Title
A Multilingual Exploration of Semantics in the Brain Using Tensor Decomposition

Author
Bardhan, Sharmistha

Publication Date
2018

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
RIVERSIDE

A Multilingual Exploration of Semantics in the Brain Using Tensor Decomposition

A Thesis submitted in partial satisfaction
of the requirements for the degree of

Master of Science

in

Computer Science

by

Sharmistha Bardhan

June 2018

Thesis Committee:

Dr. Evangelos E. Papalexakis, Chairperson
Dr. Vagelis Hristidis
Dr. Ahmed Eldawy

The Thesis of Sharmistha Bardhan is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I am grateful to my advisor Dr. Evangelos E. Papalexakis, for his constant support throughout this project. It is because of his patience and encouragement that, I could work on this project. I would also like to thank him for being my mentor and always guiding me whenever I needed.

I would also like to thank Dr. Brian Murphy for providing the data for this experiment. At the very beginning of this project, his advise helped me to get acquainted with the basics and he always provided important insights on the experimental results.

I want to thank Dr. Vagelis Hristidis and Dr. Ahmed Eldawy for being in my committee and giving me important feedback on the work.

I want to thank my family members and friends for their constant support and blessings that motivated me throughout my whole time at UCR.

To my parents, Nitya Nanda Bardhan and Rita Bardhan.

ABSTRACT OF THE THESIS

A Multilingual Exploration of Semantics in the Brain Using Tensor Decomposition

by

Sharmistha Bardhan

Master of Science, Graduate Program in Computer Science
University of California, Riverside, June 2018
Dr. Evangelos E. Papalexakis, Chairperson

The semantic concept processing mechanism of the brain shows that different neural activity patterns occur for different semantic categories. Multivariate Pattern Analysis of the brain fMRI data shows promising results in identifying active brain regions for a specific semantic category. Unsupervised learning technique such as tensor decomposition discovers the hidden structure from the brain data and proved to be useful as well. However, the existing methods are used for analyzing data from subjects who speak in one language and do not consider the cultural effect on it. This thesis presents an exploratory analysis of the neuro-semantic problem in a new dimension. The brain fMRI tensors of subjects who speak in Chinese or Italian language are analyzed both individually and together to discover the hidden structure. The Chinese and Italian tensors are jointly analyzed by coupling them along the *stimuli object* mode to discover the cultural effect. Moreover, the joint analysis of semantic features and brain fMRI tensor using the Advanced Coupled Matrix Tensor Factorization (ACMTF) method finds latent variables that explain the correlation between them. The results of the joint analysis of the tensors support the preliminary pre-

dictive analysis and find meaningful clusters for the different categories of stimuli object. Moreover, for a rank 2 decomposition, the prediction of brain activation pattern given semantic features gives an accuracy of 71.43%. It is expected that, the proposed exploratory and predictive analysis will improve existing approaches of analyzing conceptual knowledge representation of brain and guide future research in this domain.

Contents

List of Figures **x**

List of Tables **xii**

1 Introduction **1**
- 1.1 Motivation . . . 3
- 1.2 Problem Statement . . . 4
- 1.3 Experimental Organization . . . 4

2 Preliminaries And Notations **6**
- 2.1 Introduction to Tensors . . . 6
  - 2.1.1 Definitions . . . 7
  - 2.1.2 Tensor products . . . 9
- 2.2 Tensor Decomposition Methods . . . 10
  - 2.2.1 Canonical Polyadic Decomposition (CPD) . . . 10
- 2.3 Joint Analysis using Tensors . . . 11
  - 2.3.1 Coupled Matrix Tensor Factorization (CMTF) . . . 12
  - 2.3.2 Advanced Coupled Matrix And Tensor Factorization (ACMTF) . . . 13
- 2.4 Human Brain . . . 14
  - 2.4.1 Definition and Preliminaries . . . 15
  - 2.4.2 Feature Selection . . . 16

3 Related Work **18**
- 3.1 Tensor decomposition and its application . . . 18
- 3.2 Tensor decomposition in Signal Processing and Machine Learning . . . 20
- 3.3 Brain data analysis . . . 22
  - 3.3.1 Brain data and Machine Learning . . . 22
  - 3.3.2 Application of Tensor in Brain data analysis . . . 24

4 Brain data Analysis and Tensor **28**
- 4.1 The Brain fMRI dataset . . . 28
  - 4.1.1 Dataset Properties . . . 29

4.1.2 Stimuli objects 32
4.2 Data Pre-processing 33

5 Preliminary Predictive Analysis using PyMVPA **36**
5.1 Multivariate Pattern Analysis using PyMVPA 36
5.1.1 Within Subject Classification 37
5.1.2 Between Subject Classification 40

6 Tensor Decomposition for Brain data Analysis **41**
6.1 Language specific and Merged tensor analysis 41
6.2 Joint analysis of language specific tensor 43
6.3 ACMTF for Joint analysis of tensor and semantic feature 44

7 Experimental Results **46**
7.1 Results from PyMVPA analysis 46
7.1.1 Within Subject Classification 46
7.1.2 Between Subject Classification 53
7.2 Results from Tensor analysis 54
7.2.1 Language specific and Merged tensor analysis 55
7.2.2 Joint analysis of language specific tensor 57
7.2.3 ACMTF for Joint analysis of tensor and semantic feature 58

8 Conclusion **61**

Bibliography **63**

List of Figures

2.1 The N-way arrays where N=1 indicates a vector, N=2 indicates matrix and N=3 indicates a tensor. 7
2.2 A three-mode tensor. 8
2.3 A three-mode tensor with rank R. 9
2.4 A three-mode tensor decomposition represented as factor matrices in each mode. 9
2.5 The Coupled Matrix Tensor Factorization (CMTF) model. 13
2.6 The different parts of the human brain [61]. 14

4.1 The change in dataset after applying the detrending and Z-scoring method for all subjects. The time series data of five voxels are presented to show the changes. 35

6.1 The graphical model representing the three different analysis performed on the dataset. 42
6.2 The graphical model representing the tensor analysis using PARAFAC decomposition. 42
6.3 The graphical model representing the tensor analysis in the joint analysis step. 43
6.4 The graphical model representing the tensor analysis in the joint analysis step using ACMTF. 44

7.1 The highly active voxels selected by the One Way Anova method. 49
7.2 Classification performance using One Way Anova Feature Selection method. The top 100, 500 and 1000 voxels are selected for the analysis. 51
7.3 Classification performance using Sum of Squares Feature Selection method. The top 100, 500 and 1000 voxels are selected for the analysis. 52
7.4 The highly active voxels selected by the One Way ANOVA method for the Between Subject Classification task. 54
7.5 The two components of Brain Voxel factor matrix obtained from the Chinese_Tensor decomposition. 55
7.6 The two components of Brain Voxel factor matrix obtained from the Italian_Tensor decomposition. 56

7.7 The two components of Brain Voxel factor matrix obtained from the Merged_Tensor decomposition. 57
7.8 The highly active voxels for each component of the joint tensor analysis. . . 59
7.9 The highly active voxels for each component of the ACMTF analysis. . . . 60

List of Tables

4.1 The properties of the brain fMRI dataset collected from different language speaking participant . . . 29
4.2 The list of stimuli objects . . . 33

7.1 Classification accuracy for Within Subject Classification and One Way ANOVA method with 100 features . . . 47
7.2 Classification accuracy for Within Subject Classification and One Way ANOVA method with 500 features . . . 47
7.3 Classification accuracy for Within Subject Classification and One Way ANOVA method with 1000 features . . . 48
7.4 Classification accuracy for Within Subject Classification and Sum of Squares method with 100 features . . . 48
7.5 Classification accuracy for Within Subject Classification and Sum of Squares with 500 features . . . 50
7.6 Classification accuracy for Within Subject Classification and Sum of Squares with 1000 features . . . 50
7.7 Classification accuracy for Between Subject Classification task . . . 54
7.8 The top 5 stimuli objects for Chinese_Tensor decomposition . . . 55
7.9 The top 5 stimuli objects for Italian_Tensor decomposition . . . 56
7.10 The top 5 stimuli objects for Merged_Tensor decomposition . . . 57
7.11 The top 5 stimuli objects for joint tensor analysis . . . 58
7.12 The top 5 stimuli objects for ACMTF analysis . . . 58
7.13 Accuracy for brain voxel prediction . . . 59

Chapter 1

Introduction

The human brain represents conceptual knowledge and processes it through activating certain neural regions. A number of different neuroscience based research has been conducted to determine how this representation is created in the brain. Moreover, it has been found that different semantic categories of objects show different neural activation. In other words, when a subject is viewing different objects, different spatial pattern of neural activation is observed. Though such behavior of the brain is already explained, the reason behind this is still unclear. An explanation towards why certain semantic concept activates specific brain regions can clarify this along with answering other research questions.

In this project, the conceptual knowledge or semantic concept processing mechanism of the brain is analyzed. When a human look at a certain object, the brain immediately sends signals from one region to another to understand and interpret the concept. Through these brain signals, the concept is presented in the brain. In the brain, there are a number of regions that are used for different concept processing task. Moreover, according to the weight and age of a person, the brain size and concept representation process varies. Therefore, it is a challenging task to provide a generalized decision on different experiments. However, different research studies show that, it is still possible to find a pattern of activation from the brain data through proper data pre-processing steps.

The mechanism of semantic concept processing is observed in the brain data that is acquired during a certain experimental session. The two most popular method for brain data acquisition is Functional Magnetic Resonance Imaging (fMRI) and Electroencephalogram (EEG). In this project, the brain fMRI data is used for different experiments. Since brain fMRI data shows the change in blood oxygenation level, it is easier to find the activation pattern from brain image. The brain fMRI images have been used for a number of different experiments. These experiments include finding brain network and its strength, detecting Alzheimer's disease, etc. Moreover, it is also the most commonly used imaging technique to understand the concept processing mechanism of the brain.

The semantic concept processing mechanism of the brain is analyzed with PyMVPA tool and tensor decomposition method to answer several research questions. In the beginning, brain fMRI data is analyzed using the PyMVPA tool. Then the learning obtained from this step is used in the next step where the tensor decomposition concept is used to find out latent structure in the data. In this project, there are two main goals. They are, answering questions like, is there any cultural effect on how people think? Also in the presence of different language speaking subject's data, is it possible to predict the brain voxels in the presence of additional information? In the following sections, the motivation and the approaches followed in this project will be discussed in a broad manner.

1.1 Motivation

In 2008, Tom Mitchell first proposed a computational model for the prediction of Human Brain Activity Associated with the Meanings of Nouns. In that work, they focused mainly on brain regions that interact differently for different categories of stimuli [67]. They considered the semantic feature of different words to predict the brain activity. In 2015, in a different experiment, the brain fMRI data is coupled with the personal images of the participant and other participants. This study shows that each person viewing certain objects, has a different way to represent them in the brain [26]. As discussed, tensors are also used for different problems related to brain starting from modeling epilepsy seizure to Brain network analysis. The Coupled Matrix Tensor Factorization (CMTF) couples the brain data with behavioral response gives better performance regarding finding latent variables that explain the activity of the brain signals more accurately. The Advanced CMTF (ACMTF) is also proved to be useful for such joint analysis, in the presence of both shared and unshared factors. However, all of these experimental results are obtained for participants who speak English.

In this thesis, we are trying to address this problem in a new dimension. Since the current analyses are applicable for English speaking people only, it is not possible to generalize the results for different language speaking people. There can be socio-cultural effect on how a person thinks and depending on that different activation pattern can be observed in the brain for different stimuli object. Moreover, when the dataset of different language speaking are jointly analyzed, is it possible to predict the brain voxels from semantic features? These are the questions that motivate us primarily to solve the problem. In this project, the dataset contains data from participants who speak in Chinese and Italian. The primary goal of this project is to answer two research questions. We want to answer questions like, is it possible to capture cultural differences using tensor decomposition on the data? Moreover, we jointly analyze the dataset together and found that such analysis helps us answer the questions better.

1.2 Problem Statement

In this project, the dataset contains brain fMRI data from Chinese and Italian language speaking subject. The primary goal of this thesis is to answer the two questions discussed above. Therefore, the problem can be defined in the following way,

Given, brain fMRI data X = {C, I} where C and I are tensors containing data for Chinese and Italian speaking people and Semantic Features Y = $\{s_1, s_2, \ldots, s_{218}\}$ for nouns, we aim to,

1. Predict the associated Brain voxels related to noun/stimuli.

2. Capture cultural differences.

1.3 Experimental Organization

In order to reach the final goal of this project, the brain fMRI data is collected from 7 subjects where 4 subjects speak in Chinese and 3 subjects speak in Italian language. Each subject was shown 84 different stimuli objects multiple times in different runs and the brain fMRI data is captured. As stimuli, tools, mammals, and different objects are considered. However, for analysis purpose, only the tools and mammals are considered. During the fMRI acquisition, the picture of the stimuli object was shown to the subjects. Since these experiments take a long time and the subject might get tired, standard procedure to capture information for each experimental run is considered with the necessary resting period between sessions. Each stimuli object is shown to the user 3 times at a stretch.

Since, brain fMRI data is prone to noise due to temporal drift, head motion, appropriate detrending and normalization method is applied to clean the data. The PyMVPA tool is used on the data for the preliminary predictive analysis. As prediction task, the type of object the user was viewing during the experiment is considered. Then the brain fMRI data is represented as tensor and different tensor decomposition model is considered to answer the research questions.

The thesis is organized in the following way: Chapter 1 discusses the problem statement and introduces the problem, Chapter 2 discusses technical concepts required to understand the thesis, Chapter 3 discusses related work, Chapter 4 discusses the problem and experimental setup, Chapter 5 presents the preliminary predictive analysis using PyMVPA tool, Chapter 6 presents the analysis using Tensor decomposition method, Chapter 7 presents the results and Chapter 8 presents the conclusion and future work.

Chapter 2

Preliminaries And Notations

In this chapter, we will discuss the common terms and notations used throughout this paper to understand the technical concepts associated with the main work. The different models and algorithms used in this work are also discussed.

2.1 Introduction to Tensors

A tensor is a multi-dimensional array. Tensors are also known as the N-way array. A vector is a one-way array and a matrix is a two-way array. Therefore, when the value of N is three or more, it is a tensor. Figure 2.1 illustrates the difference between vector, matrix, and tensor.

In case of tensors, there are certain concepts that need to be considered while performing a decomposition or any such operation on it. In this section, the different tensor products and operations are discussed.

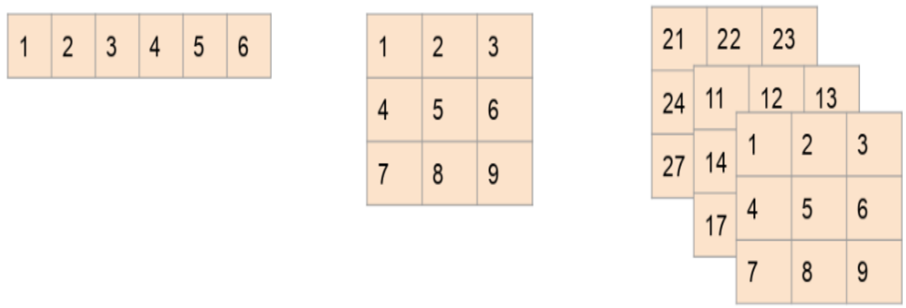


Figure 2.1: The N-way arrays where N=1 indicates a vector, N=2 indicates matrix and N=3 indicates a tensor.

2.1.1 Definitions

As discussed above, tensors are three or more dimensional arrays containing numeral values. Tensors are most commonly used to represent multi-aspect data. Starting from online social network's user-interaction data to the analysis of brain signal obtained in different trials, tensors have always been useful to present the multiple aspects of these huge datasets.

Tensor: A tensor is a multi-dimensional array with three or more modes. It contains numerical values that represent the relationship among multiple aspects. A tensor is usually represented as $\mathcal{X}$.

Tensor Order or Mode: The order of a tensor is its dimension. It is also known as *Mode*. Therefore, 1-dimensional array or vector is the first-order tensor, a 2-dimensional array is a second-order tensor, 3 or more dimensional array is a tensor. Figure 2.2 is showing a three-mode tensor.

