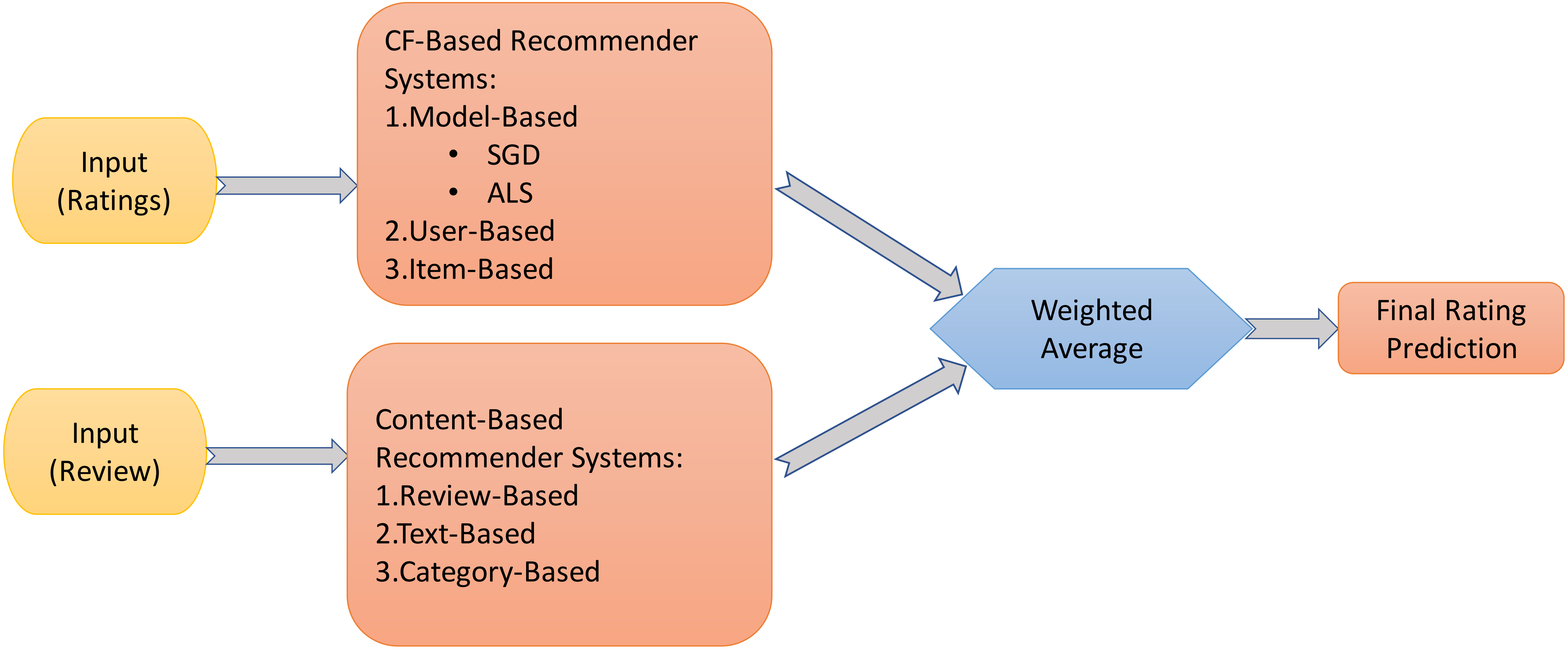
Hybrid Recommender System: Recommending Restaurants to Users

