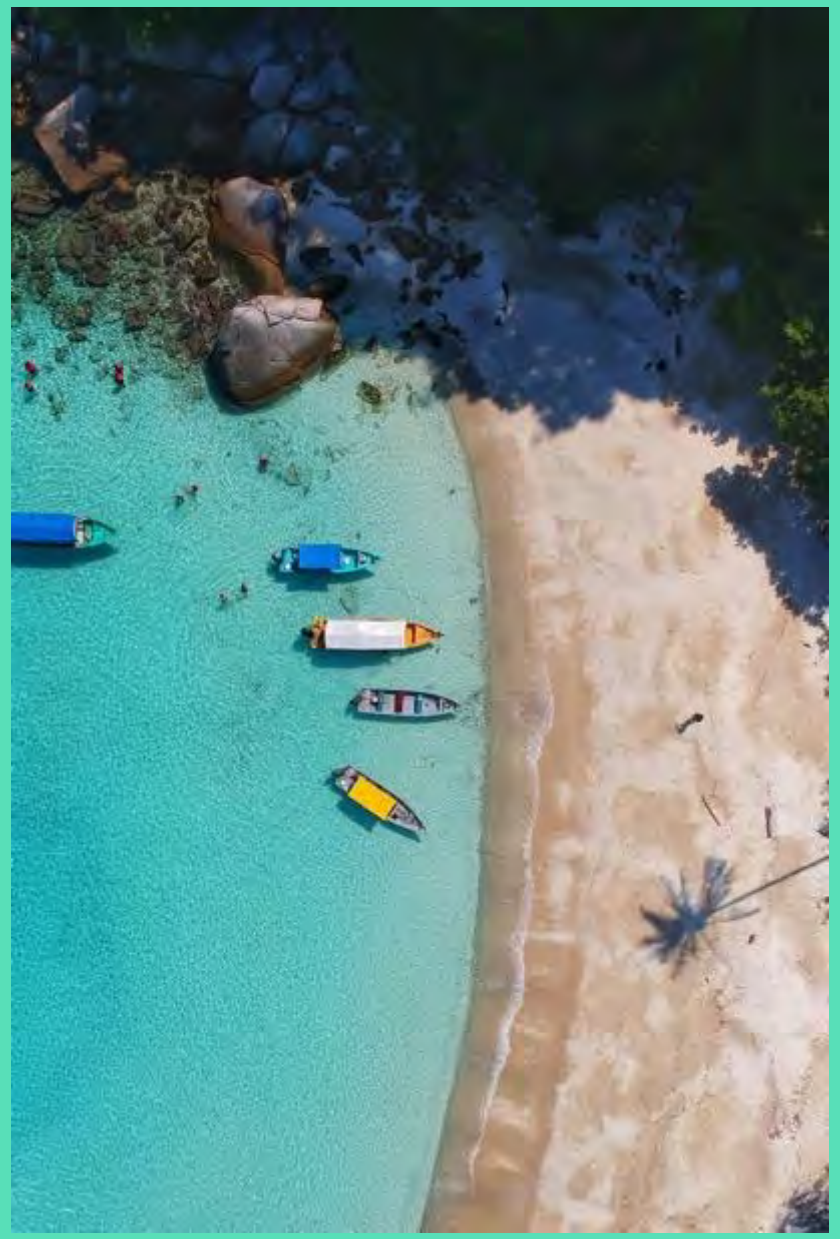


Hotel2vec: Learning Hotel Embeddings from User Click Sessions with Side Information