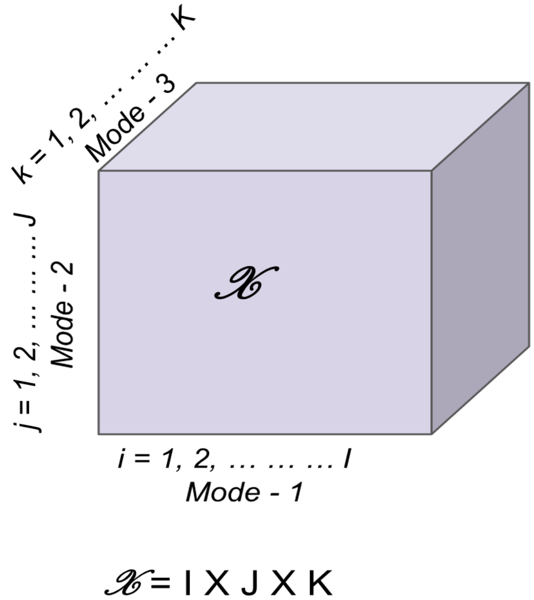


Figure 2.2: A three-mode tensor.

Tensor Rank: The rank of a tensor, $\mathcal{X}$ is the minimum number of rank-one tensors whose sum can form the tensor. The rank of a tensor is denoted by R. Figure 2.3 is showing a three-mode tensor with rank-R. In the figure, $a_1 \ldots a_R$, $b_1 \ldots b_R$ and $c_1 \ldots c_R$ are known as *Factors.* Each of these factors are rank-1 vectors.

The individual rank-1 vectors in each mode can be combined together to form a matrix where each column will hold one rank-one component. These matrices are called *Factor matrices.* Figure 2.4 is showing the same decomposition of Figure 2.3 in terms of factor matrices. In the figure, matrix $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are factor matrices for three different modes.

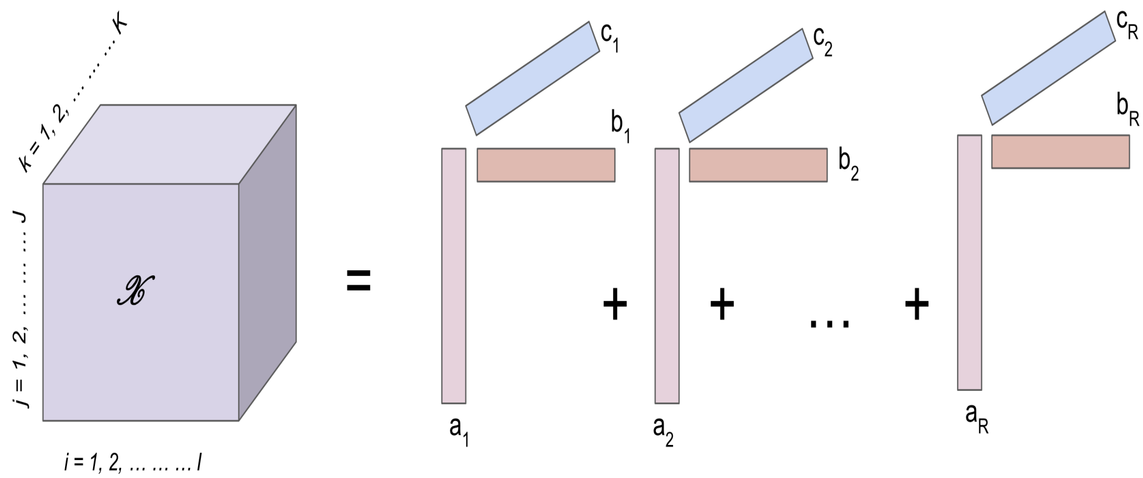


Figure 2.3: A three-mode tensor with rank R.

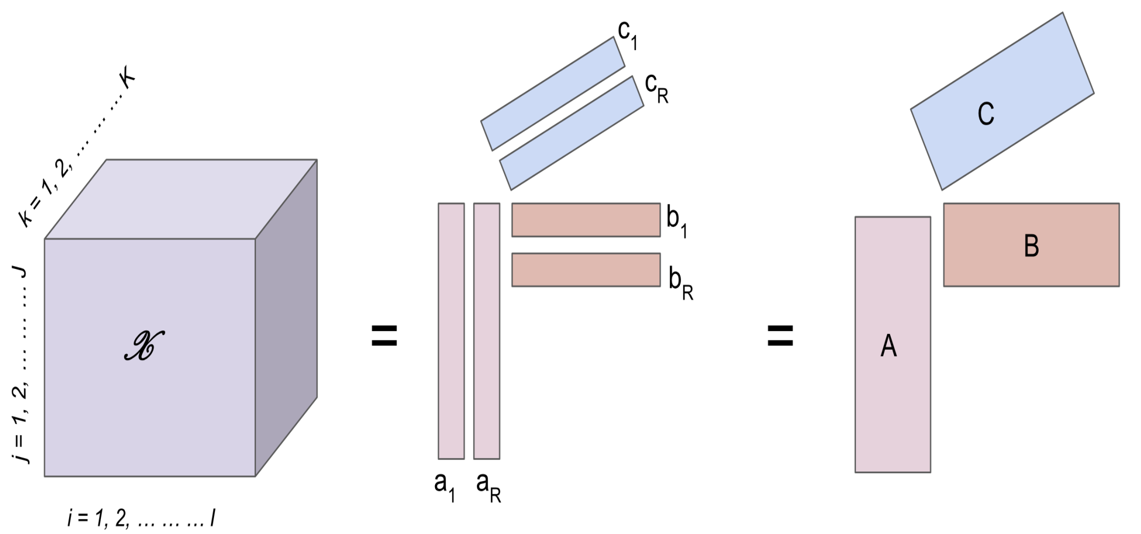


Figure 2.4: A three-mode tensor decomposition represented as factor matrices in each mode.

2.1.2 Tensor products

Kronecker Product: The Kronecker Product between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ with dimension $I\ X\ J$ and $K\ X\ L$ is defined as $A \otimes B$ with dimension $IK\ X\ JL$.

$$A \otimes B := \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}B & a_{I2}B & \cdots & a_{IJ}B \end{bmatrix}$$

Khatri-Rao Product: The Khatri-Rao Product between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ with dimension $I\ X\ K$ and $J\ X\ K$ is defined as $A \odot B$ with dimension $IJ\ X\ K$. Khatri-Rao Product is basically the column-wise Kronecker product.

$$A \odot B := \begin{bmatrix} a_1 \otimes b_1 & a_2 \otimes b_2 & \cdots & a_K \otimes b_K \end{bmatrix}$$

There are a few more tensor product methods available. They are *N-mode Product* and *Hadamard Product*.

2.2 Tensor Decomposition Methods

There are two major types of tensor decomposition. They are Canonical Polyadic Decomposition (CPD) and Tucker Decomposition.

2.2.1 Canonical Polyadic Decomposition (CPD)

The Canonical Polyadic Decomposition is used for *rank decomposition*. It is also known as ***CANDECOMP/PARAFAC*** decomposition. It decomposes a tensor into the sum of rank - 1 tensors. For example, let $\mathcal{X}$ is a three-mode tensor of dimension $I\ X\ J\ X\ K$, then the CP decomposition is computed using the following formula,

$$\mathcal{X} = \sum_{r=1}^{R} a_r \circ b_r \circ c_r$$

The tensor is basically decomposed as the sum of an outer product of three vectors. These three vectors are factors, where $a_r \in R^I$, $b_r \in R^J$ and $c_r \in R^K$. These factors are vectors that are merged together in each mode to form the factor matrices where $A \in R^{IXR}$, $B \in R^{JXR}$ and $C \in R^{KXR}$ where R is rank of the decomposition.

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_R \end{bmatrix}$$

$$B = \begin{bmatrix} b_1 & b_2 & \cdots & b_R \end{bmatrix}$$

$$C = \begin{bmatrix} c_1 & c_2 & \cdots & c_R \end{bmatrix}$$

There are a number of algorithms exist that can compute the CP decomposition. They are *Alternating Least Square (ALS)* algorithm, *Jennrichs algorithm*, *Tensor Power Method*, etc.

2.3 Joint Analysis using Tensors

The joint analysis of Tensor means coupling a tensor with one or more tensors or matrices along one or more modes and then jointly factorize them. This is also known as *Structured data fusion (SDF)*. When a number of dataset are analyzed together, it can identify the hidden structure of the data in a broad manner. This Structured Data Fusion (SDF) problem can be presented as a *Coupled Matrix Tensor Factorization (CMTF)* problem. Moreover, there is another variation of CMTF which is known as *Advanced CMTF*

(ACMTF). In this section, the CMTF and ACMTF model will be discussed.

2.3.1 Coupled Matrix Tensor Factorization (CMTF)

In CMTF, tensors and matrices are jointly analyzed. Suppose, $\mathcal{X} \in \mathbb{R}^{IXJXK}$ is a tensor with three modes and $Y \in \mathbb{R}^{IXL}$ is a matrix. If the given tensor and the matrix is coupled in the first mode then the common latent structure from both the dataset can be extracted. Therefore, the CMTF model for a rank-R decomposition can be presented as Equation 2.1.

$$f(A, B, C, D) = \|\mathcal{X} - [[A, B, C]]\|^2 + \|Y - AD^T\|^2 \tag{2.1}$$

Where $A \in \mathbb{R}^{IXR}$, $B \in \mathbb{R}^{JXR}$ and $C \in \mathbb{R}^{KXR}$ are factor matrices for $\mathcal{X}$. In this equation, the CP decomposition is used. Moreover, the $A \in \mathbb{R}^{IXR}$ and $D \in \mathbb{R}^{LXR}$ are the factor matrices obtained from the matrix factorization. Equation 2.1 can be generalized for any tensors and matrices. When multiple datasets are analyzed jointly this way, it is easier to capture underlying structure of the dataset through clusters. Figure 2.5 is showing the CMTF model.

There are a number of algorithms available to solve the CMTF for a variety of loss functions. The two most popular algorithms are based on *Alternating Least Square (ALS)* method and the Optimized version of it. In the $CMTF\_ALS$ algorithm, the general ALS method is used to compute the decomposition where the computation of one factor matrix is done by keeping the others fixed. The algorithm stops when there is no change in the loss function, in other words, it is minimized. The ALS method has certain limitations. If

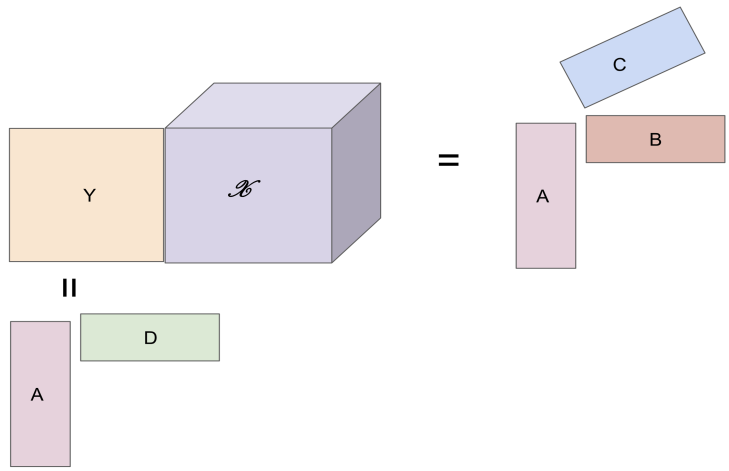


Figure 2.5: The Coupled Matrix Tensor Factorization (CMTF) model.

there is missing data, it converges poorly. Moreover, if the number of components or rank is not determined properly, it may not find the latent structure of the data.

In order to solve the above problems, the *CMTF_OPT* is proposed. In this paper, the *CMTF_OPT* algorithm is used. In this algorithm, the gradient is computed first. Afterward, a first-order optimization algorithm is used.

2.3.2 Advanced Coupled Matrix And Tensor Factorization (ACMTF)

The CMTF model works well when all of the factors are shared across the dataset. However, if there are both shared and unshared factors, it may not capture the underlying structure of the data. Therefore, the CMTF is reformulated in such a way so that the columns of the factor matrices have unit norm. The new version of the CMTF is known as *Advanced Coupled Matrix Tensor Factorization (ACMTF)* that can effectively identify shared and unshared factors.

2.4 Human Brain

The Human brain consists of four regions. They are *Frontal lobe*, *Parietal lobe*, *Occipital lobe* and *Temporal lobe*. These lobes have different functions and contribute equally to the overall brain functioning. Figure 2.6 is showing the different parts of the brain.

Frontal Lobe: This is the front part of the brain. It manages the attention, planning and reasoning activities.

Parietal Lobe: This is the middle portion of the brain. It manages the perception, arithmetic calculation activities.

Occipital Lobe: This resides at the back of the brain. It manages the vision.

Temporal Lobe: This resides at the bottom of the brain. It manages the language comprehension and semantic stimuli processing i.e. interpret the meaning of visual stimuli and establish object recognition. This region is basically where we are interested in for this experiment.

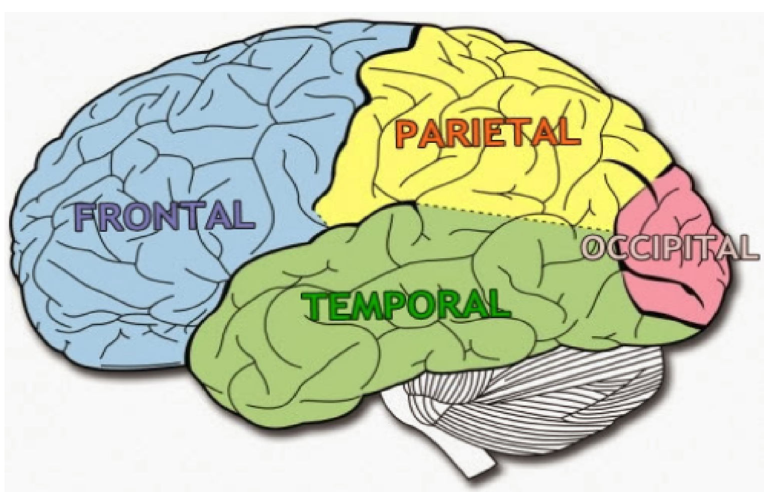


Figure 2.6: The different parts of the human brain [61].

2.4.1 Definition and Preliminaries

Blood Oxygen Level Dependent (BOLD): A particular brain region becomes active when it is in use. The neurons of that brain region need more energy to be active than others. The active neurons get this energy from blood since it provides more oxygen to these neurons. This process is known as the *Hemodynamic response.*

Functional magnetic resonance imaging (fMRI): fMRI uses *BOLD-contrast imaging* technique [108] to determine which regions of the brain are the most active. It measures the change in blood oxygen levels in the brain. A change in the blood oxygen level is observed in a particular region of the brain when that region is in use or active. In other words, the change in the blood oxygenation level actually indicates neural activity in the brain. The fMRI technique scans the brain and maps the neural activity by considering the change in the blood oxygenation level. Therefore, the fMRI scan is images of hemodynamic response, corresponding to the neural activity in the particular region of the brain.

A fMRI image is a 3D volume of the brain. In this image, each point is known as a *voxel.* If a voxel is active, it represents, that particular place in the brain is in use and vice versa. Moreover, the number of voxels are not fixed for all users. It depends on the size and shape of the brain of a particular person. In different studies that have used fMRI brain images to train machine learning models, considers voxels as features.

PyMVPA: PyMVPA stands for MultiVariate Pattern Analysis (MVPA) in Python [108]. PyMVPA is a python package that is used to analyze larger datasets. It is mostly used for brain fMRI dataset. However, it can be used for different types of dataset as well. In case of neural data analysis, the statistical learning methods of PyMVPA are useful.

Temporal Resolution (TR): Temporal resolution basically means the time between fMRI volume acquisition during an experiment. It is always recommended to have less than 2 seconds of TR. It is also known as *Time to Repetition*.

Insula: Insula is a region of the brain that resides at the very deep of the brain and in the cerebral cortex.

Echo-Planar Imaging (EPI): EPI is considered the standard method for collecting fMRI images. The basic way of collecting fMRI images is to go from top to bottom of the brain. The scanner goes from top to the bottom of the brain and collects the fMRI images slice by slice.

