Travel better together

Deep Personalized Retargeting

Meisam Hejazinia, Pavlos Mitsoulis-Ntompos, Serena Zhang
Vrbo, Part of Expedia Group, Austin, Texas and London, UK

- Introduction
- Business Context
- Research Overview
- Use case
- Research Position
- Model
- Neural Nets
- Conclusion

Travel better together

Largest Travel Platform in the World ...

High Volume and diversity of **Customers**

750M+ monthly visits[1]

2M+ active corporate travelers

Powering **~100K** offline travel agents[2]

35K+ B2B partners leveraging our platform

50M+ contacts handled annually[3]

Broad and diversified **Supply Partners**

1.1M+ properties on core lodging platform[4]

2M+ Vrbo online bookable listings

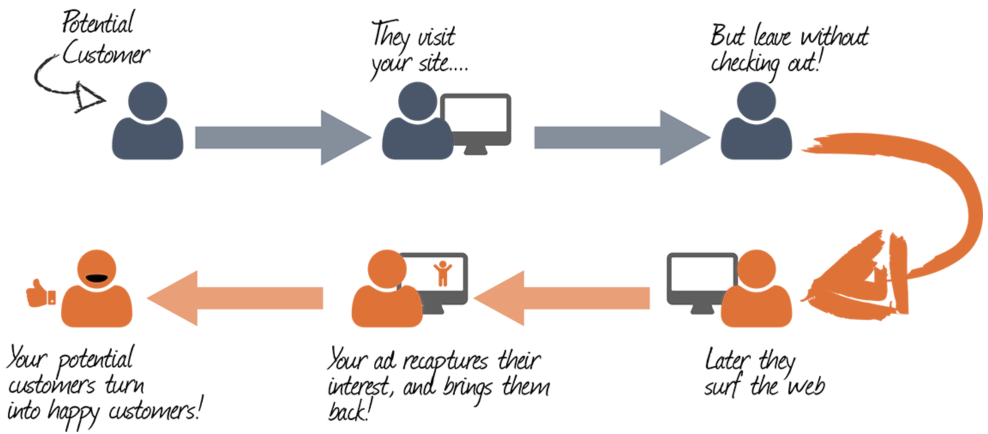
500+ airlines

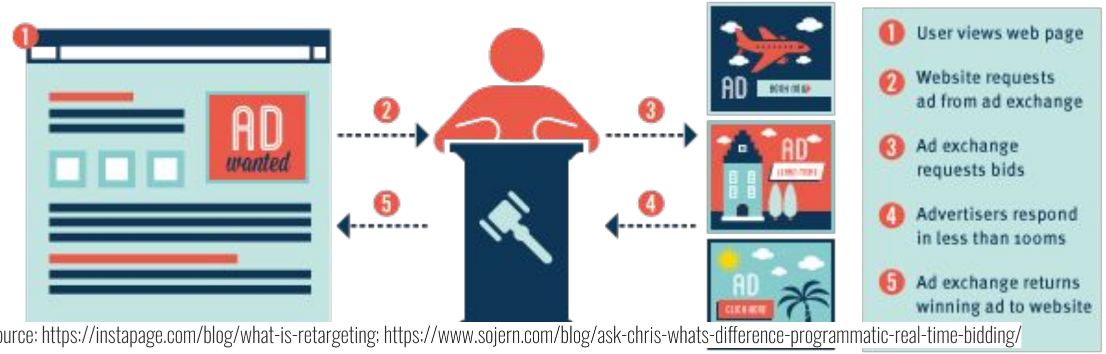
175+ car rental companies

35K+ unique activities

Notes: Expedia Group data shown as of 3/31/19, unless otherwise noted. 1. Monthly visits based on data for Brand Expedia, Hotels.com, Orbitz, Travelocity, Wotif, Vrbo, trivago and Hotwire combined during 2018. 2. Offline travel agents based on number of sales agents in Global Customer Operations, Expedia Partner Solutions (EPS), Vrbo, Classic Vacations, CruiseShipCenters, Travel Agent Affiliate Program (TAAP). 3. Contacts handled annually include calls, emails, chats and social media. 4. Includes more than 460,000 integrated Vrbo listings. This information is based on Q2 19' Earning Report.