By,
Dharmik Ghoghari
Lakshya Kejriwal
Rishab Kumar

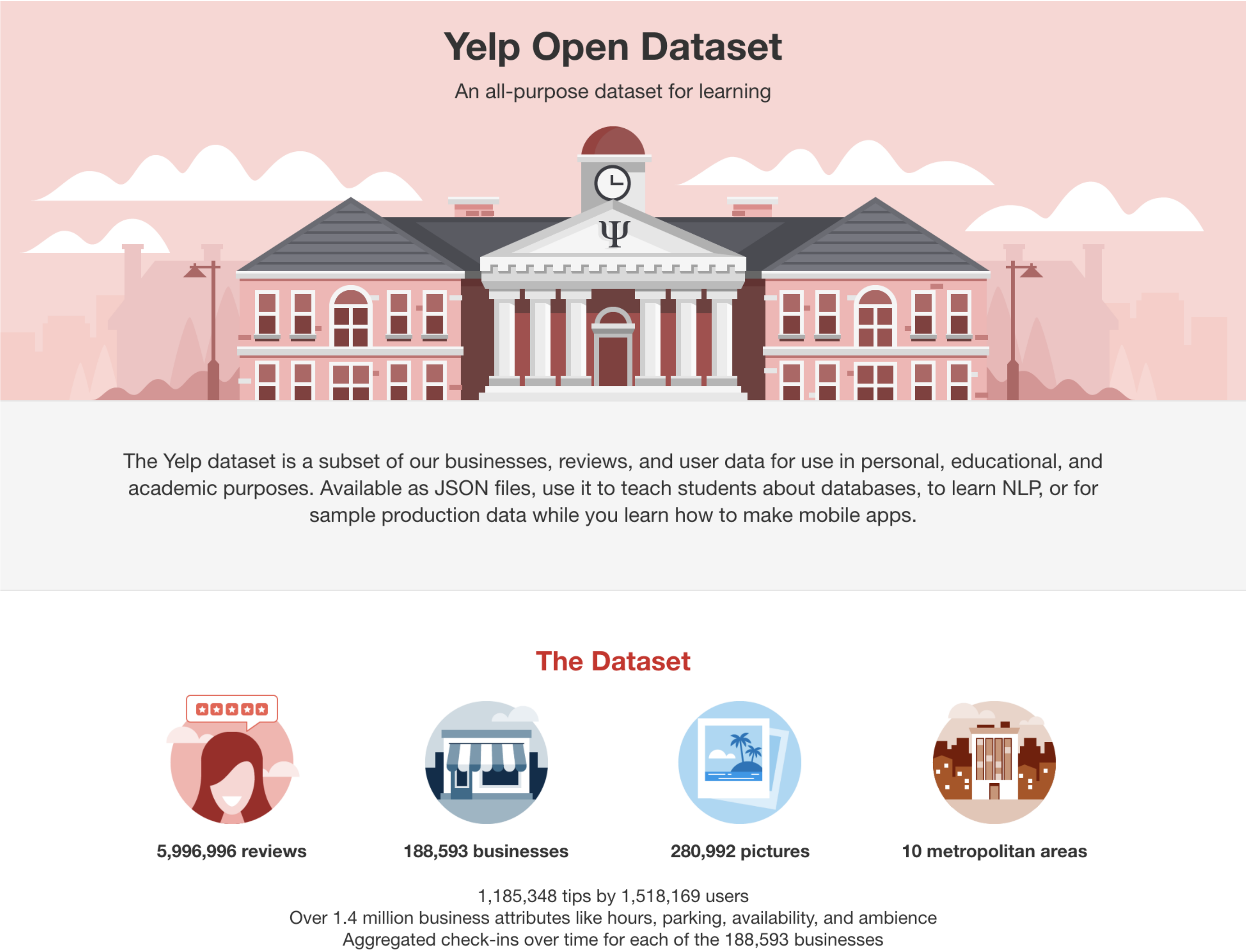
Hybrid-Model Predictions



Dataset

Pittsburgh
- 50,893 reviews
- 987 restaurants
- 3192 users
- Users > 20 reviews
- Faster testing and debugging

Las Vegas
- 158,880 reviews
- 8309 restaurants
- 5272 users
- Users > 50 reviews
- Proof of models built



Methodologies

Collaborative Filtering Based Models

- Model Based CF
  - Stochastic Gradient Descent (SGD)
  - Alternating Least Squares (ALS)
- User Based CF
- Item Based CF

Content Based Models

- Review based
- Text Based
- Category Based

Methodologies

Collaborative Filtering Based Models

- Model Based CF
  - Stochastic Gradient Descent (SGD)
  - Alternating Least Squares (ALS)
- User Based CF
- Item Based CF

Content Based Models

- Review based
- Text Based
- Category Based

Model Based

Ratings Matrix: $X \in R^{U \times I}$

X

$5 \times 5\ matrix$

X_{11}	?	X_{13}	?	?
X_{21}	?	?	X_{24}	X_{25}
?	X_{32}	X_{33}	X_{34}	?
?	X_{42}	?	?	X_{45}
X_{51}	?	X_{53}	?	?

