ML-based Visualization Recommendation: Learning to Recommend Visualizations from Data

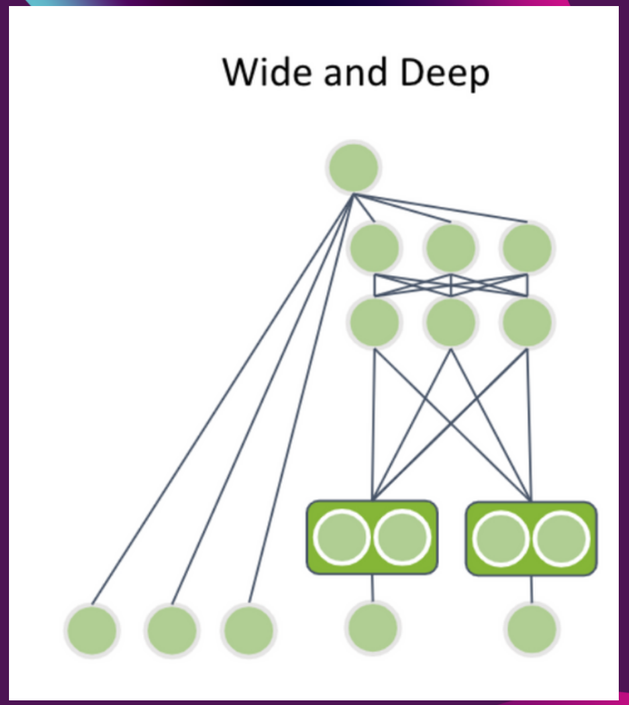
Authors: Xin Qian, Ryan A. Rossi, Fan Du, Sungchul Kim, Eunyee Koh, Sana Malik, Tak Yeon Lee, Joel Chan

Team Members:

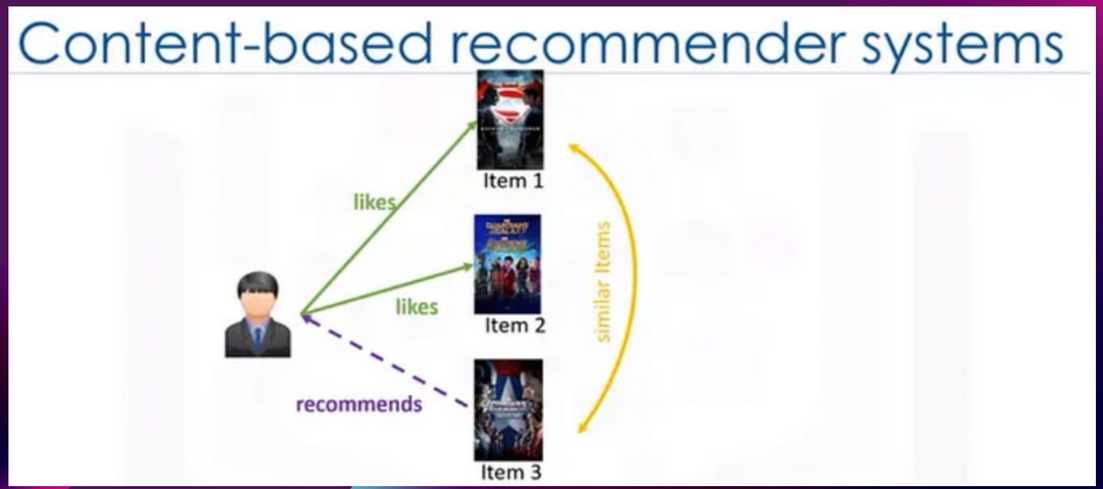
1. Arihant Dugar (B00917961) ar968345@dal.ca
2. Abhinav Acharya Tirumala Vinjamuri (B00929073) ab806657@dal.ca

Introduction

- Rule-Based Recommender Algorithms
- Wide and Deep Learning
- Content-based recommendation model



Rule Based Learning Model



Problems Addressed

- Problem: Existing visualization recommendation systems are inefficient and ineffective
- Approach: Machine learning-based method for visualization recommendation using wide-and-deep learning
- Goal: To capture the latent characteristics of datasets and visualizations, and to generate, score, and recommend useful visualizations
- Evaluation: Modified nDCG metric and user study to measure the effectiveness and utility of the recommended visualizations
- Model: Wide and Deep Learning Content-Based Visualization Recommendation Model.

Note:

- nDCG - normalized Discounted Cumulative Gain

Research Paper Questions

- How to automatically generate, score, and recommend useful visualizations from a large corpus of datasets and visualizations?
- How to learn a general and data-driven model for visualization recommendation that can adapt to new datasets and user preferences?
- How to evaluate the effectiveness and utility of the recommended visualizations using quantitative and qualitative methods?
- How to leverage machine learning techniques such as wide-and-deep learning to capture the latent characteristics of datasets and visualizations?

Relevance

	Research Paper	**Our Project**
Focus	Recommendations	Recommendations & Analysis
Dataset	Large corpus of datasets	Spotify Music Dataset
Solution	Recommend useful and effective visualizations for unseen data.	Interactive music-feature, genre exploration, and User matching through effective and useful visualizations.
Inference	Providing insight to users	

Dataset

- Training corpus of 1K datasets (and their visualizations) from the Plot.ly corpus
- 12 attributes each
- Source: https://chart-studio.plotly.com/feed/#/