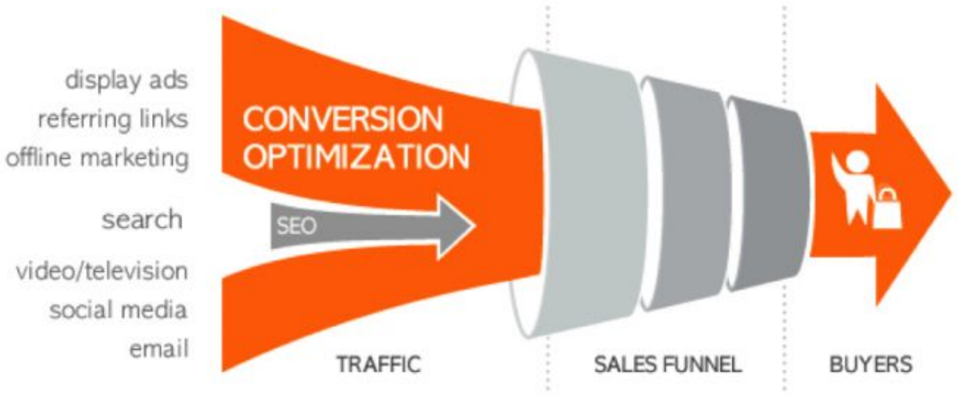
Retargeting and Real Time Bidding

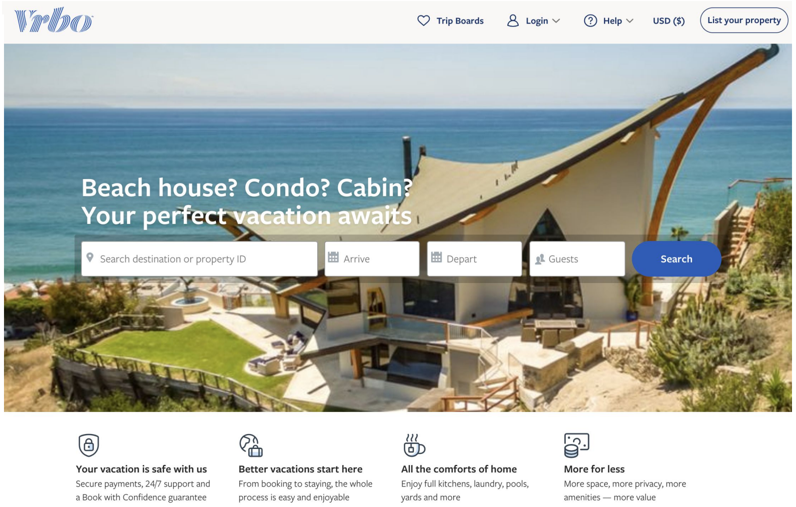




Source: https://instapage.com/blog/what-is-retargeting; https://www.sojern.com/blog/ask-chris-whats-difference-programmatic-real-time-bidding/