=

P

$3 \times 5\ matrix$

		a		
		b		
		c		

Q

$5 \times 3\ matrix$

e	f	g

$$X_{32} = (a, b, c).(e, f, g) = a * e + b * f + \mathrm{c} * \mathrm{g}$$

rating prediction

$$\hat{r}_{ui} = p_u^T q_i$$

item preference vector

user preference vector

Model Based

minimizing cost function

squared error: e

$$\min \sum_{(u,i)\in D} (r_{ui} - p_u^T q_i)^2 + \lambda(\|P\|^2 + \|Q\|^2)$$

known rating entries

regularization to prevent overfitting

Algorithms to minimize cost function

1. Stochastic Gradient Descent (SGD)

$$q_i = q_i + \alpha(2ep_u + \lambda q_i)$$
$$p_u = p_u + \alpha(2eq_i + \lambda p_u)$$

2. Alternating Least Squares (ALS)

Methodologies

Collaborative Filtering Based Models

- Model Based CF
  - Stochastic Gradient Descent (SGD)
  - Alternating Least Squares (ALS)
- User Based CF
- Item Based CF

Content Based Models

- Review based
- Text Based
- Category Based

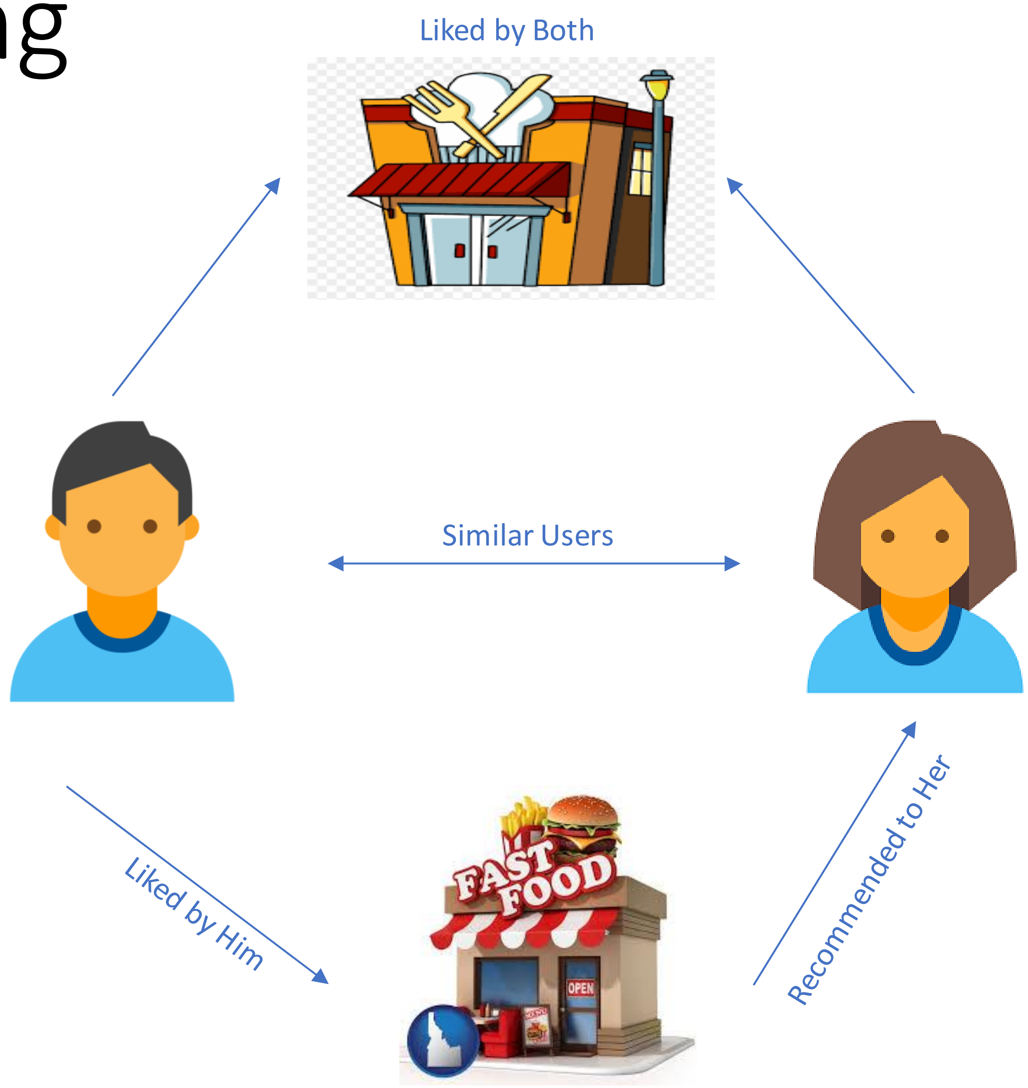
Collaborative-Filtering Based Models

- ## User-Based CF

  Focuses on taking people who have rated similarly to a user, and predicting rating depending upon how others have rated that item