2.4.2 Feature Selection

There are different types of methods that are used for feature selection [63, 11, 114, 112, 78, 94, 109, 110]. In case of brain fMRI data analysis, different feature selection methods are used. The One Way ANOVA method is used for univariate analysis. The Search Light and Recursive Feature Elimination (RFE) are the most popular multivariate feature selection methods for brain fMRI images. In this work, the One Way ANOVA method is used for feature selection.

Analysis of Variance

ANOVA method is used to compute and test the means of several groups of features. It usually tests whether the difference between these means are equal or not across all the groups. The reasoning about the means is basically computed by analyzing the variance of the groups of features, justifying the name [78].

There are two major types of ANOVA test. They are, *One-way ANOVA* and *Two-way ANOVA*. The One-way or Two-way ANOVA tests are distinguished by the number of independent variables that are present in the analysis. One-way ANOVA has one independent variable with 2 different groups in it. On the other hand, Two-way ANOVA has two independent variables with 2 or more different groups in it [94].

Chapter 3

Related Work

Brain fMRI data has been used in different experiments to answer important research problems. Moreover, a number of important questions have been solved using fMRI data and tensor decomposition methods. In this chapter, a brief overview of the previous research works has been discussed.

3.1 Tensor decomposition and its application

Tensor and different tensor decomposition methods are now very popular in the field of data mining, machine learning, and signal processing. The tensor was first proposed in 1900. Gregorio Ricci-Curbastro and Tullio Levi-Civita are the two mathematicians who proposed this [116]. The tensor decomposition concepts are first introduced in 1927 by Frank L. Hitchcock [50, 89]. However, this introduction gives a high-level idea of tensor decomposition. Tensor decomposition method has great potential in modeling latent variables and it is first introduced by Cattell in Psychometrics [25]. In this study, linear tensor decomposition is discussed which is basically the decomposition of tensor in rank-1 factors. As a result of the first work, tensor decomposition was also used in chemometrics and econometrics. Then in 1960, the concept of multi-linear tensor decomposition is introduced by Tucker and Levin [98, 97, 60].

In 1970, Carroll and Chang introduced CANDECOMP model [24]. This work proposes an approximate linear rank model for specifically three-mode tensors. At the same time, Richard A.Harshman proposed another approximate linear rank model, PARAFAC [24]. Though both of these research studies are based on Catell's principle, they are separate work. In 1980's Kroonenberg proposed Tucker3, Tucker2 and 3-mode PCA. He discussed these methods as three-mode Principal Component Analysis (PCA) dimensionality reduction methods [99, 57, 58]. He used Alternating Least Square (ALS) approach for this work. In 1996 Rasmus Bro published a research work where he discussed different applications of PARAFAC in chemometrics where he solved problems like missing data, variance analysis, etc [20]. In 2000, Lathauwer, Moor, and Vandewalle proposed Higher-Order SVD (HOSVD) [34]. This work basically generalizes the SVD concept of matrix decomposition to tensor or higher-order matrices. The same year, these three researchers came up with Higher-Order Orthogonal Iteration (HOOI) [35]. This work discusses about computing the best approximation of a tensor given another tensor whose column and row rank value is known.

At the beginning of 2000, tensor and different tensor decomposition methods came to researcher's attention. There were several works that have been done since then, for example, new tensor decomposition methods [106, 119], optimization, improvements or extension of the existing methods [88, 83, 3, 120, 73], assessing the quality of the decomposition [21], etc. Moreover, there are several works that are discussing the application of these methods for solving different problems, for example, face recognition [105, 104], Video compression [91], brain data analysis [29]. In the beginning, the tensor decomposition methods were used only in Chemometrics and Psychometrics. However, tensor decomposition methods are now used in different fields like data science, machine learning, Neuroscience, and statistics.

3.2 Tensor decomposition in Signal Processing and Machine Learning

In the year 2000 and later, there has been a lot of research work happened in signal processing and machine learning involving tensor decomposition. There are research works where a combination of DS-CDMA signals was separated without any training using PARAFAC model [32]. When using only one antenna, the signal is always corrupted by noise. However, in all of the practical designs, we have to use several antennas and for such array of antennas coupled tensor methods can be used [96]. There are several works where tensor is used for multi-array blind source separation problems [62, 28]. Moreover, tensors have been used in signal processing for Multiple Input and Multiple Output (MIMO) radar as well [75]. In signal processing, a tensor is used for blind source separation problems of speech and audio signals [76]. The PARAFAC model is used for sensor array processing problem as well [92].

Tensor decomposition has become a very popular method for solving different machine learning problems. In case of social media data or recommender system, tensor used to model the data. In case of social media, it tries to find out the underlying pattern of the data that detects a community depending on various factors [14, 117, 80]. Moreover, for the nature of social media and user's interaction, data coming is in the form of a stream. In that case, computing the whole tensor over again is expensive and inefficient. This problem is also solved using dynamic tensor analysis methods [95]. In case of a recommender system, evaluating user's selection, finding any anomalies tensor decomposition methods have been used [79, 37]. In case of social network data and computer vision, a common issue is missing data. A modified version of the CP decomposition method is CP Weighted OPTimization, that takes the weighted least squares for the decomposition is used to solve the problem [4]. This problem is addressed and analyzed in different ways as well [52].

Since one of the most important applications of tensor is modeling latent variables, it has been used for problems in topic modeling [13], Independent component analysis (ICA) [16], discovering patterns, etc. It has also been used in Hidden Markov Models as well [70]. Tensor decomposition methods are used for identifying the latent variables in different fields such as data mining [68], big data analysis while dealing with the curse of dimensionality [107], etc. There are research works that focus on tensor's immense capacity to model latent variables and its application in the field of machine learning [12].

There are several studies that discussed the application of tensor decomposition for different problems of signal processing and machine learning [28, 55, 93, 33]. There are a number of surveys where different tensor decomposition methods are discussed [30, 56, 31].

3.3 Brain data analysis

There are a number of research work where brain data has been analyzed. fMRI and EEG are the two major type of brain data that is used to analyze brain network, human emotional state, and diseases. In case of brain fMRI data analysis, MultiVariate Pattern Analysis (MVPA) was proved to be effective. The reason behind this is, it captures different patterns in the fMRI data depending on different object category shown to the participant during the recording. James V. Haxby first introduced MVPA analysis for brain fMRI data [48, 46] and showed that in Ventral Temporal Cortex, the representation of different object categories shows different patterns like distributed, overlapping, etc. Then MVPA analysis is used for different problems, for example, understanding and decoding the neural activity related to visual object categories, memory search, line orientation, etc. [77, 47]. Brain data analysis is an active research topic and different machine learning algorithms, tensor decomposition methods are introduced and used in this field. In all of these research studies, the MVPA method is used.

3.3.1 Brain data and Machine Learning

Machine learning algorithms have been used for solving different problems using brain data [39, 118, 59]. Both EEG and fMRI data have been used for these experiments. Machine learning algorithms have been used to classify emotional states of human subjects where they have used the EEG brain data [18]. Then SVM is used as a classifier to classify happy and unhappy emotions from Brain EEG signal and an average accuracy of 75.62% and 65.12% have been obtained for subject dependent and independent model respectively [53].

Machine algorithms have also been used for mental-state monitoring systems as well where for feature extraction is performed using spatial filters and a framework of regularized linear discriminant analysis (LDA) is used for classification [71]. Moreover, a neural-weighting approach has been applied to guide machine learning algorithms [38]. A high-dimensional model representing the ventral temporal (VT) cortex of the brain shows promising results in common model space for the Between-Subject Classification problem. In this problem, SVM is used for classification [49]. Particularly, using fMRI brain images, there have been a number of research works that discuss how different concepts are processed in the brain [41].

When the brain data is correlated with behavioral data about specific nouns, it gives better insight about the brain signals. The first such experiment was performed by building a computational model using 11 participants' fMRI data and 60 nouns that show a promising result in the prediction of brain voxels from semantic features of noun [67]. The dataset is also available online [66]. Usually, in such experiments, the semantic features are collected from the participants by asking some questions. These semantic features can be obtained automatically from a text as well [85]. However, Wikipedia has also been used to generate such semantic features in different experiments [86]. There has been a research study where the brain fMRI data is coupled with the personal images of the participant and other participants. This study shows that each person viewing certain objects, has a different way to represent them in the brain [26]. Moreover, a framework for multi-variate pattern analysis on the fMRI brain data and stimuli supports the theory of unique representation of brain through dissimilarity matrices [27]. In such experiments, multivariate classifiers like Naive Bayes, SVM, and KNN have been used and Naive Bayes outperforms the other two methods. Moreover, it shows that from the activation pattern of the brain, it is possible to find out the category of the object shown to the participant [17].

There are tools available for such analysis. EEGLAB is used for such analysis where mostly EEG data is used [101]. ERPWAVELAB is another such tool that performs PARAFAC decomposition specifically on EEG and MEG data [69]. Moreover, there are a number of tools that are developed for the statistical analysis of the brain fMRI data, for examples, PyMVPA [42, 43, 44]. PyMVPA is a toolbox written in python that is used for Multivariate Pattern Analysis using brain fMRI data. Analysis of Functional NeuroImages (AFNI) [74] and Statistical Parametric Mapping (SPM) [100] are tools used for fMRI analysis as well.

3.3.2 Application of Tensor in Brain data analysis

Tensor decomposition has already been used for brain data analysis in problems including detecting diseases, analyzing brain network, etc. PARAFAC model is used for multi-way analysis of EEG brain data of patients with epilepsy seizure where it successfully modeled the seizure [1]. Brain network analysis is an active research problem that is important to find out the temporal and spectral connection between brain regions [102]. Tensors are highly used for such representation and analysis. SemiBAT is a semi-supervised approach that is based on constrained tensor factorization method [23]. Tensor-based Brain Network Embedding (t-BNE) is another such approach for brain network analysis problems [22]. Moreover, it is also possible to identify the states of the dynamic functional brain network through tensor decomposition from EEG brain data [64].

As discussed before, the brain fMRI data is correlated with behavioral data about specific nouns which are also known as the semantic features, coupling the brain data and behavioral response gives better performance regarding finding latent variables that explain the activity of the brain signals more accurately [9]. Coupled Matrix Tensor Factorization (CMTF) is used to analyze such dataset together [5, 6, 2, 40]. Moreover, CMTF is also used for problems where the factors are partially shared [36]. However, there can be unshared factors in the dataset, Advanced CMTF (ACMTF) has also been used for such experiments and it captures the underlying structures better than other methods [7]. Relaxed ACMTF (RACMTF) is another method that is used with EEG recordings to detect eye movements that are associated with the experimental data [90]. The general CMTF solvers are Alternating Least Square (ALS) based methods which are slower while applying on a large dataset. SCOUP-SMT is a new CMTF solver that is 50-100 times faster because of running in parallel. Given the semantic features, this algorithm was successful in identifying the activity in premotor cortex [84]. Turbo-SMT is another algorithm that solved the CMTF specifically for Sparse Matrix and Tensors with significantly good speed [81].

The fusion of both EEG and fMRI brain data can give a better spatio-temporal overview of the brain recordings of the epilepsy patients. Joint Independent Component Analysis (Joint-ICA) and Coupled Matrix Tensor Factorization have been used for such experiments where the fusion of the EEG and fMRI data were achieved using joint Blind Source Separation (BSS) process [51]. CMTF has also been used for analyzing the fusion of EEG and fMRI data to detect the neurological changes in the brain of patients with Schizophrenia [8]. A general framework has also been proposed for the fusion of multimodal brain data where the CMTF like tensor structures are used to capture underlying structure of the data [54]. There are more applications of tensors for such analysis [15, 103].

Brain fMRI data has been analyzed using different classifiers to see which machine learning algorithm works better with them. These studies show that, in such experiments, the brain voxels are directly considered as features to train the model and the class label is the stimulus object that a participant was looking at [87]. However, it is also possible to consider the average of a number of voxels that resides in a single ROI as a single voxel. We can also consider the voxel at each time point in a trial as a single feature [65]. Moreover, in case of class label also, there are exceptions depending on the requirement of the research study. It is also a common practice to ask participants questions regarding the shown stimuli for example, in one of such experiments the subject was shown an apple and were asked questions like, "Is it edible?" or "Does it fit in hand?", etc. [82].

The size of the dataset plays an important role on the selection of a classifier [87]. In case of fMRI data analysis, it is a common scenario that data will be noisy and it is necessary to remove the noise from the data. While removing the noise, the size of the dataset might reduce. Though having many examples might help training the model, it is better to use few examples that does not include any noise to train the model. In brain data analysis, noise is a common issue and always misleads the analysis. Therefore, the selection of the good examples is important. Moreover, in case of brain data, it is often the case that there are a few examples, but thousands of voxels or features. In that case, the trained model might suffer from overfitting problem. The research studies also proposed linear classifiers to solve their problem. However, it is not always true for all experiments.

Therefore, the choice of the classifier depends primarily on the size of the dataset and research problem.

In case of brain fMRI data, it is always a common practice to preprocess the data. The main reason behind this is, brain fMRI data contains noise. This noise is induced by different sources such as head motion, eye-blinks, etc. Moreover, as discussed above, it is also necessary to reduce the number of features if the ratio between the number of features and number of samples is really high. In data preprocessing step, different approaches have been followed. In some approaches, a General Linear Model (GLM) is used to capture the variance in the data and in other approaches, an Independent Component Analysis (ICA) method is used in the preprocessing step [19]. The fMRI data is first converted to examples where the definition of examples varies depending on the goal of the experiment. For example, the linear model activity estimate images using a GLM are created first and then the examples are created from the activation signals from these images. In other experiments, the example is generated as the average of several TRs of images in a single trial [87]. In some of the works, the number of features is reduced using masking technique [87] (considering only the region of interest) and SVD or PCA technique.

Chapter 4

Brain data Analysis and Tensor

Tensor analysis has been used for different problems involving brain fMRI data. In this thesis, a new problem is studied and analyzed using tensor decomposition and Coupled Matrix Tensor Factorization (CMTF). In this chapter, the experimental setup and the different features of the acquired brain fMRI data will be discussed. Moreover, the data pre-processing steps will be discussed as well.

4.1 The Brain fMRI dataset

In this experiment, the brain fMRI data has been collected from 7 participants where 4 of them speaks in Chinese and the rest of them speaks in Italian. Brain fMRI needs to be pre-processed and remove the noise before training the model. Moreover, these brain data come with additional information that needs to be considered to find hidden structure from them. In this section, the various features and attributes of the acquired brain fMRI data are discussed.

4.1.1 Dataset Properties

The brain fMRI data collected from different participants have a different number of samples and features. However, all of the participants viewed the same 84 stimuli objects during the experiment. Since the human brain size varies for different individuals according to their age and weight [111], the brain fMRI data obtained from different language speaking people also contains a different number of voxels or features. Table 4.1 lists the number of samples and features for each participant.

Table 4.1: The properties of the brain fMRI dataset collected from different language speaking participant

Participant ID	**Number of samples**	**Number of features**
Chinese_Subject_1	3681	286850
Chinese_Subject_2	3158	291910
Chinese_Subject_3	3681	291570
Chinese_Subject_4	4141	295590
Italian_Subject_1	3681	288670
Italian_Subject_2	3681	294690
Italian_Subject_3	3681	261710

The collected brain fMRI data actually shows the original mapping of the brain. However, for this experiment, the analysis of the whole brain is not necessary. According to the goal of this project, it is more interesting to study only those portions of the brain that deals with visual memory or language comprehension. As discussed previously, *temporal lobe* is the brain region that interprets visual objects and tries to recognize them. More specifically, the ventral temporal cortex is assigned for such object identification job in the brain. Therefore, in order to utilize only the useful voxels, a brain mask is applied to the raw data so that only the voxels of the ventral temporal cortex get selected for further analysis. There are total 379360 voxels correspond to the ventral temporal cortex of the brain. Though all of the individual datasets have their own mask file, in order to generalize the analysis across all dataset, a common mask file has also been used. This mask file is created by performing a logical OR among all of the seven mask files. This mask file is created using the SPM8 software.

In different experiments, it is required to know more information about the brain fMRI data. Therefore, all brain fMRI dataset usually contain three different types of attributes. They are *Sample Attribute*, *Feature Attribute*, and *Dataset Attribute*. The attribute values are stored as vectors. The dataset that has been used in this experiment contains a different number of sample, feature and dataset attributes.