Short-term Vacation Rental Two-sided Marketplace Platforms



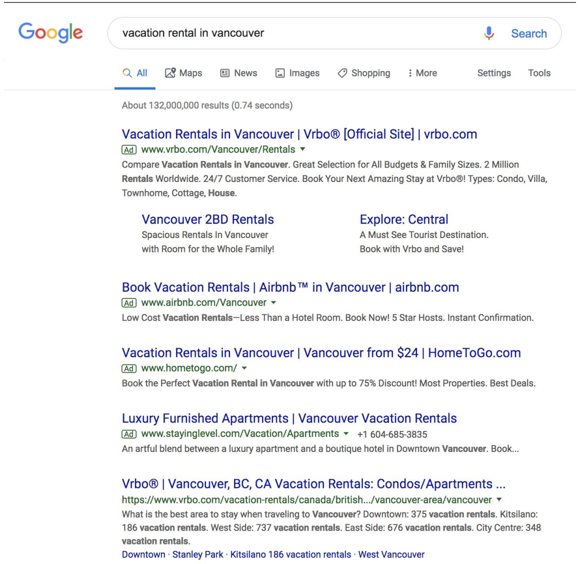


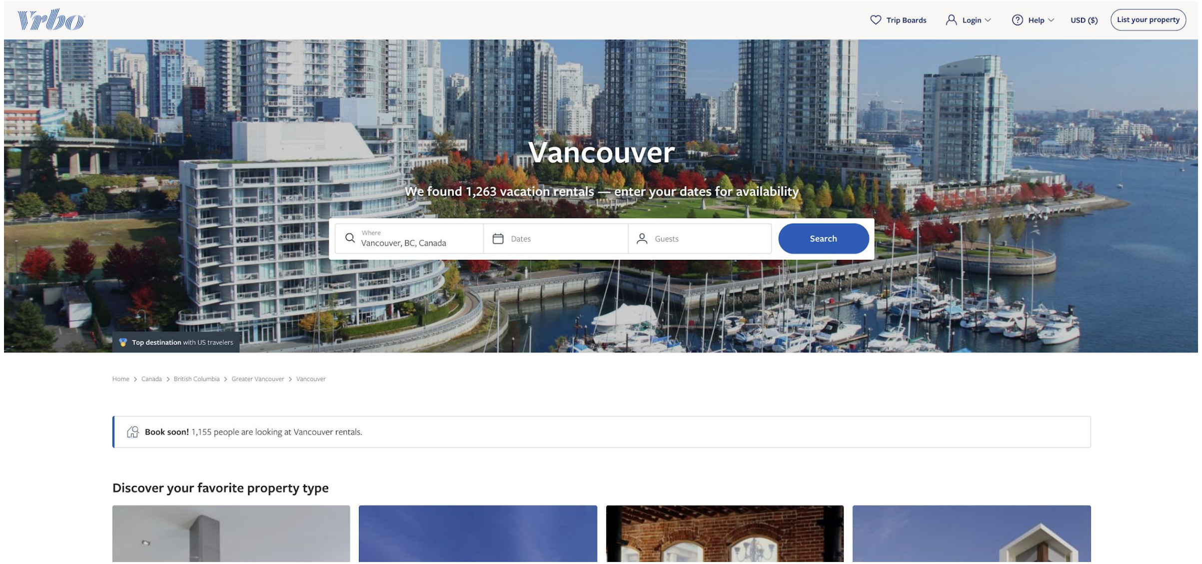
Source: https://www.upwork.com/hiring/for-clients/google-search-when-organic-and-paid-search-efforts-align/

Research Questions

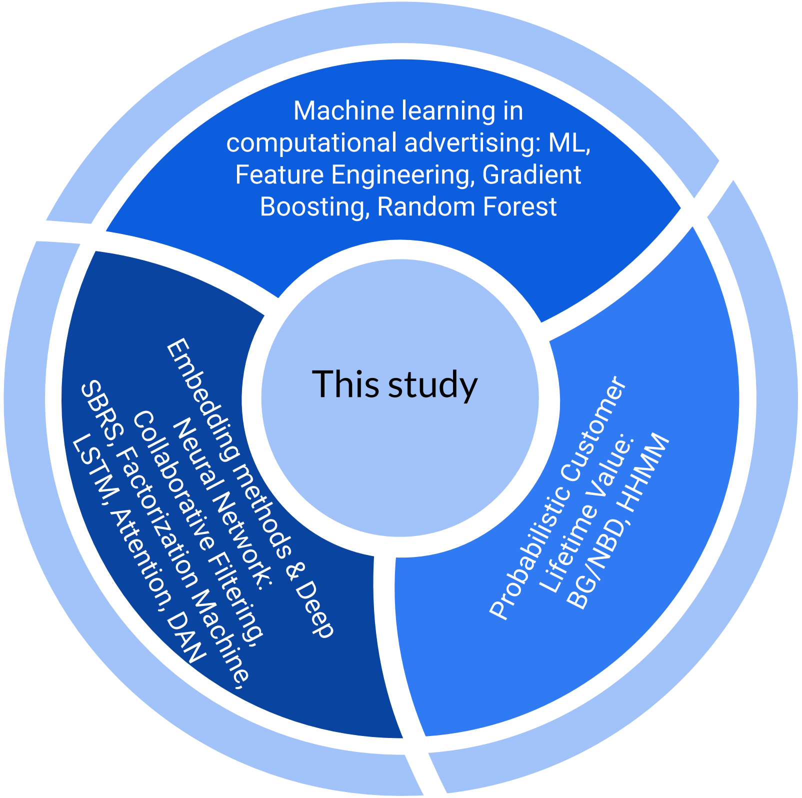
- How to predict conversion probability and value for real time bidding?
- How to improve the prediction of conversion probability and value?
- How to automatically extract features and account for unobservable features?
- How to productionalize such system?
- How is the performance of the designed system compared with alternatives?

