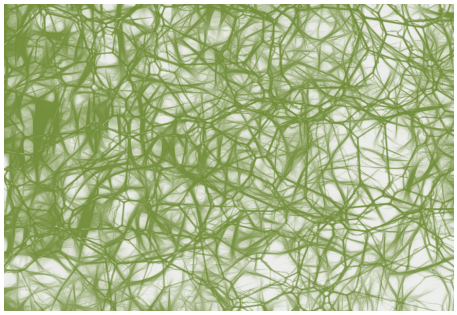
COMSM0018: Applied Deep Learning (2018-19)

# Your Team Project

(task for *up to* 3 student teams, summative, 60%)

## Re-produce a 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/qyxksy3x9cwituhq
*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 SAFE by the deadline. The report (see below) should note the full names and UoB emails of all members of the team.*

*It is up to each team to decide their best strategy to tackle this problem, i.e. whether to divide the tasks below, or to work together on all tasks. Contributions of team members need not be explicitly stated.*

**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 these results.*

*Gathering all the knowledge you acquired from the lectures and labs, read the paper below carefully and replicate its 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:**

**A. Schindler, T. Lidy, and A. Rauber. Comparing shallow versus deep neural network architectures for automatic music genre classification. In *Proceedings of the 9th Forum Media Technology (FMT2016)*, St. Poelten, Austria. URL: https://publik.tuwien.ac.at/files/publik_256008.pdf**

Note that our choice for paper is based on its simplicity and clarity in all the hyper-parameters used, rather than its superior performance or exceptional novelty. This is not typical for published papers.

A few clarifications of the paper are worth noting to avoid confusion:

- Use the paper for all details of the two architectures and hyper-parameter choices. You are attempting to reproduce the results, not try a different solution.
- We will only be using ONE of the four datasets tested in the paper: **GTZAN**. You are not expected to download or test any of the other datasets. Using another dataset is also not a valid extension (Section K below).
- Instead of the four-fold cross validation, we are offering you **one train/test split**. While this makes the results not directly comparable to the paper, *any random split is unlikely to be directly comparable to the published results*.

**Dataset and Helpful Code:**
You can find resources we've prepared for you for this project at: https://goo.gl/VcTvNq

**Important Note:**

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 thus provide below the corresponding table that we could produce, using the provided train/test split for your reference. We do not provide base results for the data augmentation, note Section K below.

| D | Model | raw | max | maj | ep |
|---|---|---|---|---|---|
| GTZAN | shallow | 62.05 | 81.20 | 75.20 | 100 |
| | deep | 69.17 | 81.20 | 81.60 | 100 |
| | shallow | 61.15 | 78.80 | 75.20 | 200 |
| | deep | 69.12 | 84.00 | 84.40 | 200 |

**Final Submission:**

1. *An original code, **based on Tensorflow 1.2** (other software engines won't be accepted – we won't accept Keras either), replicating the published paper. You can use your lab code from any or all group members. We aim to run your code on BC4 using Tensorflow 1.2, so ensure it compiles and runs.*

2. *A report in the IEEE conference format (https://www.ieee.org/conferences_events/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 Schindler et al (in your own words)*

   C. ***Related Work**: A summary of published papers attempting to address the same problem (up to 3 works). These could be from the references of the paper itself, or otherwise.*

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

E. ***Spectrogram Introduction****: Explain the concept of spectrograms and how these encode the audio file. Give 1-2 examples visually.*

F. ***Method (Schindler et al)****: up to 20 lines summarising the method/architecture and contributions of the work. Do not re-draw the architectures.*

G. ***Implementation Details****: Summary of the steps you have undertaken to replicate the architecture, 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 <u>your results</u> for the table above. (You do not need to implement the data augmentation at this step)*

I. ***Training curves****: Include your training/test loss (and accuracy) curves for your models, and comment on any overfitting in your 100 and 200 epoch results. The tables 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 spectrogram(s) in each case. Particularly: (a) find one or more examples that are incorrectly classified by the 'raw' measure and are corrected by 'max' or 'major'. (b) find at least one failure case where the correct class is the second/third class in the predictions. (c) find at least one failure case where the confidence of the correct class is low. For each case, discuss whether you can explain the failure.*

K. ***[60+] Improvements****: You have one of two options for this section:*
   i. *Implement the data augmentation approach from the paper, and <u>one additional</u>* improvement to the results in the paper (not mentioned in the original manuscript)
   ii. *Or, implement <u>up to two</u> potential improvements you made to improve the results in the paper (not mentioned in the original manuscript. [You do not need to do data augmentation].*

   *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. <u>Note:</u> any improvements should be made using the same dataset, training/testing split and evaluation metrics used earlier. Improvements can include changes to architecture, hyper-parameters and learning algorithm. Your choice should be justified theoretically and experimentally. Your choices could improve one or more of the three metrics 'raw', 'max', 'major'.*

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.*

**<u>Deadline:</u>**

*Deadline for submission via SAFE is*

***Wed 16 Jan 2019 17:59***

*Each member of the team should submit a copy of the final submission (code + report).*

**<u>Marking Guideline.</u>**

*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 Tensorflow 1.2) 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 10% error of expected accuracy (62% for Shallow/Raw and 81% for Shallow/Max), or better.*

*55-59*
*In addition to the above, sections 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. Reported errors are within 5% of the expected accuracy (62% for Shallow/Raw Raw and 81% for Shallow/Max), or better.*

*60-64*
*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.*

*65-69*
*In addition to the above, the presentation given was to a very good standard with almost no areas of weakness. The proposed improvements are far from random and have 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.*

*70-79*
*In addition to the above, the report should be submit-able to a B-class peer review conference, 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.*

*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. Code is commented, and could be easily understood and re-used by a reader.*