Holistic AI × The Alan Turing Institute

Trade-offs of Bias with other verticals in Trustworthy AI Part III

Turing Course

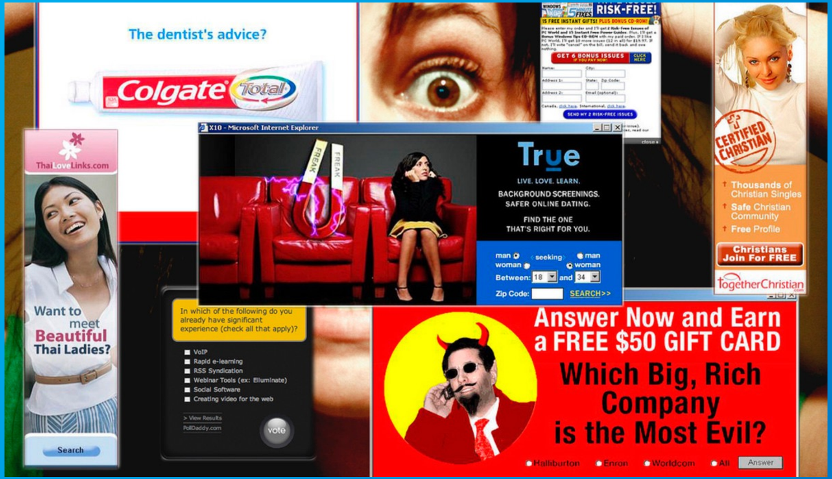
Contents

Reminders

- 1) What is Recommendation
- 2) Why should we ensure Recommender Systems are built with **trustworthiness** in mind?

Recommendation

- A recommender system is a subclass of information filtering system that seeks to predict the rating a user would give to an item.
- The predicted ratings are then used to recommend new items to each user, that they are likely to enjoy/buy/interact with.
- These systems are trained using the past interactions of the users.



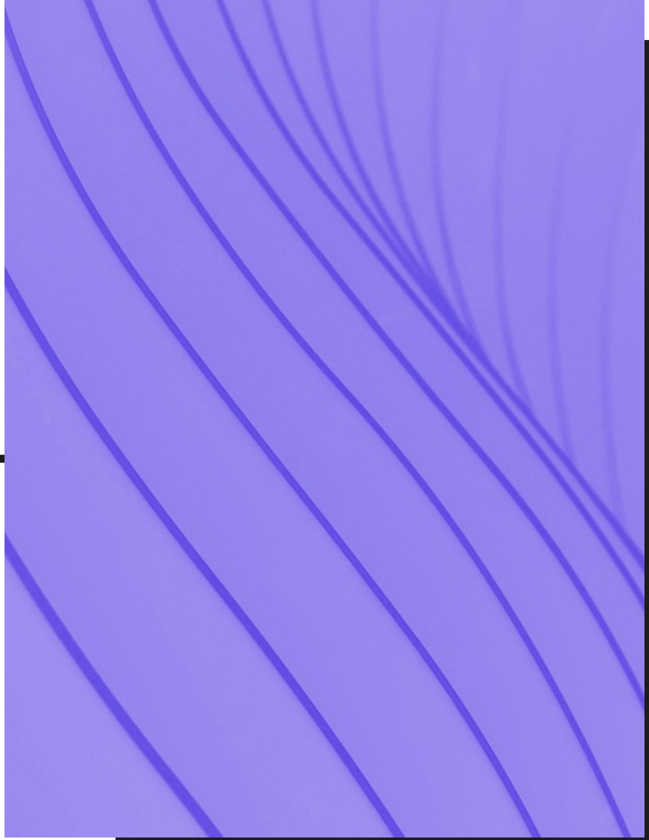
Importance

- E-commerce, social networks, search engines, news portals, hiring platforms, intelligent assistants, smart home, smart city services, healthcare, financial applications, etc.
- Recommender system is the frontier of Human-centered AI research and works as the bridge between humans and AI.
- This is the era of information overload. Hence the critical need to make systems trustworthy!

