 4.3: A sample of cluster centers produced by k-means on the raw time series vectors in the dataset. Note that three cluster centers ((a), (b) and (d)) have similar shapes (i.e. increasing intensity) but different average intensity (values in the y-axis).

mean time series and the resulting cluster centers demonstrated a variety of shapes. A set of samples is given in Figure 4.4 where we can observe some clear shapes, i.e. trends, such as decreasing (Figure 4.4a), increasing (Figure 4.4b), decreasing and then increasing (Figure 4.4c), or increasing and then decreasing (Figure 4.4d).

Based on the results of our exploratory analysis, we asked the following questions which helped us design our experimental work:

- Does average intensity per voxel have any discriminative information?
- Does zero-mean time series per voxel (i.e. only the intensity trend) have any discriminative information?



(a) A decreasing trend.

(b) An increasing trend.

(c) A decreasing and then increasing trend.

(d) An increasing and then decreasing trend.

Figure 4.4: A sample of cluster centers produced by k-means on the zero-mean time series vectors in the dataset. Different shapes, i.e. trends, become obvious when the time series vectors are made zero-mean and unit-variance.

- Which of the two items above has more discriminative information? And if we combine them, will there be a gain in terms of classification accuracy?
- Does including spatial or functional neighbors to the modeling increase the classification accuracy?
- Does spatial pooling (e.g. with respect to anatomical brain regions) increase the classification accuracy?
- Does voxel selection together bag-of-words modeling help improve the classification accuracy?

In the following sections, we first present baseline MVPA results with which we compare our various bag-of-words modeling results.

4.3 Baseline MVPA Results

We decode cognitive brain states, i.e. which class of image the subject was looking at, by automatically classifying subjects' fMRI data using linear SVM. In the following, we describe how we feed the data to the linear SVM without processing the data using the bag-of-words model. In each SVM experiment reported in this work, we selected the optimal C parameter value by doing 5-fold cross-validation on the training set.

We established two baseline performances by training linear SVMs on raw voxel intensity values without applying any BoW encoding.

4.3.1 Linear SVM on Raw Time Series

Here, we used voxel intensity time series without any pre-processing, i.e. the pre-processing function is the identity function: $p(x) = x$. We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating all the $\mathbf{v}_{ij}$ vectors (for $j = 1 \ldots M$). In this case, for each subject the training data was a 210x137502 matrix where $N = 210$ is the number of training neuroimages, each neuroimage has $M = 22917$ voxels, each

Table 4.1: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on raw time series.

Subjects	**Accuracies**
S1	32.86%
S2	33.81%
S3	26.19%
S4	30.00%
S5	25.24%
S6	52.86%
S7	41.90%
S8	33.81%
S9	29.05%
S10	28.57%
S11	35.24%
S12	35.24%
AVERAGE	$\mathbf{33.73}\%$

voxel has 6-dimensional time series of intensity values and 22917x6 $= 137502$. We trained a linear SVM on $(\mathbf{f}_i, y_i)$ pairs in the training set. The average classification accuracy[1] over 12 different datasets was $\mathbf{33.73}\% \pm \mathbf{2.10}\%$ where 2.10% represent one standard-error[2]. The percent correct classification rates per subject are presented in Table 4.1.

4.3.2 Linear SVM on Average of Time Series

Here, we set another baseline performance where we used the *average* intensity of the time series to represent a voxel. That is, the pre-processing function in this case is $p(\mathbf{v}_{ij}) = \frac{1}{6}\sum_{l=1}^{6} \mathbf{v}_{ij}(l)$. We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating

[1] Percent correctly predicted testing labels.

[2] Standard error is equal to the bootstrapped standard error of the mean. That is, firstly, new samples are generated, then, means of these samples are calculated, finally, the standard deviation of these means gives the bootstrapped standard error.

Table 4.2: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series.

Subjects	**Accuracies**
S1	34.76%
S2	35.24%
S3	33.81%
S4	31.43%
S5	32.86%
S6	68.57%
S7	51.90%
S8	40.00%
S9	29.05%
S10	25.24%
S11	39.52%
S12	42.38%
AVERAGE	$\mathbf{38.73}\%$

the average intensity value at each voxel, i.e. $\mathbf{f}_i = [p(\mathbf{v}_{i1}), p(\mathbf{v}_{i2}), \ldots, p(\mathbf{v}_{iM})]$. This reduced the training matrix size to the number of images times the number of voxels, i.e. 210x22917. The classification accuracy was $\mathbf{38.73}\% \pm \mathbf{3.22}\%$. The result is shown in Table 4.2.

4.4 Bag-of-words Modeling Results

Looking at the baseline results, one can hypothesize that the average intensity over time per voxel has more discriminative information than the intensity time series. In this section, we explore whether the same is true when we apply the BoW encoding. In our preliminary experiments on this dataset, hard-encoding (Subsection 3.3.2.1) consistently yield better performance than soft-encoding (Subsection 3.3.2.3), for this reason, all of the results we report on this dataset in the following are based on hard-encoding.

4.4.1 Average of Time Series + BoW

Table 4.3: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series with BoW.

Subjects	**Accuracies**
S1	38.10%
S2	51.43%
S3	44.29%
S4	49.05%
S5	38.57%
S6	69.52%
S7	55.24%
S8	50.00%
S9	35.24%
S10	35.24%
S11	42.86%
S12	41.90%
AVERAGE	$\mathbf{45.95}\%$

Here, we applied BoW encoding to average intensities of individual voxels. Specifically, the pre-processing function is $p(\mathbf{v}_{ij}) = \frac{1}{6}\sum_{l=1}^{6}\mathbf{v}_{ij}(l)$ and we run the k-means method on all $p(\mathbf{v}_{ij})\ \forall i, j$ (in the training set) to obtain the code-words. The activation vector $\mathbf{a}_{ij}$ for voxel $\mathbf{v}_{ij}$ is computed as described in Section 3.3.2.1, Equation (3.6). We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating the activation vectors, i.e. $\mathbf{f}_i = [\mathbf{a}_{i1}, \mathbf{a}_{i2}, \cdots, \mathbf{a}_{iM}]$.