Ioannis Partalas & Anne Morvan

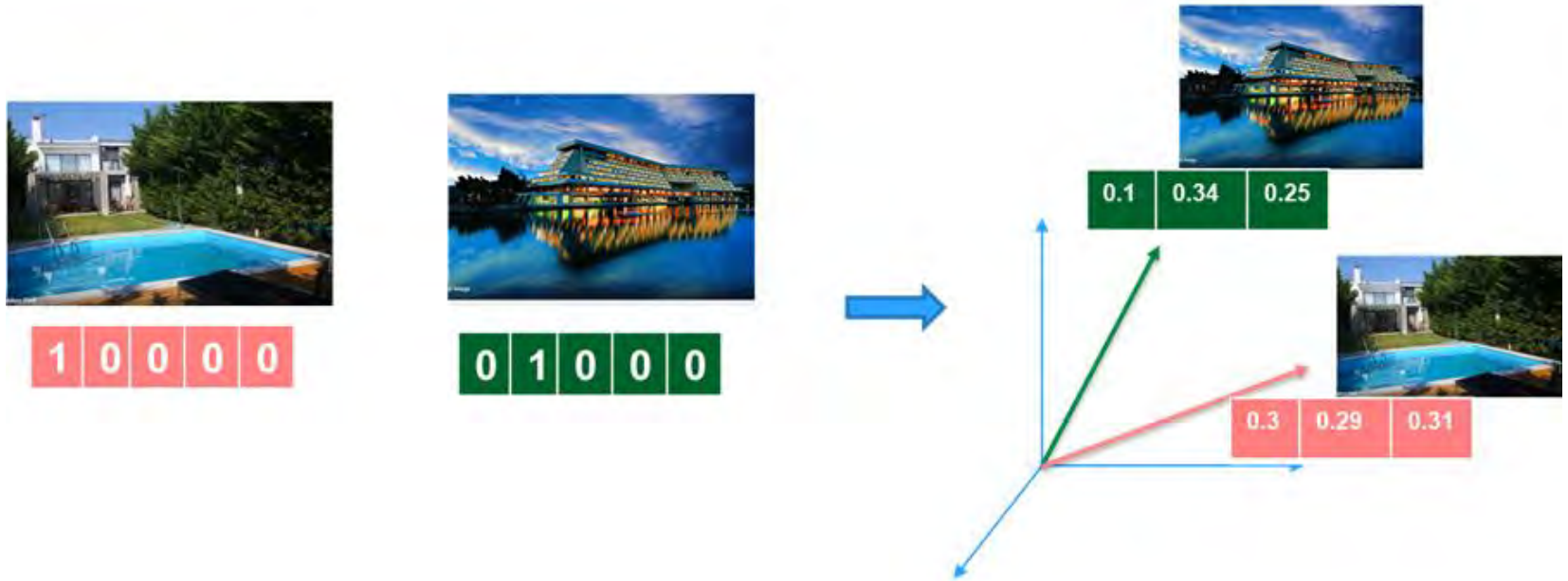
Joint work with A. Sadeghian, S. Minaee, X. Li, B. Cowan and D. Z. Wang

Sep., 26th 2021 – RecTour'2021 Workshop @RecSys'2021

Learning dense embeddings for properties

Advantages

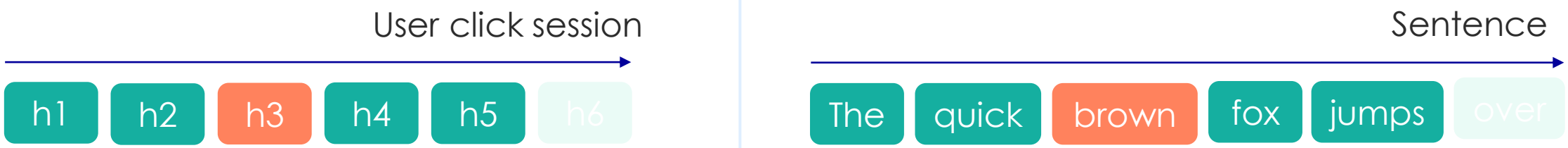
- Represent a property as a dense instead of a sparse vector with lower dimension
- Easier to compute/extract meaningful similarities between properties



Learning dense embeddings for properties

User search click sessions

- Captures semantic similarity
- Assumption: Hotels that appear in the same context should have similar vectors



Approach borrowed from the NLP community : Word2vec

- Distributional hypothesis: Words appearing frequently in similar contexts share statistical properties
- Result: Semantically similar words have geometrically closer vectors
- Applications: Prod2Vec[1], Node2Vec[2], **Airbnb listing embeddings**[3]