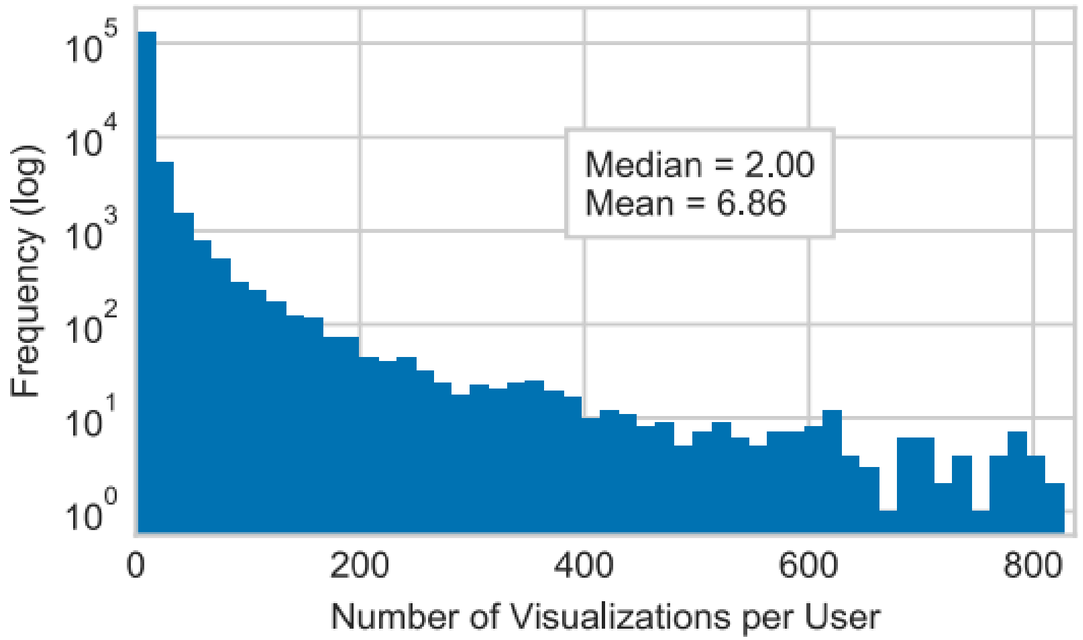
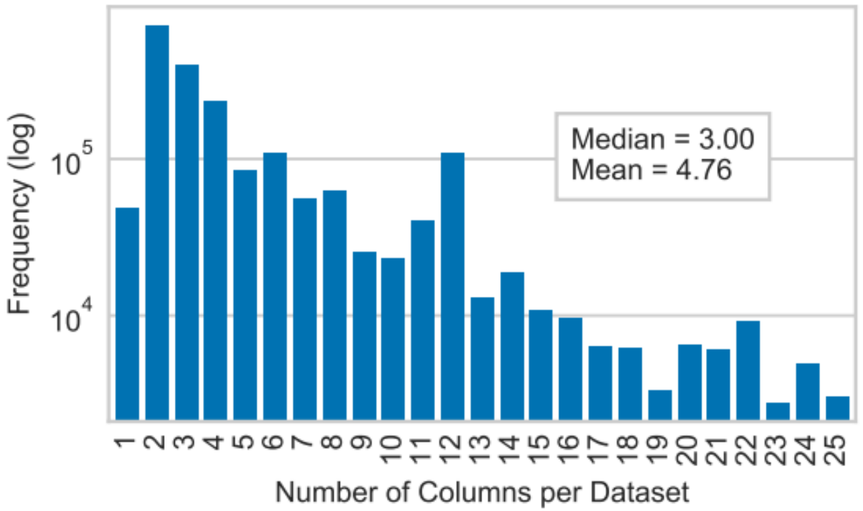
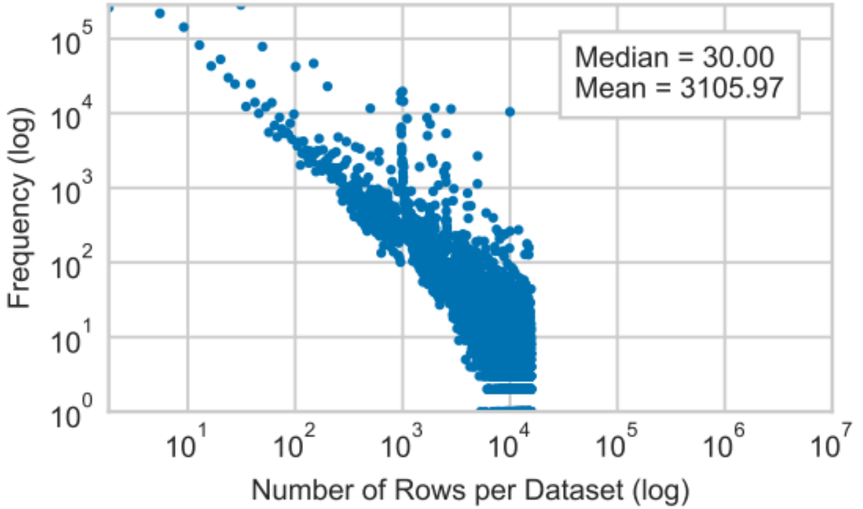


Fig. 4: Distribution of plots per user, visualized on a log-log scale.



(a) Distribution of columns per dataset, after removing the 5.03% of datasets with more than 25 columns, visualized on a log-linear scale.



(b) Distribution of rows per dataset, visualized on a log-log scale.

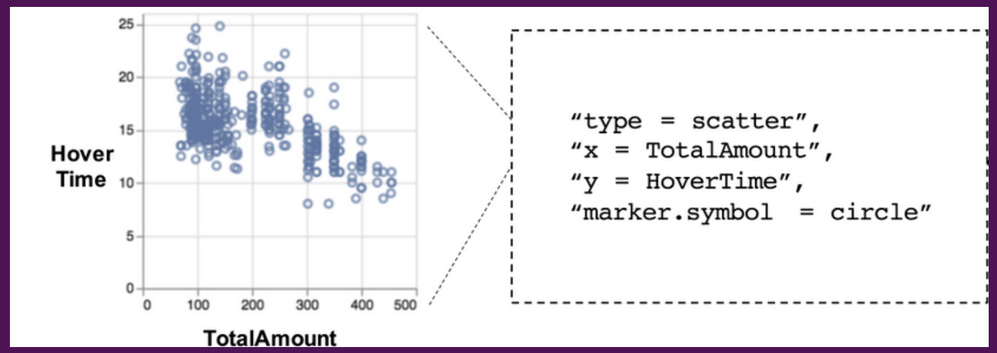
Fig. 5: Distribution of dataset dimensions in the Plotly corpus.

