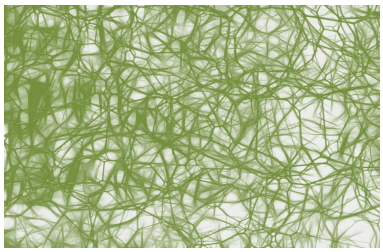
COMSM0098: Applied Deep Learning (2021-22)

Your Coursework

(task for *up to* 3 student teams, coursework, 100%)

Re-produce Published Research Paper



Forming your Teams:
Register your team of up to 3 people (i.e. one, two or three students) online at:
https://doodle.com/poll/zku8kh57a3atenbe
Post registration, teams can split but cannot merge, to avoid any copying of code or ideas.

Each member of the team should submit an exact copy of the final submission on Blackboard by the deadline. The report (see below) should note the full names and usernames of all members of the team.

It is up to each team to decide their best strategy to tackle this coursework, i.e. whether to divide the tasks below, or to work together on all tasks. Contributions of team members need not be explicitly stated.
***However,** by submitting a group coursework, you are implicitly acknowledging that all members of the team contributed equally. If this is not the case, you should email unit director with details of any issues encountered during the coursework.*

Task Brief:
This assignment gives you the opportunity to appreciate the work required in replicating published research from a publicly available dataset and manuscript. It allows you to reflect on the experience of reproducing published results and potentially outperforming on your replication.

Gathering all the knowledge you acquired from the lectures and labs, read the paper below carefully and replicate the required results (Note: you are not required to re-produce all the paper's results). Feel free to take any pieces of code from the labs as a baseline, but the rest of the code should be originally yours.

The Paper:

Michele Valenti, Aleksandr Diment, Giambattista Parascandolo, Stefano Squartini, Tuomas Virtanen. DCASE 2016 ACOUSTIC SCENE CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS. In Detection and Classification of Acoustic Scenes and Events, 2016.
http://dcase.community/documents/workshop2016/proceedings/Valenti-DCASE2016workshop.pdf

Note that our choice for paper is based on its simplicity and similarity to your labs, rather than its superior performance or exceptional novelty.
Please read the following information carefully before attempting the replication.

1) Architecture:

For this coursework, you will only be asked to implement and replicate the **convolutional** architecture (Fig 2 in the paper). You should not replicate the MLP or GMM proposals in the paper.

2) Selected Dataset:

While the paper has been evaluated on four-fold cross validation, we ask you not to do that. Instead, **we provide you with one train/test split from the dataset**, already formulated for your use. [see **Dataset and Useful Code**]

3) Required Results:

In replicating the results, we expect you to provide code and your results for:

1. We ask you to produce results for the two-layer log-mel CNN with sequence length of 3, marked here of Table 2 in the paper.

   Table 2: Accuracy comparison for different systems and training modes ("non-full" and "full"). Neural architectures are identified by their number of hidden layers.

system	*seq. len.* (s)	*accuracy* (%)	
		non-full	*full*
two-layer MLP (log-mel)	-	66.6	69.3
one-layer CNN (log-mel)	3	70.3	74.8
two-layer CNN (log-mel)	3	75.9	**79.0**
two-layer CNN (MFCC)	5	67.7	72.6
baseline GMM (MFCC)	-	-	72.6

2. Fig 3: You should be able to produce the confusion matrix of all classes from your code. **The code to produce the confusion matrix should be part of your submission.**

4) Other details:

There are a few implementation details that were left unspecified by the authors. We specify these below to resolve ambiguities:

- The paper uses max pooling to perform "destruction of the time axis". We instead used Adaptive Max Pooling for this, and therefore the frequency bands are not guaranteed to be non-overlapping. We advise that you use this and only specify the number of frequency bands. https://pytorch.org/docs/stable/generated/torch.nn.AdaptiveMaxPool2d.html

- The paper refers to "sequence splitting". Please check the code README for guidance on how this can be achieved.

5) Our replicated results:
Replicating papers rarely produces exact results as those reported in the published papers. It is highly advisable to publish one's code with the paper, however this is very infrequently adopted by researchers.

