

# What Investments are Right for You?

Supporting Financial Advisors with Customer Risk Aware Investment Recommendations

Richard McCreadie, 16/03/2021



# Contents

---



Personalized Investment and Why we need AI Assistants



Can we use recommendation solutions from other domains to create an effective financial assistant?



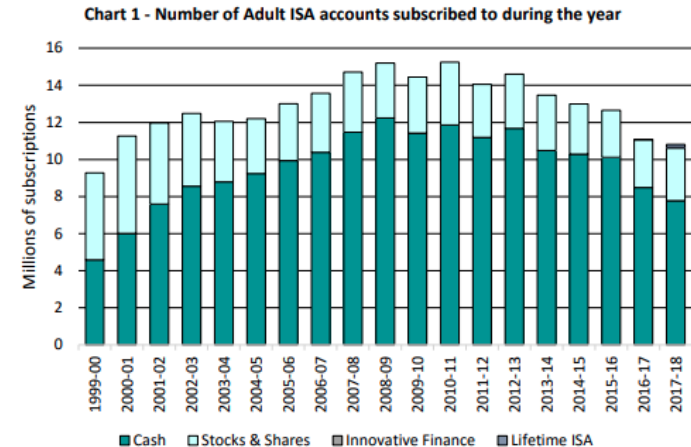
What are the key opportunities moving forward?

# Personalized Investment and Why we need AI Assistants



# Personal Investment Problem

- We are currently living in a world where only a small fraction of the public sufficiently invest for the future
  - **35%** (18.4 million) of the UK adult population say they don't have a pension.
  - **43%** of the population admit they don't know how much they will need for retirement.
- One way to tackle this is to encourage people to invest spare cash rather than have it lie un-used in current accounts
  - But only around **4-5%** of adults in the UK have stocks and shares ISAs (one of the most cost-effective ways of doing this)





# Challenges to Investment

---

- There are a wide array of reasons that people choose not to invest in financial assets even if they have the money to do so, including:
  - **Complexity:** Finance is complicated, and many people are not sufficiently educated to understand the consequences of their decisions (...and the large volumes of financial jargon does not help!)
  - **Time:** Investing successfully takes time and effort to research and understand the target markets, most people don't have the time for this
  - **Risk:** There are a wide range of investment risks, and some of those are difficult to effectively quantify
  - **Choice:** The range of possible investments is so large that choice paralysis is a barrier
  - **Advice:** Its not clear to a new investor where the should go to get advice, and who they can trust

# Insurance Recommendation

---

- Given the aforementioned problems, we cannot expect an average member of the public to become a savvy investor on their own
- They need a financial advisor to **analyze their position** and **recommend assets** to invest in personalized to them



# Insurance Recommendation Task

- Input
  - User Profile  $u \in U$ 
    - Questionnaire data about their circumstances and preferences,
    - Banking history, e.g. purchase histories, direct debits/re-occurring payments, salary, mortgages, etc.
    - (In a few cases) Investment history, i.e. previous and current investments
  - Financial Assets/Instruments  $a \in A$ 
    - Price History, Volume
    - Risk Estimators, e.g. Alpha, Beta, Value Standard Deviation or Sharpe Ratio
  - Environment Data  $E$ 
    - Market Segment History/Trends
    - Tax Breaks
- Output
  - A score for a single asset, representing investment suitability with respect to the person at the current time ( $t$ )

$$f(u_t, a_t, E) \rightarrow [0,1]$$

# The Personalization Problem

One of the keys for good financial investments is that they need to be personalized to the situation of the particular customer

How much can they invest?	How long can they invest for?	How volatile is their income?	Are there significant future known costs (e.g. starting a family)?	How much can they afford to lose?
---------------------------	-------------------------------	-------------------------------	--	-----------------------------------



In an ideal world everyone would have a personal financial advisor, but this is simply not scalable...





Can we use recommendation solutions from other domains to create an effective financial assistant?