- ## Item-Based CF

  Focuses on taking items which are similar to the given item, and predicting rating depending on those items ratings as given by the user



Methodologies

Collaborative Filtering Based Models

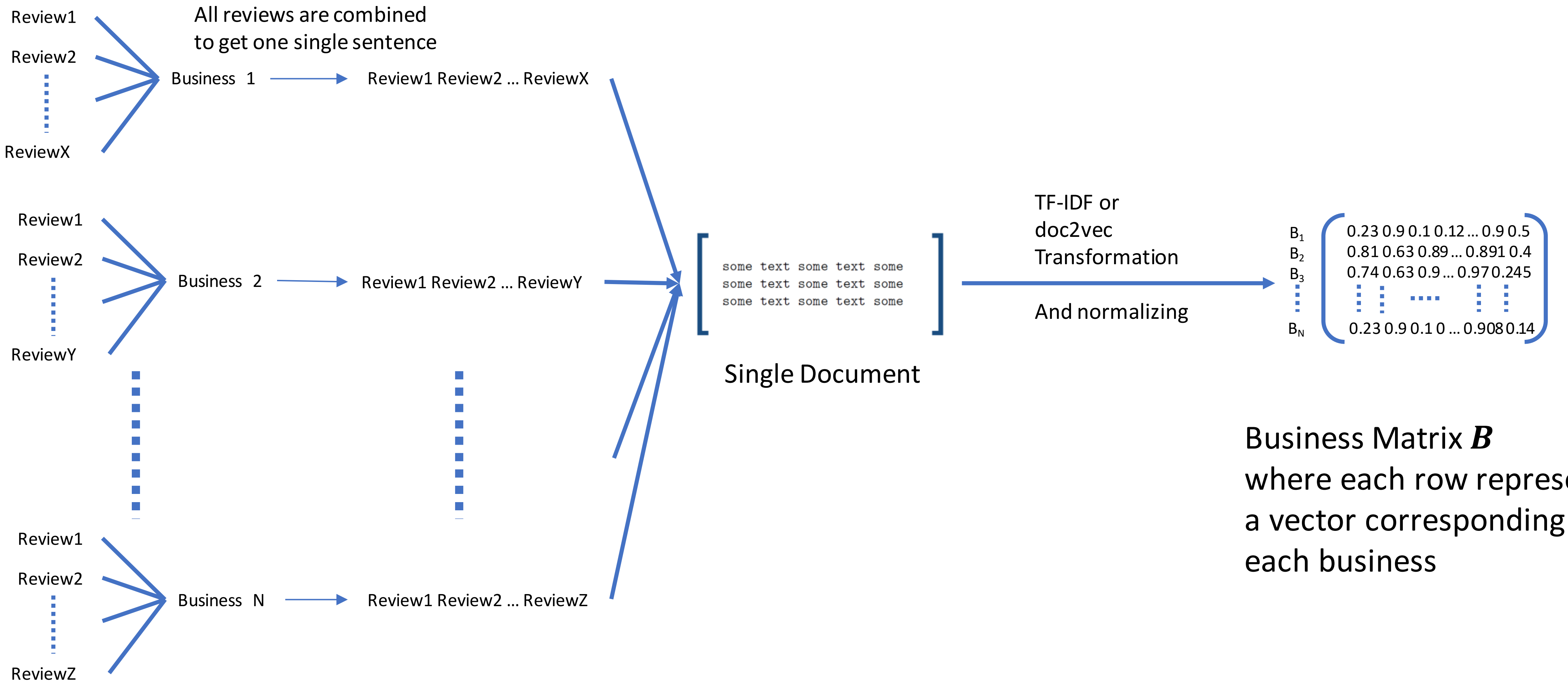
- Model Based CF
  - Stochastic Gradient Descent (SGD)
  - Alternating Least Squares (ALS)
- User Based CF
- Item Based CF