I – Explainability & Recommendations

– Discuss the benefits of explainable recommendation.

– Introduce Explicit factor Models for explainable recommendation.



Motivation

- More explainable recommendations allow for accountability of the system. We can know for certain how/why a recommendation was made.
- Hence it is much easier to make sure the recommendations are ethical (or spot if they are not).
- We can also provide users with explanations of what they are recommended and that can even incentivise them to buy products (so it can also be good for sales!).

Explicit Factor Models

- Paper by Zhang et al, 2014, Explicit Factor Models for Explainable Recommendation
- In this paper, the explainability is built into the way the model is devised.
- Recall the matrix factorization methods we introduced in the Bias in Recommender Systems section of the course. The method of matrix factorization works by learning latent factors describing items.
- In **Explicit Factor Models,** the features of items are manually set, and the user's preferences are learnt with metadata and sentiment analysis.

Example

- Suppose we have an e-commerce website where users search for and buy mobile phones.
- We could learn a recommendation model using matrix factorization, but the latent factors would be highly abstract and obscure.
- This method proposes that we set the latent factors of items by hand: for phones this could be (screen size, battery life, memory size, brand, camera quality, etc).

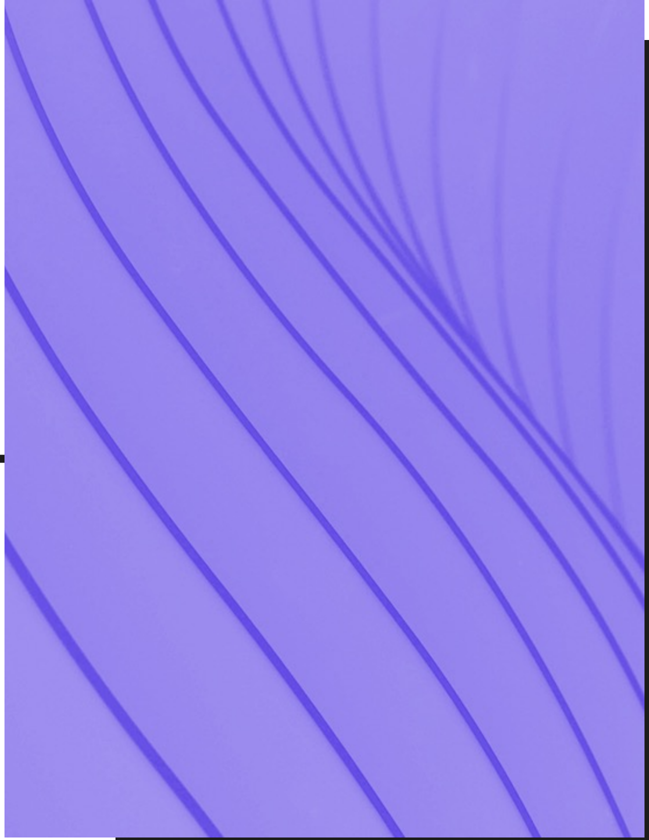
Example

– We then learn the user's sentiment towards each of these explicit features using their searches, reviews and other metadata obtained from their behaviour.

– Note the model is still a matrix factorization, only the latent features are explicit now!

– If a user likes small screen size, long battery life, large memory size, Samsung brand, low camera quality. The model can easily deduce scores for each phone.

II – Robustness & Recommendations



1) Explain importance of robust recommendation.
2) Explain one method of Attack.
3) Explain one method of Defense.

Motivation

– Recommender Systems use models that can learn from the preferences of users and use those to make new suggestions.
– But the strength of these learning methods is also what makes them liable to **attacks**.
– These attacks are made to alter or diminish the performance of the system.