Solutions

Training Dataset Corpus

ID	Timestamp	HoverTime	Currency	Browser	Disp	...	TotalAmount
2E073CA78 5310000	2018/12/10 14:33	18	USD	Mozilla (iPhone)	en-gb		121
2E073CA78 5310001	2018/12/10 14:15	15	EUR	Chrome (Mac OS)	de-DE		97
...							
2E073CA78 5320000	2018/12/10 14:36	31	GBP	Safari (iPhone)	en-US		238

Visualizations on the training dataset



Model

Positive Visualization

A configuration

```
{
    "type = scatter",
    "x-type = quantitative",
    "y-type = quantitative",
    "marker.symbol = circle",
}
```

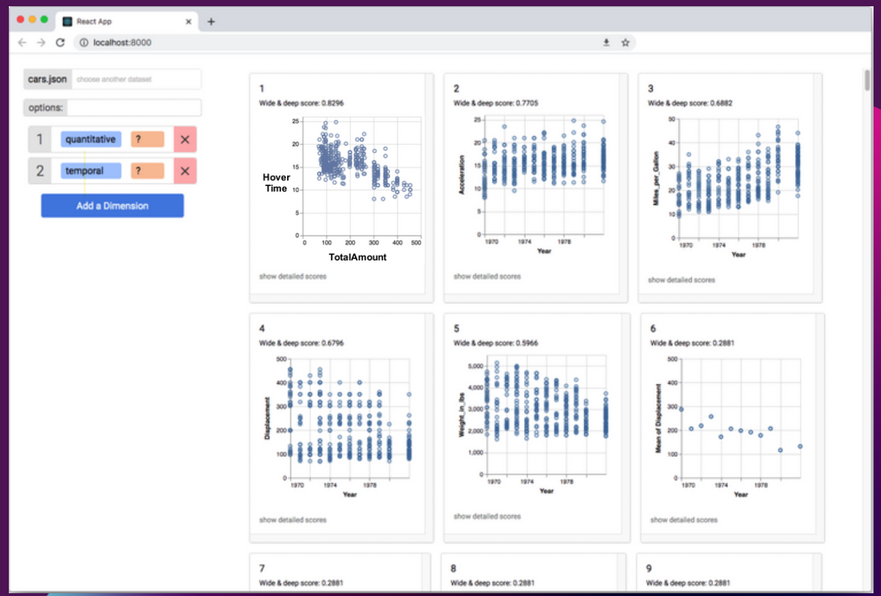
Attribute combination

[TotalAmount, HoverTime]

Similar unseen dataset corpus

ID	Timestamp	HoverTime	Currency	Browser	Disp	...	TotalAmount
2E073CA78 5310000	2018/12/10 14:33	18	USD	Mozilla (iPhone)	en-gb		121
2E073CA78 5310001	2018/12/10 14:15	15	EUR	Chrome (Mac OS)	de-DE		97
...							
2E073CA78 5320000	2018/12/10 14:36	31	GBP	Safari (iPhone)	en-US		238

Recommended visualizations dashboard



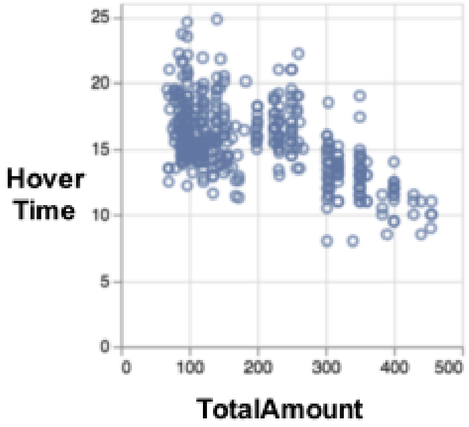
Visualization Data Extraction Process

ML-based Visualization Recommendation:
Learning to Recommend Visualizations from Data

A dataset with ***many*** attributes (top) and ***one of*** its visualizations (bottom left)

ID	Timestamp	HoverTime	Currency	Browser	Disp	...	TotalAmount
2E073CA78 5310000	2018/12/10 14:33	18	USD	Mozilla (iPhone)	en-gb		121
2E073CA78 5310001	2018/12/10 14:15	15	EUR	Chrome (Mac OS)	de-DE		97
...							
2E073CA78 5320000	2018/12/10 14:36	31	GBP	Safari (iPhone)	en-US		238

The visualization uses ***a subset of*** attributes, *HoverTime* and *TotalAmount*




```
"type = scatter",
"x = TotalAmount",
"y = HoverTime",
"marker.symbol = circle"
```