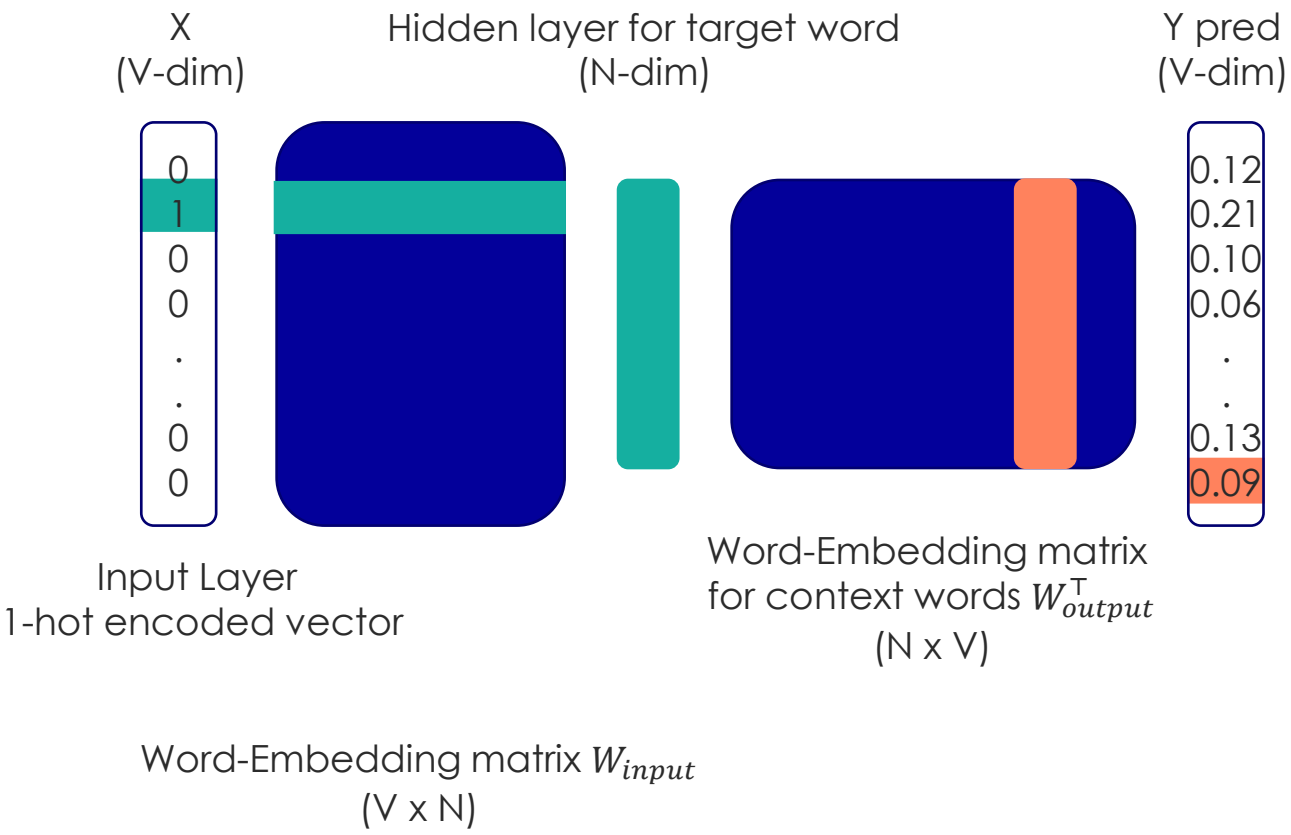
[1] Mihajlo Grbovic, Vladan Radosavljevic, Nemanja Djuric, Narayan Bhamidipati, Jaikit Savla, Varun Bhagwan, and Doug Sharp. 2015. E-commerce in your inbox: Product recommendations at scale. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. 1809–1818.

[2] Aditya Grover and Jure Leskovec. 2016. Node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 855–864.

[3] Mihajlo Grbovic and Haibin Cheng. 2018. *Real-time Personalization using Embeddings for Search Ranking at Airbnb*. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD, 311–320.

Word2Vec

- Let $\{w_0, w_1, \dots, w_T\}$ be a sequence of words in a document with a vocabulary of size V



- Skip-gram cost function:

$$-\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\ne 0}\log p(w_{t+j}|w_t;\theta)$$

where $p(w_{t+j}|w_t;\theta)$ is the probability of observing w_{t+j} given w_t
with parameters $\theta = [W_{input}, W_{output}]$.

