LAHORE UNIVERSITY OF MANAGEMENT SCIENCES
Department of Electrical Engineering

EE514/CS535 Machine Learning
Spring 2023

Project Description

Total Marks: 50

Contribution to Final Assessment: 20%

1 Objectives

In this course project, we will serve the following objectives:

- Learn to extract the features from audio (speech) recordings or images
- Understand the role of feature extraction in machine learning
- Learn to use the different algorithms covered in the course to the problems of

  Project 1: Sentiment Analysis from Audio Recordings

  Project 2: Surface Defect Detection and Classification from Images

  Project 3: Reproduce Research Paper Results

- Learn to evaluate and compare the performance of different algorithms
- Learn to document, report and present the solution of a machine learning problem

We assume that the execution of the project will help the retention of the material and significantly enhance the depth of your understanding.

We require you to work in a group of (maximum) three students (every student in the group will receive same score).

2 Project 1 - Sentiment Analysis from Audio Recordings

2.1 Motivation

Sentiment analysis from audio recordings is the process of using natural language processing techniques to analyze the emotional tone of spoken language in audio files. This can be done using machine learning algorithms that identify patterns and features in the audio, such as intonation, pitch, and tempo, to infer the underlying sentiment of the speaker. Sentiment analysis from audio recordings has a wide range of applications, including in customer service, where it can be used to monitor customer sentiment and satisfaction during phone calls, and in social media monitoring, where it can help to identify trends and sentiment around specific topics. It can also be used in the entertainment industry, where sentiment analysis can help to gauge audience reactions to films, TV shows, and other forms of media.

2.2 Project Overview

In this project, you will work on the development of a classifier to identify the emotion/sentiment for each recorded audio. You will first remove noise from the audio files and then extract the features of the given audio signal and use these features for classification.

2.3 Data Set Description

CREMA-D is a data set of 7,442 original clips from 91 actors. These clips were from 48 male and 43 female actors between the ages of 20 and 74 coming from a variety of races and ethnicities (African America, Asian, Caucasian, Hispanic, and Unspecified).

The actors spoke from a selection of 12 sentences. The sentences were presented using one of six different emotions (Anger, Disgust, Fear, Happy, Neutral and Sad) and four different emotion levels (Low, Medium, High and Unspecified).

You can access the dataset from here: link

2.4 Project Requirements

Before processing the audio files, you are required to select audio files from the dataset. You must create training, validation and testing sets that are best suited to train and evaluate your classifier(s). Then remove the noise from the selected audio files using low pass filtering.

We require you to review the literature and identify a list of features that can be used for synthetic speech attribution. For your convenience, we have listed below the features that can be potentially used for the problem under consideration.

- Fourier transform
- Mel frequency cepstral coefficients (MFCCs): Features that describe the overall shape of a spectral envelope of the signal.

- Melspectrogram : A spectrogram where the signal frequencies are converted to Mel scale.
- Chromagram: Also known as chroma features, this is a representation for audio or speech signal in which the entire signal spectrum is projected onto 12 bins known as pitch classes.
- Bicoherence: Mean, variance, skewness and kutrosis of magnitude and phase of bicoherence.
- Spectral centroid: Location of center of mass of the spectrum.
- Spectral bandwidth: Difference between the highest and the lowest frequency in the spectrum.
- Spectral contrast: The decibel difference between peaks and valleys in the spectrum.

2.5 Classifiers

Followed by feature extraction, we require you to evaluate the performance of following different classifiers.

- K-Nearest Neighbors
- Logistic Regression
- Naive Bayes
- Support Vector Machines
- Neural Network

You are allowed to use Scikit-learn implementations of the algorithms.

3 Project 2 - Surface Defect Detection and Classification from Images

3.1 Motivation

In recent years, thanks to advances in hardware technologies and the equally growing need for real-time quality inspection of production processes, there is a demand for artificially intelligent solutions geared towards improving the robustness of machine vision systems. Defect detection on surface using machine vision involves the use of computer algorithms and image processing techniques to automatically detect and classify defects on the surface of a material. This technology can be applied in various industries, such as manufacturing, quality control, and automotive, where the detection of surface defects is crucial for ensuring the safety and functionality of the product. The machine vision system captures images of the material's surface and analyzes them to identify anomalies such as scratches, dents, and cracks. By automating the defect detection process, machine vision can significantly improve production efficiency and reduce the likelihood of human error. It can also save time and resources compared to manual inspection, which is more time-consuming and prone to errors. Additionally, defect detection using machine vision can provide valuable data that can be used to optimize production processes and prevent future defects from occurring.

Feature extraction and learning is an essential part of machine vision system these days. Although most surface defect detection or localization techniques out there employ self-adaptive neural networks, nevertheless, it is important to establish an acute understanding of the classical approach to isolate an object in an image or a video stream. This project will give you a hands-on comprehension of the way in which a feature identification pipeline is designed and use the features for classification tasks.

3.2 Project Overview

In this project, you will work on the development of a classifier to detect defects in an image. This can be interpreted as a multi-class classification problem in which we have three classes: no defect, hole and mark. You will extract features of the object from the given data-set to train your classifier. And use the trained classifier to search for defects in images of a reel of paper.

3.3 Data Set Description

In the data-set, there are 1500 images of a reel of paper. The images belong to three classes,

- defect-free paper
- paper with a hole
- paper with a pen mark on it

You can access the dataset from here: link
You are encouraged to explore data augmentation to get more training data (optional).