We set the value of k using cross-validation on the training set as follows. We randomly split the training set of each subject into 70% training and 30% evaluation sets. Then, for various values of k, we trained a linear SVM on the training set and evalu-

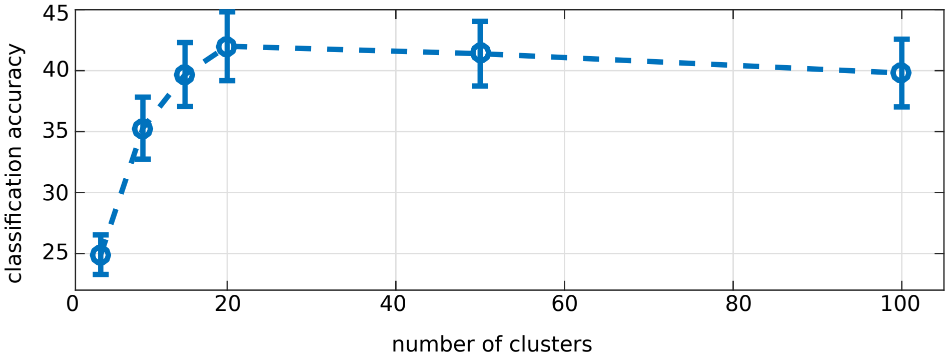


Figure 4.5: The number of clusters (k) versus classification accuracy on a random 70%-30% split of the training sets of each subject. Error bars denote one standard error (over 12 measurements).

ated the SVM on the evaluation set. Figure 4.5 gives the cross-validation results. The optimal value of k turned out be 20. Values lower than 20 and larger than 20 resulted in worse validation performances.

Using the optimal setting ($k = 20$), on the whole training and testing sets, linear SVM's average accuracy over 12 subjects was $\mathbf{45.95}\% \pm 2.73\%$. This result is shown in Table 4.3.

4.4.2 Trend (zero-mean time series) + BoW

Here, we applied the BoW encoding to time series trends. The pre-processing function used in this case was $p(\mathbf{v}_{ij}) = \mathbf{v}_{ij} - \frac{1}{6}\sum_{l=1}^{6}\mathbf{v}_{ij}(l)$. The optimal number of clusters in this case was 50. The average classification accuracy was $25.24\% \pm 0.84\%$ (shown in Table 4.4).

4.4.3 Average of Time Series + Trend + BoW

Here, we concatenated the two BoW models described above, i.e. the average intensity BoW ($k = 20$) and the trend BoW ($k = 50$). The average classification accuracy for this case was $32.66\% \pm 0.77\%$ (shown in Table 4.5).

Table 4.4: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the time series trends with BoW.

Subjects	**Accuracies**
S1	26.67%
S2	25.71%
S3	24.29%
S4	29.05%
S5	25.71%
S6	20.48%
S7	21.43%
S8	30.48%
S9	27.62%
S10	25.24%
S11	21.90%
S12	24.29%
AVERAGE	$\mathbf{25.24}\%$

4.4.4 Raw Time Series + BoW

Here, we used voxel intensity time series without any pre-processing, i.e. the preprocessing function is the identity function: $p(x) = x$, and we run the k-means method on all $p(\mathbf{v}_{ij})$ $\forall i, j$ (in the training set) to obtain the code-words. The activation vector $\mathbf{a}_{ij}$ for voxel $\mathbf{v}_{ij}$ is computed as described in Subsection 3.3.2.1, Equation (3.6). We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating the activation vectors, i.e. $\mathbf{f}_i = [\mathbf{a}_{i1}, \mathbf{a}_{i2}, \cdots, \mathbf{a}_{iM}]$.

We set the value of k using cross-validation on the training set as follows. We randomly split the training set of each subject into 70% training and 30% evaluation sets. Then, for various values of k, we trained a linear SVM on the training set and evaluated the SVM on the evaluation set. The optimal value of k turned out be 50. Values lower than 50 and larger than 50 resulted in worse validation performances.

Table 4.5: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the time series + time series trends with BoW.

Subjects	**Accuracies**
S1	30.00%
S2	34.29%
S3	31.43%
S4	32.86%
S5	29.05%
S6	38.57%
S7	33.33%
S8	36.19%
S9	32.86%
S10	30.95%
S11	29.52%
S12	32.86%
AVERAGE	$\mathbf{32.66}\%$

Using the optimal setting ($k = 50$), on the whole training and testing sets, linear SVM's average accuracy over 12 subjects was $\mathbf{49.25}\% \pm 2.48\%$. This result is shown in Table 4.6.

This method gave the most significant results in this work since the performance increase is $\mathbf{15.52}\%$ compared to the results in Subsection 4.3.1. This is the largest classification accuracy increase over its MVPA counterpart in this work.

4.4.5 BoW with Spatial Pooling

In all the previous experiments we reported above, BoW encoding was applied to individual voxels and the resulting activation vectors were concatenated to form the image feature vector. In this section, we spatially pool the activation vectors within each anatomical region of the brain. Specifically, we form the feature vector for the

Table 4.6: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the raw time series with BoW.

Subjects	**Accuracies**
S1	42.86%
S2	55.24%
S3	47.14%
S4	50.95%
S5	45.24%
S6	70.95%
S7	57.14%
S8	52.86%
S9	40.95%
S10	37.62%
S11	44.29%
S12	45.71%
AVERAGE	$\mathbf{49.25}\%$

i^{th} image as follows

$$\mathbf{f}_i = \left[\frac{1}{|R_1|} \sum_{j \in R_1} \mathbf{a}_{ij}, \frac{1}{|R_2|} \sum_{j \in R_2} \mathbf{a}_{ij}, \ldots, \frac{1}{|R_{116}|} \sum_{j \in R_{116}} \mathbf{a}_{ij} \right], \tag{4.1}$$

where R_l is an anatomical region and $|R_l|$ represents the number of voxels in that region. Activation vectors were computed on average voxel intensities. This method gave an average classification accuracy of $25.95\% \pm 0.72\%$ (shown in Table 4.7).

4.4.6 BoW with Voxel Selection

Here, we first performed voxel selection using the MATLAB's relieff function [4], [24]. Optimal setting for this case was selecting 3000 voxels. After that we applied the model in Subsection 4.4.1. The average classification accuracy for this case was

Table 4.7: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series with spatial pooling BoW.

Subjects	**Accuracies**
S1	28.57%
S2	25.24%
S3	26.19%
S4	26.19%
S5	25.24%
S6	25.24%
S7	23.33%
S8	30.95%
S9	24.29%
S10	27.14%
S11	28.10%
S12	20.95%
AVERAGE	$\mathbf{25.95\%}$

$44.21\% \pm 3.10\%$ (shown in Table 4.8). As this model did not outperform the model in Subsection 4.4.1, we did not apply it on raw time series.

4.4.7 BoW with Spatial and Functional Neighborhood

We also tried to utilize neighborhood information in decoding the cognitive states. Both spatial and functional neighborhood experiments were performed. Since the most significant classification accuracy increase was obtained in Subsection 4.4.4, these experiment were carried out only with voxel intensity time series.

Table 4.8: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series with voxel selection and BoW.

Subjects	**Accuracies**
S1	38.57%
S2	49.52%
S3	46.67%
S4	42.38%
S5	30.00%
S6	71.90%
S7	55.24%
S8	39.05%
S9	35.71%
S10	32.86%
S11	43.81%
S12	44.76%
AVERAGE	**44.21%**

4.4.7.1 Spatial Neighborhood

Definition 4.1. Spatial Neighborhood: For each voxel v, we set v as the center voxel and define the neighborhood of voxel v within a radius r by $n(v,r) = \{u \in V | d(v,u) < r\}$. Here V is the set of voxels and $d(v,u)$ is the Euclidean distance between the spatial coordinates of v and u in 3-dimensional space.