- Conditional distribution is defined as:

$$p(w_{context}|w_{target}) = \frac{\exp(W_{output_{context,:}} \cdot W_{input}^{\mathsf{T}} \cdot w_{target})}{\sum_{i=1}^{V}\exp(W_{output_{i,:}} \cdot W_{input}^{\mathsf{T}} \cdot w_{target})}$$

- Word2Vec optimizes an alternative objective using Negative Sampling by drawing negative samples from a noise distribution

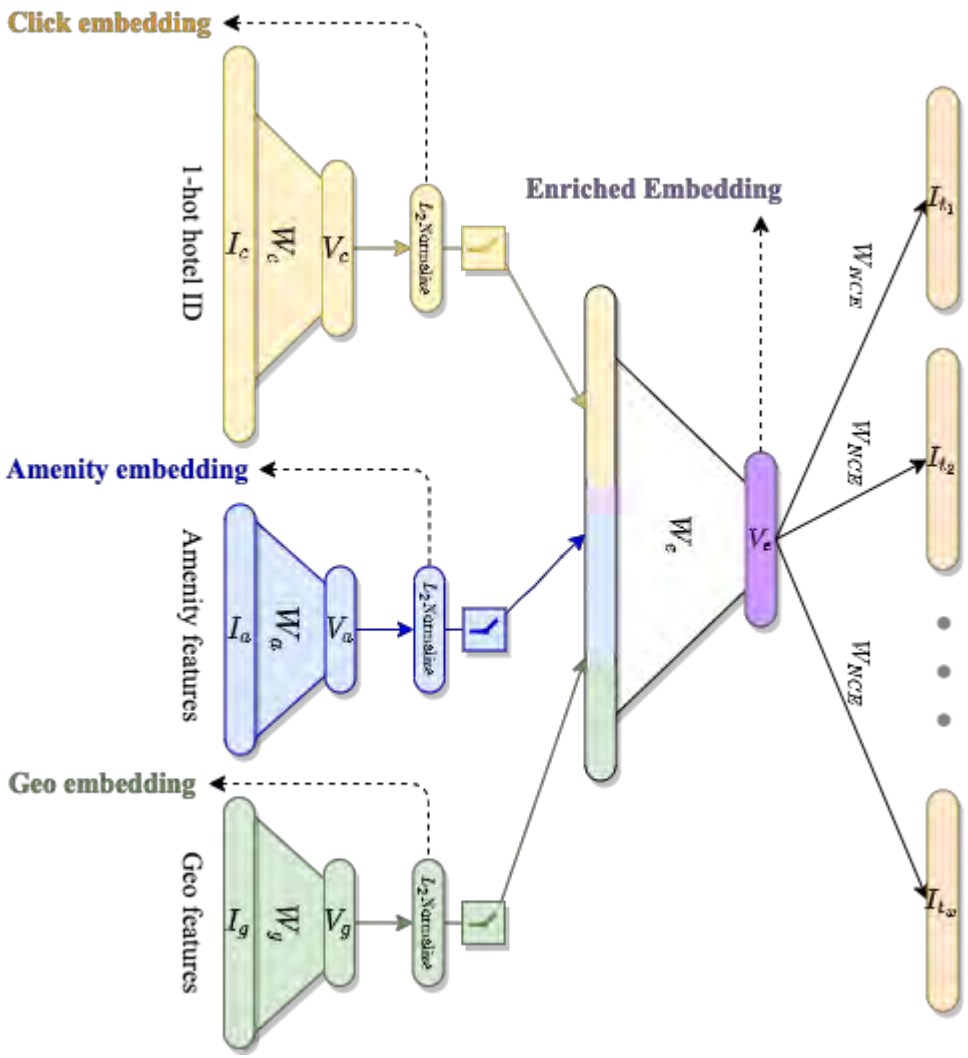
Proposed model: from Session-32 to Hotel2Vec

User click session

Hotel2Vec

- Click sessions are very sparse (7 days for attribution)
- Session-32: "*Real-time Personalization using Embeddings for Search Ranking at Airbnb*", Grbovic & Cheng. KDD 2018.
- The idea is to complement with attribute features
- Learn richer representations
- Remedy to the cold-start problem

- $V_c = ReLU(\frac{I_c \boldsymbol{W}_c}{||I_c \boldsymbol{W}_c||_2})$
- $V_a = ReLU(\frac{I_a \boldsymbol{W}_a}{||I_a \boldsymbol{W}_a||_2})$
- $V_g = ReLU(\frac{I_g \boldsymbol{W}_g}{||I_g \boldsymbol{W}_g||_2})$
- $V_e = ReLU([V_c, V_a, V_g]^{\top} \boldsymbol{W}_e)$



Proposed model

Clicked 1-hot hotel id

$I_c \in \{0,1\}^{1.4Mio}$,

$V_c \in \mathbb{R}^{32}$

Amenity features

Guest rating

Has pool

Are pets allowed

Has spa services

Etc.

