



Udacity Machine Learning Engineer Capstone- Fintech App

The project I would like is in the domain of Finance. The motivation behind the project is very personal. Last Black Friday, there were very nice laptop deals and I had the intention to buy one. After two days of intense research, I found the one that matches my criteria and went online on a certain Canadian bank's website that was offering loans at discounted interest rate and applied for a loan. The application process itself took a couple of minutes, but the decision of loan acceptance took two full weeks. Obviously, I wasn't able to buy the laptop that I wanted and felt completely frustrated. After digesting my frustration, I thought to myself. My background is in Finance and I know that banks use machine learning algorithms to make their loan approval decisions. But these algorithms are running on the banks infrastructure which is often legacy technology and very bureaucratic in decision making. What if the machine learning algorithm is brought closer to the customer and an instant decision is made. There are challenges to this solution which I'll talk about below, nevertheless, a clear gap exists between application time and approval time and this gap to me is worth exploring.

The apparent reason why these approval decision take so much time is due to extensive credit checks, document gathering and complicated proceedings to wire the money to the customer in case they get approved (Finder.com). However, with the rise of Open Banking and the use of Open Data APIs in Canada, information sharing is becoming significantly easier and a simple loan approval decision cannot take this much time. Additionally, the long processing time is a clear burden on customer experience that can be partially or completely alleviated. Keeping in mind the tip given in the Future of Machine Learning interview about ML being run on customer's mobile device to improve customer experience, I decided to explore this gap. The solution I propose is to build a machine learning-powered web application that processes loan applications and give instant decisions with maximized accuracy on false negatives. From a bank's perspective, I stand to lose a lot both in customer experience and brand reputation if I give an approved loan decision but after reviewing documents and submitting the money transfer I realize they shouldn't have been approved. So, this is the evaluation metric I will seek to minimize.

In terms of a benchmark, I did some industry research on the type of lenders that currently exist in Canada. The industry is dominated by three players, the online lenders, peer-to-peer lenders and banks & credit unions. In terms of processing times, online lenders and P2P lenders give approval decisions in minute to days intervals. On the other side, banks and credit unions take a few weeks to approve a loan. In terms of model accuracy, this would be undisclosed information and I would not be able to know exactly. However, I could safely assume that banks have the know-how and expertise to achieve a minimum 70% accuracy on their small loans. During my study program at university we did some research and conducted interviews and arrived at this number. It's an average rather than a precise one as banks have the capacity to achieve higher number. Therefore, there are two benchmarks I should maximize, so a combined metric would include the processing time as well as accuracy of decision. For the processing time, I would develop the web app to be part of the bank's website or mobile application so decision will be made instantly (similar to online lenders) but also verify that the model parameters used are achieving high accuracy without compromising model-generalality (overfit).

For the dataset, I found a Kaggle dataset that is most similar to a standard bank application data. It has some basic profiling questions such as income, education, marital status and employment. In terms of

design, I follow the agile methodology to develop the machine learning application. I indicate below the scrum board with user stories and related tasks.

As a ML dev, I would like to know what historical data I have, patterns, customer segments, etc.

Task 1: Data Cleaning & wrangling
Task 2: Data visualization and plotting

As a ML dev, I would like to know which machine learning algorithms I can use depending on the data/ As a bank I would like to use an accurate model

Task 3: Modelling the data according to patterns (account for imbalanced class)
Task 4: Choose the best performing model

As a ML dev, I would like to use the model outside of Jupyter labs and test it for production

Task 5: Pickle the model that performs best
Task 6: Get random data and test it as if it's in production

As a ML dev, I would like to have an API to connect directly to my model and make predictions

Task 7: Develop a RESTful POST and GET operations to be able to make predictions

As a customer, I would like to have a UI to enter my information

Task 8: Make a data schema in a database that matches the data input fields
Task 9: Create UI template in HTML

As a ML, I would like to have the UI connect to backend API and database to make the prediction

Task 10: connect the web app to the backend through routing

As a customer, I would like to be able to access the product easily

Task 11: Build a web page with the web app to be integrated into an existing website

As a customer, I would like the app to be interactive and user friendly

Task 12: Continue building on the web app using CSS for styling and better user experience

As a bank, I would like to be able to access customer data after for post-production analysis and model monitoring

Task 13: Deploy the web app and check data for model monitoring