Here, for each voxel v, we first detect the n many nearest neighbors(voxels) as explained in Definition 4.1 and concatenate them with the voxel v. After that we apply the model mentioned in Subsection 4.4.4. Here, the optimal number of neighbors is 4. The average classification accuracy for this case was $45.71\% \pm 2.43\%$ (shown in Table 4.9).

Table 4.9: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the raw time series with spatial neighborhood and BoW.

Subjects	**Accuracies**
S1	40.00%
S2	53.81%
S3	43.33%
S4	47.62%
S5	40.95%
S6	66.67%
S7	51.90%
S8	48.57%
S9	36.19%
S10	34.76%
S11	43.81%
S12	40.95%
AVERAGE	$\mathbf{45.71}\%$

4.4.7.2 Functional Neighborhood

Definition 4.2. Functional Neighborhood: Let $X_v(t)$ denote the intensity time series of voxel v. We first set the similarity measure metric as Pearson correlation. Now to detect the functionally nearest neighbor of voxel v, we calculate the similarity between intensity time series $X_v(t)$ and $X_u(t)$ for all the voxel pairs $v, u \in V$. Then we select the voxels having p of the highest similarity with v.

Here, for each voxel v, we first detect the n many nearest neighbors(voxels) as explained in Definition 4.2 and concatenate them with the voxel v. After that we apply the model mentioned in Subsection 4.4.4. Here, the optimal number of neighbors is 4. The average classification accuracy for this case was $46.07\% \pm 2.43\%$ (shown in Table 4.10).

Table 4.10: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the raw time series with functional neighborhood and BoW.

Subjects	**Accuracies**
S1	43.33%
S2	51.43%
S3	41.90%
S4	46.67%
S5	40.95%
S6	68.57%
S7	52.86%
S8	49.52%
S9	37.62%
S10	35.71%
S11	42.38%
S12	41.90%
AVERAGE	$\mathbf{46.07}\%$

Here, we also notice that nearly half of the functionally nearest neighbors are also the half of the spatially nearest neighbors of the specific voxel v.

4.4.8 Raw Time Series + BoW with Pearson Correlation

Here we applied the model in Subsection 4.4.4 with a minor difference. It is in the phase of dictionary construction. Until this experiment, we used Euclidean distance as the distance metric of k-means algorithm. However, here we used Pearson correlation as the distance metric. What we tried to answer is how changing the distance metric affects the classification performance. The average classification accuracy for this case was $24.29\% \pm 0.82\%$ (shown in Table 4.11).

Table 4.11: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the time series with Pearson Correlation BoW.

Subjects	**Accuracies**
S1	26.19%
S2	21.90%
S3	22.86%
S4	29.05%
S5	23.33%
S6	18.10%
S7	25.24%
S8	26.67%
S9	27.62%
S10	25.24%
S11	22.86%
S12	22.38%
AVERAGE	**24.29%**

4.4.9 Summary

In this section, we summarize the bag-of-words modeling results on the emotional memory retrieval dataset (Section 4.1). In Table 4.12, we present all of the subject-averaged results. As can be seen here, the most promising results were obtained when we applied the BoW model with hard encoding to the raw time series and the average of the time series intensities of voxels. We see that extending the experiments with spatial pooling, voxel selection, adding neighborhood and using Pearson correlation as distance matrix resulted in losing discriminative information.

4.4.10 Deep Learning Experiments Results

We also did some preliminary deep learning experiments on this dataset. We used Multilayer Perceptron (MLP) [46] and Convolutional Neural Network (CNN) [28]

Table 4.12: Classification accuracies (percent correct on testing labels) of baseline and proposed methods.

Method		**Accuracy**
Baseline	Raw time series	$\mathbf{33.73}\% \pm 2.10\%$
	Average of time series	$\mathbf{38.73}\% \pm 3.22\%$
BoW	Time series + BoW	$\mathbf{49.25}\% \pm 2.48\%$
	Average + BoW	$\mathbf{45.95}\% \pm 2.73\%$
	Trend + BoW	$25.24\% \pm 0.84\%$
	Average + Trend + BoW	$32.66\% \pm 0.77\%$
	Average + Spatial pooling BoW	$25.95\% \pm 0.72\%$
	Average + Voxel selection + BoW	$44.21\% \pm 3.10\%$
	Time series + Spatial neighborhood + BoW	$45.71\% \pm 2.43\%$
	Time series + Functional neighborhood + BoW	$46.07\% \pm 2.43\%$
	Time series + BoW with Pearson Correlation	$24.29\% \pm 0.82\%$

models with several different architectures and parameters. The results reported here are by no means complete or comprehensive. Nonetheless, they are at least useful for a machine learning practitioner who aims to use deep learning models on fMRI images.

4.4.10.1 Multilayer Perceptron (MLP) experiments

We used the data of two different subjects and experimented with

1. different number of layers (between 2 and 4, inclusive),

2. different number of hidden neurons per layer (8, 16, 32, 64, 128),

3. different activation non-linearities (ReLU, leaky ReLU and sigmoid),

4. different batch sizes (16, 32, 64),
5. different learning methods (Adam, SGD, RMSProp) and
6. using or not using batch-normalization layers.

Among all these different variants, only the batch-normalization made a significant difference. In short, we observed two distinct network behaviors: (1) no or almost zero learning, (2) learning but overfitting.

As a representative result for the first behavior (i.e. no or almost zero learning), we report the performance of a 3 layer MLP without any batch-normalization layers. The details of this architecture can be found in Figure 4.6 and the accuracy, loss plots can be seen in Figure 4.7.

When we added a batch-normalization layer *right after the input* (Figure 4.8), the behavior of the MLP changed qualitatively; it started to learn, however, it showed overfitting (Figure 4.9). Adding batch-normalization to any other place (e.g. in between a linear dense layer and its subsequent activation non-linearity) did not change the behavior.

4.4.10.2 Convolutional Neural Network (CNN)

Here, we used the CNN model with similar architectural and parametrical variations as we did with the MLPs, however, we could not get any CNN to obtain better-than-chance accuracy on the test set. We believe that this extreme overfitting is due to the fact that the number of training images (210) is very low. A representative architecture is given in Figure 4.10 and the accuracy, loss plots are presented in Figure 4.11.

