# Requirement for projects:

We want to make sure each of the team members takes care of one model or a pipeline. So that everyone can benefit from the task. You can treat a group-based project as a learning group given a specific task, but each of you still needs to implement your own DL pipeline.

The project proposal must clearly describe which member is taking care of what task and get it approved by TAs.

There are multiple goals for the group-based projects:

1. Get familiar with experiment design and dataset preparation for a specific deep-learning task.
2. Get familiar with SOTA models for the specific task.
3. Able to replicate SOTA models and apply them to new datasets.
4. Able to compare and conclude on the performance of SOTA models.
5. Collaborate with others to boost your knowledge on the specific task.
6. Presentation (we won’t ask for a written report, we ask for a recorded presentation and a replicate package for all experiments, your code will tell the methodology, and your results will tell your findings).

—-----------------------------------------------------------------------------------------

# **Topic 1: Recommender System**

Dataset: Tenrec

Paper: Tenrec: A large-scale Multipurpose Benchmark Dataset for Recommender Systems

This is a new benchmark dataset built for recommender system tasks. You can download the data from:<https://drive.google.com/file/d/1GEdxyHTJz_BN_X_0WbThIjurhjtFC8UI/view?usp=share_link>

The data (zipped) is 5 GB.

I would suggest you take a look at the paper from the authors describing how they collect the data and some tasks can be directly explored using this dataset.

This is an ideal dataset who are interested in general recommender system algorithms. It includes cross-platform user behaviors and involves both users' interaction with news articles and videos.

Candidate Projects:

* P1: Session-based Recommendation
* P2: Multi-task Learning for Recommendation
* P3: Transfer Learning for Recommendation
* P4: Cold-start Recommendation

**Fairness in Recommendation:**

**Objective:** Recommender systems, an essential part of various services, such as social media and e-commerce websites, increase customer satisfaction by customizing ads or recommending content according to the customer’s interests and provide benefits to the business as well. Fairness in these systems is essential as biased systems can have consequences such as perpetuating existing inequalities.

**Problem Statement:** Using the Tenrec dataset, investigate how gender and age bias might occur in the recommender systems. The dataset can be used for cold start recommendations as well as session-based recommendations. Biases can be investigated for both of them.

Step 1: You need to get familiar with the data, associated features and the four subsets of the Tenrec dataset. Perform feature analysis and report the findings.

Step 2: Divide data by age group and gender for the target dataset, e.g., session-based recommendations. Pick two/three models and generate results for different age groups and gender for each model.

Model selection: There are available SOTA models that have been used for the Tenrec dataset (<https://tenrec0.github.io/#leaderboard>). Models can be picked from here.

Step 3: To compare the performance of the models in each group, a baseline model can be trained on all the data. To identify if the model is biased towards a specific group, analyze the model performance by subgroups.

# **Topic 2: Fraud Detection**

Paper: Turning the Tables: Biased, Imbalanced, Dynamic Tabular Datasets for ML Evaluation

Dataset: Bank account Fraud Detection:

<https://github.com/feedzai/bank-account-fraud/tree/main>

Labels: Binary, 30 features:

<https://github.com/feedzai/bank-account-fraud/blob/main/documents/datasheet.pdf>

Detailed information about the dataset: <https://www.kaggle.com/datasets/sgpjesus/bank-account-fraud-dataset-neurips-2022>

The introduced datasets regard the detection of fraudulent online bank account opening applications in a large consumer bank. In this scenario, fraudsters will attempt to either impersonate someone via identity theft, or create a fictional individual in order to gain access to the banking services. After being granted access to a new bank account, the fraudster quickly maxes out the accompanying line of credit or uses the account to receive illicit payments. All costs are sustained by the bank, as there’s no way of tracing the fraudster's true identity.

Each instance (row) of the dataset represents an individual application. The label of each instance is stored in the "is\_fraud" column. A positive instance represents a fraudulent attempt, while a negative instance represents a legitimate application. The data is highly imbalanced, i.e., only about 1% of applications are labeled as positive. The data spans eight months of applications, which can be identified in the column “month”.

In addition to the base dataset, five dataset variants each containing specific types of data bias are created to allow practitioners to stress test both performance and fairness of ML methods. Each dataset consists of a total of one million instances of individual applications, using a total of thirty features.

**Problem statement:** Given a highly imbalanced labeled dataset, i.e. Bank Account Fraud (BAF), which provides 30 features for each account opening application, you need to provide ML/DL models that accurately predict the frauds.

Step 1: start with the baselines provided in this notebook:

<https://www.kaggle.com/code/lennart4711/baselinemodels-roc>

Try to improve each baseline with some hyper parameter tuning. Also add two more ML models.

Step 2: Investigate how the preprocessing applied on the data has improved the performance. Try to apply some other preprocessing on the data and check how they affect the performance of each model in step 1.

Step 3: Investigate SOTA approaches to handle high imbalanced data. Apply them on the data and compare the results with step 1 and 2.

Step 4: apply your best approach on all the five dataset variances and interpret the results.

Candidate Projects:

You can focus on improving the results for one of the six challenges.

