Cairo University
Faculty of Engineering
Computer Engineering Department

CMPN450
Fall 2018

Pattern Recognition and Neural Networks
Project Document

Project Description:

In this project, you are required to implement a **writer identification system** that will be able to identify a given writer from their handwritten script or text. This writer identification system will prove very useful in a wide range of applications, such as the fields of security, biometrics and forensics. Not limited to that, but it also raised interest in the analysis of historical texts, historical musical scores with unknown composers, forged signatures in official documents, and many other applications.

You should implement a complete Machine Learning pipeline, i.e. the project should include (but not limited to) the following modules:

- Preprocessing Module.
- Feature Extraction/Selection Module.
- Model Selection and Training Module.
- Performance Analysis Module.

You will need to research about the topic, read scientific papers that tackle this application and similar applications, and do a literature survey that can help you identify the best approaches/ techniques that you can start with in order to improve the accuracy of your results.

You will work with the IAM Handwriting Database containing a large number of handwritten English scripts which you will divide it into:

- Training set to train your model.
- Validation set to tune the parameters of your model.
- Test set to report the results of your models.

Note that the dataset contains many forms of handwritten text including complete form text, extracted sentences and extracted words. **You will work with complete form text only.**
The dataset can be found here:
http://www.fki.inf.unibe.ch/databases/iam-handwriting-database

FAQ:

1. **What programming language(s) can I use?**
You can use either Python or R for this purpose. Both languages are very popular for machine learning.

2. **Is there any restriction on the techniques or approaches to use in any phase of the project?**
You are free to use any approach or technique you find appropriate for the problem in hand. It’s actually your task to find out the best combination of techniques that will

yield the best results. However, you are ***limited only*** *to classical machine learning methods (Bayesian Classifiers, K-NN, Linear/Logistic Regression, Neural Networks, Support Vector Machines, Principal Component Analysis, etc.). You are* ***not allowed*** *to use deep learning techniques.*

3. **How to find research papers for this topic?**
   There are many resources on the web tackling this problem. You can start by creating an account on EKB (Egyptian Knowledge Bank) with your university-provided student's email, which will give you wide access to a huge number of research papers and journals.

4. **How will our project be tested?**
   We will test with our own test set in order to standardize the way all teams are graded and ranked.

5. **How will our project be ranked?**
   Please refer to the "Grading" section for the grading criteria. The teams will be ranked according to some formula comprising the results accuracy, as well as, the time taken to generate the results, with the larger weight for the results accuracy.
   For example, Team X had results with accuracy 92.0%, with running time = 5 seconds, will be ranked above team with accuracy 80.0%, with running time = 2 seconds.

Team Formation:

A team is formed of 3-4 members. Please fill in your names in this Google Sheet: https://docs.google.com/spreadsheets/d/1IfzcNotRsgXJaJSun0XzeoU6KyYhg_gR_cn6nsmEpfs/edit?usp=sharing

Deliverables:

1. **Final Report**:
   The **body of the report** should include the following sections fully and clearly described, with all work done and approaches adopted. You should include also the unsuccessful trials.
   I. **Project Pipeline.**
   II. **Preprocessing Module.**
   III. **Feature Extraction/Selection Module**
   IV. **Model Selection and Training Modules.**
   V. **Performance Analysis Module.**
   VI. **(Optional) Any other developed modules.**
   VII. **Enhancements and Future work.**
   The report should include also a workload distribution between the team members.
2. **Codes:**
   - A zipped folder with the format ***code_[team number].zip*** containing all Python/R codes developed with a readme file including all packages or libraries needed to run this code.

3. **A readme text file** containing the names and IDs/BNs of the team members.

Project Schedule:

Phase	Week	Due date
Team formation.	Week 6	Friday 2nd November, 11:59 PM.
Final Delivery.	Week 13	Sunday 16th December, 11:59 PM.

Delivery Details:

- Deliverables should be sent by e-mail to *submissions.pr@gmail.com* using the subject:
  - **[Credit – Pattern Recognition][Team Number]**: for credit students.
  - **[Semester – Pattern Recognition][Team Number]:** for semester students.
- Don't print any document or submit the project on a CD. All submissions are electronic.
- There will be a **late penalty** for submissions in any of the two mentioned phases.
- **Any sign of cheating or plagiarism will not be tolerated. If one team got caught cheating or plagiarizing from other team, both teams will receive a ZERO grade in the project.**
- **Workload should be distributed fairly and equally. Team members who did not contribute effectively in the project will be penalized.**
- **All team members should attend the final discussion. A team member who fails to show up will get a ZERO in the discussion grade.**

Grading Criteria:

Item	Percentage	Notes
Report and Discussion	20%	This point includes the report submission, the quality of the report and the discussion with all team members. It also evaluates how all team members fully understand the details of the project and can elaborate on the examiner questions.
Results Accuracy	65%	Both results accuracy and running time will be taken into consideration by the given weights. However, it's not a linear formula but the ranking procedure will be based only on these two factors.
Running Time	15%	