- **Sample Attributes**: In case of sample attributes, there is one attribute value per sample. The sample attributes are stored as *Collectable* and do not allow inappropriate attributes. There are total 9 sample attributes in the dataset. They are,

  1. ***time_indices***: It stores the index of the volume.
  2. ***targets***: It stores the class label or stimuli value for each sample.
  3. ***chunks***: It stores information about independent groups of samples or corresponding experimental run. Each dataset contains samples that are obtained from a different experimental run. This attribute contains the id of the experimental run. There are total 9 chunks in the dataset. The last chunk contains only one sample with target value ***ignore*** which usually indicates the end of the experiment.
  4. ***time_coords***: It contains information about volume acquisition time.

5. ***duration***: It stores how long the stimuli were shown to the participant while collecting the brain fMRI data during a particular experimental run. The time is stored in seconds.

There are a few more attributes, they are, ***event_chunks***, ***onset***, ***event_targets***, and ***event_onsetidx***.

- **Feature Attributes**: There is one attribute value per feature and they are known as feature attributes. There are 2 feature attributes, ***event_offsetidx***, ***voxel_indices***. The ***voxel_indices*** feature attribute contains information about where the voxels actually reside in the brain. This is important to again create and plot the brain fMRI data. The ***voxel_indices*** are similar for all users after using the same mask.

- **Dataset Attributes**: The datatset attributes are, ***imgaffine***, ***imghdr***, ***imgtype***, ***mapper***, ***voxel_dim*** and ***voxel_eldim***.

1. ***mapper***: Mapper is one of the most important components of PyMVPA. Mappers actually transform the data in different ways depending on the requirement of the experiment. For example, in order to apply different algorithms, the 3D brain data needs to be flattened in a 1D feature vector. A mapper can be used for this purpose. Therefore, *mapper* stores what data pre-processing method has been applied to the data to transform it.

2. ***voxel_eldim***: It stores the size or dimension of 1 voxel in millimeter. In this experiment, *voxel_eldim* across all voxels were 3 X 3 X 6.

3. ***imghdr***: It stores information about the fMRI image or NIfTI header.

4. ***imgtype***: It stores information about the class of the image. The brain data used in this experiment is of *Nifti1Image* type.

5. ***voxel_dim***: The brain voxels are of 3 dimension. ***voxel_dim*** stores information about the size of the 3D voxels in each dimension as a vector. Therefore, the total number of voxels per volume can be computed from here. In this experiment, the total number of voxels are 53 X 63 X 23.

There are a few more dataset attributes like, ***imgaffine***, etc.

In PyMVPA, when a specific portion of the brain is selected for certain analysis using the masking concept, the corresponding sample, feature and dataset attributes are extracted automatically. Each of the sequential brain samples is collected in 1s TR.

4.1.2 Stimuli objects

The stimuli objects used in this experiments are from 3 different categories. They are Mammals, Tools, and Objects. There are 30 different Mammals and Tools. The number of objects used is 24. All of the 7 participants were shown this stimuli objects. Table 4.2 lists the target stimuli objects used in this experiment. The stimuli objects were shown to the participants on a regular interval. In between each stimuli viewing task, there were ***rest*** or ***base***, so that the participants get time to reset their brain and concentrate on the next stimuli without any impact of the previous stimuli. In the end there was ***ignore*** state.

Table 4.2: The list of stimuli objects

Worktool	**Land Mammal**	**Object**
worktool-paint_brush_big	vanillaLandMammal-hippopotamus	margins-banana
worktool-screw_driver	vanillaLandMammal-camel	margins-tiger
worktool-garden_trowel	vanillaLandMammal-monkey_unid2	margins-shell
worktool-axe	vanillaLandMammal-squirrel	margins-mountain
worktool-pneumatic_drill	vanillaLandMammal-mouse	margins-frisbee
worktool-nail	vanillaLandMammal-koala	margins-building
worktool-chain_saw	vanillaLandMammal-mole	margins-helicopter
worktool-hammer	vanillaLandMammal-chamois	margins-boulder
worktool-crow_bar	vanillaLandMammal-skunk	margins-scorpion
worktool-scissors	vanillaLandMammal-elephant	margins-puppy
worktool-screw	vanillaLandMammal-kangaroo	margins-mobiletelephone
worktool-spanner	vanillaLandMammal-llama	margins-skateboard
worktool-plaster_trowel	vanillaLandMammal-ibex	margins-hamster
worktool-paint_roller	vanillaLandMammal-hare	margins-rose
worktool-saw	vanillaLandMammal-gorilla	margins-cactus
worktool-scraper	vanillaLandMammal-beaver	margins-computer
worktool-power_drill	vanillaLandMammal-deer	margins-boat
worktool-pen_knife	vanillaLandMammal-hedgehog	margins-robot
worktool-hack_saw	vanillaLandMammal-badger	margins-table
worktool-rubber_mallet	vanillaLandMammal-zebra	margins-tree
worktool-rake	vanillaLandMammal-bison	margins-pebbles
worktool-plunger	vanillaLandMammal-giraffe	margins-handbag
worktool-pliers	vanillaLandMammal-ant_eater	margins-hail
worktool-garden_fork	vanillaLandMammal-fox	margins-cloud
worktool-pick_axe	vanillaLandMammal-panda	
worktool-tape_measure	vanillaLandMammal-armadillo	
worktool-craft_knife	vanillaLandMammal-boar	
worktool-allen_key	vanillaLandMammal-otter	
worktool-file	vanillaLandMammal-rhinoceros	
worktool-sickle	vanillaLandMammal-chimpanzee	

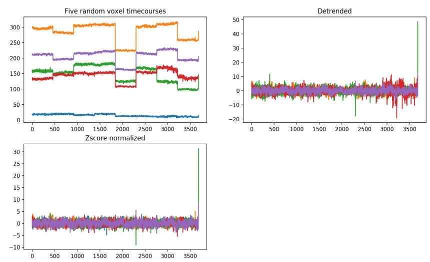
4.2 Data Pre-processing

When the brain fMRI data specific to the ventral temporal cortex has been masked out, next step is to pre-process the data to remove noise. In fMRI brain data, noise can be induced in different ways. For example, when the fMRI scanner gets warmer because of operating for a long time, temporal drifts occur. Moreover, eye-blink, head-motion can also affect the data. These sources induce variance that is unnecessary and misleads the analysis. There are several ways to remove the noise from the data. Since the data collected is already motion-corrected, pre-processing like removing temporal trend and normalization is required to apply. Therefore, the detrending and normalization methods are applied.

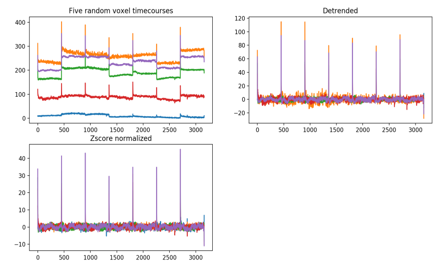
- **Detrending**: The detrending method is used to remove polynomial trends or temporal drift from the data. In this experiment, the linear detrending method is used. In linear detrending, linear regression is used to fit a straight line to each voxel's time series. The residuals obtained this way is considered as new feature values. The linear detrending method is performed in a chunk-wise manner.

- **Normalization**: In order to remove inhomogeneous voxel intensities, a chunk-wise Z-scoring is applied to the data. It scales all the features into approximately the same range. It also removes the mean.

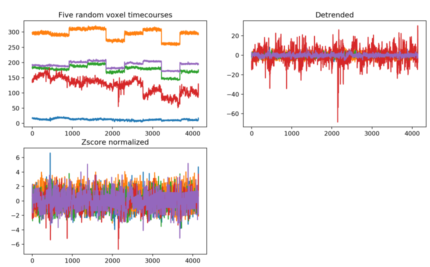
In this experiment, after applying the detrending and Z-scoring method, the data is fed into different machine learning algorithm. Figure 4.1 is showing the changes in the real dataset after applying the detrending and then the Z-scoring method. From the entire dataset, the time series data of five voxels are plotted.



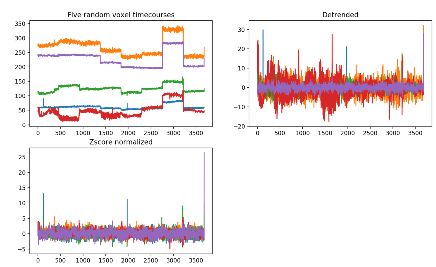
(a) Chinese Dataset 1



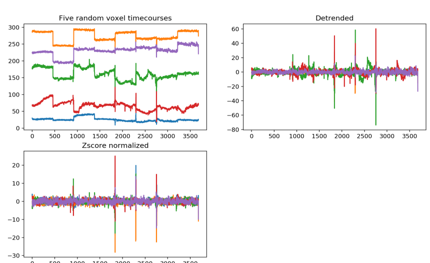
(b) Chinese Dataset 2



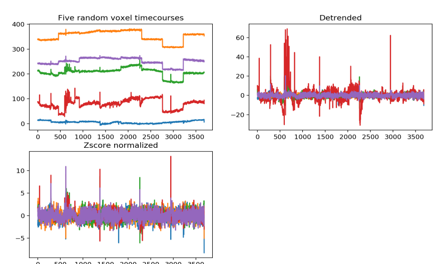
(c) Chinese Dataset 3



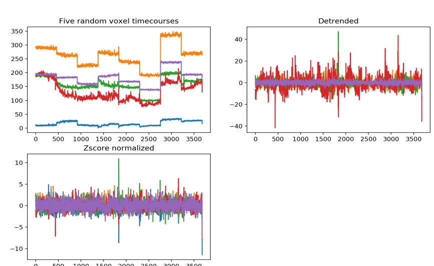
(d) Chinese Dataset 4



(e) Italian Dataset 1



(f) Italian Dataset 2



(g) Italian Dataset 3

Figure 4.1: The change in dataset after applying the detrending and Z-scoring method for all subjects. The time series data of five voxels are presented to show the changes.

Chapter 5

Preliminary Predictive Analysis using PyMVPA

The brain fMRI data has been analyzed by two different methods. They are PyMVPA and Tensor decomposition. In this section, the techniques applied to the data using PyMVPA will be discussed.

5.1 Multivariate Pattern Analysis using PyMVPA

In PyMVPA, two different Machine Learning analysis has been applied to the data. They are, *Within Subject Classification* and *Between Subject Classification.* For all of the following analysis, the clean and masked data has been used.

5.1.1 Within Subject Classification

In the Within Subject classification process, each of the 7 participants is considered individually and the classification task is to determine whether the participants are looking at a worktool or land mammals. Since there are 8 chunks with appropriate samples, an eight-fold cross-validation is used for each individual dataset. For example, the *Chinese_Subject_1* dataset contains 460 samples in each of the eight-chunks. Therefore, in each of the eight-fold cross-validation steps, one whole chunk is used for testing and the remaining seven chunks are used for training. This way, all of the chunks are tested once. Each of the stimulus objects is shown 12 times across 4 different chunks in each participant's session.

In order to select the important features for training, the *SensitivityBasedFeatureSelection* method of PyMVPA has been used. This method selects the important features depending on the given parameters. It computes a sensitivity map for the features and only the important features depending on the map are selected. There are two different methods that have been used to select the important features. For both of these methods, top 100, 500 and 1000 features are considered for training. For all of these datasets, their own mask is used.

- **ANOVA method**: The *FeaturewiseMeasure* has been computed using the ANOVA method and generate the sensitivity map. The F-scores are within the range of *0* to *Infinity*, where a high value indicates high activation.
- **Sum of Squares Method**: In this method, at first a row-wise sum of squares has been computed for each column or voxel. In other words, for one feature/voxel, the square of all the values of samples are summed up. Then, the features that have high

value are selected as important features. Since particular voxels get activated and participate for object recognition, they are supposed to be activated across all stimuli objects. Therefore, this method will help to identify those voxels.

The classification results are computed as the mean of correct prediction for each chunk. For example, for *Chinese_Subject_1*, one chunk contains 460 samples and the predictions are made for each of these samples. Then a mean is computed that represents, on average how many correct predictions were made i.e. how many semantic categories were correctly identified. Then for all the 8 chunks, the 8 classification accuracies are considered and the mean of all the intermediate results are computed for the overall accuracy measure.

Since in fMRI brain data, there are usually a small number of samples and thousands of features, the choice of classification algorithm should take the situation of overfitting into account. Therefore, three classification algorithms are used for classification. They are *Penalized Logistic Regression (PLR)*, *K-Nearest Neighbor (KNN)* and *Support Vector Machine (SVM)*.

1. **Penalized Logistic Regression (PLR)**: In PLR, it performs the classification task based on a logistic function [113]. The model is trained to predict the category of the target and returns the prediction as a binary value.

   The general logistic regression is regularized by penalizing the likelihood. The performance of prediction is improved by penalizing the classification that resembles overfitting attributes. Equation 5.1 is showing the penalized log-likelihood equation.

$$logL(\beta) = logL(\beta) - \frac{\lambda}{2}J(\beta) \tag{5.1}$$

In the above equation $J(\beta)$ is the ridge penalty where, $J(\beta) = \|\beta\|^2 = \sum_{j=1}^{n} \beta_j^2$ and β is the regression co-efficient.

2. **K-Nearest Neighbor (KNN)**: The KNN takes classification decision based on the distance measure of the one test sample and all the training samples at a time. When the distance is computed, depending on the K-value, the top K closest samples are chosen and the test sample is classified as the label of the samples that got the highest number of votes [72].

   In this work, the distance was measured between different samples. The activation of brain voxels is considered for the computation. The Euclidean distance measure is used for computing the distance. Equation 5.2 is showing the equation for computing euclidean distance between two samples x and y where each sample has k number of voxels.

$$EuclideanDistance = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{5.2}$$

3. **Support Vector Machine (SVM)**: The SVM builds a hyperplane that separates the data points of two different classes. The best hyperplane is chosen in a way so that it stays in the farthest position from the closest data-points of each of the classes [115].

   In this work, the C parameter is used in the Linear-CSVM version of the SVM algorithm. With linear kernel, higher C indicates stricter margin in SVM. This parameter plays an important role between the margin and support vectors.

The main reason for choosing these three classifiers is, they can appropriately deal with brain fMRI data where the data can be noisy. Moreover, such dataset also has few samples and a large number of features. For such dataset, these classifiers can identify the patterns of activation and then analyze it appropriately.

5.1.2 Between Subject Classification

The Between Subject Classification basically means that the dataset of all 7 users will be considered at a time i.e. the 7 datasets will be stacked up as a single dataset. Then the entire dataset for one user will be considered at a time as test dataset and the remaining dataset will be used for training, since the dataset will be partitioned depending on the number of users. In this way, the dataset for each subject will be considered once for testing. The classification task is to again predict the class label of unseen data as Worktool or Land mammals.

Since, in the *Within Subject Classification* portion, the *Sum of Squares* feature selection method did not work well, the ANOVA based feature selection method has been applied with the same classification algorithms. Moreover, for this experiment also top 100, 500 and 1000 voxels are selected for training. In this part of the project, a common mask has been used which is a logical OR of all seven masks.

Chapter 6

Tensor Decomposition for Brain data Analysis

In this section, the tensor decomposition methods used to analyze the data will be discussed in detail. Three different analysis has been performed. Figure 6.1 is showing the graphical view of the approaches followed in this work.