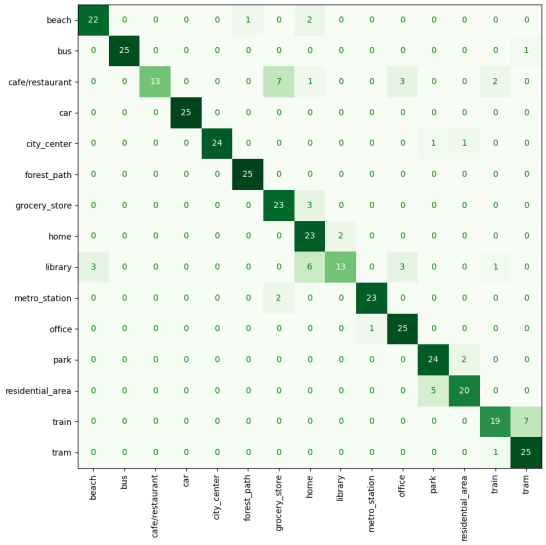
We have replicated the paper's results in PyTorch for you. We provide the corresponding table that we could re-produce,

using the data files and train/test split available to you. These are the results you are attempting to reproduce.

System	Seq len (s)	Accuracy (%)	
		non-full	full
Two-layer CNN (log-mel)	3	84.5	87.2

As our reimplementation produces better results on the chosen data splits, this is the replication we expect you to reproduce.

Also our replication of Fig 3 (from the paper) to match the results above is here:



Dataset and Helpful Code:

You can find resources we've prepared for you for this project at: ADL OneDrive
https://github.com/luke-who/adl-cw/tree/main/ADL_DCASE_DATA

The OneDrive directory includes a README file, which you should refer to understand what each file contains. The dataset contains a development and evaluation set. All of your final results should be produced on the evaluation set. **Do not train your model on the data in the evaluation set.**

Final Submission:

1. *An original code, **based on PyTorch** (other software engines won't be accepted – we won't accept Keras or Tensorflow), replicating the published paper. You can use your lab code from any or all group members. **We aim to run your code on BC4, so ensure it compiles and runs**.*

2. *A report in the IEEE conference format (https://www.ieee.org/conferences/publishing/templates.html) of up to 5 pages including references, submitted in PDF format. The report should include the following sections:*

   A. ***Title** and **Team members** (names and usernames)*

   B. ***Introduction**: Definition of the problem addressed by the paper Pan et al (in your own words)*

   C. ***Related Work**: A summary of **more recent** published papers (i.e. after Valenti et al was published in 2016) attempting to address the same problem (up to 3 works).*

   D. ***Dataset**: A description of the dataset used, training/test split size, labels and file formats.*

   E. ***Input**: Explain what log-mel spectrograms are. Give 1-2 examples, by plotting these as spectograms, and showing the ground-truth for these. We have provided code to help you visualise the log-mel spectogram.*

   F. ***CNN Architecture (Valenti et al)**: Describe through a table the architecture and all its details.*

   G. ***Implementation Details**: Summary of the steps you have undertaken to replicate the results, train the data and obtain the results, including any decisions you needed to make along the way. Do not include any pieces of code, but you can include pseudo-codes if needed.*

   H. ***Replicating Quantitative Results**: You need to present your results for table 2 as above, as well as Fig 3*

   I. ***Training curves**: Include your training/test loss curves for your models, and comment on any overfitting in your training. The curves here should correspond to the same run as those in the reported table (Section H). These curves could be directly retrieved from Tensorboard.*

   J. ***Qualitative Results:** This section should include sample success and failure cases based on your algorithm. In presenting these examples, you can plot/display the input log-mel spectrogram, ground-truth and predicted classes. Particularly: (a) find 1 good example, where your prediction works. (b) find 2 problematic examples where your prediction can be criticised. These should represent two different modes of failure.*

   K. ***[65+] Improvements**: Using the same CNN architecture, propose, implement and test one improvement you made to your results).*
   ***Note:** if you describe multiple improvements, we will give you the lower mark (rather than the higher one), so choose the one you believe in. Cover any implementation details required to understand and replicate your modifications. Report your improved results in tabular format for all metrics. Do not include any pieces of code, but you can include pseudo-codes if needed.*
   ***Note:** Your improvement should be made using the same dataset, train/test split and evaluation metrics used earlier. Improvements can include changes to architecture, hyper-parameters, data augmentation or learning algorithm. Your choice should be justified theoretically and experimentally.*

   L. ***Conclusion and Future Work**: Summarise what your report contains in terms of content and achievements. Suggest future work that might extend, generalise or improve the results in your report.*

Marking Guideline.