Content Based Models

- Review based
- Text Based
- Category Based

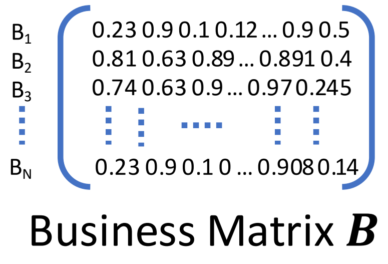
Review-Based Model

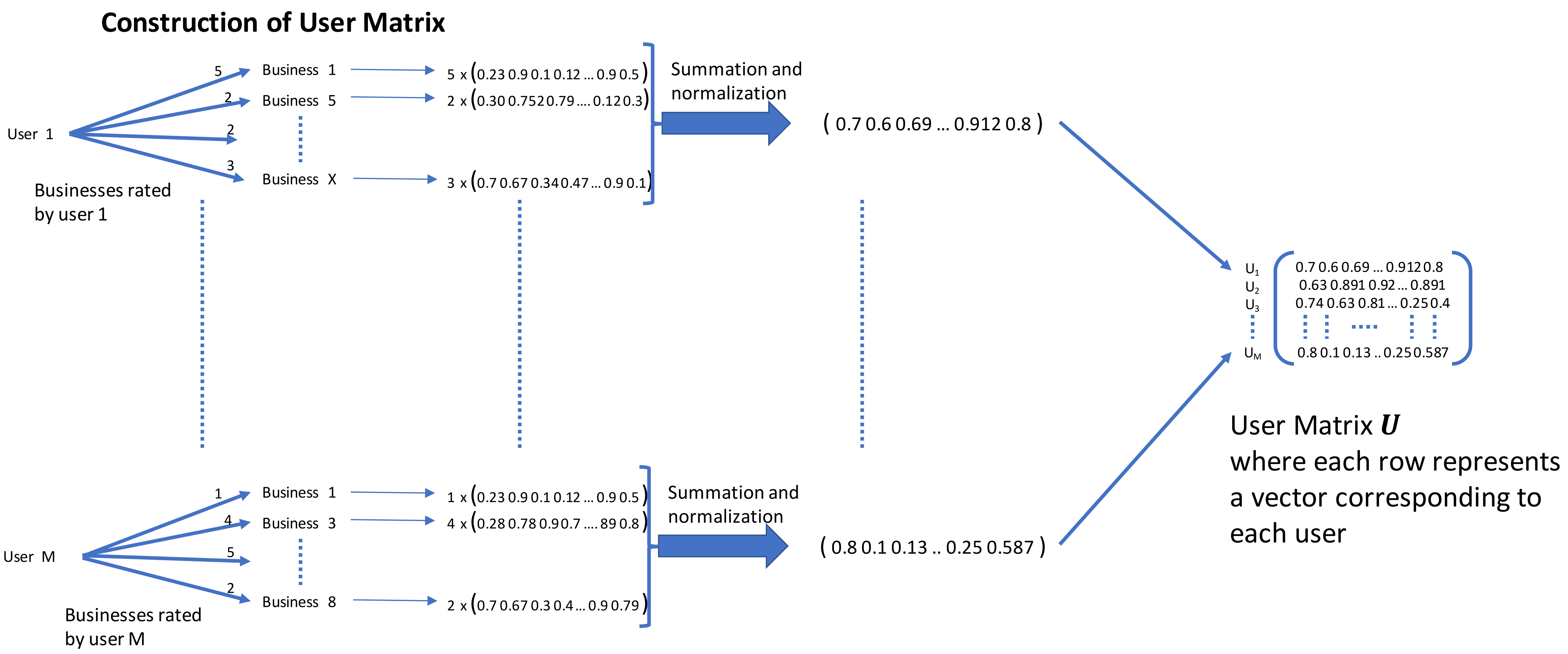
Construction of Business Matrix



Business Matrix $\boldsymbol{B}$
where each row represents
a vector corresponding to
each business

Review-Based Model





Review Based

- To compute the cosine similarity between a User and a Business we compute $S \in R^{I \times Q}$ (I unique Businesses and Q unique Users) by,

User Matrix

$$S = UB^{\mathrm{T}}$$

Similarity Matrix

Business Matrix

	B_1	B_2	B_3	B_4	B_5
U_1	-0.12	0.76	-0.56	0.33	0.36
U_2	0.37	0.21	0.45	0.36	-0.98
U_3	0.83	-0.25	.074	0.11	0.65
U_4	0.26	0.52	0.73	-0.84	0.36
U_5	-0.64	0.28	0.57	0.18	0.82

- To transform the similarity values (-1,1) into ratings (1,5) we build a regression model to regress ratings over similarity values.