4.4.11 The results of other studies conducted on this dataset

In the work done by Ertugrul et al. [42], functionally local meshes are constructed around each voxel of the brain volume using a functional neighborhood system. They

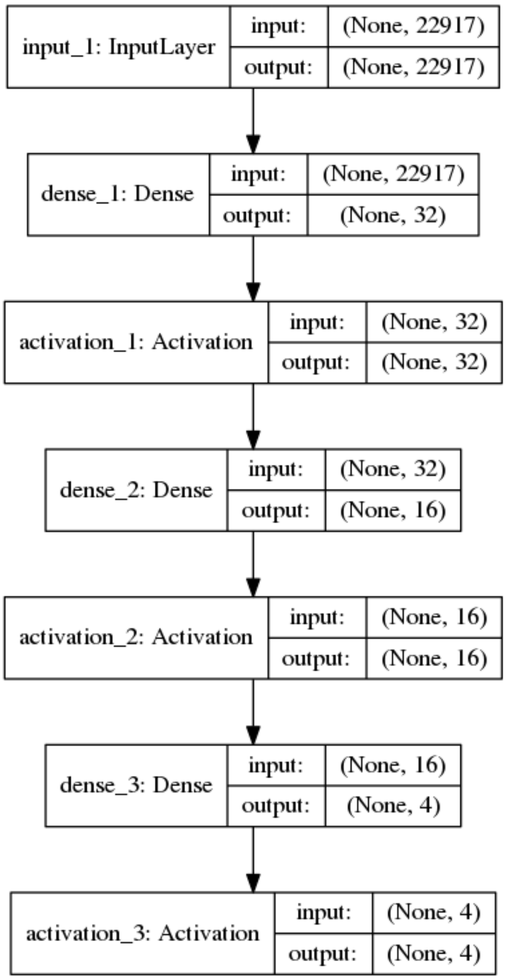


Figure 4.6: The architecture of the Multilayer Perceptron model. Here, the activation layers use the ReLU (rectified linear unit) non-linearity.

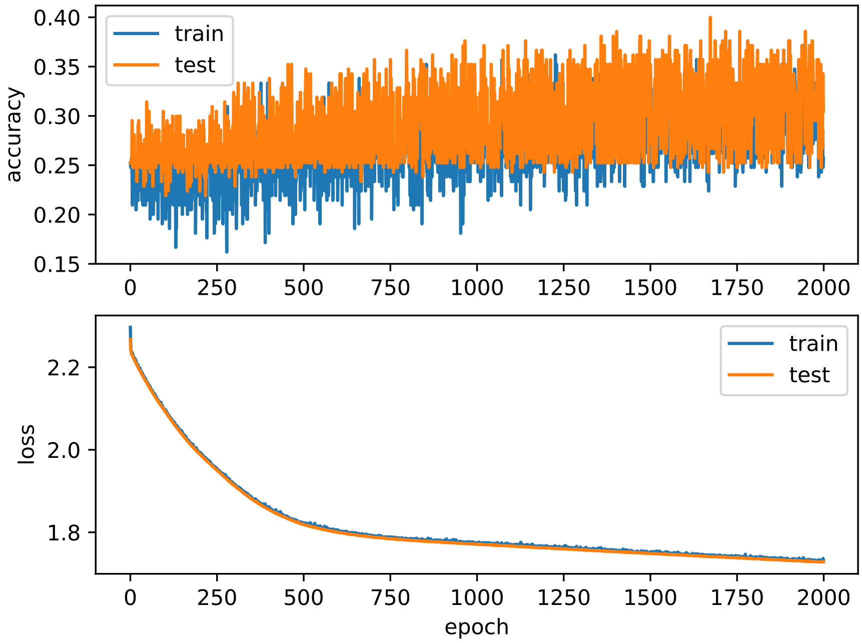


Figure 4.7: Accuracy and loss when we used MLP without batch normalization.

use Pearson correlation to specify the functional neighbors. After that, they estimate the edge weights of these meshes and feed these weights directly to the classification algorithm. The average classification accuracy obtained with this method is 66.56%.

4.5 Validation of Our Methods on Another Dataset

Given the success of our proposed method on an fMRI dataset (see Section 4.1), we asked whether we can replicate the same success on another fMRI dataset. Below, we describe this additional dataset used in this work. We did not do a comprehensive bag-of-words evaluation on this dataset due to time constraints. We evaluated one of our best performing BoW method (hard-encoding on average of time series) to see if it produces a similar improvement as it did on the emotional memory retrieval dataset. We also evaluated soft-encoding on average of time series and compared the results.

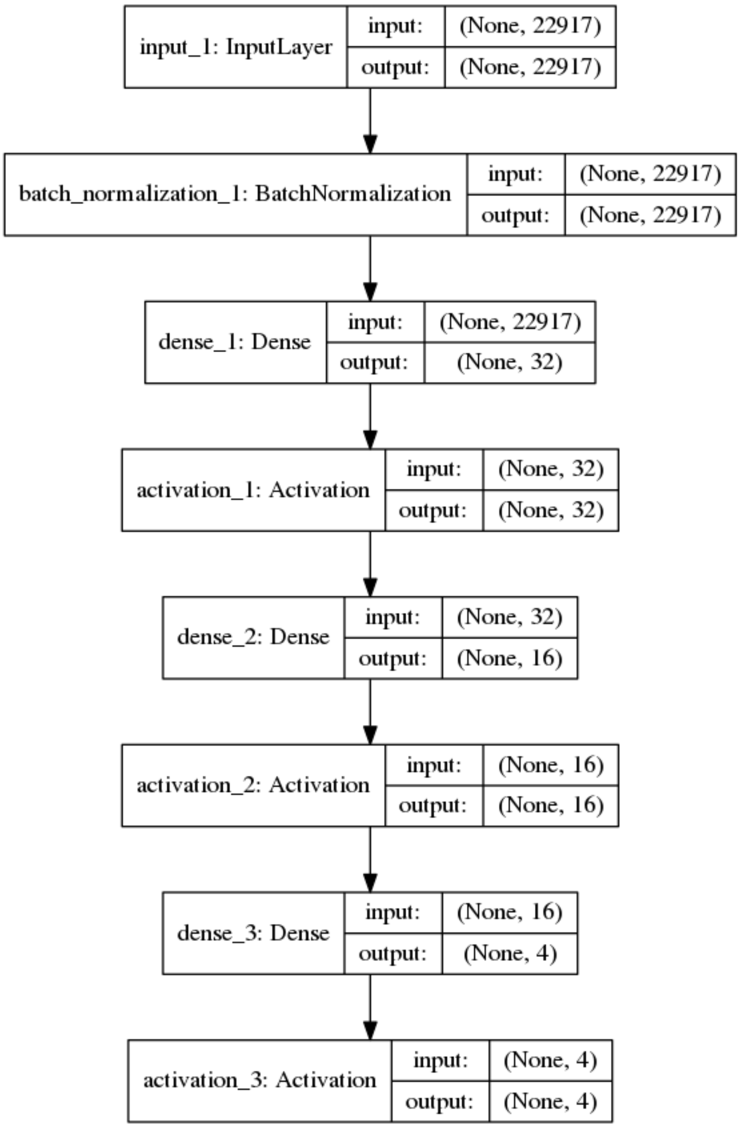


Figure 4.8: MLP model with batch normalization.