$I_a \in \mathbb{R}^{58}, V_a \in \mathbb{R}^{15}$

Geographical features

Space2Vec [1]: lat long encoded as concatenation of multi-scale representations

4*16=64-dim

H3 embeddings [2]: 8-dim

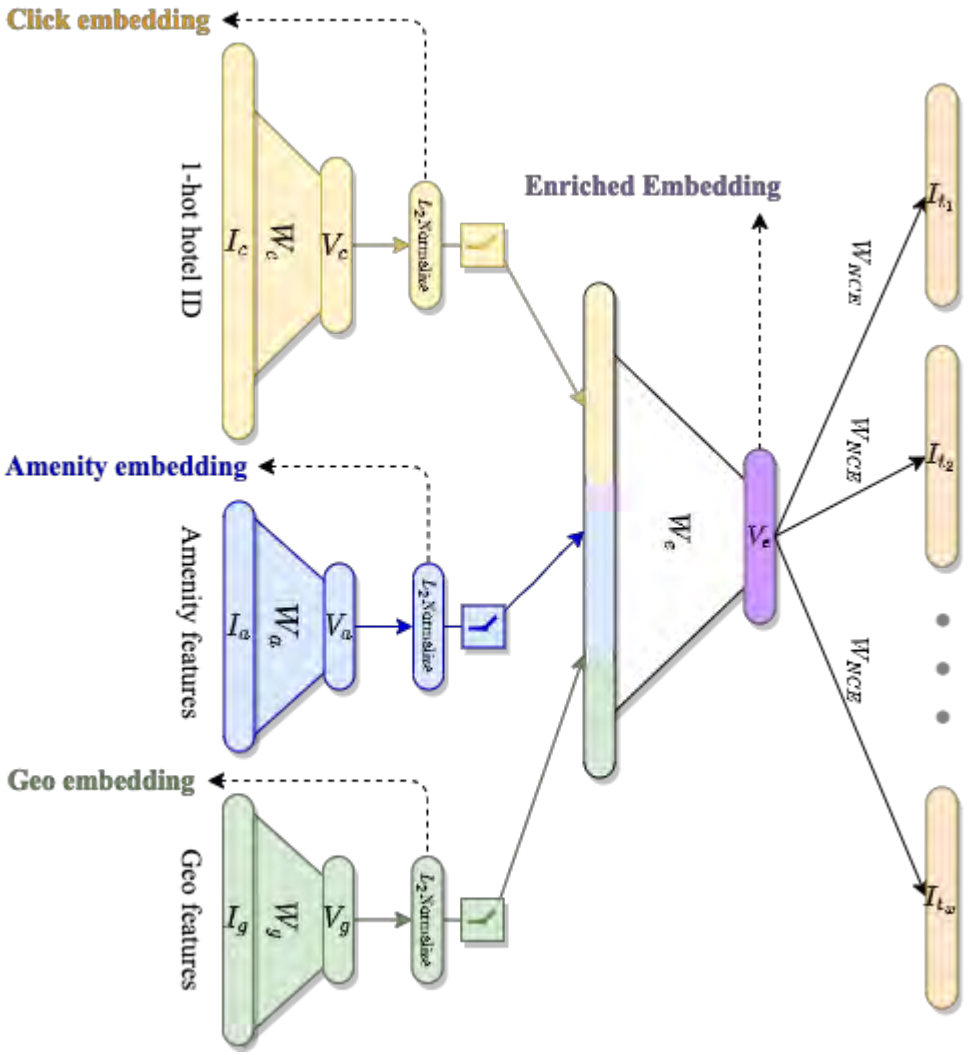
$I_g \in \mathbb{R}^{72}, V_g \in \mathbb{R}^{36}$

$$V_c = ReLU\left(\frac{I_c \boldsymbol{W}_c}{||I_c \boldsymbol{W}_c||_2}\right)$$

$$V_a = ReLU\left(\frac{I_a \boldsymbol{W}_a}{||I_a \boldsymbol{W}_a||_2}\right)$$

$$V_g = ReLU\left(\frac{I_g \boldsymbol{W}_g}{||I_g \boldsymbol{W}_g||_2}\right)$$

$$V_e = ReLU([V_c, V_a, V_g]^{\top} \boldsymbol{W}_e)$$



[1] Gengchen Mai, Krzysztof Janowicz, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. 2020. *Multi-Scale Representation Learning for Spatial Feature Distributions using Grid Cells.* In International Conference on Learning Representations. https://openreview.net/forum?id=rJljdh4KDH