6.1 Language specific and Merged tensor analysis

In this step, the simple PARAFAC decomposition is applied on the single tensors and merged tensor. There are two single tensors, one consists of all the dataset of the Chinese subjects (Chinese_Tensor) and the other consists of all the dataset of the Italian subjects (Italian_Tensor). Then both of these tensors are merged in the third mode to form a single tensor with all dataset (Merged_Tensor). The dimension of $Chinese\_Tensor$ is 60

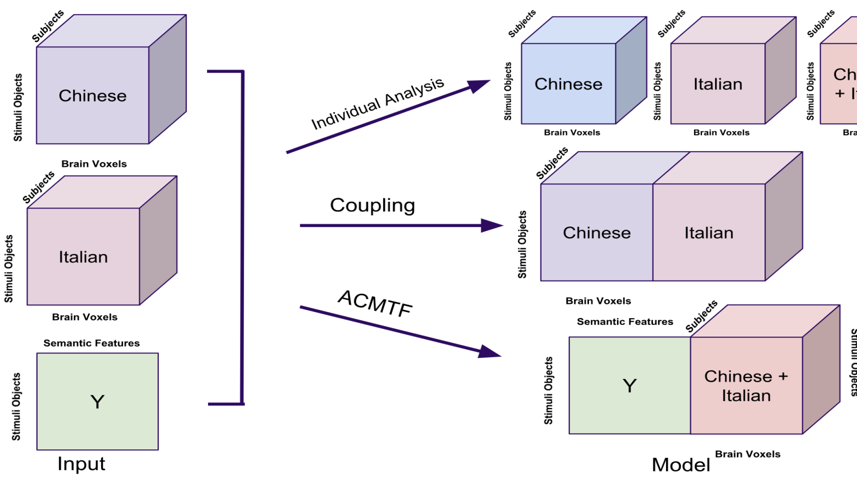


Figure 6.1: The graphical model representing the three different analysis performed on the dataset.

X 379360 X 4, *Italian_Tensor* is 60 X 379360 X 3 and *Merged_Tensor* is 60 X 379360 X 7. Then all of these three tensors are decomposed using PARAFAC decomposition. Figure 6.2 is showing the decomposition method applied to the data.

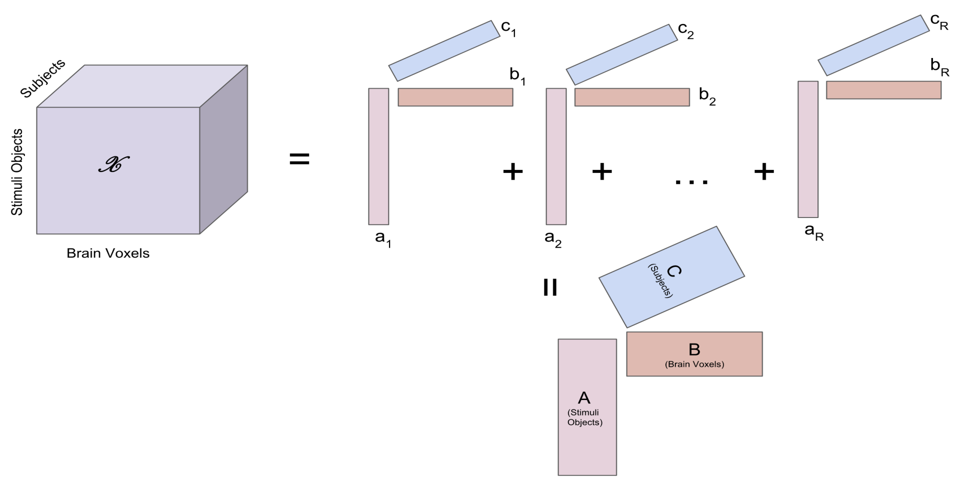


Figure 6.2: The graphical model representing the tensor analysis using PARAFAC decomposition.

In this figure, $A \in \mathbb{R}^{60XR}$ is the factor matrix in Stimuli Object Mode, $B \in \mathbb{R}^{379360XR}$ is the factor matrix in Brain Voxel Mode, $C \in \mathbb{R}^{4XR}$ is the factor matrix in Subject Mode for $Chinese\_Tensor$. These factor matrices are used for further analysis.

6.2 Joint analysis of language specific tensor

In this step, the Chinese tensor and the Italian tensors are coupled on the Stimuli object mode and decomposed using the CMTF decomposition. Figure 6.3 is showing the graphical representation of the decomposition. In this figure, $A \in \mathbb{R}^{60XR}$ is the factor matrix in Stimuli Object Mode, $B \in \mathbb{R}^{379360XR}$ is the factor matrix in Chinese Brain Voxel Mode, $C \in \mathbb{R}^{4XR}$ is the factor matrix in Chinese Subject Mode, $D \in \mathbb{R}^{379360XR}$ is the factor matrix in Italian Brain Voxel Mode and $E \in \mathbb{R}^{3XR}$ is the factor matrix in Italian Subject Mode. Here, R is the rank of the decomposition.

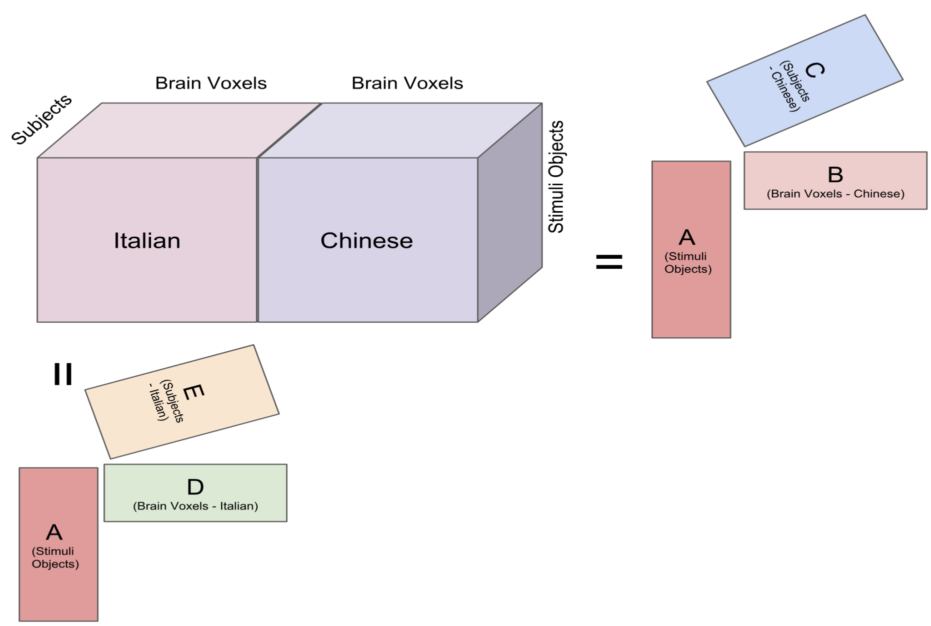


Figure 6.3: The graphical model representing the tensor analysis in the joint analysis step.

6.3 ACMTF for Joint analysis of tensor and semantic feature

In case of tensor analysis, it has always been proved to be useful when additional information is added for the decomposition. In this step, the $Merged\_Tensor$ is coupled with semantic features in the *Stimuli Object* mode to see whether brain data and semantic features can be correlated together or not. Figure 6.4 is showing the graphical representation of the decomposition. In this figure, $A \in \mathbb{R}^{58XR}$ is the factor matrix in Stimuli Object Mode, $B \in \mathbb{R}^{379360XR}$ is the factor matrix in Brain Voxel Mode, $C \in \mathbb{R}^{7XR}$ is the factor matrix in Subject Mode, $D \in \mathbb{R}^{218XR}$ is the factor matrix for Semantic Features. Here, R is the rank of the decomposition.

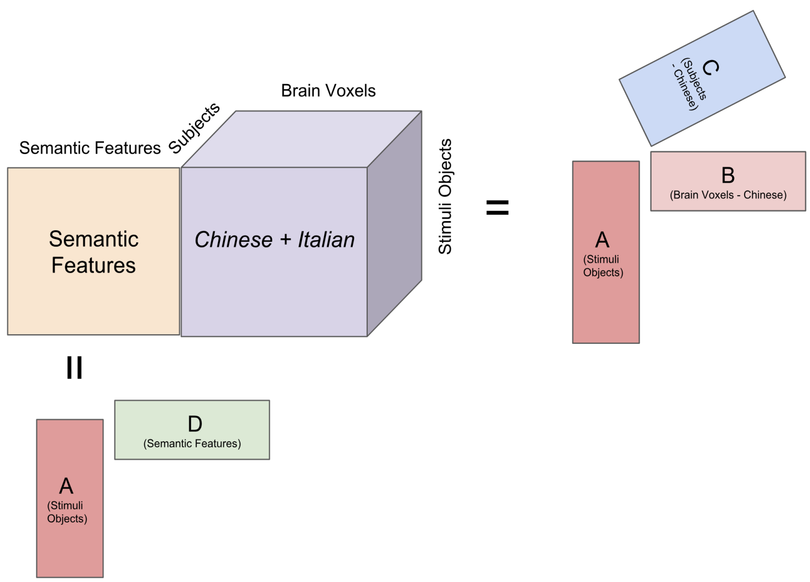


Figure 6.4: The graphical model representing the tensor analysis in the joint analysis step using ACMTF.

This analysis gave good results and shows promise that if data from different language speaking subject is analyzed with semantic features, it can give a better response.

In this experiment, a prediction task has been performed where, given the semantic features, the brain voxels are predicted. In the cross-validation, the *leave-two-out* concept is used to evaluate the model. Two records of stimuli objects are removed from the tensor and the rest of the 58 stimuli are used for training. Then the two words are used for testing where the activation of the brain voxels are measured and then compared with the actual value. In order to compute the activation of brain voxels, the B and D factor matrices will be used. Let, there are i stimuli objects where $i = 1, 2, \ldots, 60$. Q^i be the question vector for object i. V_i is the brain activity associated with the object i. Equation 6.1 is showing the formula used to compute the brain activation.

$$\tilde{V}_i = BD^T Q^i \tag{6.1}$$

The question vector is first projected to the latent space by the left multiplication between D^T and Q^i. Then it is projected on the brain voxel space by the second left multiplication between the B factor matrix and previous result. The brain activation is predicted for both the remaining objects and the classification is done according to the rule in equation 6.2. In the rule, $\tilde{V}_1$ and $\tilde{V}_2$ are the predicted brain activation for stimuli object 1 and 2. The accuracy results show promising performance with such approach.

$$\|V_1 - \tilde{V}_1\|_2 + \|V_2 - \tilde{V}_2\|_2 < \|V_1 - \tilde{V}_2\|_2 + \|V_2 - \tilde{V}_1\|_2 \tag{6.2}$$

Chapter 7

Experimental Results

The results obtained in two major sections of the study shows promising results. The results are presented and discussed in this chapter.

7.1 Results from PyMVPA analysis

As discussed before, two different analysis has been performed in this section. The results for *Within Subject Classification* and *Between Subject Classification* are presented below.

7.1.1 Within Subject Classification

The classification task is to distinguish the brain fMRI samples as viewing land mammals or worktools. Each of the participants' session is considered individually with their own mask. For classification, the SVM, PLR, and KNN (for K = 3, 5 and 7) algorithms are used.

One Way ANOVA Method

The One Way Anova method is used to select the highly active top 100, 500 and 1000 features. The classification accuracies obtained for the seven dataset with 100, 500 and 1000 feature are listed in Table 7.1, Table 7.2 and Table 7.3 respectively.

Table 7.1: Classification accuracy for Within Subject Classification and One Way ANOVA method with 100 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	90.82	95.49	95.03	95.02	94.98
Chinese_Subject_2	90.53	95.82	92.51	92.04	92.04
Chinese_Subject_3	95.51	95.04	93.71	93.05	93.28
Chinese_Subject_4	86.65	86.78	87.61	88.85	85.45
Italian_Subject_1	87.10	88.35	88.26	88.32	85.82
Italian_Subject_2	87.83	91.58	93.13	93.52	93.90
Italian_Subject_3	91.38	93.85	92.99	92.58	91.73

Table 7.2: Classification accuracy for Within Subject Classification and One Way ANOVA method with 500 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	90.45	95.43	95.89	94.61	96.26
Chinese_Subject_2	90.50	93.90	89.14	89.17	89.57
Chinese_Subject_3	96.27	95.84	92.68	92.35	92.32
Chinese_Subject_4	79.91	88.01	85.25	85.19	86.51
Italian_Subject_1	85.93	91.56	88.50	88.56	88.22
Italian_Subject_2	93.98	94.11	94.04	93.96	93.57
Italian_Subject_3	91.35	95.07	95.45	95.48	95.48

The one-way anova method selects the top voxels according to *F-scores*. Figure 7.1 is showing the highly active voxels that are selected by the one-way anova method for each individual subject.

Table 7.3: Classification accuracy for Within Subject Classification and One Way ANOVA method with 1000 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	94.15	93.79	91.20	92.87	94.21
Chinese_Subject_2	94.40	94.85	88.68	88.28	87.27
Chinese_Subject_3	97.34	95.78	91.87	91.89	91.97
Chinese_Subject_4	85.61	89.51	86.50	83.95	85.22
Italian_Subject_1	86.97	89.17	83.79	83.80	84.68
Italian_Subject_2	94.68	94.50	93.92	94.00	94.37
Italian_Subject_3	92.98	97.13	95.44	95.84	95.84

Sum of Squares Method

The results obtained for the Sum of Squares method is discussed here. The sum of squares for each column is calculated and features that show high activity are selected. The highly active top 100, 500 and 1000 features selected this way is used for training. The classification accuracies obtained for the seven dataset with 100, 500 and 1000 feature are listed in Table 7.4, Table 7.5 and Table 7.6 respectively.

Table 7.4: Classification accuracy for Within Subject Classification and Sum of Squares method with 100 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	58.69	52.57	57.53	57.25	55.12
Chinese_Subject_2	62.51	50.79	50.07	54.63	53.59
Chinese_Subject_3	51.81	44.57	45.98	48.09	46.08
Chinese_Subject_4	51.78	50.04	50.23	49.54	48.17
Italian_Subject_1	49.78	49.52	47.72	51.58	49.32
Italian_Subject_2	49.89	42.27	48.16	50.62	49.09
Italian_Subject_3	49.58	44.53	47.57	47.01	47.92

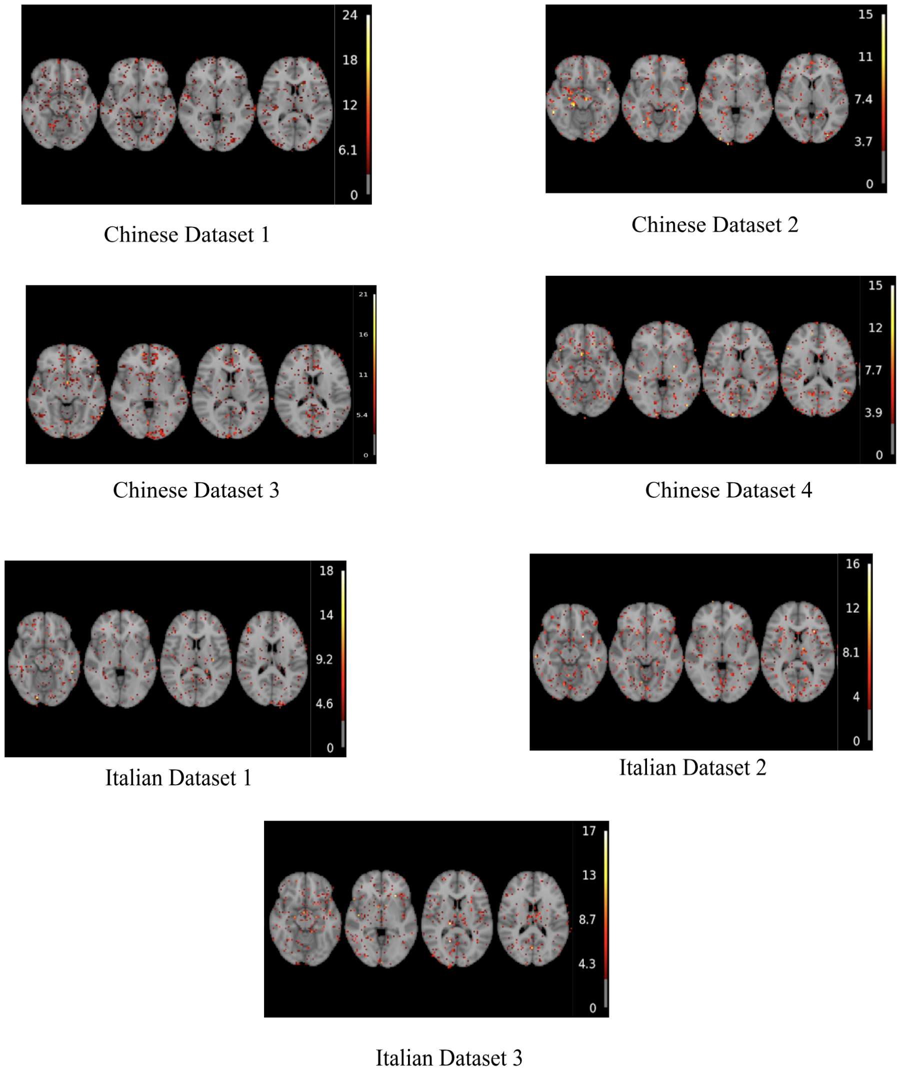


Figure 7.1: The highly active voxels selected by the One Way Anova method.