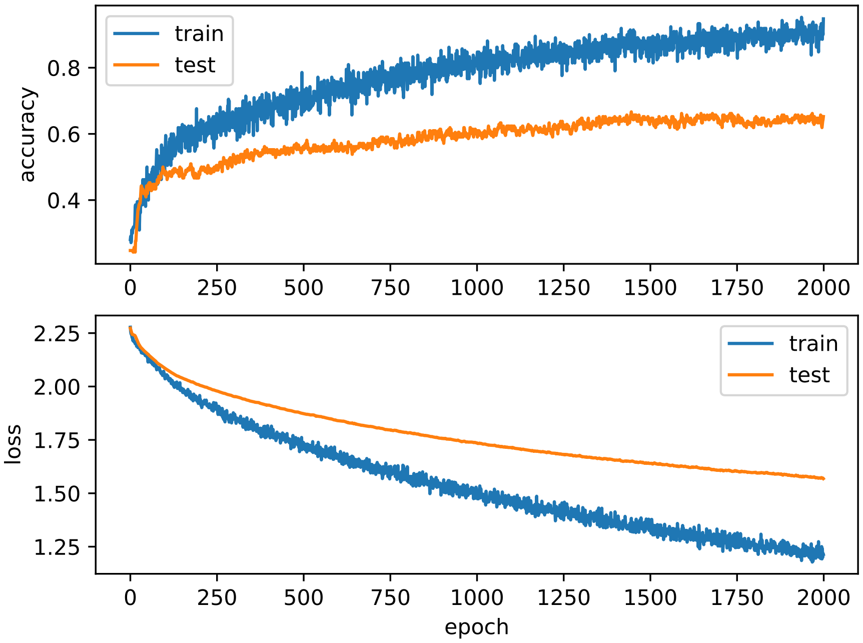


Figure 4.9: Accuracy and loss when we used MLP with batch normalization.

4.5.1 Visual Object Recognition Experiment Dataset

This dataset is the secondary dataset we used in this work. It was used to validate the success we obtained in the first dataset on which we applied our model. It is a neuroimage dataset collected from 5 human subjects. Each neuroimage in the dataset belongs to one of two classes. For each subject, there are 216 images. Each neuroimage was normalized by the standard normalization step of preprocessing of the raw fMRI data. This normalization is done by Matlab's Statistical Parametric Mapping (SPM) toolbox [5]. We normalized the brain of each subject to a standard template [6]. The voxel size was set to $4\text{x}4\text{x}4 = 64\text{mm}^3$.

The dataset was originally collected for a visual object recognition experiment where the stimuli consisted of gray-scale images belonging to two categories, namely, birds and flowers. Participants performed a one-back repetition detection task in this experiment. Each trial started with a 4-second stimulus presentation period followed by a 8-second rest period. A Siemens 3T Magnetom TRIO MRI System was used for this