Real time bidding use case in Vrbo





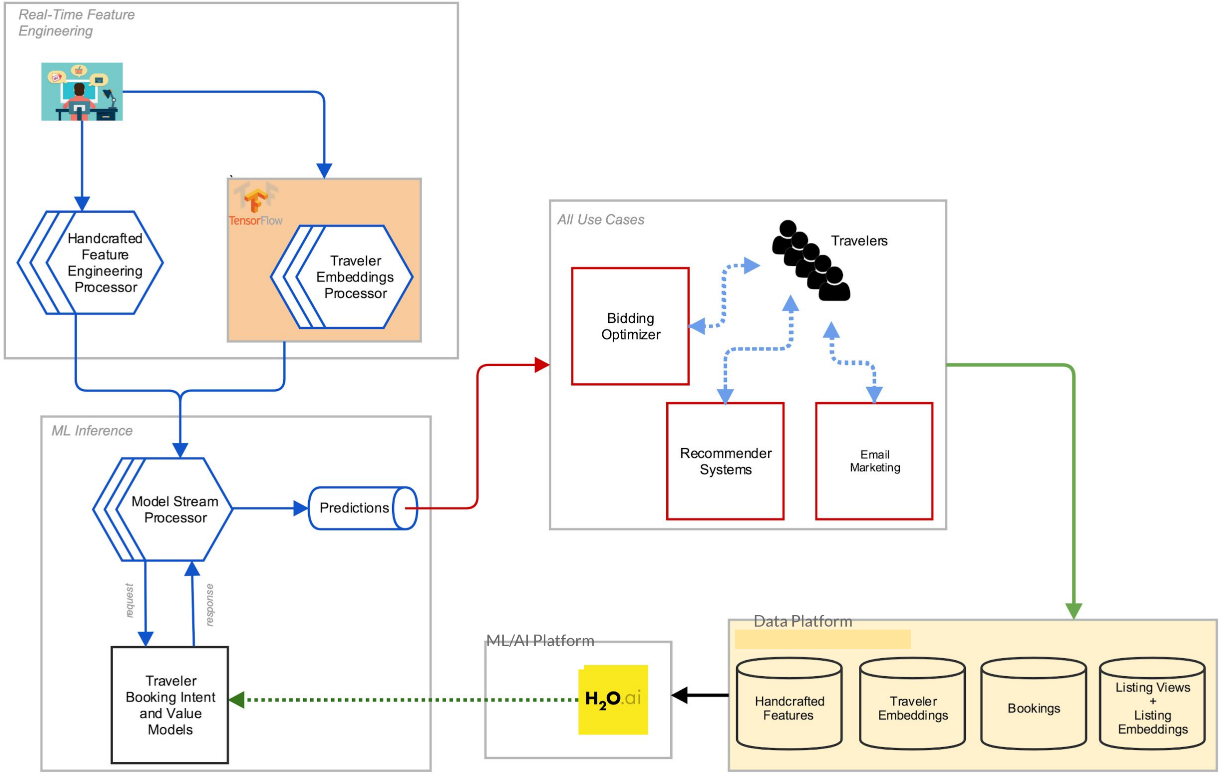
Position of this research in literature



This study in the context of literature

- Leverage embeddings to improve the intent prediction solution
- Extend embedding features from linear to non-linear
- Sweet spot with low cost and high value for combining deep neural networks with shallow neural networks and tree-based method
- We deployed our solution into a production system