Note: Code and report will be checked for plagiarism. Proven plagiarism will result in a 0 grade on this coursework for the whole team.
50-54
To pass this assignment, you must produce original complete (compiles and runs on BC4 using batch-mode command and PyTorch) code that replicates the results in the paper. You should produce a report with sections A-F correct and satisfactory. A partially-complete and correct attempt to

address sections G, H, I and L is included (i.e. excluding J and K). Any errors or misses do not significantly affect a "replication of results" effort. Replication results (Section H) are within 5% accuracy on all metrics.

55-64
In addition to the above, sections F, G, H, I and J would be complete, correct and reflective of your understanding of the code and the implementation. All sections (except K) are completed to an acceptable standard. Replication results (Section H) are within 1% accuracy on all metrics.

65-70
In addition to the above, a satisfactory attempt to provide improvements (K) on the published results have been achieved, correctly described, with improvements to the results. Marginal improvements will be accepted.

70-75
In addition to the above, the presentation given was to a very good standard with almost no areas of weakness. The proposed improvement is far from random and has been carefully thought of in light of the problem and misclassification errors. Section J should include interesting (rather than random) success and failure cases, with explanations of failure cases. The report's organisation and structure should be very good.

75-80
In addition to the above, the report should be submit-able to a B-class peer review conference or workshop, i.e. it shows excellent understanding, correct and complete showcasing of the approach. Statements are concise, and any jargon out of implementation details is avoided. The chosen related work reflects state of the art on this problem. Extensive evidence of analysis, creativity & originality in concise content presentation should be shown. Code is commented, and could be easily understood and re-used by a reader.

80-100
In addition to the above, the produced code and report are exemplary, and could be given as an example for an attempt to replicate this published work. Improvements in results are beyond marginal.

General Guidelines (Department Regulations):

Deadline

The deadline for submission of all optional unit assignments is 13:00 on Friday 10th of December.
The intention is that you submit by 12:00pm and keep the last hour as emergency reserve for e.g. technical problems. In case of problems with your submission, you must e-mail coms-info@bristol.ac.uk before the 1pm final deadline to avoid your work being counted as late.
Students should submit all required materials to the "Assessment, submission and feedback" section of Blackboard for the coursework unit **COMSM0098**- it is essential that this is done on the Blackboard page related to the "With Coursework" variant of the unit.

Time commitment

You are expected to work on both your courseworks in the 3-week period from Week 9 to Week 11 as if it were a working week in a regular job, that is 5 days a week for no more than 8 hours a day. It is up to you how you distribute your time and workload between the two units within those constraints.

You are strongly advised not to try and work excessive hours during the coursework period: this is more likely to make your health worse than to make your marks better. If you need further pastoral/mental health support, please talk to your personal tutor, a senior tutor, or the university wellbeing service.

For this unit **COMSM0098 Applied Deep Learning**, the following support is available during the coursework period:

Wk 9 Drop-in Session: Thur 25 Nov 10-11am
Wk10 Drop-in Session: Thur 2 Dec 10-11am

This information is available on your unit's webpage.

Academic Offences

Academic offences (including submission of work that is not your own, falsification of data/evidence or the use of materials without appropriate referencing) are all taken very seriously by the University. Suspected offences will be dealt with in accordance with the University's policies and procedures. If an academic offence is suspected in your work, you will be asked to attend an interview with senior members of the school, where you will be given the opportunity to defend your work. The plagiarism panel are able to apply a range of penalties, depending the severity of the offence. These include: requirement to resubmit work, capping of grades and the award of no mark for an element of assessment.

Extenuating circumstances

If the completion of your assignment has been significantly disrupted by serious health conditions, personal problems, periods of quarantine, or other similar issues, you may be able to apply for consideration of extenuating circumstances (in accordance with the normal university policy and processes). Students should apply for consideration of extenuating circumstances as soon as possible when the problem occurs, using the following online form:

If your application for extenuating circumstances is successful, it is most likely that you will be required to retake the assessment of the unit at the next available opportunity