Machine learning for vision and multimedia
(01URPOV)

Project templates, requirements and resources

2025-2026

Politecnico di Torino

Project timeline

Proposal

- Project scoping: identifying the problem/dataset/metrics
- Proposed by the students and approved by the instructors

Intermediate

- Dataset management: data split, annotation, augmentation and loading
- Opportunity to gather feedback during the project

Final delivery

- Model implementation and training, analysis of experimental results
- Written report + presentation and discussion

Project discussion

- Two dates per session:
  - the day of the written exam(s)
  - end of each exam session
- Booking will be managed via a form shared prior to each exam session
- A preliminary check will be performed on the project prior to the oral discussion

Applications

- Valid applications include the **training** of a ML model for multimedia data of your choice
- Draw inspiration from **other courses you are following**, real-life problems, or
- … have a look at recent publications from top-tier conferences, or challenges

Inspiration (I): papers

- CVPR: IEEE Conference on Computer Vision and Pattern Recognition
- ICCV: International Conference on Computer Vision
- ECCV: European Conference on Computer Vision
- NIPS: Neural Information Processing Systems
- ICML: International Conference on Machine Learning
- arXiv: Largest preprint archive for ML/CV research. Preprints are freely accessible, but are not peer-reviewed and vary drastically in quality. If you are relying on an arXiv submissions, make sure that it is a valid paper (when in doubt, ask)
- PapersWithCode: Collection of ML papers with links to code and comparison table on many benchmark datasets
- TwoMinutesPapers: YouTube channel with short videos summarizing recent papers in ML/AI

Inspiration (II): challenges

- Kaggle is the largest platform machine learning competition datasets. In most cases, the challenges and datasets remain available after the challenge is over
- Many challenges are related to the analysis of images or multimedia data, such as:
  - https://www.kaggle.com/c/rfcx-species-audio-detection: Automate the detection of bird and frog species in a tropical soundscape
  - https://www.kaggle.com/c/yelp-restaurant-photo-classification: Yelp restaurant photo classification challenge

Inspiration (III): datasets

- In the past five years, hundreds of datasets have been published for different tasks. Besides the course slides and Kaggle, several web sites index existing datasets:
  - https://www.visualdata.io/discovery https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research
  - http://www.cvpapers.com/datasets.html
  - http://yacvid.hayko.at/
  - https://computervisiononline.com/datasets
  - https://datasetsearch.research.google.com/ Google search engine for datasets

Final project delivery

- The following content must be provided
  - The project code/notebooks
  - The weights of a trained model
  - A copy or link to the dataset, if not publicly available
  - A 6-8 pages paper that describes the application, machine learning techniques, hyper-parameters, dataset, performance metrics and results
  - The self-assessment checklist
  - (optional) Experiment log
  - (optional) Supplementary material

Project discussion

- The project discussion typically lasts 15–30 minutes per group and is organized as follows:
  - 10 minutes presentation of the approach and main results
  - 5–15 minutes Q&A
- Discussions may be in person or remote
- During the discussion individual contributions will be assessed and a live demo can be requested

Final report template

- The final report should be 6–8 pages and follow the structure of a scientific paper
- **Use the template provided** so we can fairly compare all student projects
- The following is the suggested structure, which is suitable for most projects.
  - **Title, Authors**
  - **Abstract** (not more than 300 words)
  - **Introduction (15%)**
  - **Data (15%)**
  - **Methods (30%)**
  - **Experiments (30%)**
  - **Conclusion (10%)**
  - **References**
- Be precise: would you be able to reproduce your own work based on this description?

Final report template (II)

- **Introduction**: this section should describe the problem, why it is important, and a brief overview of the approach you followed
- **Dataset**: this section should describe the data you have used for the project.
  - What type of data is it? Where did it come from?
  - If you collected the data, describe the collection and annotation process in detail. Describe problems and challenges you encountered during the process and how you fixed them
  - If you used a public dataset, include a brief description, a reference to the original source, and describe any additional operations you performed: did you select a subset of the data? Did you use different or additional labels? Did you mix data from different datasets and how?
  - How much data are you working with? How did you split the data into a training / validation / testing split?
  - Did you have to do any preprocessing, filtering, or other special treatment to use the data in your project?
  - Did you use a subset of the training set to do the hyper-parameter search in order to reduce the computational requirements?
  - How did you perform data augmentation?

Final report template (III)

- **Methods**: this section should describe your approach for solving the problem that you described in the introduction. Show yow you applied skills and techniques from the classes and labs to solve it. It may be helpful to include figures or diagrams to illustrate your network architecture, your training methodology or a comparison with other methods
  - Why do you believe this approach is correct? Did you consider alternative approaches?
  - If you used an existing implementation as starting point, insert a reference (as a footnote) and discuss any high-level changes you have made.
  - Describe the model and loss used.
  - How did you select the hyper-parameters? What process did you follow?
  - What was the most challenging part in the design/implementation and how did you solve it?
  - How did you evaluate performance? Which metrics did you use? Did you use an independent test set, K-fold cross-validation, or other?

Final report template (IV)

- **Experiments**: this section should report the results of the experiments that you have performed. The exact structure will depend on the project, but it should include at least the performance of the best performing configuration, along with the loss curve on the training and validation set.
- More experiments are expected for larger groups and/or if the dataset and methodology is simple/based on available code.
- If your goal is to replicate a previously published paper, include a comparison with the original published results
- Examples of results that could be included:
  - Comparison with previously published methods
  - Removing parts of the systems to evaluate their impact on performance (ablation study)
  - Comparing different hyperparameters/architectural choices
  - Use visualization techniques to gain insight on how the model works
  - Discussion of cases where the model fails and possible reasons

Final report template (V)

- **Conclusions**: this section should summarize the key results
  - Focus not only on the final performance, but mostly on **what you have learned**
  - If the results are negative (you were not able to achieve high performance, or you were not able to replicate the results from the literature), focus on the challenges you faced and how you solved them/could be solved in the future
  - Suggest ideas for future extensions

A note on performance

- Achieving high performance on a given problem is often a matter of resources ($$$ and time)
- Reproducibility of ML papers is notoriously challenging and likely to fail
  - Different random initialization, environment, hyper-parameters may lead to very different results
  - Crucial details may not be reported
  - Experiments that fail are usually not reported
- In the evaluation, the obtained performance will be balanced against the complexity of the problem, data and network. Overall, the **quality** of the experiments and training will be assessed. Examples of quality indicators include, but are not limited to:
  - how carefully were the hyper-parameters selected?
  - do the experiments show obvious signs of overfitting?
  - Are the performances properly validated on a validation and test set?
- In the report, focus on **what you have learned**, not only on the raw performance.

Project assessment

- Project assessment will be roughly as follows:
  - Introduction (5%)
  - Dataset (10–20%)
  - Methods (30%)
  - Experiments (35–45%)
  - Presentation (10%)
- Up to 2 extra points may we awarded for projects characterized by additional complexity, e.g., using techniques not seen during the course and/or with a substantial amount of experiments