A positive visualization

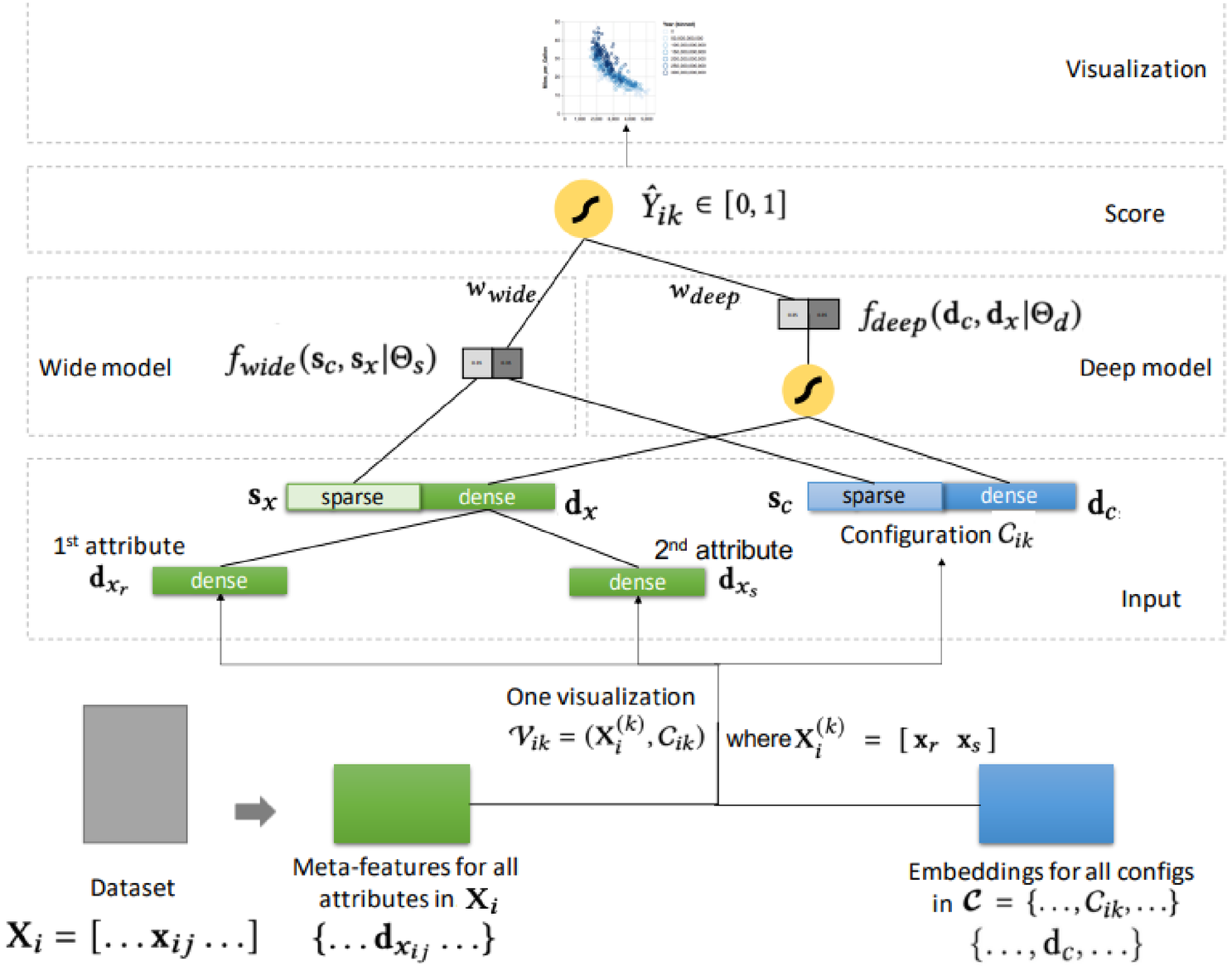
A configuration

```
{
    "type = scatter",
    "x-type = quantitative",
    "y-type = quantitative",
    "marker.symbol = circle",
}
```