# **Topic 3: Fact Verification**

Dataset: <https://tabfact.github.io/>

Paper: TabFact : A Large-scale Dataset for Table-based Fact Verification

**Objective:** In the future, AI-generated content will likely appear very often on online platforms. Thus, the importance of able to verifying AI-generated content is increasing too. In this project, we will explore the performance of LLM-based models for table-based fact verification given chatgpt-based table summaries, i.e., we aim to automatically verify the generated content from chatgpt based on tabular data.

**Problem statement:** ChatGPT has the potential to summarize information within Tables and answer questions based on the tables. In this project, we would like to explore the potential of checking the statements generated by ChatGPT leveraging Table-based Fact Verification technology.

There are three steps to achieve the goal:

Step 1: compare two different types (graph-based, fine-tuned LLM, prompt-based LLM) of SOTA fact-verification models on the TabFact dataset. The goal is to understand the pipeline for table-based fact verification and get familiar with existing SOTA methods that can be applied to this task. You can pick models from: <https://paperswithcode.com/sota/table-based-fact-verification-on-tabfact>

Step 2: design a new evaluation task by asking ChatGPT to reason about tabular data. Something like <https://chatexcel.com/>. You can also prepare the data manually. Think about a specific type of tabular data that you feel would be beneficial for industrial companies and ask relevant questions. There must be a strong motivation behind the design of the new evaluation dataset.

Step 3: Evaluate the performance of SOTA fact-verification models on the new benchmark you created in Step 2 and report your findings.

# **Topic 4: Time Series Forecasting**

# Paper: Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2022). Are transformers effective for time series forecasting?. arXiv preprint arXiv:2205.13504.

Benchmark datasets:

<https://github.com/cure-lab/LTSF-Linear>

**Objective**: Transformer-based approaches are popular in NLP and CV classification tasks, which motivates their applications in time series forecasting. Time series forecasting is a long-standing task with many real-world applications. In the above paper, the authors compare a simple linear model, i.e., LTSF-Linear, and report that it outperforms existing complex transformer-based time series forecasting approaches. However, there are new types of transformer-based approaches have come out.

**Problem statement**: for time series containing C variates, give historical data, predict future values at T future time steps. Your task is to further compare a simple linear model for time series forecasting with a recently published transformer-based approach, i.e., PatchTST (Nie, Yuqi, et al. "A Time Series is Worth 64 Words: Long-term Forecasting with Transformers." arXiv preprint arXiv:2211.14730 (2022).)

There are four steps to achieve the goal:

Step 1: Get familiar with the benchmark tests used in “Are transformers effective for time series forecasting?”, and replicate their results by selecting two models, i.e., LTSF-Linear and FEDformer.

Step 2: Replicate PatchTST following the evaluation pipeline replicated in step 1. You can find the code for PatchTST here: <https://github.com/yuqinie98/PatchTST>

Step 3: design a new evaluation task from one existing dataset on Kaggle, including competitions. Think about a specific type of data that is suitable for time series forecasting. There must be a strong motivation behind the design of the new evaluation dataset.

Step 4: Evaluate the performance of LSTF-Linear approaches, FEDformer and PatchTST and perform qualitative and quantitative studies.

# 

# **Topic 5: Sentiment analysis on Reviews considering text and image**

Dataset page: <https://jiachengli1995.github.io/google/index.html>

Google Local Dataset contains review information on Google map (ratings, text, images, etc.), business metadata (address, geographical info, descriptions, category information, price, open hours, and MISC info), and links (relative businesses) up to Sep 2021 in the United States.

Paper: “Visual Sentiment Analysis for Review Images with Item-Oriented and User-Oriented CNN” Code: <https://github.com/PreferredAI/vs-cnn>

**Objectiv**e: “A picture is worth a thousand words”, today’s reviews contain more than text. For instance, reviews on restaurants often contain pictures showing specific food that people commented on. Thus those reviews are naturally “multimodal”. The goal of this project is to explore sentiment expressed in reviews with images for multiple types of business from Google Local Dataset.

**Problem statement**: The training data contains <textual content in review, review rating, and images in review> for different types of business from Google Map. Your task is to explore visual only, text only and visual-text multimodal sentiment analysis models.

There are four steps to achieve the goal:

Step 1: Get familiar with the Google Local Dataset and prepare a dataset for the visual sentiment analysis task, following “Visual Sentiment Analysis for Review Images with Item-Oriented and User-Oriented CNN”. Think about two specific types of businesses that are suitable for visual sentiment analysis. There must be a strong motivation behind the selected types of businesses. For instance, you can check the “category” field from metadata and rank categories by the number of reviews containing images.

Step 2: Replicate the two models and case studies in “Visual Sentiment Analysis for Review Images with Item-Oriented and User-Oriented CNN” on the dataset created from Step 1. Report your findings, i.e., the kind of review images that connote positive or negative sentiment visually.

Step 3: Explore the potential of Multimodal aspect-based sentiment analysis on the created dataset from Step 1. Following Ling, Yan, and Rui Xia. "Vision-language pre-training for multimodal aspect-based sentiment analysis." arXiv preprint arXiv:2204.07955 (2022). <https://github.com/NUSTM/VLP-MABSA>. Report your findings.

Step 4: Explore the potential of aspect-based sentiment analysis on the created dataset from Step 1, using <https://github.com/yangheng95/PyABSA>. You just need to pick one SOTA model. Report your findings.