In this Sum of Squares method, the voxels are selected that have the highest sum. When the top voxels are plotted on the brain image, it gives a completely distorted image which indicates that this approach can not determine the highly active voxels.

Table 7.5: Classification accuracy for Within Subject Classification and Sum of Squares with 500 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	53.27	54.31	52.94	50.38	57.06
Chinese_Subject_2	58.79	48.19	51.85	52.86	47.97
Chinese_Subject_3	57.08	44.53	58.55	59.18	57.09
Chinese_Subject_4	51.77	50.88	48.08	49.17	53.27
Italian_Subject_1	49.59	49.92	51.64	53.69	49.10
Italian_Subject_2	46.58	41.77	45.30	44.73	45.92
Italian_Subject_3	47.93	46.81	50.77	53.24	50.36

Table 7.6: Classification accuracy for Within Subject Classification and Sum of Squares with 1000 features

Dataset	**Linear C-SVM (%)**	**PLR (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
Chinese_Subject_1	48.37	54.24	50.21	51.58	51.24
Chinese_Subject_2	60.28	55.26	53.29	56.15	58.53
Chinese_Subject_3	56.78	59.75	54.39	52.57	53.22
Chinese_Subject_4	49.71	50.94	45.75	45.92	49.85
Italian_Subject_1	51.51	48.26	50.30	49.67	48.22
Italian_Subject_2	55.43	42.11	48.30	53.43	49.69
Italian_Subject_3	48.39	45.49	46.66	45.53	46.43

A clear view of the classification performance can be obtained when the results are plotted. Figure 7.2 and 7.3 are showing the classification performance for both One Way ANOVA and Sum of Square feature selection method.

From the figure, it is evident that the *One Way ANOVA* method works really well since the accuracy is always above 80%. Moreover, the *Linear C-SVM* classifier works well when the features are selected using *One Way ANOVA* method. Whereas, the *Sum of Squares* method doesn't perform well, the reason behind this is over-fitting. Moreover, in this approach, the brain signal values are squared and summed together. Therefore, if there is a negative value indicating the signal strength of a voxel, due to the squaring process, it

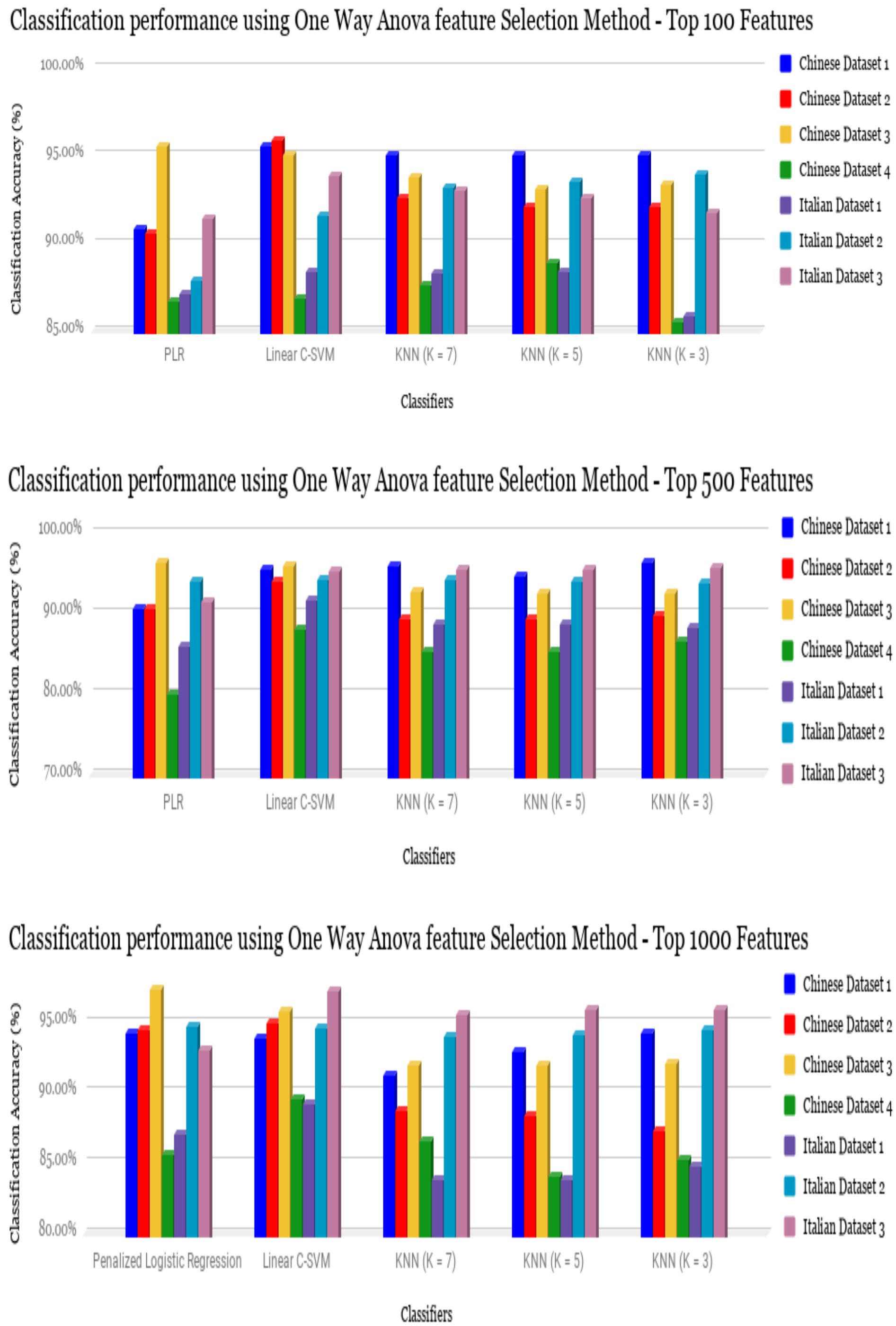


Figure 7.2: Classification performance using One Way Anova Feature Selection method. The top 100, 500 and 1000 voxels are selected for the analysis.

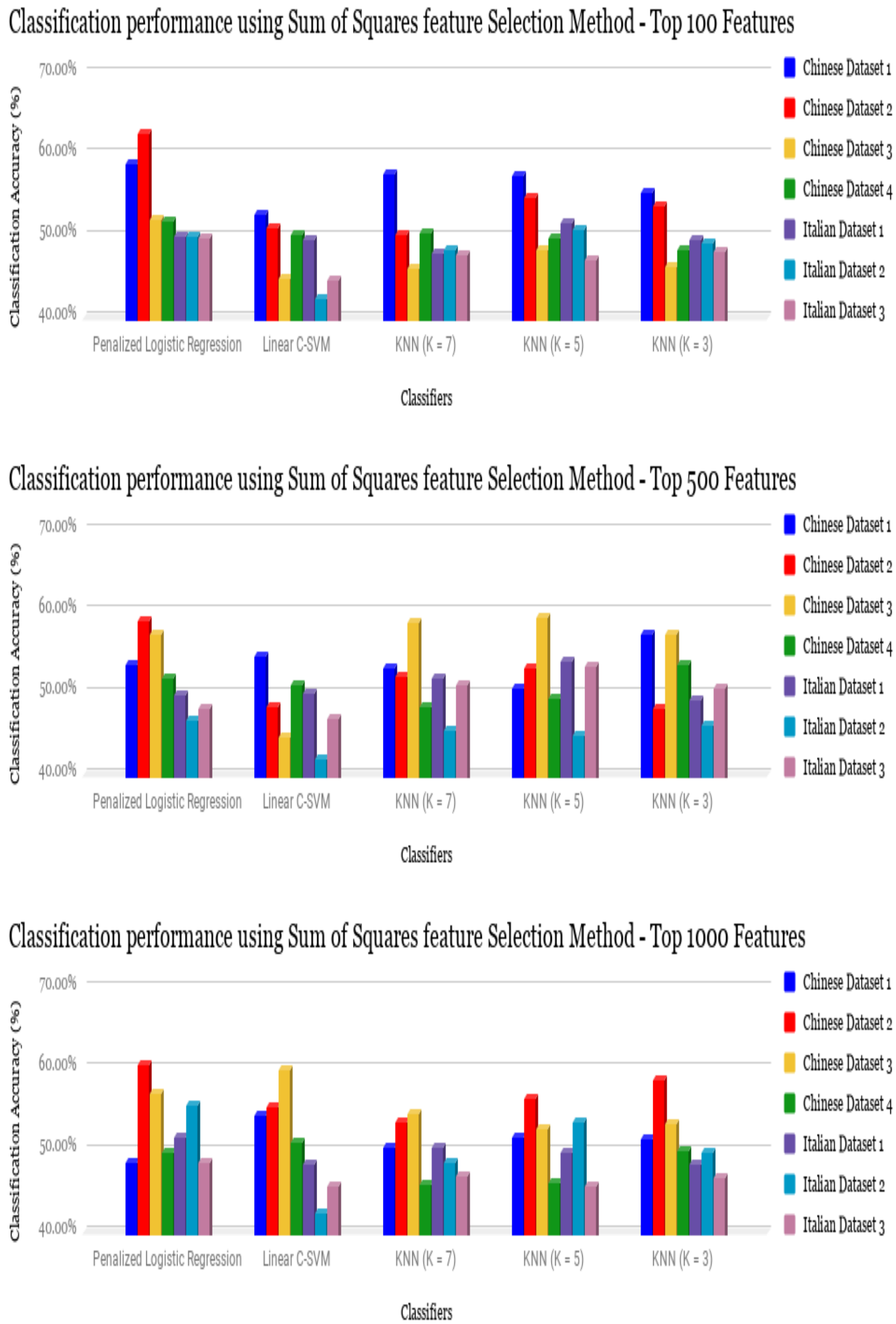


Figure 7.3: Classification performance using Sum of Squares Feature Selection method. The top 100, 500 and 1000 voxels are selected for the analysis.

will become positive and rather than reducing the overall strength of that particular voxel, it will increase it. Therefore, it turns out that voxels that might have less strength appear to be active in this feature selection process which misleads the analysis. The classification accuracy with this approach is around 45%. Moreover, the *Penalized Logistic Regression (PLR)* method performed better than the other methods.

7.1.2 Between Subject Classification

In case of Between Subject Classification, the dataset of 7 subjects are considered as a whole. The classification task is identical to Within Subject Classification i.e. to distinguish the brain fMRI samples as viewing land mammals or worktools. However, in cross-validation step the dataset is partitioned according to the number of subjects. In each step of cross-validation, the dataset of one subject is totally removed, trained with the rest of the dataset and then tested with the previously removed dataset. Since for *Within Subject Classification* step, the anova feature selection method performed well, in this step this particular feature selection method is used. In this method, again the top 100, 500 and 1000 features are considered for training. The classification accuracies obtained for the whole dataset are listed in Table 7.7. From the table, it is evident that the accuracies are really low and *Linear C-SVM* shows better performance than the other classifiers. Figure 7.4 shows the highly active voxels selected by the anova method, plotted on the brain.

Table 7.7: Classification accuracy for Between Subject Classification task

Number of Features	**Linear C-SVM (%)**	**KNN with K = 7 (%)**	**KNN with K = 5 (%)**	**KNN with K = 3 (%)**
100	0.55	0.50	0.46	0.39
500	0.55	0.49	0.46	0.41
1000	0.55	0.48	0.46	0.41

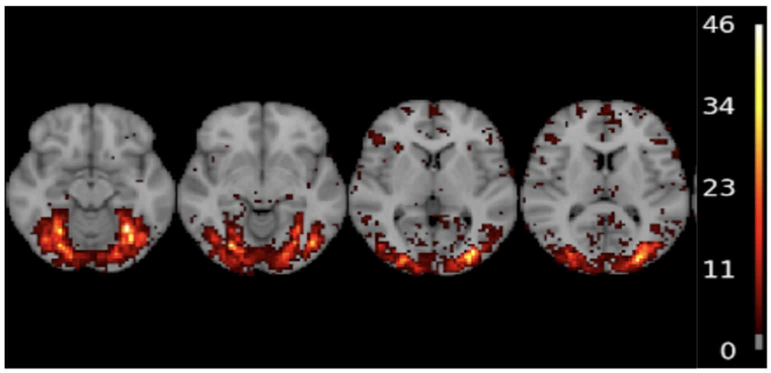


Figure 7.4: The highly active voxels selected by the One Way ANOVA method for the Between Subject Classification task.

7.2 Results from Tensor analysis

In this section, the results obtained from the tensor analysis will be discussed. The tensor analysis is performed with different ranks. All of the analysis used rank 2, 3, 4, 5, 7 and 10. However, the analysis of all these decompositions shows that, rank 2 gives a better approximation of the correlation.

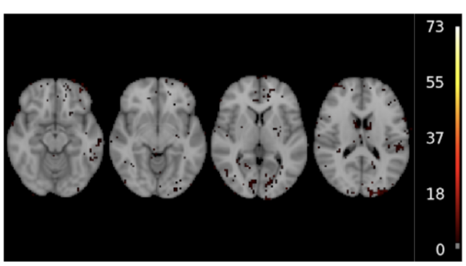
7.2.1 Language specific and Merged tensor analysis

As discussed previously, there are three different tensors that contain the brain fMRI data. *Chinese_Tensor*, *Italian_Tensor* and *Merged_Tensor* are analyzed using PARAFAC decomposition.

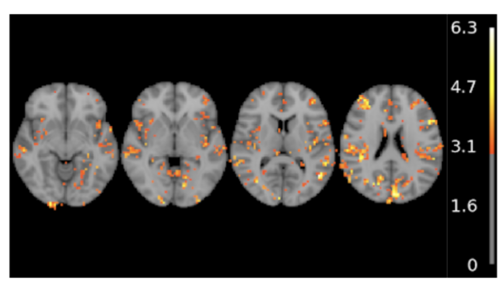
- *Chinese Tensor*: This tensor consists of data that are collected from the Chinese subjects. The PARAFAC decomposition for rank 2 gives 2 components. The top 5 stimuli objects for both of these components are listed in Table 7.8. Moreover, the highly active voxels are plotted on the brain for both of these components in Figure 7.5. The decomposition could not find an appropriate cluster from the data.

Table 7.8: The top 5 stimuli objects for Chinese_Tensor decomposition

Top 5 Stimuli for component 1	**Top 5 Stimuli for component 2**
worktool-plaster_trowel	vanillaLandMammal-panda
vanillaLandMammal-fox	worktool-saw
vanillaLandMammal-ibex	worktool-nail
worktool-paint_roller	worktool-scissors
worktool-garden_trowel	vanillaLandMammal-mole



(a) Component 1



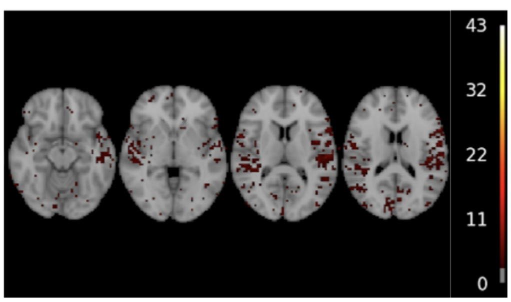
(b) Component 2

Figure 7.5: The two components of Brain Voxel factor matrix obtained from the Chinese_Tensor decomposition.