Attack: Shilling

- Paper by Shyong et al. 2004.
- In the simplest sense, shilling attacks can be created to **push** or **nuke** an item.
- **Inject** a collection of new users into the system, each of which has rated a set of items to try to look like real users.
- Also rate the items being attacked **very low** in order to nuke them or **very high** in order to push them.

Defense: Clustering

- Paper by Bhaumik et al, 2011.
- The method works by extracting 5 descriptive features of a user profile from the user's ratings.
- One example is Length Variance: it is introduced to capture how much the length of a given profile varies from the average length in the database.
- We use this new 5D embedding of our users to cluster the profiles into 2 clusters (using 2-means clustering).
- We assume the smaller cluster is the fake profiles. We can then remove the supposedly fake profiles from the training.

III – Privacy & Recommendations

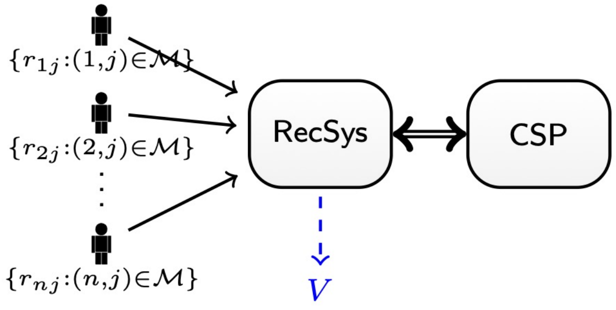
- Explain importance of privacy in recommendation
- Explain one method for privacy preserving recommendation

Motivation

- Modern Recommender Systems access and make use of a lot of **personal data** (e.g., gender, age, and address) beyond the ratings given to items.
- Most of the time, users are not even aware of the data they are giving away, for instance because of accepting obscure terms and conditions.
- This sensitive user data can be misused, resold or leaked if the System is not built with Privacy concerns in mind.

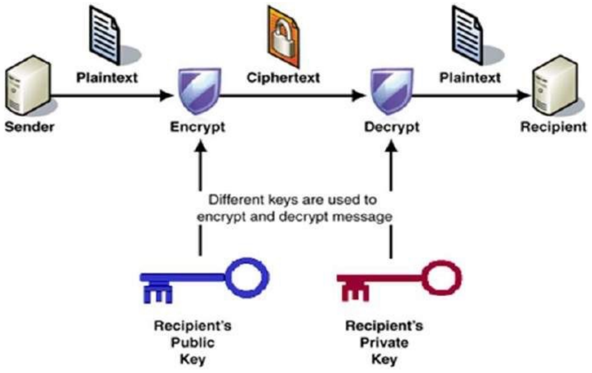
Privacy-preserving matrix factorization

- Here we give an overview of a method by Nikolaenkoet al, 2013.
- This paper introduces a way to implement Matrix Factorization while ensuring users do not reveal their ratings to the owner of the system.



Privacy-preserving matrix factorization

- To achieve private computation, the owner of the recommender system must make use of a **crypto-service provider** (CSP).
- The CSP is a module that is apart from the main system that is in charge of implementing all the encryption and decryption functionalities (e.g., RSA public key cryptography).



Privacy-preserving matrix factorization

– This method makes use of an encryption method called **Garbled Circuits**.

– A garbled circuit is a way to encrypt a computation that reveals only the output of the computation.

– This method reveals nothing about the inputs, or any intermediate values so that the owner of the system never has access to any information on the user data.

References

- [1] Ge et al, 2022, A Survey on Trustworthy Recommender Systems
- [2] Zhang et al, 2014, Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis
- [3] Gunes et al, 2014, Shilling attacks against recommender systems, a comprehensive survey
- [4] Shyong et al. 2004. Shilling Recommender Systems for Fun and Profit.
- [5] Bhaumik et al, 2011. A clustering approach to unsupervised attack detection in collaborative recommender systems.
- [6] Friedman et al, 2015, Privacy aspects of recommender systems
- [7] Nikolaenkoet al, 2013. Privacy-preserving matrix factorization.