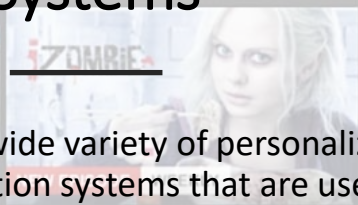
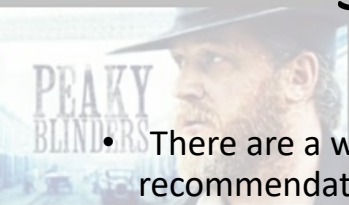


## Emmy-winning US TV Shows



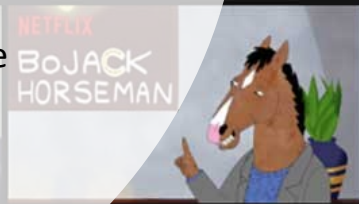
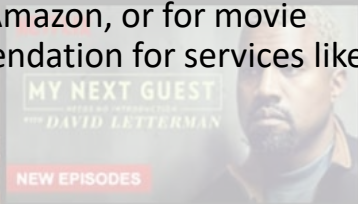
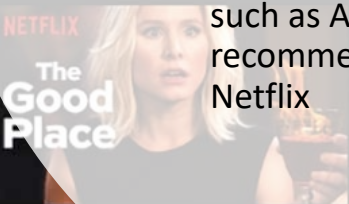
## Existing Recommender Systems

### Police Detective TV Dramas



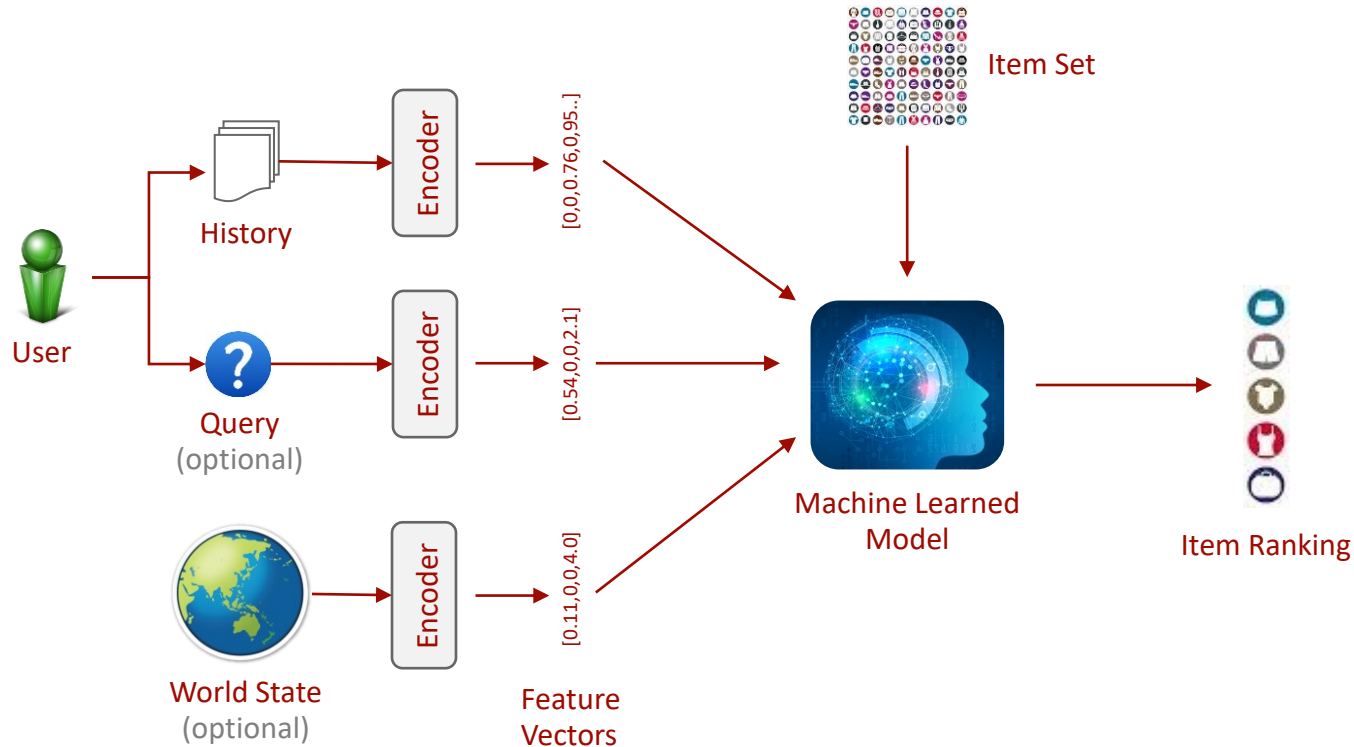
- There are a wide variety of personalized recommendation systems that are used world-wide