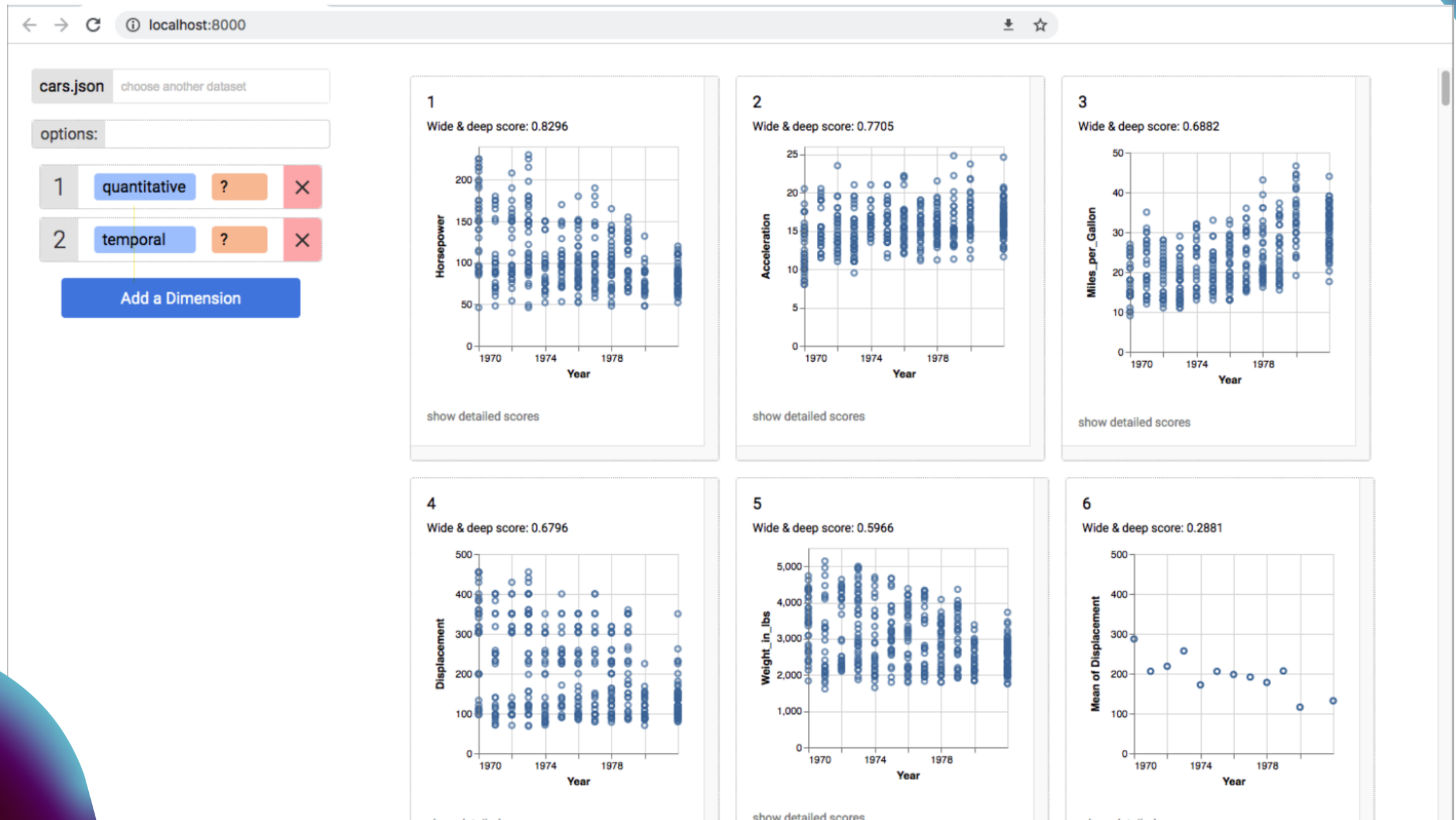
Attribute combination

[TotalAmount, HoverTime]

Model Design



Dashboard


localhost:8000
cars.json
choose another dataset
options:
1
quantitative
?
2
temporal
?
Add a Dimension
1
Wide & deep score: 0.8296
Horsepower
Year
show detailed scores
2
Wide & deep score: 0.7705
Acceleration
Year
show detailed scores
3
Wide & deep score: 0.6882
Miles_per_Gallon
Year
show detailed scores
4
Wide & deep score: 0.6796
Displacement
Year
5
Wide & deep score: 0.5966
Weight_in_lbs
Year
6
Wide & deep score: 0.2881
Mean of Displacement
Year

Evaluation - Quantitative Analysis

The modified normalized Discounted Cumulative Gain (nDCG) at k ∈ {1, 2, 5, 10, 20} for the different top-k visualization recommendations (nDCG@k).

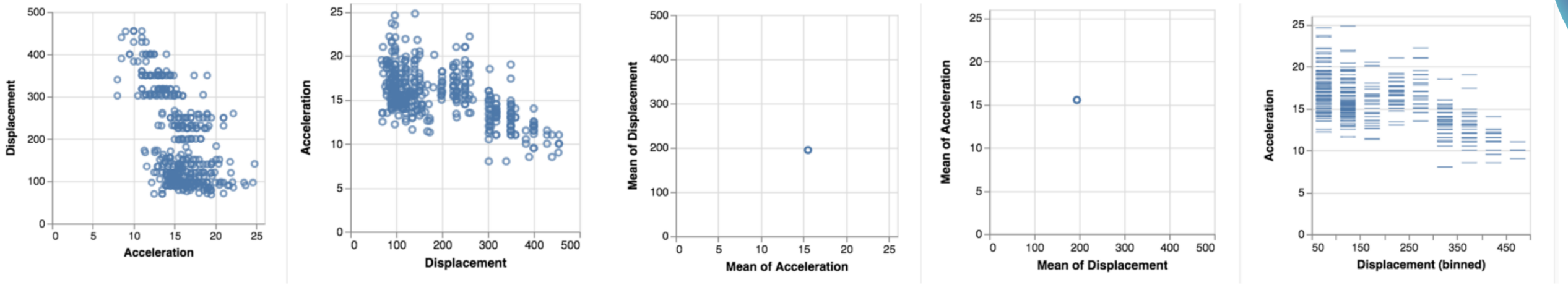
Table 4: Training Corpus Statistics. # Config/Dataset denotes the average number of configurations used by each dataset.

#Datasets	#Vis. Configs	#Attributes	#Visualizations	#Attribute/Dataset	# Vis. Configs/Dataset
925	60	11,778	4,865	11.93	5.89

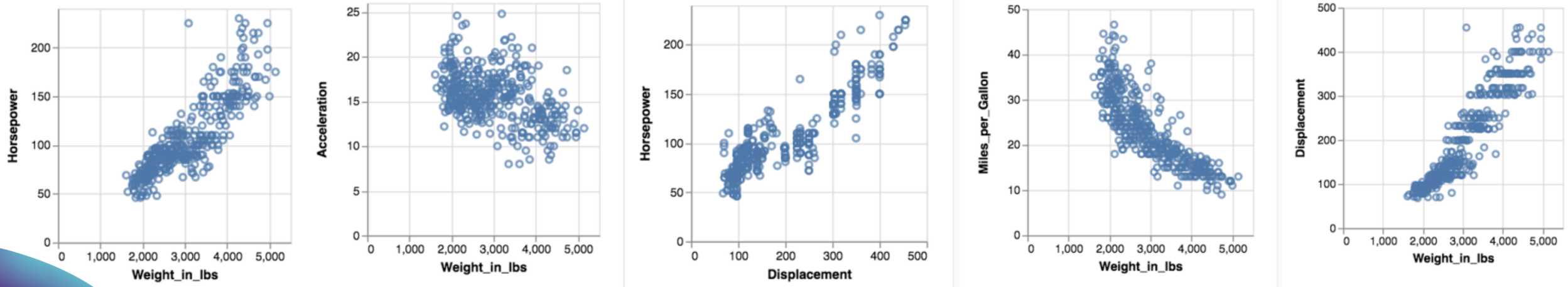
Table 5: Quantitative Results for Visualization Recommendation. See text for discussion.

	nDCG					
Model	@1	@2	@5	@10	@20	Rank
Random	0.207	0.206	0.253	0.311	0.457	5
ConfigPop	0.366	0.532	0.671	0.691	0.693	4
Ours	**0.827**	**0.827**	**0.867**	**0.882**	**0.897**	1
Ours (Deep-only)	0.804	0.807	0.851	0.866	0.887	2
Ours (Wide-only)	0.721	0.714	0.768	0.801	0.839	3