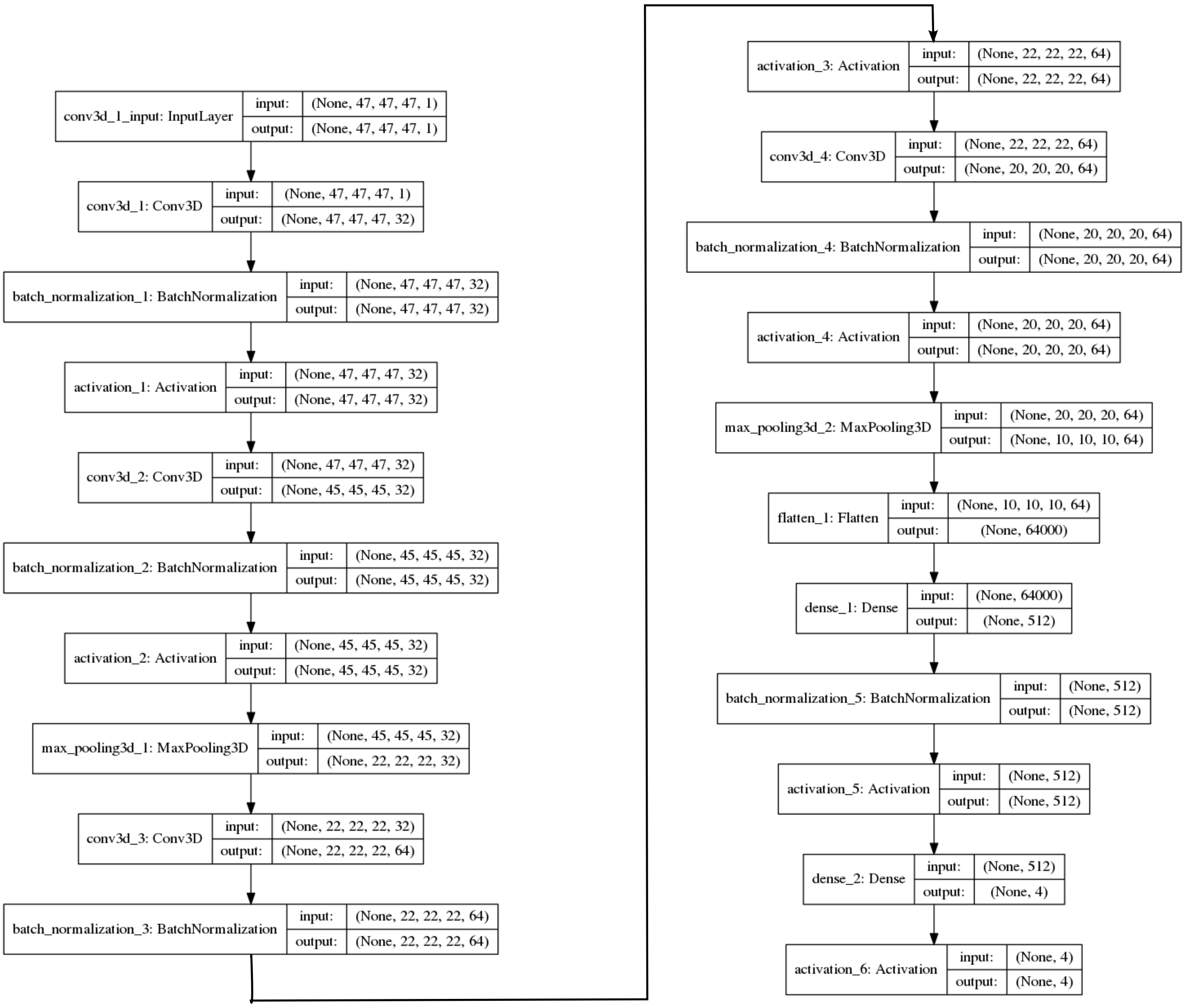

conv3d_1_input: InputLayer
input: (None, 47, 47, 47, 1)
output: (None, 47, 47, 47, 1)
conv3d_1: Conv3D
input: (None, 47, 47, 47, 1)
output: (None, 47, 47, 47, 32)
batch_normalization_1: BatchNormalization
input: (None, 47, 47, 47, 32)
output: (None, 47, 47, 47, 32)
activation_1: Activation
input: (None, 47, 47, 47, 32)
output: (None, 47, 47, 47, 32)
conv3d_2: Conv3D
input: (None, 47, 47, 47, 32)
output: (None, 45, 45, 45, 32)
batch_normalization_2: BatchNormalization
input: (None, 45, 45, 45, 32)
output: (None, 45, 45, 45, 32)
activation_2: Activation
input: (None, 45, 45, 45, 32)
output: (None, 45, 45, 45, 32)
max_pooling3d_1: MaxPooling3D
input: (None, 45, 45, 45, 32)
output: (None, 22, 22, 22, 32)
conv3d_3: Conv3D
input: (None, 22, 22, 22, 32)
output: (None, 22, 22, 22, 64)
batch_normalization_3: BatchNormalization
input: (None, 22, 22, 22, 64)
output: (None, 22, 22, 22, 64)
activation_3: Activation
input: (None, 22, 22, 22, 64)
output: (None, 22, 22, 22, 64)
conv3d_4: Conv3D
input: (None, 22, 22, 22, 64)
output: (None, 20, 20, 20, 64)
batch_normalization_4: BatchNormalization
input: (None, 20, 20, 20, 64)
output: (None, 20, 20, 20, 64)
activation_4: Activation
input: (None, 20, 20, 20, 64)
output: (None, 20, 20, 20, 64)
max_pooling3d_2: MaxPooling3D
input: (None, 20, 20, 20, 64)
output: (None, 10, 10, 10, 64)
flatten_1: Flatten
input: (None, 10, 10, 10, 64)
output: (None, 64000)
dense_1: Dense
input: (None, 64000)
output: (None, 512)
batch_normalization_5: BatchNormalization
input: (None, 512)
output: (None, 512)
activation_5: Activation
input: (None, 512)
output: (None, 512)
dense_2: Dense
input: (None, 512)
output: (None, 4)
activation_6: Activation
input: (None, 4)
output: (None, 4)

Figure 4.10: CNN model.

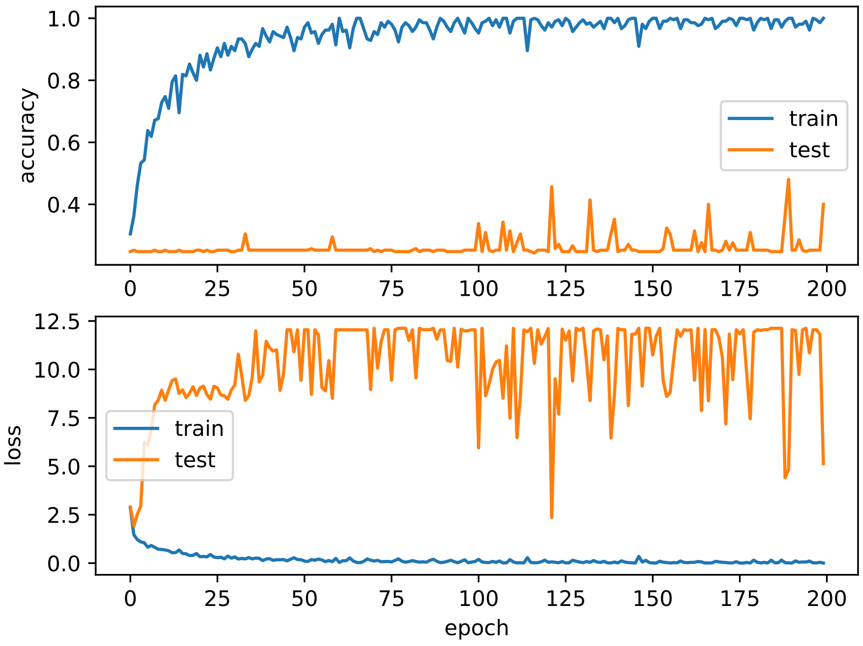


Figure 4.11: Accuracy and loss when we used CNN.

experiment. Functional images were acquired using a gradient EPI sequence (TR = 2000 msec, TE = 30 msec, flip angle = 90, 34 interleaved axial slices). Since our TR = 2000 msec, for each trial we get 6 measurements. In this experiment there were 6 sessions each of which consists of 36 trials. Therefore, for a single subject 216 trials in total were obtained.

A pre-defined train/test split is not given for this dataset. For this reason, we randomly split the dataset into 50% training and 50% testing. We repeated this 10 times.

4.6 Baseline MVPA Results

Here, we set the baseline performance where we used the *average* intensity of the time series to represent a voxel. That is, the pre-processing function in this case is $p(\mathbf{v}_{ij}) = \frac{1}{6}\sum_{l=1}^{6} \mathbf{v}_{ij}(l)$. We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating the average intensity value at each voxel, i.e. $\mathbf{f}_i = [p(\mathbf{v}_{i1}), p(\mathbf{v}_{i2}), \ldots, p(\mathbf{v}_{iM})]$. This

Table 4.13: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series.

Subjects	**Accuracies**
S1	76.39%
S2	75.74%
S3	79.07%
S4	82.22%
S5	69.91%
AVERAGE	$\mathbf{76.67}\%$

reduced the training matrix size to the number of images times the number of voxels. The classification accuracy was $76.67\% \pm 1.83\%$. The result is shown in Table 4.13.

4.7 Bag-of-words Modeling Results on Visual Object Recognition Experiment Dataset

4.7.1 Soft Encoding Results

Table 4.14: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series with soft encoding BoW.

Subjects	**Accuracies**
S1	74.54%
S2	77.41%
S3	78.98%
S4	82.56%
S5	69.72%
AVERAGE	$\mathbf{76.64}\%$

Here, we applied BoW encoding to average intensities of individual voxels. Specifically, the pre-processing function is $p(\mathbf{v}_{ij}) = \frac{1}{6}\sum_{l=1}^{6} \mathbf{v}_{ij}(l)$ and we run the k-means method on all $p(\mathbf{v}_{ij})$ $\forall i, j$ (in the training set) to obtain the code-words. The activation vector $\mathbf{a}_{ij}$ for voxel $\mathbf{v}_{ij}$ is computed as described in Section 3.3.2.3, Equation (3.9). We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating the activation vectors, i.e. $\mathbf{f}_i = [\mathbf{a}_{i1}, \mathbf{a}_{i2}, \cdots, \mathbf{a}_{iM}]$. Linear SVM's average accuracy over 5 subjects was 76.64% ± 1.93%. This result is shown in Table 4.14.

Here, we notice that when compared to the baseline results (Section 4.6), the classification accuracies of two subjects (Subject2 and Subject 4) were increased with soft encoding.

4.7.2 Hard Encoding Results

Table 4.15: Classification accuracies (percent correct on testing labels) for each subject, when we apply linear SVM on the average of time series with hard encoding BoW.