Text-Based Model

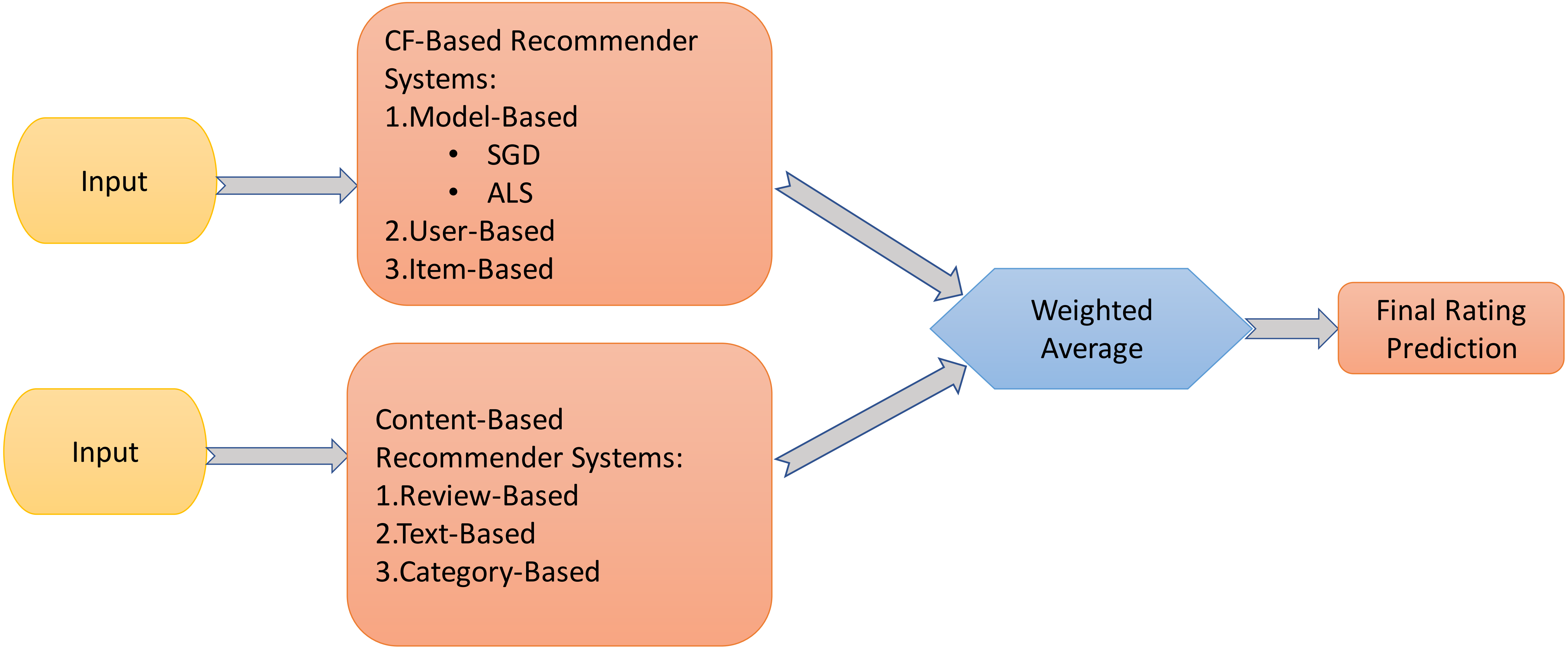
- We compute the Business Matrix B similar to Review-based Model we discussed earlier.
- Here we follow the similar approach to compute the User Matrix U as we did for the Business Matrix.
- Rest we follow the same procedure as in Review Based Model once we have User and Business Matrices.
- We compute the similarity matrix $S = UB^{\mathrm{T}}$, $S \in R^{I \times Q}$ and regress ratings over similarity values.

Category-Based Model

- This method is similar to Review-based Model with different input.
- In Review-based Model we used the review as the input here we use business specific information as features.