[2] https://eng.uber.com/h3/

Training

User click session

h1	h2	h3	h4	h5	h6
h1	h2	h3	h4	h5	h6
h1	h2	h3	h4	h5	h6
h1	h2	h3	h4	h5	h6

…

Training examples

{(h1,h2), (h1,h3)}

{(h2,h1), (h2,h3), (h2,h4)}

{(h3,h1), (h3,h2), (h3,h4), (h3,h5)}

{(h4,h2), (h4,h3), (h4,h5), (h4,h6)}

…

Target hotel h_t

Context hotel h_c in window = 2

Another context hotel outside window

Negative examples within the market

$$J(\theta) = -\frac{1}{T}\sum_{t=1}^{T}\{\ \log P(h_{c_t}|h_t) + \sum_{h_i \in N_c} \log \sigma(-V_{e_t}^{\top} W_{h_i,:})\}$$

$$\log P(h_c|h_t) = \log \sigma(V_{e_t}^{\top} W_{c,:})$$

Dataset statistics

65M
User click sessions

1.4M
Unique hotels

6
Average number of clicks per user session

2
Minimal number of clicks per user session

50
Maximal number of clicks per user session

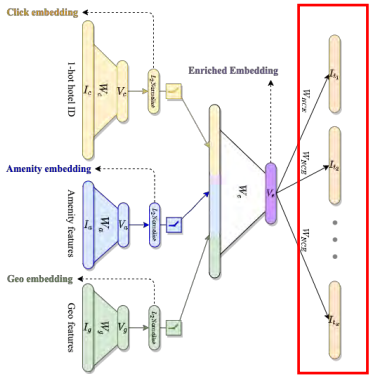
Experimental framework

- 8:1:1 ratio for training, validation and test
- We use a system with 64GB RAM, 8 CPU cores, and a Tesla V100 GPU for training

Hyperparameters

- Learning rate: {0.01, 0.05, 0.1, **0.5**, 1.0, 2.5}
- Embedding dimensions: {**32**, 64,128}
- Batch size: {256, 512, **1024**, 4096}
- Optimization: {**SGD**, Adam, RMSProp}
- SGD with exponential decay (power=0.99 and staircase steps per 40k training steps)
  - Number of negative samples: 2000

Hits@k for hotel context prediction - using softmax probability



On all inventory

Methods	Hits@10	Hits@100	MRR@10	MRR@100
Session-32[1]	0.1565	0.5352	0.0512	0.0689
Hotel2Vec no geo	0.1763	0.5585	0.0587	0.0728
Hotel2Vec	**0.1807**	**0.5671**	**0.0604**	**0.0746**

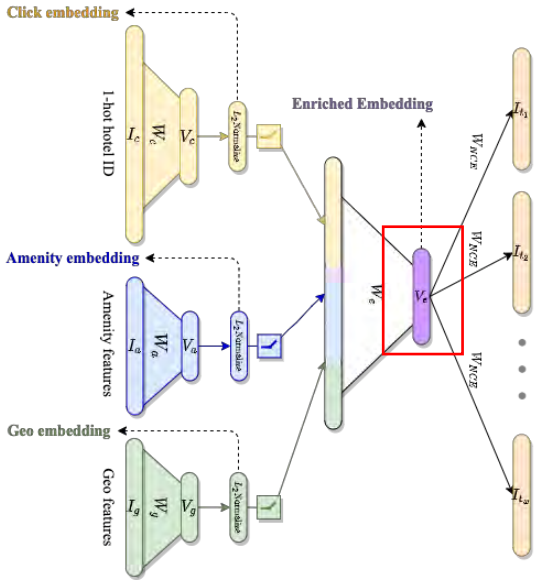
Within the same market

Methods	Hits@10	Hits@100	MRR@10	MRR@100
Highest Rated	0.0158	0.0102	0.0029	0.0032
Most Popular (last year)	0.0739	0.1789	0.0187	0.023
Most popular (last week)	0.0928	0.2397	0.0233	0.0292
Session-32[1]	0.1583	0.5562	0.0605	0.0752
Hotel2Vec	**0.1998**	**0.5787**	**0.0675**	**0.0825**

[1] Mihajlo Grbovic and Haibin Cheng.. 2018. *Real-time Personalization using Embeddings for Search Ranking at Airbnb.* In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD, 311–320.