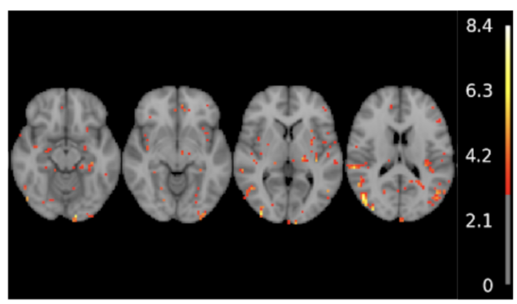
- *Italian Tensor*: This tensor consists of data that are collected from the Italian subjects. The top 5 stimuli obtained in this case are listed in Table 7.9. Moreover, the highly active voxels are plotted on the brain for both of these components in Figure 7.6. The decomposition could not find an appropriate cluster from the data.

Table 7.9: The top 5 stimuli objects for Italian_Tensor decomposition

Top 5 Stimuli for component 1	**Top 5 Stimuli for component 2**
vanillaLandMammal-panda	vanillaLandMammal-chamois
worktool-rake	vanillaLandMammal-boar
vanillaLandMammal-rhinoceros	vanillaLandMammal-kangaroo
worktool-garden_trowel	worktool-pneumatic_drill
worktool-plaster_trowel	worktool-plaster_trowel



(a) Component 1



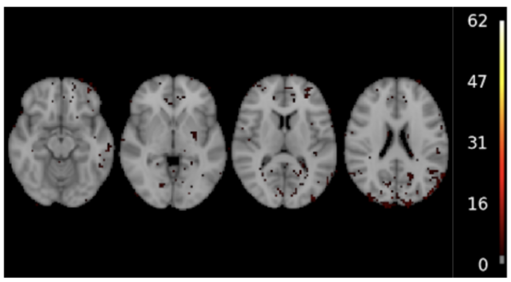
(b) Component 2

Figure 7.6: The two components of Brain Voxel factor matrix obtained from the Italian_Tensor decomposition.

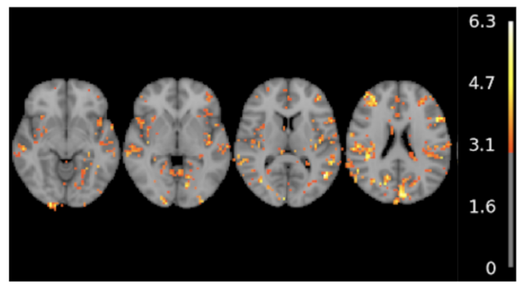
- *Merged Tensor*: This tensor consists of data that are collected from both the Chinese and Italian subjects. The top 5 stimuli obtained in this case are listed in Table 7.10. Moreover, the highly active voxels are plotted on the brain for both of these components in Figure 7.7. The decomposition could not find appropriate cluster.

Table 7.10: The top 5 stimuli objects for Merged_Tensor decomposition

Top 5 Stimuli for component 1	**Top 5 Stimuli for component 2**
worktool-plaster_trowel	vanillaLandMammal-panda
vanillaLandMammal-fox	worktool-nail
vanillaLandMammal-ibex	worktool-saw
vanillaLandMammal-chamois	worktool-scissors
worktool-garden_trowel	worktool-hack_saw



(a) Component 1



(b) Component 2

Figure 7.7: The two components of Brain Voxel factor matrix obtained from the Merged_Tensor decomposition.

Since none of the above three approaches could find a cluster from the decomposition, it can be said that neither Chinese nor Italian subjects show specific brain activation through which they can be distinguished.

7.2.2 Joint analysis of language specific tensor

The *Chinese_Tensor* is jointly analyzed with the *Italian_Tensor* by coupling them in the *Stimuli Object* mode. The top 5 stimuli objects obtained in this case are listed in Table 7.11. Moreover, the highly active voxels are plotted on the brain for both of these components and, for both chinese and italian subjects in Figure 7.8. The decomposition could not find an appropriate cluster from the data.

Table 7.11: The top 5 stimuli objects for joint tensor analysis

Top 5 Stimuli for component 1	**Top 5 Stimuli for component 2**
vanillaLandMammal-badger	worktool-plaster_trowel
vanillaLandMammal-chamois	worktool-file
vanillaLandMammal-kangaroo	vanillaLandMammal-chamois
worktool-plunger	vanillaLandMammal-deer
vanillaLandMammal-llama	worktool-garden_trowel

Since, in this approach, the decomposition could not find a cluster, it can be said that cultural difference does not have an impact on how people think. However, this is a broad problem and more data is needed to prove this claim.

7.2.3 ACMTF for Joint analysis of tensor and semantic feature

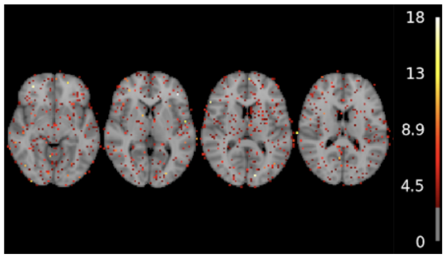
The $Margerd\_Tensor$ is jointly analyzed with the *Semantic Features* matrix by coupling them along the *Stimuli Object* mode and it shows promising results. It is possible to find clusters in the data and the expected brain voxels are active.

The top 5 stimuli obtained in this case are listed in Table 7.12. Moreover, the highly active voxels are plotted on the brain for both of these components in Figure 7.9.

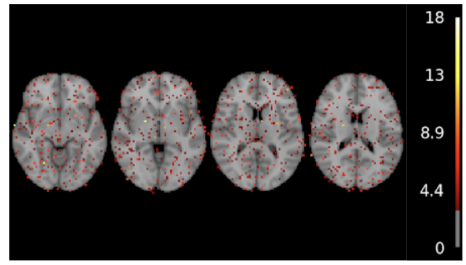
Table 7.12: The top 5 stimuli objects for ACMTF analysis

Top 5 Stimuli for component 1	**Top 5 Stimuli for component 2**
worktool-pick_axe	vanillaLandMammal-chimpanzee
worktool-garden_trowel	vanillaLandMammal-llama
worktool-axe	vanillaLandMammal-squirrel
worktool-plaster_trowel	vanillaLandMammal-camel
worktool-rake	vanillaLandMammal-chamois

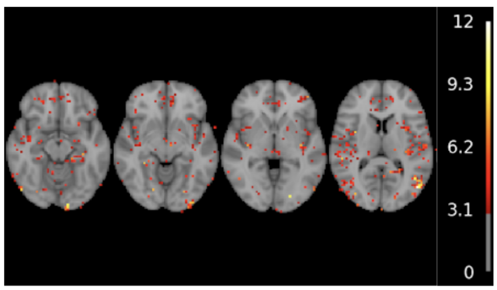
Moreover, after the decomposition, the predictive analysis shows that, it is successful in identifying the important brain voxels depending on semantic stimuli as well. In



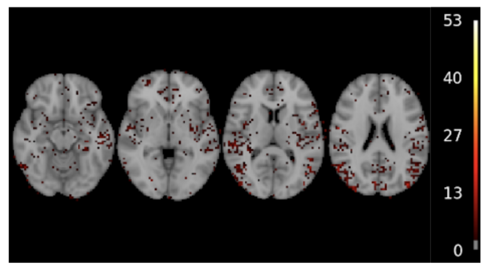
(a) Component 1 of Chinese Factor Matrix



(b) Component 2 of Chinese Factor Matrix



(c) Component 1 of Italian Factor Matrix



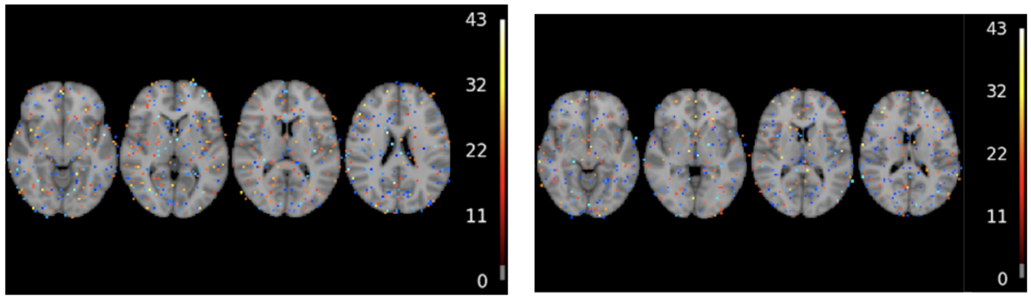
(d) Component 2 of Italian Factor Matrix

Figure 7.8: The highly active voxels for each component of the joint tensor analysis.

this case, two different noun pairs are removed from the dataset and then trained the model. Then the brain voxels for the removed noun pairs are predicted. Table 7.13 is showing the accuracy obtained for two different noun pairs.

Table 7.13: Accuracy for brain voxel prediction

Rank	**zebra/tape_measure**	**squirrel/spanner**
R=2	71.43%	71.43%
R=3	57.14%	57.14%
R=4	85.714%	71.43%
R=5	71.43%	71.43%



(a) Component 1

(b) Component 2

Figure 7.9: The highly active voxels for each component of the ACMTF analysis.

The two different noun pairs that are used for this experiments are, {*vanillaLandMammal-zebra*, *worktool-tape_measure*} and {*vanillaLandMammal-squirrel*, *worktool-spanner*}. The results show that, for different ranks the accuracy is similar for both of the noun pairs. Since, there are two classes of *Stimuli Objects* in the dataset, the results for rank 2 should be considered as correct, and thus the accuracy of the brain voxel prediction task is 71.43%. Since this approach is showing promising results, it is evident that, if additional information is jointly analyzed with brain fMRI tensor, hidden structures can be found out.

Chapter 8

Conclusion

The current approaches have already shown promising results. However, there is a number of scopes to improve.

- In the joint analysis approach, more data sequence can be considered to see whether tensor decomposition methods can identify the latent structure that can differentiate subjects according to different languages. In other words, by adding more dataset for different language speaking subjects, it can be claimed whether it is possible to find out cultural effect or not.

- In Advanced CMTF analysis, if more than two types of stimuli objects are considered then more data sequence should be added. In that case, it will be possible to generalize the method for clustering brain voxels for different categories of the stimuli object. Though the current approach for voxel activation computation and classification approach works well, a concrete classification approach should be considered for this task.

The conceptual knowledge processing mechanism of the brain can answer important research questions that are required to understand the human brain. There are a number of reasons for which it is challenging to solve this problem completely. In this thesis, this problem is addressed on a new dimension. The two different problems are analyzed using PyMVPA and tensor decomposition methods. Though the joint analysis of *Chinese_Tensor* and *Italian_Tensor* cannot identify cultural difference, the joint analysis of *Merged_Tensor* and *Semantic Feature* matrix shows promising result.

The dataset for the Chinese and Italian subject is studied with different types of tensor analysis. The reason behind this is to find out which method captures the correlation properly. Among all of the applied method, the joint analysis using ACMTF gives a good prediction of brain voxels. If questions related to the existence of cultural differences are answered properly, then many important problems in this domain will be resolved. Moreover, if this problem is again addressed with the proposed future extension, it can definitely create ways to understand the brains clearly. It is expected that, in order to solve different neuro-semantic problems, the proposed methods will work as an ideal model and contribute effectively to future research in this domain.

Bibliography

[1] Evrim Acar, Canan Aykut-Bingol, Haluk Bingol, Rasmus Bro, and Bülent Yener. Multiway analysis of epilepsy tensors. *Bioinformatics*, 23(13):i10–i18, 2007.

[2] Evrim Acar, Rasmus Bro, and Age K Smilde. Data fusion in metabolomics using coupled matrix and tensor factorizations. *Proceedings of the IEEE*, 103(9):1602–1620, 2015.

[3] Evrim Acar, Daniel M Dunlavy, and Tamara G Kolda. A scalable optimization approach for fitting canonical tensor decompositions. *Journal of Chemometrics*, 25(2):67–86, 2011.

[4] Evrim Acar, Daniel M Dunlavy, Tamara G Kolda, and Morten Mørup. Scalable tensor factorizations with missing data. In *Proceedings of the 2010 SIAM international conference on data mining*, pages 701–712. SIAM, 2010.

[5] Evrim Acar, Tamara G Kolda, and Daniel M Dunlavy. All-at-once optimization for coupled matrix and tensor factorizations. *arXiv preprint arXiv:1105.3422*, 2011.

[6] Evrim Acar, Anders J Lawaetz, Morten A Rasmussen, and Rasmus Bro. Structure-revealing data fusion model with applications in metabolomics. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pages 6023–6026. IEEE, 2013.

[7] Evrim Acar, Yuri Levin-Schwartz, Vince D Calhoun, and Tülay Adali. Acmtf for fusion of multi-modal neuroimaging data and identification of biomarkers. In *Signal Processing Conference (EUSIPCO), 2017 25th European*, pages 643–647. IEEE, 2017.

[8] Evrim Acar, Yuri Levin-Schwartz, Vince D Calhoun, and Tülay Adali. Tensor-based fusion of eeg and fmri to understand neurological changes in schizophrenia. In *Circuits and Systems (ISCAS), 2017 IEEE International Symposium on*, pages 1–4. IEEE, 2017.

[9] Evrim Acar, Evangelos E Papalexakis, Gözde Gürdeniz, Morten A Rasmussen, Anders J Lawaetz, Mathias Nilsson, and Rasmus Bro. Structure-revealing data fusion. *BMC bioinformatics*, 15(1):239, 2014.

[10] Hiroyuki Akama, Brian Murphy, Li Na, Yumiko Shimizu, and Massimo Poesio. Decoding semantics across fmri sessions with different stimulus modalities: a practical mvpa study. *Frontiers in neuroinformatics*, 6:24, 2012.

[11] Analytics Vidhya. Introduction to Feature Selection methods with an example. Last accessed: January 3, 2018.

[12] Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research*, 15(1):2773–2832, 2014.

[13] Animashree Anandkumar, Daniel Hsu, and Sham M Kakade. A method of moments for mixture models and hidden markov models. In *Conference on Learning Theory*, pages 33–1, 2012.

[14] Miguel Araujo, Spiros Papadimitriou, Stephan Günnemann, Christos Faloutsos, Prithwish Basu, Ananthram Swami, Evangelos E Papalexakis, and Danai Koutra. Com2: fast automatic discovery of temporal (comet) communities. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 271–283. Springer, 2014.

[15] Hanna Becker, Pierre Comon, and Laurent Albera. Tensor-based preprocessing of combined eeg/meg data. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 275–279. IEEE, 2012.

[16] Christian F Beckmann and Stephen M Smith. Tensorial extensions of independent component analysis for multisubject fmri analysis. *Neuroimage*, 25(1):294–311, 2005.

[17] Mehdi Behroozi and Mohammad Reza Daliri. Predicting brain states associated with object categories from fmri data. *Journal of integrative neuroscience*, 13(04):645–667, 2014.

[18] Lachezar Bozhkov, Petia Georgieva, and Roumen Trifonov. Brain neural data analysis using machine learning feature selection and classification methods. In Valeri Mladenov, Chrisina Jayne, and Lazaros Iliadis, editors, *Engineering Applications of Neural Networks*, pages 123–132, Cham, 2014. Springer International Publishing.

[19] Molly G Bright and Kevin Murphy. Is fmri noise really noise? resting state nuisance regressors remove variance with network structure. *NeuroImage*, 114:158–169, 2015.

[20] Rasmus Bro. Parafac. tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171, 1997.

[21] Rasmus Bro and Henk AL Kiers. A new efficient method for determining the number of components in parafac models. *Journal of chemometrics*, 17(5):274–286, 2003.

[22] Bokai Cao, Lifang He, Xiaokai Wei, Mengqi Xing, Philip S Yu, Heide Klumpp, and Alex D Leow. t-bne: Tensor-based brain network embedding. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 189–197. SIAM, 2017.

[23] Bokai Cao, Chun-Ta Lu, Xiaokai Wei, S Yu Philip, and Alex D Leow. Semi-supervised tensor factorization for brain network analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 17–32. Springer, 2016.

[24] J Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. *Psychometrika*, 35(3):283–319, 1970.

[25] Raymond B Cattell. parallel proportional profiles and other principles for determining the choice of factors by rotation. *Psychometrika*, 9(4):267–283, 1944.

[26] Ian Charest, Rogier A Kievit, Taylor W Schmitz, Diana Deca, and Nikolaus Kriegeskorte. Unique semantic space in the brain of each beholder predicts perceived similarity. *Proceedings of the National Academy of Sciences*, 111(40):14565–14570, 2014.