Overview of the System



Overview of the Methodology

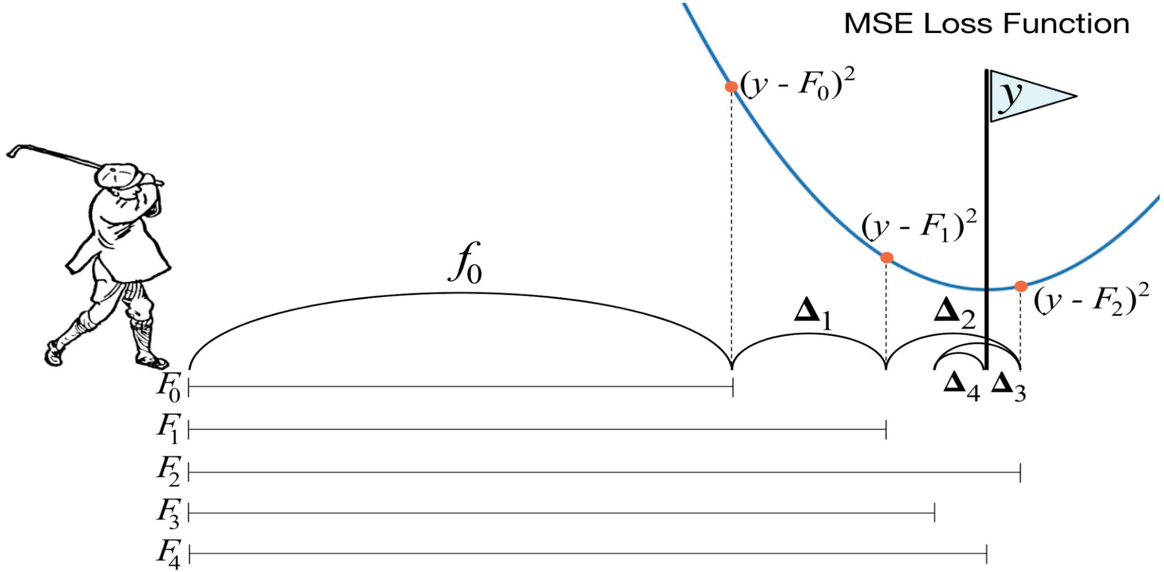
- Internal anonymized clickstream data is collected for millions of users from two different seven-day periods:
    - The 1st dataset was used to generate embeddings using Deep Average Network and the LSTM with Attention
    - The 2nd dataset was used to evaluate the learned embeddings on the Traveler Booking Intent Model
- Three baselines methods:
    - Random Listing Embedding Selection
    - Averaging Embeddings
    - LSTM with Attention

Overview of Results

Algorithm	**AUC**	**Precision**	**Recall**	**F1-Score**
Random	0.973	0.821	**0.633**	0.715
Averaging Embeddings	0.971	0.816	0.628	0.71
LSTM + Attention	0.976	0.877	0.62	0.727
DAN	**0.978**	**0.888**	0.628	**0.735**

Settings	**AUC**	**Precision**	**Recall**	**F1-Score**
Only handcrafted Feat.	0.975	0.817	**0.651**	0.724
Handcrafted + DAN Feat.	**0.978**	**0.888**	0.628	**0.735**

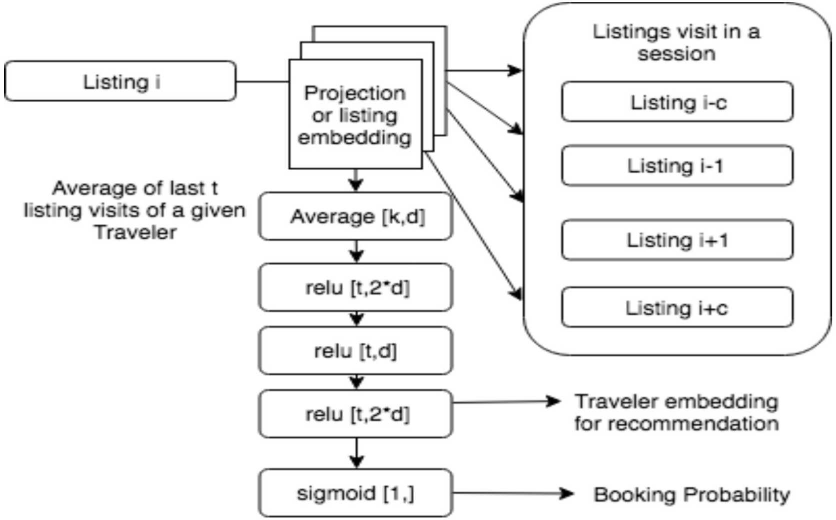
Model -1



$$\hat{y}_j \equiv F(\mathbf{x}_j) = \sum_{k=1}^{K} f_k(x_i), \text{ where } f_k \in \mathbb{F}$$
$$L^{(t)} = \sum_j l(y_j, \hat{y}_j^{(t-1)} + f_t(\mathbf{x}_j)) + \Omega(f_t)$$