### Critically Acclaimed Witty TV Shows



- These are most commonly seen supporting shopping tasks on sites such as Amazon, or for movie recommendation for services like Netflix

# How do Recommenders Function?



# How do Recommenders Function?



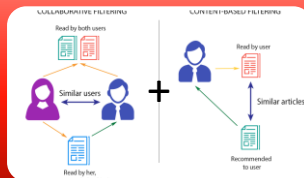
## Content-based filtering

- These models compare the similarity of a **user's preferences** to the **attribute profiles of the items**



## Collaborative filtering

- These models recommend items on the basis of **past interactions from other similar users** with these items
- Later models also can integrate content-based features as side information

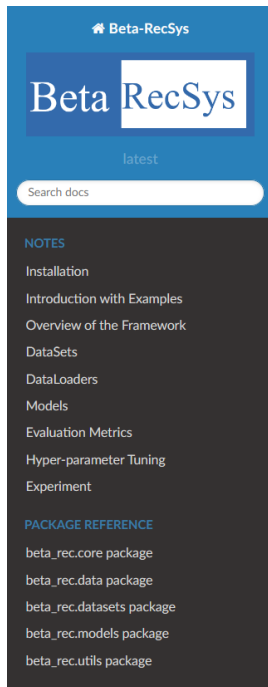


## Hybrid models

- Combine more than one approach, for example combining the recommendations of a content-based and collaborative filtering approach together, such as via voting

# Recommender Models

- **Simple content-based (CB) model**, using the demographic information and spending history of the users, along with the volatilities of the assets
- **Association rule mining**, which finds patterns of co-occurring item sets that users tend to interact with
- Collaborative filtering (CF) models:
  - **Matrix factorisation**, an embedding model which decomposes the user-item interaction matrix into lower-dimensionality constituents
  - **Neural graph collaborative filtering (NGCF)**, which aims to exploit the user-item graph structure and inject this into the embedding process
  - **LightGCN**, a model that uses the essential part of a graph convolutional network, the neighborhood aggregation, to influence user and item embeddings
  - **Neural matrix factorisation (NeuMF)**, which combines generalised matrix factorisation with implicit feedback from a multi-layer perceptron model
  - **HIRE**, a CF model with auxiliary information on users and items
- **Hybrid models**, which combine the CB model with the CF models above



Beta-RecSys

latest

Search docs

NOTES

- Installation
- Introduction with Examples
- Overview of the Framework
- DataSets
- DataLoaders
- Models
- Evaluation Metrics
- Hyper-parameter Tuning
- Experiment

PACKAGE REFERENCE

- beta\_rec.core package
- beta\_rec.data package
- beta\_rec.datasets package
- beta\_rec.models package
- beta\_rec.utils package

Docs » Indices and tables

[Edit on GitHub](#)



Beta-RecSys an open source project for Building, Evaluating and Tuning Automated Recommender Systems. Beta-RecSys aims to provide a practical data toolkit for building end-to-end recommendation systems in a standardized way. It provided means for dataset preparation and splitting using common strategies, a generalized model engine for implementing recommender models using Pytorch with a lot of models available out-of-the-box, as well as a unified training, validation, tuning and testing pipeline. Furthermore, Beta-RecSys is designed to be both modular and extensible, enabling new models to be quickly added to the framework. It is deployable in a wide range of environments via pre-built docker containers and supports distributed parameter tuning using Ray.

codecov 60% CI passing

## Notes

- [Installation](#)
- [Introduction with Examples](#)
- [Overview of the Framework](#)
- [DataSets](#)
- [DataLoaders](#)

<https://beta-recsys.readthedocs.io>

# Datasets

- **Investment history of users with assets**
  - **965** users, **449** assets across **11** categories
- **Demographic and non-investment financial history of users**
  - **Age, gender, income, deposit account balances, card spends**, and many other KPIs besides
  - **51442** unique users
- **Historical asset prices from 2018-2021**
  - **2.49 million** datapoints (23.7% from 2018, 25.12% from 2019, 28.35% from 2020, 22.85% from 2021)
  - Spread across four key asset classes – **bonds, mutual funds, stocks, and time deposits**

# Recommendation Performance for Financial Asset Recommendation

## Normalized Discounted Cumulative Gain (nDCG):

a metric that assesses how many relevant results were obtained at the top rankings of the recommender

Figure shows  $k = 3$ .