Evaluation - User Study

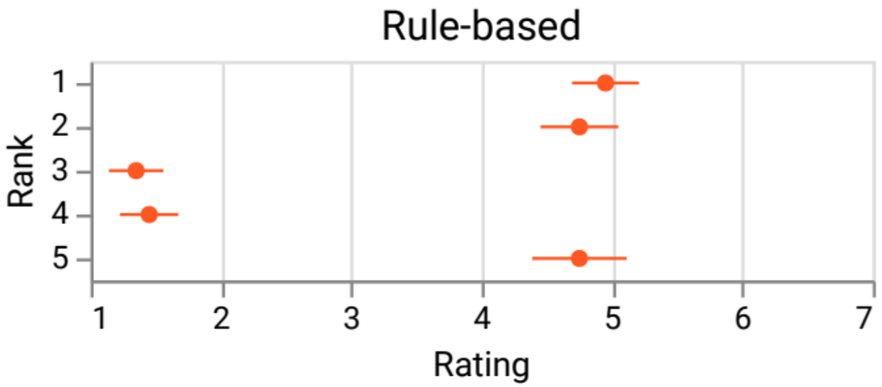


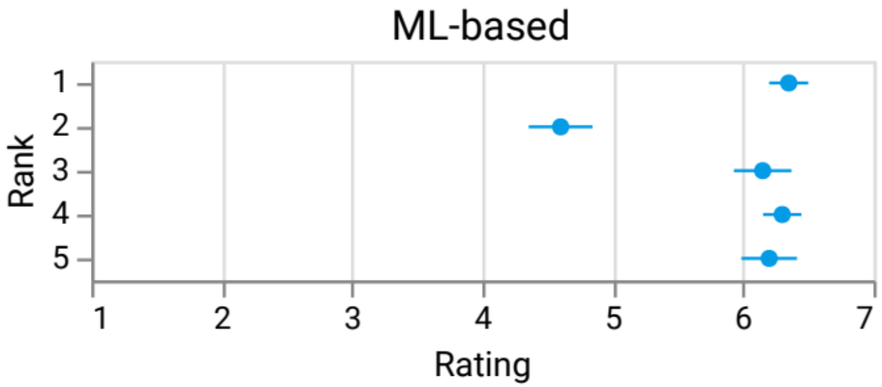
(a) Top 5 Rule-based Vis Rec.



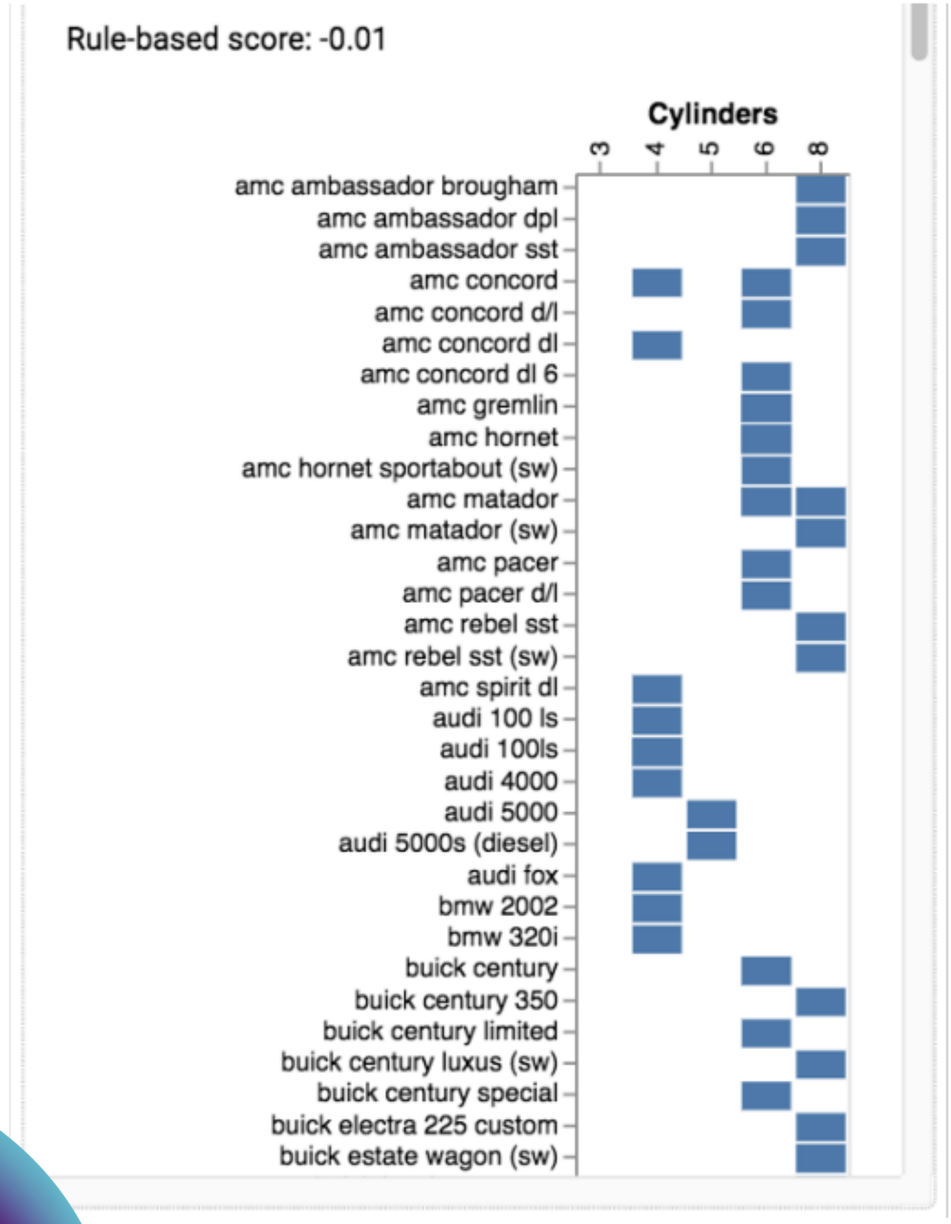
(b) Top 5 ML-based Vis. Rec. (Ours)