Hits@k for hotel context prediction - using cosine similarity

On all inventory



Methods	Hits@10	Hits@100	MRR@10	MRR@100
graphSAGE[1]	0.0040	0.0212	0.0009	0.0014
Matrix Factorization[2]	0.964	0.4690	0.0297	0.0417
Cleora[3]	0.1360	0.4160	0.0350	0.0449
Session-32[4]	0.141	0.491	0.0389	0.0512
Hotel2Vec	**0.1676**	**0.5341**	**0.0413**	**0.0546**

[1] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. *Inductive Representation Learning on Large Graphs*. In Advances in Neural Information Processing Systems, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.pdf

[2] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. *Matrix Factorization Techniques for Recommender Systems*. Computer 42, 8 (Aug. 2009), 30–37. https://doi.org/10.1109/MC.2009.263

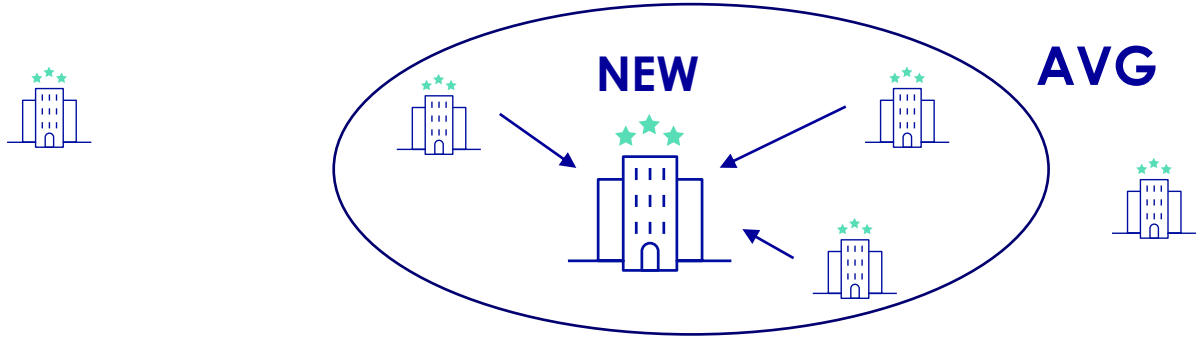
[3] Barbara Rychalska, Piotr Bąbel, Konrad Gołuchowski, Andrzej Michałowski, and Jacek Dąbrowski. 2021. *Cleora: A Simple, Strong and Scalable Graph Embedding Scheme*. arXiv:2102.02302 [cs.LG]

[4] Mihajlo Grbovic and Haibin Cheng. 2018. *Real-time Personalization using Embeddings for Search Ranking at Airbnb.* In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD, 311–320.

Cold start experiments

Same-market prediction results when the target hotel is an unseen hotel, click embeddings imputed by avg top-100 similar hotel embeddings in market

Methods	Hits@10	Hits@100	MRR@10	MRR@100
Matrix Factorization[1]	0.0196	0.0380	0.0053	0.0061
Cleora[2]	0.0131	0.0067	0.0019	0.0022
Session-32[3]	0.0296	0.0632	0.0079	0.0093
Hotel2Vec	**0.0513**	**0.1248**	**0.0132**	**0.0162**



[1] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. *Matrix Factorization Techniques for Recommender Systems*. Computer 42, 8 (Aug. 2009), 30–37. https://doi.org/10.1109/MC.2009.263

[2] Barbara Rychalska, Piotr Bąbel, Konrad Gołuchowski, Andrzej Michałowski, and Jacek Dąbrowski. 2021. *Cleora: A Simple, Strong and Scalable Graph Embedding Scheme*. arXiv:2102.02302 [cs.LG]

[3] Mihajlo Grbovic and Haibin Cheng. 2018. *Real-time Personalization using Embeddings for Search Ranking at Airbnb.* In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD, 311–320.

A/B test results in Lodging ranking model

Ranking model: Factorization Machines

Search features

Destination

Dates

Number of travelers

Etc.

Property features

Price

Ratings

Geographical information

Etc.

+

Property embedding

Property embedding

Matrix Factorization — Test 1

Hotel2Vec — Test 2

A/B test results in Lodging ranking model

Test 1

- Matrix Factorization embeddings

Neutral
Convergence Rate

Positive
Gross Profit

- Duration: 1 month
- Total of ~25M of user ids

Test 2

- Hotel2Vec embeddings

Positive
Convergence Rate

Neutral
Gross Profit

Current and future work

Online updates of the model

How to ensure that the vectors do not drift?

