Project Description

The course project is to give the students hands-on experience on applying what you have learned in the class to the problems of your interest. **You may choose one of the default course projects,** we will provide you with the corresponding dataset or links.

Default Projects

Image
Imbalanced Fish Image Classification
Fine-Grained Classification for Salmon Images

Text
Fake News detection
Author Identification

Grade Policy (Report + Presentation)

Final Report (15%)
We will provide a template, which is like a formal conference paper. Each team needs to submit a report for your project. The instructors and TAs will evaluate the report from the three perspectives: methodology (10%), results (e.g., test accuracy, 10%), writing (10%). **The report is due on 27 May 2021, no late day is allowed.**

Project Milestone (5%)
A 2-5 pages report for the progress and preliminary results of your project

Presentation (10%)
A 10-15 minutes short presentation for each team. We may invite other faculties or students to attend the presentation.

Notice

The **trained models and source code** should be submitted as well.

Team size

Students may do final projects solo or in teams of **up to 5 people**. Teamwork is encouraged. Larger teams are expected to do larger projects.
Contribution: In the final report we ask for a statement of what each team member contributed to the project. Team members will typically get the same grade, but we may differentiate in extreme cases of unequal contribution. You can contact us in confidence in the event of unequal contribution.

Resources

- You can use any deep learning/machine learning framework you like (PyTorch, TensorFlow, Chainer, MXNet, etc.)
- You may use any existing code, libraries, and refer to any papers, books, online references, etc. for your project. However, you must cite your sources in your report and indicate which parts of the project are your contribution and which parts were implemented by others.
- Do not look at another CS412 group's code. Plagiarism is **NOT** allowed, and you will be graded as 0 for this project part.

Imbalanced Fish Image Classification

Project Introduction

In this project, we aim at designing an image classifier on the Fish2016 dataset. There are 42 classes in the dataset. The number of each class has a large variant. You need to come up with some algorithms that can train a robust classifier on the imbalanced data.

Data Preparation

1. Download data from https://zjuintl-my.sharepoint.com/:u:/g/personal/gaoangwang_intl_zju_edu_cn/Ecj6wLTmfLlPhUH5oV7R-ikBcX9x-z4DQZrCEPPTiimeKw?e=nSMblp.
   (Do **NOT** spread the data. This may contain a license issue.)
2. Split the dataset into training and validation. Specifically, the first 3/4 images of each class are used in the training. The rest are used as validation.

Classifier

1. Pytorch is recommended for this project. You can use pre-trained models from https://pytorch.org/vision/0.8/models.html.
2. Try to come up with some ideas for imbalanced data classification. For example, you are free to design some loss function.

Evaluation

the mAP is the evaluation metrics. Report the overall mAP and AP of each class on the validation set. You can check https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html for computing the score. Compare with the baseline method that does not include any imbalanced data processing strategy.

Some Image Examples

Fine-Grained Classification for Salmon Images

Project Introduction

In this project, we aim at designing an image classifier on the Salmon images. Different salmon species look very similar. You are required to design a classifier to distinguish different salmon species.

Data Preparation

1. Download the data from https://zjuintl-my.sharepoint.com/:u:/g/personal/gaoangwang_intl_zju_edu_cn/EdT_p6M4LjxJtLYe1pqUKhIBOKCkaL0QSrHhZdbD46_dyA?e=SdS21J.
   (Do **NOT** spread the data. This may contain a license issue.)
2. Split the dataset into training and validation. Specifically, the first 3/4 images of each class are used in the training. The rest are used as validation.

Classifier

1. Pytorch is recommended for this project. You can use pre-trained models from https://pytorch.org/vision/0.8/models.html.
2. Try to come up with some ideas for fine-grained classification.

Evaluation

the mAP is the evaluation metrics. Report the overall mAP and AP of each class on the validation set. You can check https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html for computing the score. Compare with the baseline method that does not look at fine-grained information.

Some Image Examples

Fake News detection

Project Introduction

In this project, we aim at developing a machine learning program to identify when an article might be fake news. Please see the details with the following link: https://www.kaggle.com/c/fake-news/overview

Data Preparation

1. Download data from the Blackboard (Fake_News_data.zip)
2. train.csv: A full training dataset with the following attributes:
   - id: unique id for a news article
   - title: the title of a news article
   - author: author of the news article
   - text: the text of the article; could be incomplete
   - label: a label that marks the article as potentially unreliable
     - 1: unreliable
     - 0: reliable
3. test.csv: A testing training dataset with all the same attributes at train.csv without the label.

Evaluation

The evaluation metric for this competition is accuracy, a very straightforward metric.

$$\text{accuracy} = \frac{\text{correct predictions}}{\text{correct predictions} + \text{incorrect predictions}}$$

Accuracy measures false positives and false negatives equally, and really should only be used in simple cases and when classes are of generally equal class size.

Notice

1. Pytorch or Tensorflow is recommended for this project.
2. Though a high accuracy or good performance is desired, we are looking forward to more creative ideas and your problem-solving skills.
3. You may refer to available solutions from the internet, but do not copy all the codes from them. Try to propose your solution and do cite the codes and references if your implementation is based on them.
4. Besides solving the tasks in these projects, I do recommend you guys to identify the current challenges in these tasks and the limitations of your methods or current state of the arts.

Author Identification

Project Introduction

In this project, you are challenged to predict the author of excerpts from horror stories by Edgar Allan Poe, Mary Shelley, and HP Lovecraft.
Please see the details with the following link: https://www.kaggle.com/c/spooky-author-identification/overview

Data Preparation

The dataset contains text from works of fiction written by spooky authors of the public domain: Edgar Allan Poe, HP Lovecraft, and Mary Shelley. Your objective is to accurately identify the author of the sentences in the test set. You can download data from the Blackboard (Author_Identification_data.zip)

File descriptions

- train.csv - the training set
- test.csv - the test set

Data fields

- id - a unique identifier for each sentence
- text - some text written by one of the authors
- author - the author of the sentence (EAP: Edgar Allan Poe, HPL: HP Lovecraft; MWS: Mary Wollstonecraft Shelley)

Evaluation

Submissions are evaluated using multi-class logarithmic loss. Each id has one true class. For each id, you must submit a predicted probability for each author. The formula is then:

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log(p_{ij})$$

where N is the number of observations in the test set, M is the number of class labels, (log) is the natural logarithm, (y_{ij}) is 1 if observation (i) belongs to a class (j) and 0 otherwise, and (p_{ij}) is the predicted probability that observation (i) belongs to a class (j).
The submitted probabilities for given sentences are not required to sum to one because they are rescaled before being scored. Each row is divided by the row sum. To avoid the extremes of the log function, predicted probabilities are replaced with $max(min(p, 1-10^{-15}), 10^{-15})$.

Notice

1. Pytorch or Tensorflow is recommended for this project.
2. Though a high accuracy or good performance is desired, we are looking forward to more creative ideas and your problem-solving skills.
3. You may refer to available solutions from the internet, but do not copy all the codes from them. Try to propose your solution and do cite the codes and references if your implementation is based on them.
4. Besides solving the tasks in these projects, I do recommend you guys to identify the current challenges in these tasks and the limitations of your methods or current state of the arts.