Alessio Petrozziello - Senior Data Scientist

I work in the AI & Data Science team (i.e., Core Data Science team) in Expedia Group and my primary role is to build new Recommender Systems for properties ranking across all platforms (i.e., desktop, mobile, etc), reaching millions of users every day. My role covers the whole data life cycle: extraction of new features from both structured and unstructured sources; creation of new machine learning models in Spark and Tensorflow; collaborating with product and engineers to productionise models and with analysts to collect insights on the online performance of the models (through A/B testing).

Current Projects

Deep learning for sort in Tensorflow and Keras

- Overview: I implemented the first production ready deep neural networks in Tensorflow to rank properties and closely worked with engineers to productionise a Spark+Tensorflow pipeline on the live environment.
- Approach: I built a framework spanning multiple technologies and programming languages to use both the distributed capacity offered by Spark (for data pre-processing); the GPU capabilities for training (in Python and Keras); and again Spark to make fast predictions at large scale. I also had to closely work with engineers and helped them to create a platform able to use Tensorflow serving graphs in production.
- Technologies: Scala, Python, Spark, pySpark, AWS, Tensorflow, Keras

Sort explainability

- Overview: I am leading on a workstream to explain the sort model to customers in order to give them more confidence on how we are making recommendations.
- Approach: I'm experimenting with offline explaination tools for non-linear models (i.e., LIME and Shapley) to produce search-level and item-level explanations, in order to identify which factors are driving the properties ranking. Such explainations can be shown to the user in order to build their confidence on our machine learning driven "Recommended" sort.
- Technologies: Scala, Python, Spark, pySpark, AWS, Tensorflow, Keras

Past Projects

Cold start problem for new properties (with particular focus on Vacation Rentals).

- Overview: I investigated the cold start problem for properties recently added to the catalogue and the impact of missing historical data on the recommender system ranking. The problem is quite important as a business perspective as newly acquired properties could never show up at the top of the ranking, hence hurting their short and long term value and both customers and property managers satisfaction.

- Approach: An initial analysis was necessary to understand the current ranking of new properties (this implied the creation of a large Spark pipeline processing +500gb data daily and summarizing statistics across different key segments). After the initial exploration, KNN, Random Forests and Gradient-Boosted Trees (already available on Spark) were applied to impute missing features such as Guest Rating and Star Rating. The imputed features have then been used in production to rank new properties.
- Results: following the use of the imputed features, further analysis showed an overall improvement in ranking for new properties.
- Technologies and Methods: Scala, Spark, AWS, KNN, RF, GBTs.
- More Details: http://alessiopetrozziello.altervista.org/papers/PetrozzielloMIC2017published.pdf

Scaling the missing data imputation to higher dimensional datasets.

- Overview: Guest Rating and Star Rating are only two of many historical features missing for a new property added to the catalogue. The use of ready Spark-ML algorithms such as KNN, RF and GBT is no longer feasible when a large number of variables need to be imputed, as only one feature can be regressed at the time.
- Approach: I implemented a mini-batch SGD Neural Network on Spark able to distribute the training phase across a cluster of machines. The neural network can have as many outputs as the number of features to impute, hence it is not necessary to train multiple models.
- Results: The imputation pipeline on Spark was scheduled as a recurrent job in order to have fresh imputation for all new properties added to the catalogue through time.
- Technologies and Methods: Scala, Spark, AWS, Scala Neuron Package (as Neural Network building blocks to build the distributed version).
- More Details: http://alessiopetrozziello.altervista.org/papers/PetrozzielloIJCNN2018.pdf

Personalisation in sort

- Overview: I drove the effort to add user level personalisation features in the sort algorithm and have more relevant customer-based ranking.
- Approach: I built large Spark pipelines to daily extract and aggregate user level features; furthermore, modelling users in a recommender system model is particularly challenging due to the high cardinality nature of the feature. Lastly, this task poses interesting engineering challenges when it comes to the productionisation of such models, as the live environment can only contain a limited amount of information. To solve this problem I had to build a “next day customer prediction model” to predict which set of customers would access the website the next day based on their past activity. This model allows to load into memory (every day) only information for a subset of customers.
- Technologies: Scala, Spark, AWS

Daily Tasks

- Extract new features for the sort model and set up Spark pipelines to have them ready every day for offline training and online scoring;
- Supervise the creation of new baseline data in order to have reproducibile offline

experimentation and comparison;
- Collaborate with different stakeholds to collect requests about new features and brand initiatives to be included in the sort model;
- Collaborate with engineers around the globe to productionise new sort models;
- Collaborate with analysts to assess performance of the models.
- Manage one Data Scientist
- I co-supervise two PhD students at the University College London (UCL)

Managerial tasks

- I manage one Data Scientist
- I co-supervise two PhD students at the University College London (UCL)