3.4 Feature Extraction and Object Localization

For your convenience, we have listed below the features that can be potentially used for the problem under consideration. We require you to review the literature and identify more features that can be used for defect detection and localization.

- Colour Histogram
- Spatial Binning
- Histogram of Oriented Gradients

3.5 Classifiers

Followed by feature extraction, we require you to evaluate the performance of the following different classifiers.

- Support Vector Machines
- Naive Bayes
- Logistic Regression
- Neural Network

You will use these trained classifiers to perform two tasks.

(i) three-class classification: classify individual images into the three classes i.e., paper, paper with hole, or paper with pen mark.

(ii) defect localization: in the images with defects, you are required to find the location of the defect and draw a bounding box around it.

You are allowed to use Scikit-learn (or any other library you prefer) implementations of the algorithms.

4 Project 3 - Reproduce Research Paper Results

4.1 Overview

In this project, we require you to reproduce the results of either of the following research papers.

Paper 1: Learning to Learn without Gradient Descent by Gradient Descent, ICML 2016

Overview: The aim of this project is to reproduce the results of meta-learning algorithm proposed in the paper that learns to learn by optimizing an inner loss function using a separate meta-learner.

Dataset: You can use the MNIST dataset, which consists of 60,000 28x28 grayscale images of handwritten digits.

Methodology:

- Reproduce the Baseline Model: Start by reproducing a baseline model, such as a simple convolutional neural network, trained on the MNIST dataset.
- Implement the Meta-Learning Algorithm: Implement the meta-learning algorithm proposed in the paper, which involves training a separate meta-learner to optimize the parameters of the inner learner, which is the baseline model trained on the MNIST dataset.
- Train and Evaluate the Models: Train the baseline model and the model with the meta-learning algorithm and evaluate their performance on the test set of the MNIST dataset.
- Compare and Analyze the Results: Compare the performance of the baseline model and the model with the meta-learning algorithm and analyze the results to understand the impact of the meta-learning algorithm on the performance of the model.

Paper 2: Variational Autoencoder for Semi-Supervised Text Classification, AAAI, 2019

Overview: The aim of this project is to reproduce the results of the paper that proposes a variational autoencoder-based approach for semi-supervised text classification, which uses a small amount of labeled data and a large amount of unlabeled data to improve the performance of text classification.

Dataset: You can use the 20 Newsgroups dataset, which consists of 20,000 newsgroup documents across 20 different categories.

Methodology:

- Implement the Baseline Model: Start by implementing a baseline model, such as a simple bag-of-words model, trained on the labeled data of the 20 Newsgroups dataset.
- Implement the Variational Autoencoder: Implement the variational autoencoder-based approach proposed in the paper, which uses the labeled and unlabeled data

to train a deep generative model that can generate latent representations of the input documents.

- Train and Evaluate the Models: Train the baseline model and the model with the variational autoencoder and evaluate their performance on the test set of the 20 Newsgroups dataset.
- Compare and Analyze the Results: Compare the performance of the baseline model and the model with the variational autoencoder and analyze the results to understand the impact of the variational autoencoder on the performance of the model, as well as its ability to leverage the unlabeled data for semi-supervised learning.

5 Expectations and Scope of Work

5.1 Scope of Work (Project 1 and Project 2)

The scope of work for Project 1 and Project 2 includes

Task 1: formulation of the problem under consideration

Task 2: carry out literature review to identify the features that can be used for the problem under consideration

Task 3: apply feature extraction/engineering

Task 4: apply dimensionality reduction (if needed or to evaluate the impact of reduction on the performance)

Task 5: implement the different classifiers for the problem under consideration.

Task 6: report the performance of different classifiers and presentation of analysis/findings

5.2 Assessment

There are three components of assessment in the project:

- Project report (20 marks)
- Project code (along with code documentation) (15 marks)
- 3 minutes video presentation in Week 14 summarizing your work (10 marks)
- Submission of deliverables on time (5 marks)

We encourage you to use a template from your favorite machine learning conference (e.g., NIPS or ICML) for report. Your report is expected to have the following sections:

- Abstract (executive summary)
- Introduction
- Mathematical Formulation
- Identification and Extraction of Features
- (Optional) Feature Engineering e.g., dimensionality reduction (optional)
- Use of different classification algorithms; a subsections on each algorithm
- Performance Evaluation (plots, tables etc.), analysis and findings
- Conclusions

Note that: We encourage Masters and PhD students to use the LaTeX template for their report.

6 Timeline of Deliverables

We want you to adhere to the following time-lines.

- **Deliverable 1:** Project selection and group formation.
  - Due: Week 10, 17th March, Friday 23:55
  - The spreadsheet for groups can be found here.
- **Deliverable 2:** Submit mid-term report and code with the following tasks completed and added in the report
  - Report with the following sections populated: Abstract, Introduction and Mathematical Formulation, Identification and Extraction of Features. You can obviously change the content in these sections in your final submission.
  - Implementation of feature extraction
  - Feature Engineering (e.g., dimensionality reduction)
  - Implementation of at least two of the classification algorithms
  - Due: Week 12: 31st March, Friday 23:55 pm
- **Deliverable 3:** Video presentation (3 minutes)
  - Due: Week 14: 13th April, Thursday, in class. Any group member may be asked to present.
- **Deliverable 4:** Submit code (documented) and final report
  - Due: Week 14: 16th April, Sunday 23:55