Add more information in the model

- Learn an embedding for the type of the property
- Add pricing information (dynamic)
- More amenities
- Add more geographical information

Thank You

Q&A

expedia group™

Hits@k for hotel context prediction

Metrics

- Hits@k: avg # of times the correct selection appears in the top k hotels
- MRR@k: evaluates avg list quality of top k items returned by looking at the rank
- K = {10, 100}

Scenarii

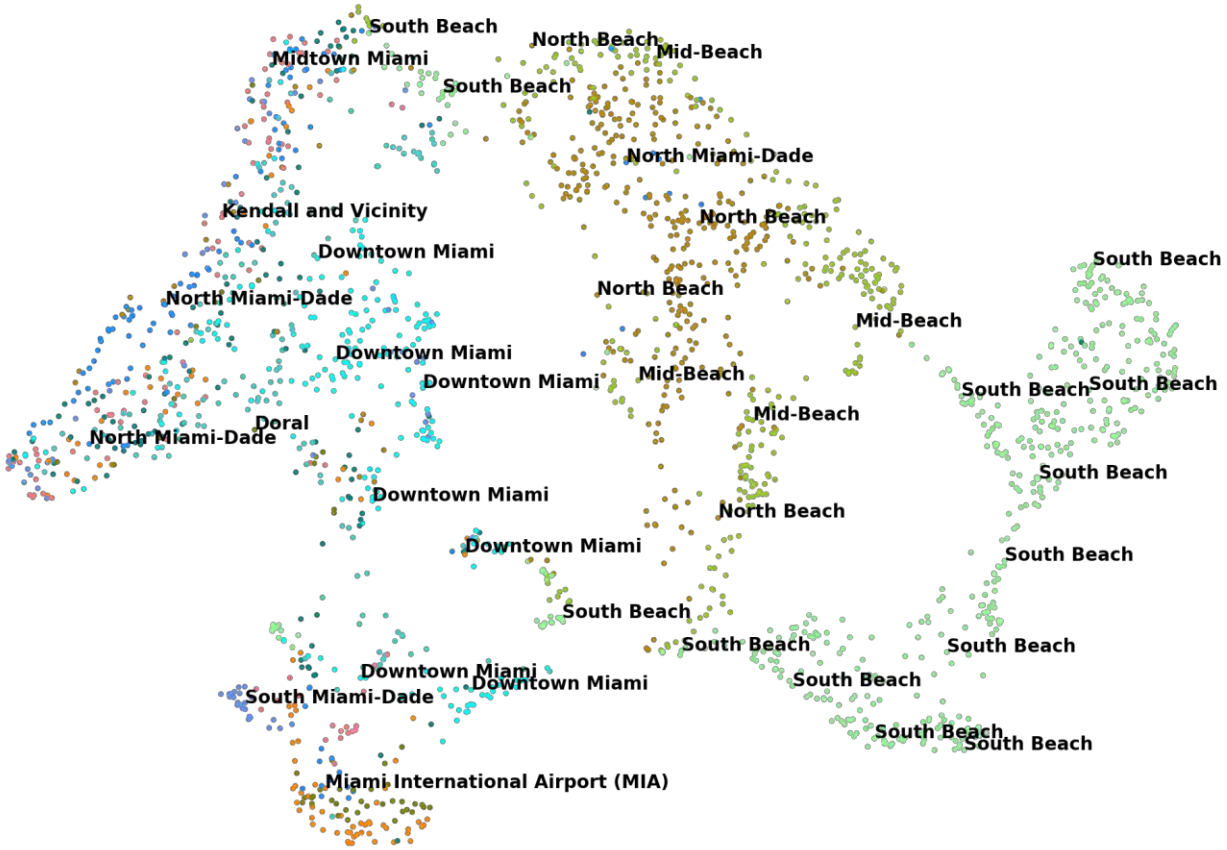
- Raw: predict among all available 1.4M hotels
- Filtered: predict only within the market

We order by

- Softmax probability
- Cosine similarity (advantage: we can test more baselines)

Visualization of embedding clusters

UMAP visualization of hotel embeddings from the Miami area. Different colors represent expert annotations of competing hotels.



Hotel2vec embeddings

Session-32 embeddings