DOPPELGÄNGER
SYNTHETIC DATA GENERATOR

INDIVIDUAL REPORT FOR **NIRAV JANAK PARIKH (A0213573J)**

GRADUATE CERTIFICATE IN PATTERN RECOGNITION SYSTEMS – PRS GROUP 19

Your personal contribution to project?

It was truly a group project which each member contributing to key aspects of the project. We all had good project proposals, and I was grateful that the team converged on my proposal. As this project was linked to some real business problems we have all encountered in real-world situations, we could all contribute from our respective experiences in our domains. While we all collaborated as a team on all project modules, specifically I lead the effort on the developing the product framework and designing the solution architecture. From a development perspective, my contribution was primarily in creating the prototypes for the synthetic data generator model using PCA inverse transform for the numeric data set . I also researched & prototyped the use of Variational AutoEncoders to generate synthetic images (although to be fair Anirban was the brains behind most of our production code). Given my familiarity with the business problems & solution architecture, I volunteered to do the final presentation video (with Prashant creating the embedded marketing video). I also contributed to the design & presentation aspects of the front-end application (which was actually developed by Prashant including productionizing the ML models). Finally, I learnt a lot from Taksh who is a DevOps expert and, amongst his other project contributions, was key to design of our deployment architecture.

What you have learnt from the project?

While it was great to get exposure to all aspects of the project, the significant practical challenges of generating good quality synthetic data was quite a learning experience. While it sounds straight-forward enough in theory, in practice it quite the opposite. The different data types like numeric, image & time-series require quite distinct approaches & modelling techniques, and even within the same data type different data sets have their own unique pre-processing challenges. It proved almost impossible to create a generic one-size fits all solution that works for every dataset. But what I did learn is that it is possible to create a 'Generator-Validator-Evaluator' type solution framework that can be applied towards most types of data – albeit with customized selection of the pre-processing techniques and ML models depending on the nature of the real data set.

On a more technical level, it was intellectually very satisfying to be able to learn & use all three ML techniques ie Unsupervised (PCA as a Generator), Semi-supervised (Auto-Encoders as Validator) and

Supervised (CNN as Evaluator) within the same project. I was also personally quite fascinated to learn about Variational Auto-Encoders (VAE). We had learnt during the lectures about Auto-Encoders and how they capture and abstract the key patterns within a dataset – which can then be used for things like anomaly detection. But VAEs take this one step further by normalizing the abstracted latent space into a distribution. By perturbing this latent distribution, it is possible to generate completely new data which is still consistent with the patterns of the original data. This technique has been well documented, and we adopted this approach to generate our synthetic image data.

How you can apply this in future work-related projects?

The Synthetic Data Generator has several practical applications in my workplace. I work on the Markets & Trading side of the bank, where we are required to archive vast amounts of historical data for regulatory audit trail purposes. We have data going back years covering not only market history but very specific data about all our key customer trading preferences and how their trading strategies adapt to different market conditions. This is a goldmine of information, but it is for most parts it has very restricted access since it contains confidential & commercially sensitive client information. But interestingly, it is not the actual transactional data but rather the broader patterns embedded within this data that are most useful for profiling the customer behaviour & their trading preferences.

Hence if we can generate a model that extracts these data patterns in a synthetic data set without exposing the details of the actual client or the specific of their trades, we can then make this synthetic data available to our market analysts and use this to create prediction models that allows us to understand our client trading preferences and to be able to be service them more proactively in changing market conditions. For example, we could identify that particular managed funds (say Pension Funds) consistently reduces risk as market volatility increases, we can then pro-actively target them with risk reducing trade ideas whenever our models predict an expected increase in volatility. On the other hands, we may find that another set of managed funds (say Hedge Funds) can be clustered differently, and their trading pattern suggests they like to increase their exposure when volatility increases. We could then target them with risk increasing strategies during those times. This would then allow us to identify two client segments each of whom react exactly opposite to each other for the same market conditions ie we have found both buyers & sellers to whom we can intermediate the transaction thus earning revenue on both sides of the trade without the Bank actually taking any market exposure!!

There are dozens of similar use-cases that can be worked upon if we can synthesise our historical data such that it can be made available to our data analysts without compromising the privacy, confidence and trust of our clients.

--- END ----