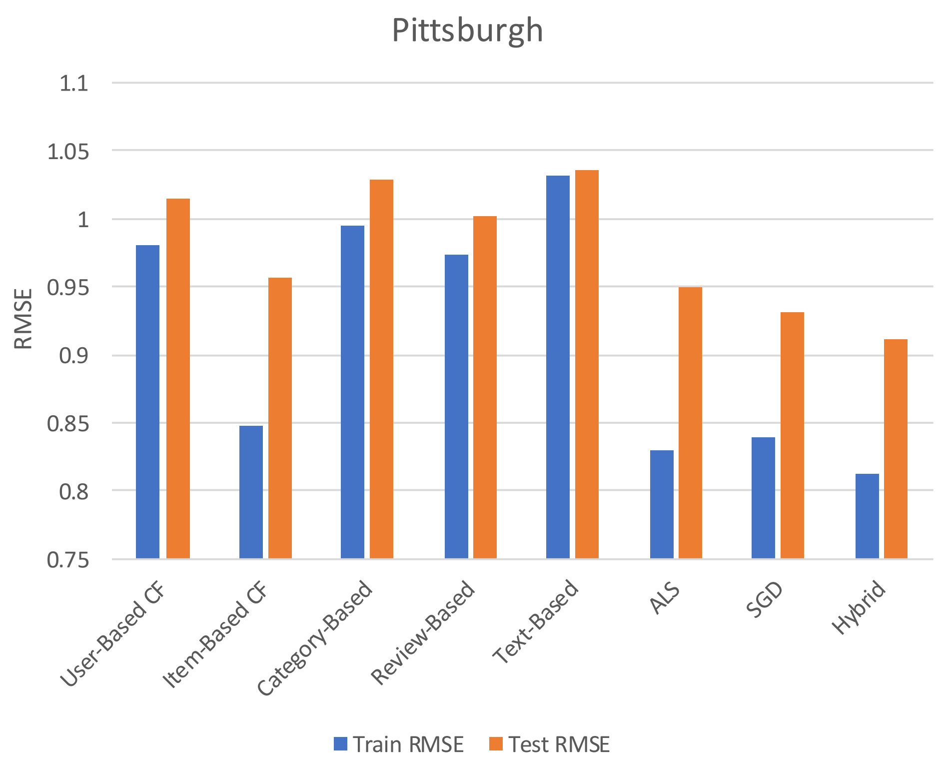
```
Old_string =
{ "attributes": {"BikeParking": "False",
"BusinessAcceptsCreditCards": "True",
"BusinessParking": "{'garage': False, 'street': True, 'validated': False, 'lot': False, 'valet': False}”,
 "NoiseLevel": "average",
"RestaurantsAttire": "casual",
"RestaurantsDelivery": "False",
"RestaurantsGoodForGroups": "True",
"RestaurantsPriceRange2": "2"
},
"categories": "Tours, Breweries, Pizza, Restaurants, Food, Hotels & Travel"
}

New_String = New_string =
businessacceptscreditcards businessparking_street noiselevel_average restaurantsattire_casual
restaurantsgoodforgroups restaurantspricerange2_2 tours breweries pizza restaurants food hotels travel
```

Hybrid-Model Predictions (RECAP)