Subjects	**Accuracies**
S1	59.54%
S2	60.65%
S3	57.22%
S4	72.11%
S5	59.44%
AVERAGE	**61.79%**

Here, we applied BoW encoding to average intensities of individual voxels. Specifically, the pre-processing function is $p(\mathbf{v}_{ij}) = \frac{1}{6}\sum_{l=1}^{6} \mathbf{v}_{ij}(l)$ and we run the k-means method on all $p(\mathbf{v}_{ij})$ $\forall i, j$ (in the training set) to obtain the code-words. The activation vector $\mathbf{a}_{ij}$ for voxel $\mathbf{v}_{ij}$ is computed as described in Section 3.3.2.1, Equation (3.6). We form a feature vector $\mathbf{f}_i$ for the i^{th} image by concatenating the activation vectors, i.e. $\mathbf{f}_i = [\mathbf{a}_{i1}, \mathbf{a}_{i2}, \cdots, \mathbf{a}_{iM}]$. Linear SVM's average accuracy over 5 subjects was 61.79% ± 2.37%. This result is shown in Table 4.15.

4.7.3 Summary

Table 4.16: Classification accuracies (percent correct on testing labels) of baseline and proposed methods.

Method		Accuracy
Baseline	Average of time series	$76.67\% \pm 1.83\%$
BoW	Average + BoW with soft encoding	$76.64\% \pm 1.93\%$
	Average + BoW with hard encoding	$61.79\% \pm 2.37\%$

Interestingly, the BoW method that produced the best result on our main dataset, did worse than the standard MVPA method on this dataset. However, when we replaced hard-encoding with soft-encoding, the performance of BoW increased but did not outperform the MVPA baseline. Since these results are based on 10 random splits of the dataset, we performed a paired t-test for each subject in order to test whether BoW and MVPA results are significantly different from each other. For all subjects, the test failed to reject the null-hypothesis (i.e. BoW and MVPA results are coming from the same distribution) at $\alpha = 0.05$ significance level ($p = 0.26$ for S1, $p = 0.93$ for S2, $p = 0.13$ for S3, $p = 0.38$ for S4, and $p = 0.23$ for S5).

CHAPTER 5

CONCLUSION

In this thesis, we used the BoW method to encode the voxel intensities of fMRI data in various ways. When we apply the hard BoW encoding model to individual voxels, there was a $\mathbf{7.22\%}$ improvement in average classification accuracy when applied on average intensity per voxel and a $\mathbf{15.52\%}$ improvement in average classification accuracy when applied to raw intensity time series per voxel compared to a classical multi voxel pattern analysis (MVPA) method. This is a promising result which encourages us to further explore the use of BoW model in encoding neuroimages.

Based on the results we reported in the previous section, we draw the following conclusions.

- Average intensity (i.e. representing each voxel with its average intensity over time) has more discriminative information than the intensity time series according to MVPA results.
- Applying the BoW encoding only to the average of the time series has discriminative information.
- Applying the BoW encoding only to the raw time series has discriminative information.
- Applying the BoW encoding only to the trend of the time series (i.e. zero-mean intensity time series vectors) does not have any discriminative information (it produces chance-level classification accuracy).

- Applying the BoW encoding to the combination of the trend of the time series and average of time series has some minor discriminative information (it produces a classification accuracy of 32.66% which is a little bit higher than the chance level).

- Applying the BoW encoding with spatial pooling using anatomical brain regions did not improve the classification accuracy. In fact, it degraded the accuracy to the chance level, which indicates the importance of the (voxel) location information in the task at hand.

- Applying the BoW encoding with voxel selection to the average of the time series has some discriminative information but it did not outperform the "average + BoW" case.

- Applying the BoW encoding with spatial neighborhood to the time series has some discriminative information but it did not outperform the "time series + BoW" case.

- Applying the BoW encoding with functional neighborhood to the time series has some discriminative information but it did not outperform the "time series + BoW" case.

- Applying the BoW encoding with Pearson correlation as the distance metric to the time series does not have any discriminative information (it produces chance-level classification accuracy).

One should note that above conclusions are valid in the context of a specific fMRI dataset and linear SVM classifiers. In order to draw more general conclusions, further experimental work should be done using additional datasets.

Possible future directions of this study can be listed as follows:

- In order to further validate the methods used here, experiments can be repeated on different and larger datasets with higher number of subjects and cognitive stimuli.

- Feature selection algorithms can be effectively applied before the dictionary construction phase of our proposed method. In this way, we can extract our code-words using the voxels having the most discriminative information.

- While constructing our model, instead of using raw voxel intensity values, we can use the Pearson correlation values. In this way, we can use the functional neighborhood in a more efficient way.

- In order to increase classification accuracy, a discriminative model representing the relationship among the voxels can be constructed.

- Also deep learning techniques can be integrated with the methods used in this thesis, as they are powerful tools used frequently in pattern recognition problems.

REFERENCES

[1] Activation maps. http://www2.fmrib.ox.ac.uk/education/fmri/introduction-to-fmri/activation-maps/. Accessed: 2017-08-20.

[2] The basics of mri. http://www.cis.rit.edu/htbooks/mri/. Accessed: 2017-08-20.

[3] Fmri functional magnetic resonance imaging lab. http://www.csulb.edu/~cwallis/482/fmri/fmri.html. Accessed: 2017-08-20.

[4] Importance of attributes (predictors) using relieff algorithm. https://www.mathworks.com/help/stats/relieff.html. Accessed: 2017-08-31.

[5] J. Ashburner, G. Barnes, C. Chen, J. Daunizeau, G. Flandin, K. Friston, D. Gitelman, S. Kiebel, J. Kilner, V. Litvak, et al. Spm8 manual. *Functional Imaging Laboratory, Institute of Neurology*, 41, 2008.

[6] J. Ashburner, K. J. Friston, et al. Spatial normalization using basis functions. *Human brain function*, 2, 2003.

[7] P. A. Bandettini, E. C. Wong, R. S. Hinks, R. S. Tikofsky, and J. S. Hyde. Time course epi of human brain function during task activation. *Magnetic resonance in medicine*, 25(2):390–397, 1992.

[8] M. K. Carroll. *fMRI "mind readers": sparsity, spatial structure, and reliability*. Princeton University, 2011.

[9] A. Coates, A. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223, 2011.

[10] A. Coates and A. Y. Ng. The importance of encoding versus training with sparse coding and vector quantization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 921–928, 2011.

[11] M. N. Coutanche, S. L. Thompson-Schill, and R. T. Schultz. Multi-voxel pattern analysis of fmri data predicts clinical symptom severity. *Neuroimage*, 57(1):113–123, 2011.

[12] D. D. Cox and R. L. Savoy. Functional magnetic resonance imaging (fmri)"brain reading": detecting and classifying distributed patterns of fmri activity in human visual cortex. *Neuroimage*, 19(2):261–270, 2003.

[13] R. C. Craddock, P. E. Holtzheimer, X. P. Hu, and H. S. Mayberg. Disease state prediction from resting state functional connectivity. *Magnetic resonance in Medicine*, 62(6):1619–1628, 2009.