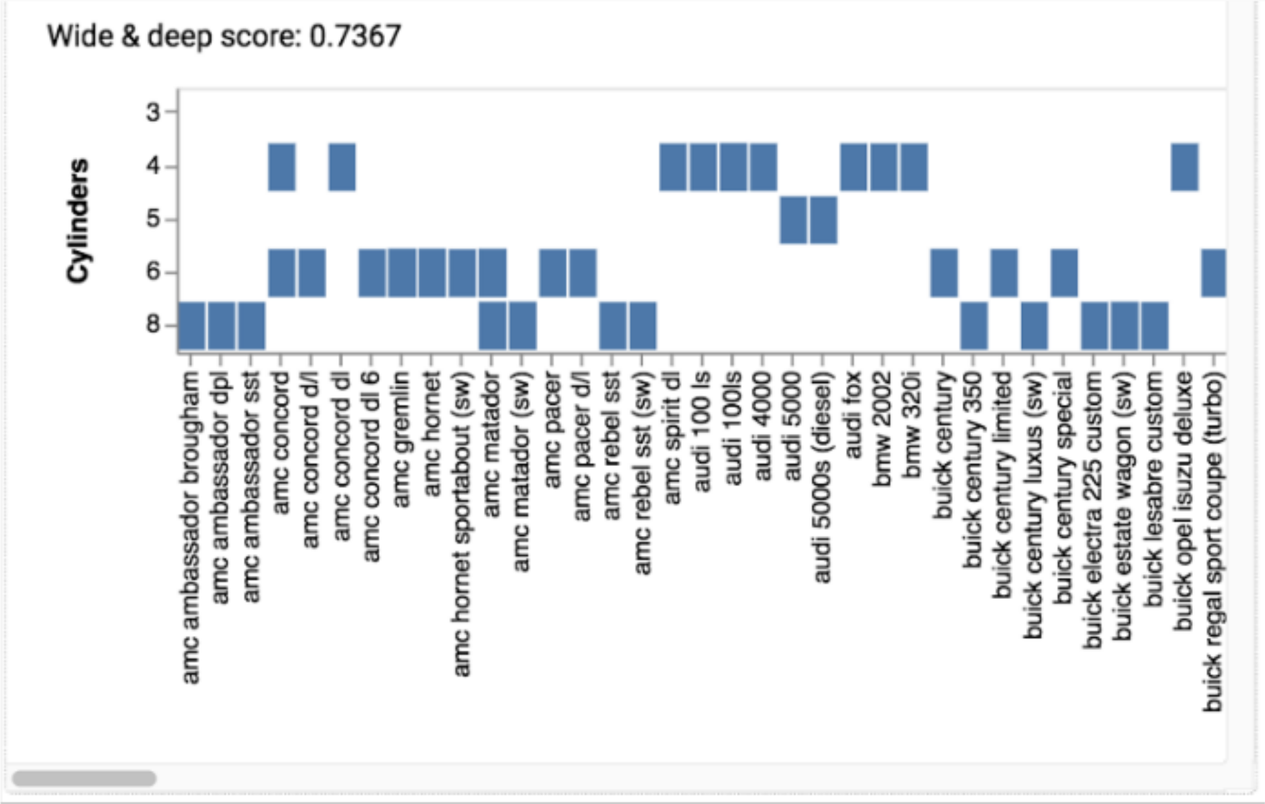
Evaluation - User Study - Expert Rankings





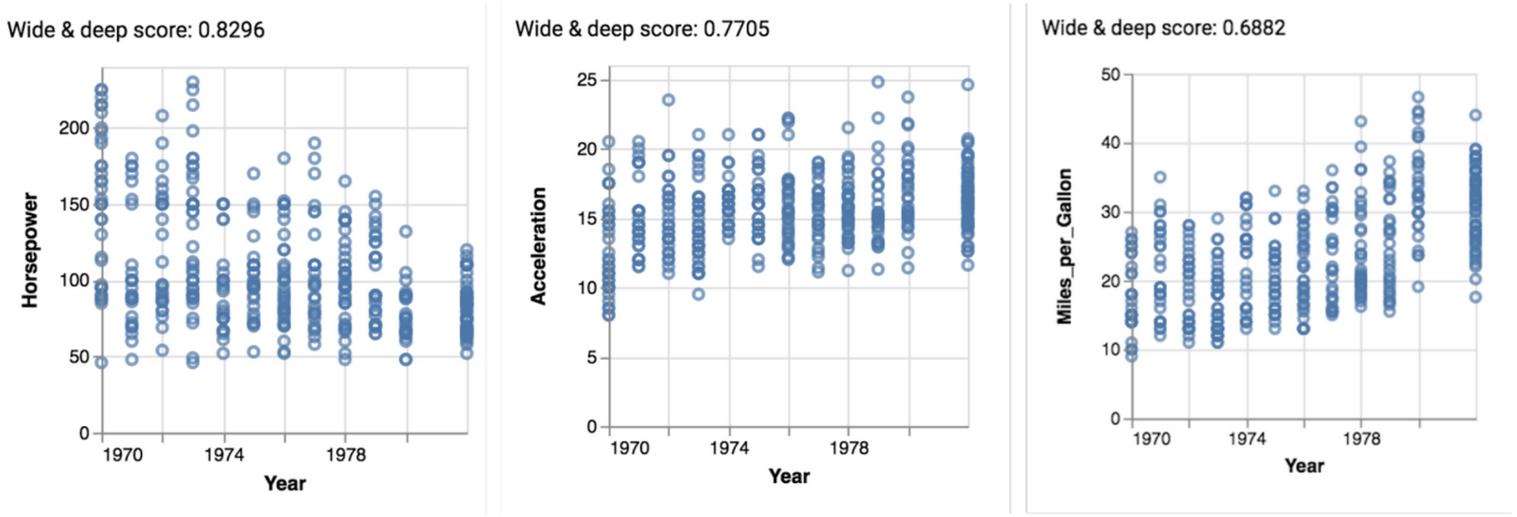
Evaluation - Qualitative Analysis - Learning to place attributes like an expert