Source: https://explained.ai/gradient-boosting/

Model - 2



$$\frac{1}{S}\sum_{s=1}^{S}\sum_{-c\leq i\leq c, i\neq 0}\log p(l_{i+j}|l_i), \text{ where } p(l_{i+j}|l_i) = \frac{\exp(\nu_{l_{i+j}}^T \nu_{l_i})}{\sum_{l=1}^{L}\exp(\nu_l^T \nu_{l_i})}$$

$$\text{However, by negative sampling } p(l_{i+j}|l_i) = \frac{exp(\nu_{l_{i+j}}^T \nu_{l_i})}{1+\exp(\nu_{l_{i+j}}^T \nu_{l_i})}$$

Model - 3

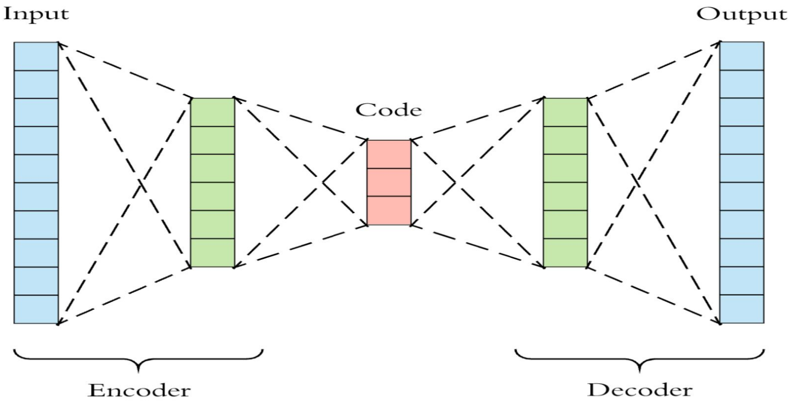
- Target to predict: $\mathrm{P}(Y_j|\mathrm{S}_j, C_j) = \text{sigmoid}(f(\nu_{j.}))$
- Encoder-Decoder Architecture (Deep Average Network - DAN):

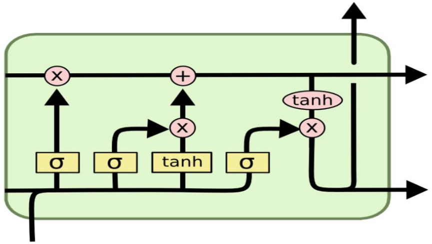
$$f(\nu_{j.}) = \text{relu}(\omega_1 \cdot h_2(\nu_{j.}) + \beta_1)$$
$$h_1(\nu_{j.}) = \text{relu}(\omega_2 \cdot h_1(\nu_{j.}) + \beta_2)$$
$$h_2(\nu_{j.}) = \text{relu}(\omega_3 \cdot \frac{1}{k}\sum_{i=1}^{t} \nu_{ji}) + \beta_3)$$

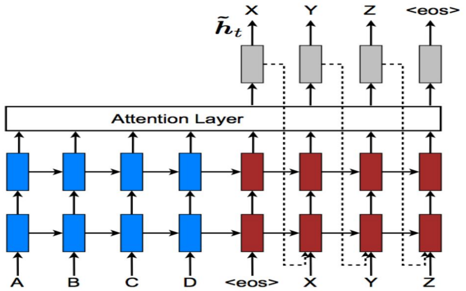
LSTM Architecture: $f(\nu_j^t) = \text{sigmoid}(\omega_f[h_t, \nu_j^t] + \beta_f) \cdot f(\nu_{j.}^{t-1})$
$$+ \text{sigmoid}(\omega_i[h_t, \nu_j^t] + \beta_i) \cdot \tanh(\omega_c[h_{t-1}, \nu_j^t] + \beta_c)$$

- LSTM + Attention: $f(\nu_i) = \text{softmax}(\omega^T \cdot h_T)\tanh(h_T)$

LSTM, Attention, and Encoder-Decoder Network Architecture







Source: https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798; https://colah.github.io/posts/2015-08-Understanding-LSTMs/; https://machinelearningmastery.com/how-does-attention-work-in-encoder-decoder-recurrent-neural-networks/

Conclusion

- Proposed a hybrid deep learning framework for a massive vacation rental marketplace.
- Deployed end-to-end manner and leveraged for re-targeting.
- Combined shallow and deep neural network embedding for listing and traveler embedding and tree based boosting methods.
- Extension by incorporating more contextual spatio-temporal information in the model.
- Full paper proceeding of ASONAM IEEE/ACM conference can be found here.

Q&A

Thank You for the Opportunity!

We're hiring!
Visit our Careers Page to join our team!
https://lifeatexpedia.com/jobs