[27] Ian Charest and Nikolaus Kriegeskorte. The brain of the beholder: honouring individual representational idiosyncrasies. *Language, Cognition and Neuroscience*, 30(4):367–379, 2015.

[28] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. PHAN. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *IEEE Signal Processing Magazine*, 32(2):145–163, March 2015.

[29] Andrzej Cichocki. Tensor decompositions: a new concept in brain data analysis? *arXiv preprint arXiv:1305.0395*, 2013.

[30] Pierre Comon. Tensors: a brief introduction. *IEEE Signal Processing Magazine*, 31(3):44–53, 2014.

[31] Lieven De Lathauwer. A survey of tensor methods. In *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pages 2773–2776. IEEE, 2009.

[32] Lieven De Lathauwer and Joséphine Castaing. Tensor-based techniques for the blind separation of ds–cdma signals. *Signal Processing*, 87(2):322–336, 2007.

[33] Lieven De Lathauwer and Bart De Moor. From matrix to tensor: Multilinear algebra and signal processing. In *Institute of Mathematics and Its Applications Conference Series*, volume 67, pages 1–16. Citeseer, 1998.

[34] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[35] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors. *SIAM journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[36] Lieven De Lathauwer and Eleftherios Kofidis. Coupled matrix-tensor factorizationsthe case of partially shared factors. In *Proc. of the Asilomar Conference on Signals, Systems and Computers*, number accepted, 2018.

[37] Daniel M Dunlavy, Tamara G Kolda, and Evrim Acar. Temporal link prediction using matrix and tensor factorizations. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(2):10, 2011.

[38] Ruth C Fong, Walter J Scheirer, and David D Cox. Using human brain activity to guide machine learning. *Scientific reports*, 8(1):5397, 2018.

[39] Elia Formisano, Federico De Martino, and Giancarlo Valente. Multivariate analysis of fmri time series: classification and regression of brain responses using machine learning. *Magnetic resonance imaging*, 26(7):921–934, 2008.

[40] Matthieu Genicot, P-A Absil, Renaud Lambiotte, and Saber Sami. Coupled tensor decomposition: a step towards robust components. In *Signal Processing Conference (EUSIPCO), 2016 24th European*, pages 1308–1312. IEEE, 2016.

[41] Giacomo Handjaras, Emiliano Ricciardi, Andrea Leo, Alessandro Lenci, Luca Cecchetti, Mirco Cosottini, Giovanna Marotta, and Pietro Pietrini. How concepts are encoded in the human brain: a modality independent, category-based cortical organization of semantic knowledge. *Neuroimage*, 135:232–242, 2016.

[42] Michael Hanke, Yaroslav O Halchenko, James V Haxby, and Stefan Pollmann. Statistical learning analysis in neuroscience: aiming for transparency. *Frontiers in neuroscience*, 3:7, 2010.

[43] Michael Hanke, Yaroslav O Halchenko, Per B Sederberg, Stephen José Hanson, James V Haxby, and Stefan Pollmann. Pymvpa: a python toolbox for multivariate pattern analysis of fmri data. *Neuroinformatics*, 7(1):37–53, 2009.

[44] Michael Hanke, Yaroslav O Halchenko, Per B Sederberg, Emanuele Olivetti, Ingo Fründ, Jochem W Rieger, Christoph S Herrmann, James V Haxby, Stephen J Hanson, and Stefan Pollmann. Pymvpa: a unifying approach to the analysis of neuroscientific data. *Frontiers in neuroinformatics*, 3:3, 2009.

[45] Richard A Harshman and Margaret E Lundy. Parafac: Parallel factor analysis. *Computational Statistics & Data Analysis*, 18(1):39–72, 1994.

[46] James V Haxby. Multivariate pattern analysis of fmri: the early beginnings. *Neuroimage*, 62(2):852–855, 2012.

[47] James V Haxby, Andrew C Connolly, and J Swaroop Guntupalli. Decoding neural representational spaces using multivariate pattern analysis. *Annual review of neuroscience*, 37:435–456, 2014.

[48] James V Haxby, M Ida Gobbini, Maura L Furey, Alumit Ishai, Jennifer L Schouten, and Pietro Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

[49] James V Haxby, J Swaroop Guntupalli, Andrew C Connolly, Yaroslav O Halchenko, Bryan R Conroy, M Ida Gobbini, Michael Hanke, and Peter J Ramadge. A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, 72(2):404–416, 2011.

[50] Frank L Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Studies in Applied Mathematics*, 6(1-4):164–189, 1927.

[51] Borbála Hunyadi, Wim Van Paesschen, Maarten De Vos, and Sabine Van Huffel. Fusion of electroencephalography and functional magnetic resonance imaging to explore epileptic network activity. In *Signal Processing Conference (EUSIPCO), 2016 24th European*, pages 240–244. IEEE, 2016.

[52] Prateek Jain and Sewoong Oh. Provable tensor factorization with missing data. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1431–1439. Curran Associates, Inc., 2014.

[53] Noppadon Jatupaiboon, Setha Pan-ngum, and Pasin Israsena. Real-time eeg-based happiness detection system. *The Scientific World Journal*, 2013, 2013.

[54] Esin Karahan, Pedro A Rojas-Lopez, Maria L Bringas-Vega, Pedro A Valdes-Hernandez, and Pedro A Valdes-Sosa. Tensor analysis and fusion of multimodal brain images. *Proceedings of the IEEE*, 103(9):1531–1559, 2015.

[55] Eleftherios Kofidis and Phillip A Regalia. Tensor approximation and signal processing applications. *Contemporary Mathematics*, 280:103–134, 2001.

[56] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[57] Pieter M Kroonenberg. *Three-mode principal component analysis: Theory and applications*, volume 2. DSWO press, 1983.

[58] Pieter M Kroonenberg and Jan De Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, 1980.

[59] Steven Lemm, Benjamin Blankertz, Thorsten Dickhaus, and Klaus-Robert Müller. Introduction to machine learning for brain imaging. *Neuroimage*, 56(2):387–399, 2011.

[60] Joseph Levin. Three-mode factor analysis. *Psychological Bulletin*, 64(6):442, 1965.

[61] Lifeinharmony. ANATOMY OF BRAIN LOBES GEOFACE. Last accessed: January 3, 2018.

[62] Lek-Heng Lim and Pierre Comon. Multiarray signal processing: Tensor decomposition meets compressed sensing. *Comptes Rendus Mecanique*, 338(6):311–320, 2010.

[63] Machine Learning Mastery. An Introduction to Feature Selection. Last accessed: January 3, 2018.

[64] Arash Golibagh Mahyari and Selin Aviyente. Identification of dynamic functional brain network states through tensor decomposition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 2099–2103. IEEE, 2014.

[65] Tom M. Mitchell, Rebecca Hutchinson, Radu S. Niculescu, Francisco Pereira, Xuerui Wang, Marcel Just, and Sharlene Newman. Learning to decode cognitive states from brain images. *Machine Learning*, 57(1):145–175, 2004.

[66] Tom M Mitchell, Svetlana V Shinkareva, Andrew Carlson, Kai-Min Chang, Vicente L Malave, Robert A Mason, and Marcel Adam Just. Supplemental web site in support of the paper: Predicting Human Brain Activity Associated with the Meanings of Nouns. Last accessed: January 3, 2018.

[67] Tom M Mitchell, Svetlana V Shinkareva, Andrew Carlson, Kai-Min Chang, Vicente L Malave, Robert A Mason, and Marcel Adam Just. Predicting human brain activity associated with the meanings of nouns. *science*, 320(5880):1191–1195, 2008.

[68] Morten Mørup. Applications of tensor (multiway array) factorizations and decompositions in data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):24–40, 2011.

[69] Morten Mørup, Lars Kai Hansen, and Sidse M Arnfred. Erpwavelab: A toolbox for multi-channel analysis of time–frequency transformed event related potentials. *Journal of neuroscience methods*, 161(2):361–368, 2007.

[70] Elchanan Mossel and Sébastien Roch. Learning nonsingular phylogenies and hidden markov models. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 366–375. ACM, 2005.

[71] Klaus-Robert Müller, Michael Tangermann, Guido Dornhege, Matthias Krauledat, Gabriel Curio, and Benjamin Blankertz. Machine learning for real-time single-trial eeg-analysis: from brain–computer interfacing to mental state monitoring. *Journal of neuroscience methods*, 167(1):82–90, 2008.

[72] Multivariate Pattern Analysis in Python. Module: clfs.knn. Last accessed: January 3, 2018.

[73] Morten Mrup, Lars Kai Hansen, and Sidse M. Arnfred. Algorithms for sparse nonnegative tucker decompositions. *Neural Computation*, 20(8):2112–2131, 2008.

[74] National Institutes of Health (NIH). Analysis of Functional NeuroImages (AFNI). Last accessed: January 3, 2018.

[75] D. Nion and N. D. Sidiropoulos. Tensor algebra and multidimensional harmonic retrieval in signal processing for mimo radar. *IEEE Transactions on Signal Processing*, 58(11):5693–5705, Nov 2010.

[76] Dimitri Nion, Kleanthis N Mokios, Nicholas D Sidiropoulos, and Alexandros Potamianos. Batch and adaptive parafac-based blind separation of convolutive speech mixtures. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1193–1207, 2010.

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

[78] Online Statistics Education: An Interactive Multimedia Course of Study. Analysis of Variance - Introduction. Last accessed: January 3, 2018.

[79] Evangelos Papalexakis, Konstantinos Pelechrinis, and Christos Faloutsos. Spotting misbehaviors in location-based social networks using tensors. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 551–552. ACM, 2014.

[80] Evangelos E Papalexakis, Leman Akoglu, and Dino Ience. Do more views of a graph help? community detection and clustering in multi-graphs. In *Information fusion (FUSION), 2013 16th international conference on*, pages 899–905. IEEE, 2013.

[81] Evangelos E Papalexakis, Christos Faloutsos, Tom M Mitchell, Partha Pratim Talukdar, Nicholas D Sidiropoulos, and Brian Murphy. Turbo-smt: Accelerating coupled sparse matrix-tensor factorizations by 200x. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 118–126. SIAM, 2014.

[82] Evangelos E Papalexakis, Christos Faloutsos, Tom M Mitchell, Partha Pratim Talukdar, Nicholas D Sidiropoulos, and Brian Murphy. Turbo-smt: Accelerating coupled sparse matrix-tensor factorizations by 200x. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 118–126. SIAM, 2014.

[83] Evangelos E. Papalexakis, Christos Faloutsos, and Nicholas D. Sidiropoulos. Parcube: Sparse parallelizable tensor decompositions. In Peter A. Flach, Tijl De Bie, and Nello Cristianini, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 521–536, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[84] Evangelos E Papalexakis, Tom M Mitchell, Nicholas D Sidiropoulos, Christos Faloutsos, Partha Pratim Talukdar, and Brian Murphy. Scoup-smt: Scalable coupled sparse matrix-tensor factorization. *arXiv preprint arXiv:1302.7043*, 2013.

[85] Francisco Pereira, Matthew Botvinick, and Greg Detre. Learning semantic features for fmri data from definitional text. In *Proceedings of the NAACL HLT 2010 First Workshop on Computational Neurolinguistics*, pages 1–9. Association for Computational Linguistics, 2010.

[86] Francisco Pereira, Matthew Botvinick, and Greg Detre. Using wikipedia to learn semantic feature representations of concrete concepts in neuroimaging experiments. *Artificial intelligence*, 194:240–252, 2013.

[87] Francisco Pereira, Tom Mitchell, and Matthew Botvinick. Machine learning classifiers and fMRI: a tutorial overview. *Neuroimage*, 45(1):S199–S209, 2009.

[88] Ioakeim Perros, Evangelos E Papalexakis, Fei Wang, Richard Vuduc, Elizabeth Searles, Michael Thompson, and Jimeng Sun. Spartan: Scalable parafac2 for large & sparse data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 375–384. ACM, 2017.

[89] Stephan Rabanser, Oleksandr Shchur, and Stephan Günnemann. Introduction to tensor decompositions and their applications in machine learning. *arXiv preprint arXiv:1711.10781*, 2017.

[90] Bertrand Rivet, Marc Duda, Anne Guérin-Dugué, Christian Jutten, and Pierre Comon. Multimodal approach to estimate the ocular movements during eeg recordings: a coupled tensor factorization method. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, pages 6983–6986. IEEE, 2015.

[91] Amnon Shashua and Anat Levin. Linear image coding for regression and classification using the tensor-rank principle. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[92] Nicholas D Sidiropoulos, Rasmus Bro, and Georgios B Giannakis. Parallel factor analysis in sensor array processing. *IEEE transactions on Signal Processing*, 48(8):2377–2388, 2000.

[93] Nicholas D Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E Papalexakis, and Christos Faloutsos. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.

[94] Statistics How To. ANOVA Test: Definition, Types, Examples . Last accessed: January 3, 2018.

[95] Jimeng Sun, Dacheng Tao, and Christos Faloutsos. Beyond streams and graphs: dynamic tensor analysis. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 374–383. ACM, 2006.

[96] M. Srensen and L. De Lathauwer. Coupled tensor decompositions for applications in array signal processing. In *2013 5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 228–231, Dec 2013.

[97] Ledyard R Tucker. Experiments in multi-mode factor analysis. In *Proceedings of the Invitational Conference on Testing Problems*, 1965.

[98] Ledyard R Tucker. *Some Mathematical Notes on Three-mode Factor Analysis [by]*. Urbana, Department of Psychology, University of Illinois, 1965.

[99] Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[100] University College London. Statistical Parametric Mapping. Last accessed: January 3, 2018.

[101] University of California, San Diego. EEGLAB. Last accessed: January 3, 2018.

[102] Martijn P Van Den Heuvel and Hilleke E Hulshoff Pol. Exploring the brain network: a review on resting-state fmri functional connectivity. *European neuropsychopharmacology*, 20(8):519–534, 2010.

[103] Simon Van Eyndhoven, Borbála Hunyadi, Lieven De Lathauwer, and Sabine Van Huffel. Flexible fusion of electroencephalography and functional magnetic resonance imaging: Revealing neural-hemodynamic coupling through structured matrix-tensor factorization. In *Signal Processing Conference (EUSIPCO), 2017 25th European*, pages 26–30. IEEE, 2017.

[104] M Alex O Vasilescu. Human motion signatures: Analysis, synthesis, recognition. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 3, pages 456–460. IEEE, 2002.

[105] M Alex O Vasilescu and Demetri Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *European Conference on Computer Vision*, pages 447–460. Springer, 2002.

[106] M Alex O Vasilescu and Demetri Terzopoulos. Multilinear independent components analysis. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 547–553. IEEE, 2005.

[107] Nico Vervliet, Otto Debals, Laurent Sorber, and Lieven De Lathauwer. Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis. *IEEE Signal Processing Magazine*, 31(5):71–79, 2014.

[108] Wikipedia. Blood-oxygen-level dependent imaging. Last accessed: January 3, 2018.

[109] Wikipedia. Chi-squared test. Last accessed: January 3, 2018.

[110] Wikipedia. Feature selection. Last accessed: January 3, 2018.

[111] Wikipedia. Human brain. Last accessed: January 3, 2018.

[112] Wikipedia. Linear discriminant analysis. Last accessed: January 3, 2018.

[113] Wikipedia. Logistic Regression. Last accessed: January 3, 2018.

[114] Wikipedia. Pearson correlation coefficient. Last accessed: January 3, 2018.

[115] Wikipedia. Support Vector Machine. Last accessed: January 3, 2018.

[116] Wikipedia. Tensor. Last accessed: January 3, 2018.

[117] Liang Xiong, Xi Chen, Tzu-Kuo Huang, Jeff Schneider, and Jaime G Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 211–222. SIAM, 2010.

[118] Haoyan Xu, Brian Murphy, and Alona Fyshe. Brainbench: A brain-image test suite for distributional semantic models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2017–2021, 2016.

[119] Tong Zhang and Gene H Golub. Rank-one approximation to high order tensors. *SIAM Journal on Matrix Analysis and Applications*, 23(2):534–550, 2001.

[120] G. Zhou, A. Cichocki, Q. Zhao, and S. Xie. Nonnegative matrix and tensor factorizations : An algorithmic perspective. *IEEE Signal Processing Magazine*, 31(3):54–65, May 2014.