# What are the key opportunities moving forward?



# Encoding Knowledge of the Future

- Current recommendation algorithms are naturally focused on the past as a means to predict the future
  - However, the customer may know of changes in their financial position well in advance, for factors within their control (having a family, moving job, etc.)
  - Companies publish information about up-coming products with release dates that affect their share value
  - The introduction of new government initiatives or international politics can affect the outlook for entire market segments
- Next generation algorithms need to account for known (or at least likely) future events



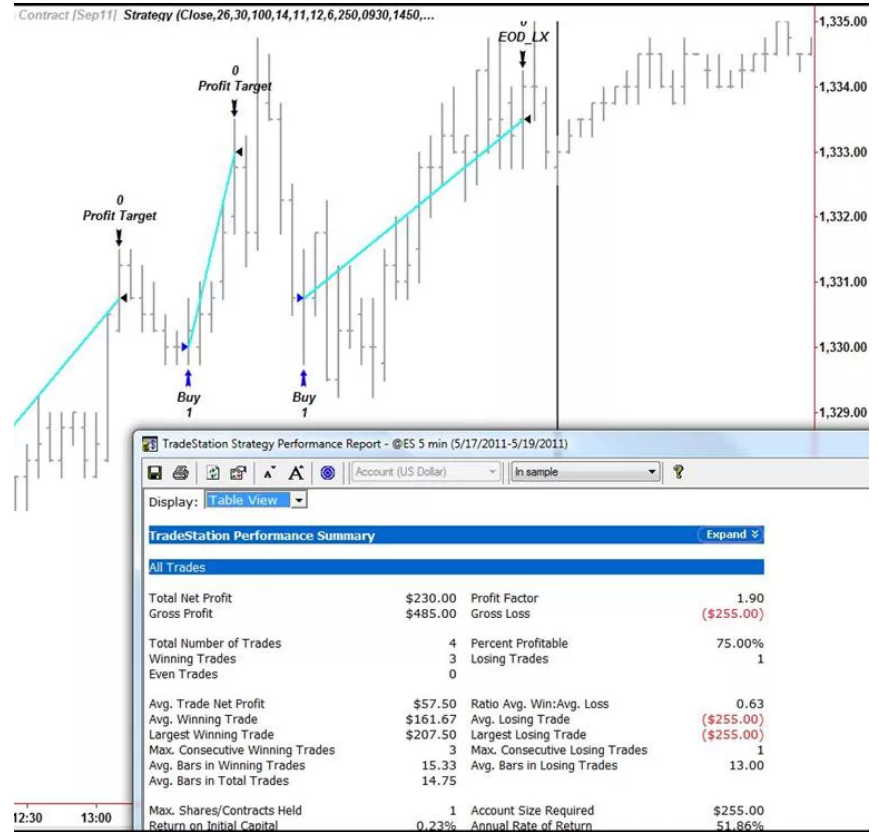
# The Textual Frontier

- Modelling of financial assets are primarily based on observations on past pricing data
  - This is by far the most useful source of information regarding the value and volatility of an asset
  - The issue is that changes in this data represents market reactions, not the cause of those reactions, meaning its impossible to 'get ahead' of market shifts
- The solution here is to instead more directly track the causes of market shifts, namely financial reports, investor call logs and news stories
  - Over the last couple of years large strides have been made in automated language understanding using deep neural networks, meaning it may be possible to have AI assistants track and reason about market changes in real-time
  - This also opens doors to more effective machine translation of local news for traders working across language boundaries



# Automated Trading Systems

- So far, we have been discussing an assistive advisor that can provide recommendations to a human
  - However, what if we wanted to place some decision-making power in the hands of the advisor?
- This is happening at scale today
  - 70% to 80% or more of shares traded on U.S. stock exchanges are estimated to come from automatic trading systems according to a study by Reuters
- This has a number of advantages, most notably the ability to quickly react to market shifts



Questions?

Contact: [richard.mccreadie@glasgow.ac.uk](mailto:richard.mccreadie@glasgow.ac.uk)