[14] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004.

[15] C. Davatzikos, K. Ruparel, Y. Fan, D. Shen, M. Acharyya, J. Loughead, R. Gur, and D. D. Langleben. Classifying spatial patterns of brain activity with machine learning methods: application to lie detection. *Neuroimage*, 28(3):663–668, 2005.

[16] I. O. Ertugrul, M. Ozay, and F. T. Y. Vural. Encoding the local connectivity patterns of fmri for cognitive state classification. *arXiv preprint arXiv:1610.05036*, 2016.

[17] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 524–531. IEEE, 2005.

[18] J. V. Haxby, M. I. Gobbini, M. L. Furey, A. Ishai, J. L. Schouten, and P. Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

[19] X. Hu and E. Yacoub. The story of the initial dip in fmri. *Neuroimage*, 62(2):1103–1108, 2012.

[20] S. A. Huettel, A. W. Song, and G. McCarthy. *Functional magnetic resonance imaging*, volume 1. Sinauer Associates Sunderland, 2004.

[21] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. *Machine learning: ECML-98*, pages 137–142, 1998.

[22] Y. Kamitani and F. Tong. Decoding the visual and subjective contents of the human brain. *Nature neuroscience*, 8(5):679–685, 2005.

[23] Y. Kamitani and F. Tong. Decoding the visual and subjective contents of the human brain. *Nature neuroscience*, 8(5):679–685, 2005.

[24] I. Kononenko, E. Šimec, and M. Robnik-Šikonja. Overcoming the myopia of inductive learning algorithms with relieff. *Applied Intelligence*, 7(1):39–55, 1997.

[25] K. K. Kwong, J. W. Belliveau, D. A. Chesler, I. E. Goldberg, R. M. Weisskoff, B. P. Poncelet, D. N. Kennedy, B. E. Hoppel, M. S. Cohen, and R. Turner. Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation. *Proceedings of the National Academy of Sciences*, 89(12):5675–5679, 1992.

[26] P. Lauterbur. Image formation by induced local interactions: examples employing nuclear magnetic resonance. 1973.

[27] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 2169–2178. IEEE, 2006.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[29] S. Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[30] T. M. Mitchell, R. Hutchinson, M. A. Just, R. S. Niculescu, F. Pereira, and X. Wang. Classifying instantaneous cognitive states from fmri data. In *AMIA annual symposium proceedings*, volume 2003, page 465. American Medical Informatics Association, 2003.

[31] T. M. Mitchell, R. Hutchinson, R. S. Niculescu, F. Pereira, X. Wang, M. Just, and S. Newman. Learning to decode cognitive states from brain images. *Machine learning*, 57(1):145–175, 2004.

[32] T. M. Mitchell, R. Hutchinson, R. S. Niculescu, F. Pereira, X. Wang, M. Just, and S. Newman. Learning to decode cognitive states from brain images. *Machine learning*, 57(1):145–175, 2004.

[33] E. Mızrak and I. Öztekin. Relationship between emotion and forgetting. *Emotion*, 16(1):33, 2016.

[34] B. Ng and R. Abugharbieh. Generalized group sparse classifiers with application in fmri brain decoding. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1065–1071. IEEE, 2011.

[35] B. Ng and R. Abugharbieh. Modeling spatiotemporal structure in fmri brain decoding using generalized sparse classifiers. In *IEEE International Workshop on Pattern Recognition in NeuroImaging, PRNI*, pages 65–68. IEEE, 2011.

[36] B. Ng, A. Vahdat, G. Hamarneh, and R. Abugharbieh. Generalized sparse classifiers for decoding cognitive states in fmri. In *International Workshop on Machine Learning in Medical Imaging*, pages 108–115. Springer, 2010.

[37] K. A. Norman, S. M. Polyn, G. J. Detre, and J. V. Haxby. Beyond mind-reading: multi-voxel pattern analysis of fmri data. *Trends in cognitive sciences*, 10(9):424–430, 2006.

[38] S. Ogawa and T.-M. Lee. Magnetic resonance imaging of blood vessels at high fields: in vivo and in vitro measurements and image simulation. *Magnetic resonance in medicine*, 16(1):9–18, 1990.

[39] S. Ogawa, T.-M. Lee, A. R. Kay, and D. W. Tank. Brain magnetic resonance imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences*, 87(24):9868–9872, 1990.

[40] S. Ogawa, T.-M. Lee, A. S. Nayak, and P. Glynn. Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields. *Magnetic resonance in medicine*, 14(1):68–78, 1990.

[41] S. Ogawa, D. W. Tank, R. Menon, J. M. Ellermann, S. G. Kim, H. Merkle, and K. Ugurbil. Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging. *Proceedings of the National Academy of Sciences*, 89(13):5951–5955, 1992.

[42] I. Onal, M. Ozay, E. Mizrak, I. Oztekin, and F. Yarman-Vural. A new representation of fmri signal by a set of local meshes for brain decoding. *IEEE Transactions on Signal and Information Processing over Networks*, 2017.

[43] L. Pauling and C. D. Coryell. The magnetic properties and structure of hemoglobin, oxyhemoglobin and carbonmonoxyhemoglobin. *Proceedings of the National Academy of Sciences*, 22(4):210–216, 1936.

[44] S. M. Polyn, V. S. Natu, J. D. Cohen, and K. A. Norman. Category-specific cortical activity precedes retrieval during memory search. *Science*, 310(5756):1963–1966, 2005.

[45] P. M. Rasmussen, L. K. Hansen, K. H. Madsen, N. W. Churchill, and S. C. Strother. Model sparsity and brain pattern interpretation of classification models in neuroimaging. *Pattern Recognition*, 45(6):2085–2100, 2012.

[46] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[47] H. Shen, L. Wang, Y. Liu, and D. Hu. Discriminative analysis of resting-state functional connectivity patterns of schizophrenia using low dimensional embedding of fmri. *Neuroimage*, 49(4):3110–3121, 2010.

[48] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision, ICCV*, page 1470. IEEE, 2003.

[49] B. Solmaz, S. Dey, A. R. Rao, and M. Shah. Adhd classification using bag of words approach on network features. In *Medical Imaging: Image Processing*, page 83144T, 2012.

[50] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

[51] R. Turner, D. L. Bihan, C. T. Moonen, D. Despres, and J. Frank. Echo-planar time course mri of cat brain oxygenation changes. *Magnetic Resonance in Medicine*, 22(1):159–166, 1991.

[52] S. Vega-Pons, P. Avesani, M. Andric, and U. Hasson. Classification of inter-subject fmri data based on graph kernels. In *IEEE International Workshop on Pattern Recognition in Neuroimaging, PRNI*, pages 1–4. IEEE, 2014.

[53] Z. Yang, F. Fang, and X. Weng. Recent developments in multivariate pattern analysis for functional mri. *Neuroscience bulletin*, 28(4):399–408, 2012.