Evaluation - Qualitative Analysis - Tie-Breaking Issue

(a) Top 3 Rule-based Vis Rec. (CompassQL/Voyager2)



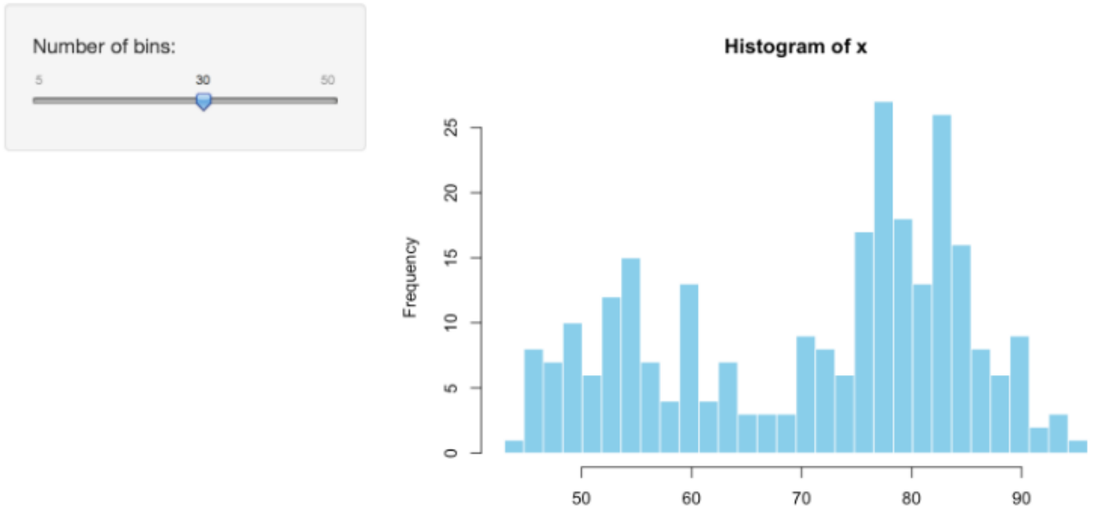
(b) Top 3 ML-based Vis. Rec. (Ours)

Limitations

- No interpretability or explainability of the model.
- No consideration of user preferences, tasks, or goals.
- Exponential visualization and negative visualization space.
- High computational cost and complexity.
- Information loss at the bottom of the recommendation list.
- No domain generalizability / extensibility.

Future Works

- Feedback learning for evaluation and relevance
- User control for complexity adjustment
- Model optimization for space and time
- User choice for visualization selection
- Knowledge base for domain generalization



Thank You

Appendix - Wide Model

- A linear model that uses sparse features of attribute combination and configuration to capture frequent patterns that lead to effective visualizations. The wide score is computed as:

$$f_{wide}(\mathbf{s}_c, \mathbf{s}_x | \Theta_s) = \mathbf{W}_s^T [\mathbf{s}, \mathbf{s}'] + \mathbf{b}_s$$

- where s and s′ are the original and cross-product sparse features, Θs is the set of parameters, and σ is the sigmoid function.
- The goal of leveraging sparse features is to capture any occurrence of feature-pairs that commonly lead to effective visualizations in the training corpus.

Appendix - Deep Model

- A non-linear model that uses dense features of attribute combination and configuration to generalize to less common patterns that may also lead to effective visualizations. The deep score is computed as

$$f_{deep}(\mathbf{d}_c, \mathbf{d}_x|\Theta_d) = a_L(\mathbf{W}_{dL}^T \mathbf{d}_{L-1} + \mathbf{b}_{dL}),$$

- where dc and dx are the dense features of configuration and attribute combination, Θd is the set of parameters, dL is the output of the last hidden layer, and σ is the sigmoid function.

Appendix - Wide and Deep Model

- A hybrid model that combines the wide and deep models to leverage both sparse and dense features. The final score is computed as:

$$f(\mathbf{X}_{\text{test}}^{(k)}, C|\Theta) = \sigma(w_{wide} f_{wide}(\mathbf{s}_c, \mathbf{s}_x|\Theta_s) + w_{deep} f_{deep}(\mathbf{d}_c, \mathbf{d}_x|\Theta_d))$$

- where wwide and wdeep are the weights for the wide and deep scores, and Θ is the entire set of parameters.
- The entire set of parameters Θ, including Θs , Θd , wwide and wdeep are learned through backward propagation

Appendix - nDCG

- A normalized discounted cumulative gain (nDCG), which is a metric to evaluate the quality of a ranking of items based on their relevance scores.
- A modified nDCG is computed as follows:

$$nDCG@K = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{Z_i^K} \sum_{j=1}^{K} \frac{2^{Y_{ij}} - 1}{log_2(j+1)}$$

$$Z_i^K = \sum_{j=1}^{\min(K, |\mathbb{V}_i|)} \frac{1}{log_2(j+1)}$$

- nDCG is a popular metric for evaluating information retrieval systems, such as search engines, recommender systems, and question answering systems. It is especially useful when the relevance of items is graded rather than binary, and when the order of items matters.
- A good end-to-end learning-based visualization recomemender system must be able to give up some of its performance at the bottom of the list of recommended visualizations to improve the performance at the top.