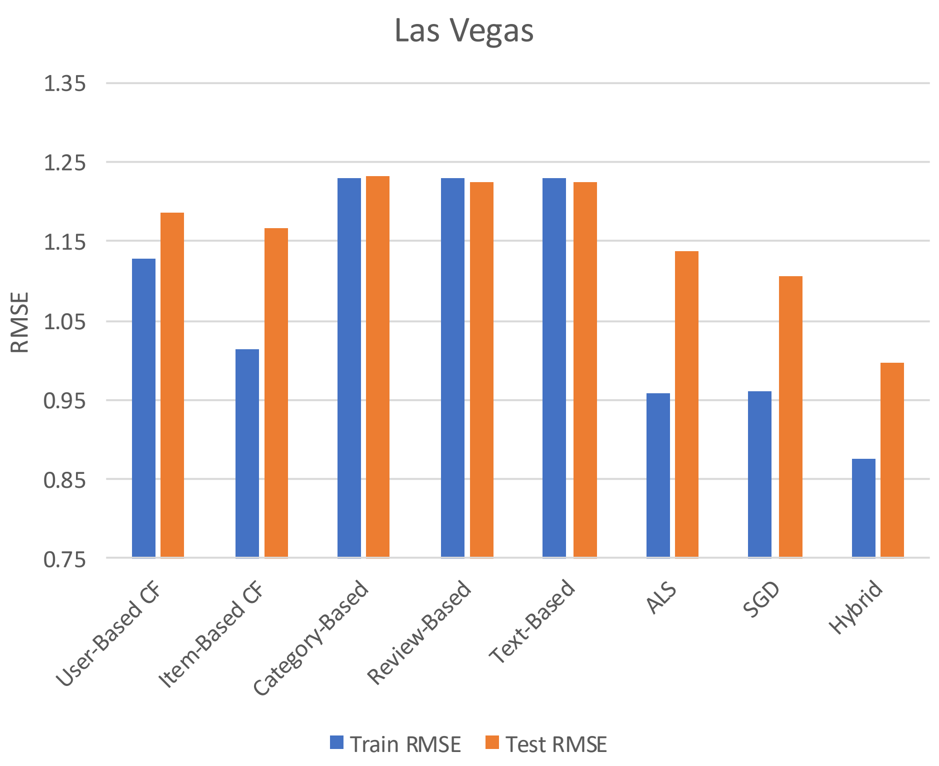


Results

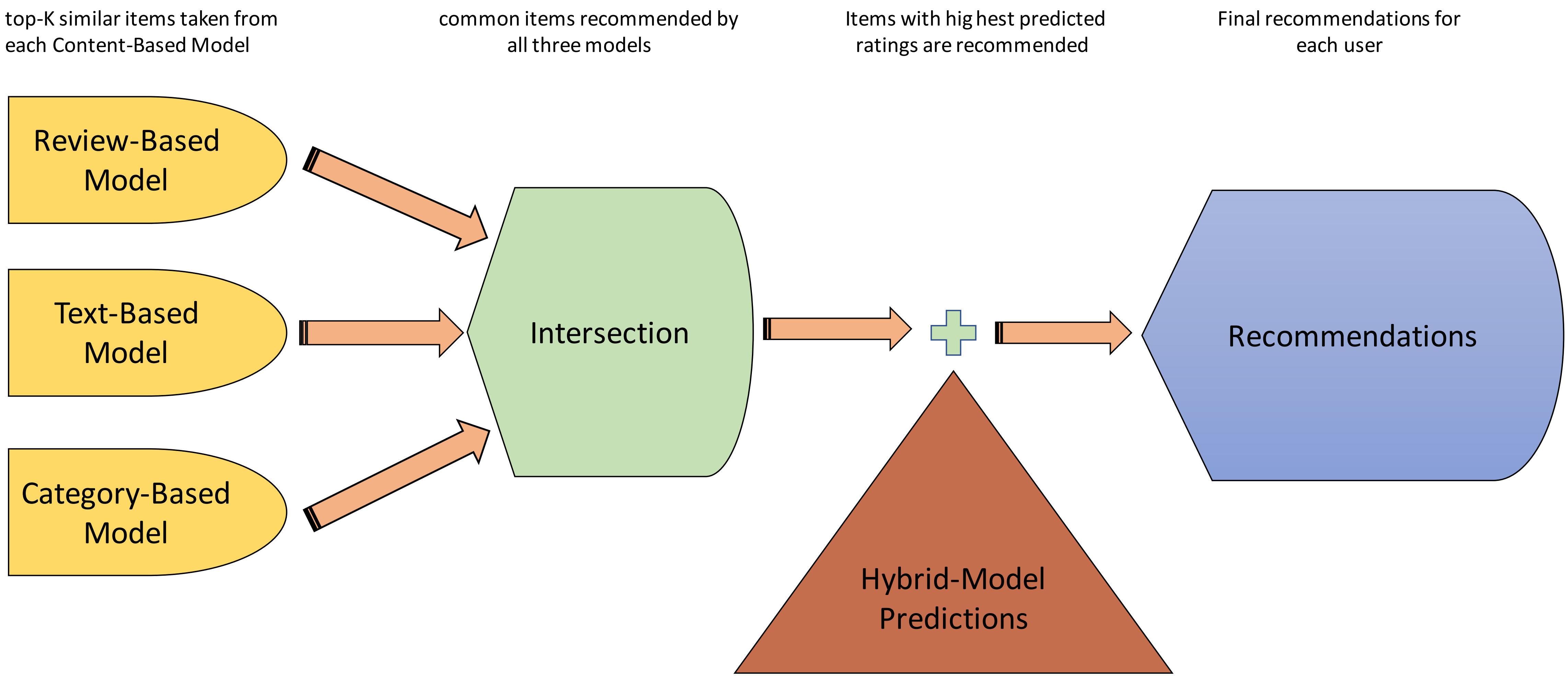
RMSEs for Pittsburgh



RMSEs for Las Vegas



Hybrid-Model Recommendations



Subjective Analysis

User Preferences

- Prefers Italian, pizza, or alcohol 67% of time
- Prefers price range of 1 or 2 (from 1-4) 98% of time
- Prefers restaurants who accept credit cards 100 % of time
- Prefer restaurants which are good for kids 70% of times
- Prefer restaurants with average noise level 65% of times

User Recommendations

- 80% of restaurants recommended served Italian, pizza, or alcohol
- 100% of restaurants recommended had price range of 1 or 2
- 100% of restaurants recommended accepted credit cards
- 65% of restaurants recommended were good for kids
- 70% of restaurants recommended have average noise level

Summary

- We implemented seven different models including 4 collaborative and 3 content based, combining them into a Hybrid Model.
- For each user, we predicted his/her rating for a restaurant as well as recommended them some restaurant that they might like.
- Finally, we analyzed that the recommendations made to each user by hybrid model were closely aligned with their preferences.

Thank You