RECIPE RECOMMENDATIONS

Northeastern University

DS 4420
AVRIL MAURO
HEIDI EREN
DENNIS HO

HOW CAN MACHINE LEARNING PERSONALIZE RECIPE RECOMMENDATIONS BASED ON USER PREFERENCES AND DIETARY PATTERNS?

INTRODUCTION

Many users struggle to find recipes that match their tastes, health needs, and dietary restrictions due to an overwhelming volume of online recipe options. This often leads to **decision fatigue or unhealthy meal choices** driven by convenience. Our motivation is to simplify this experience by delivering personalized, nutritious, and enjoyable recipe recommendations tailored to individual preferences using machine learning.

OBJECTIVE

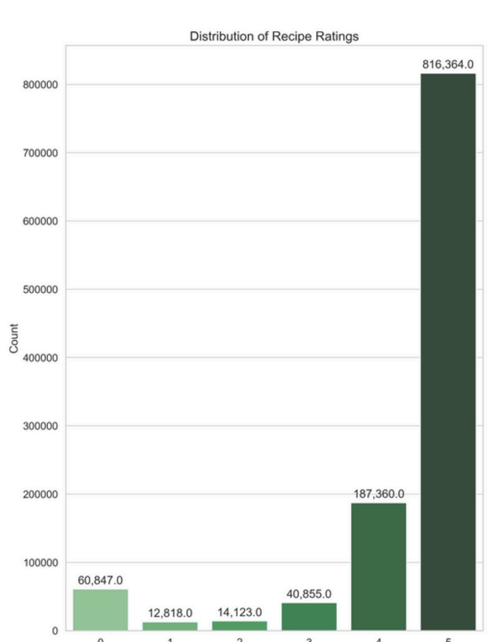
- **Predict user ratings** for unrated recipes
- **Recommend personalized recipes** based on user behavior and recipe features

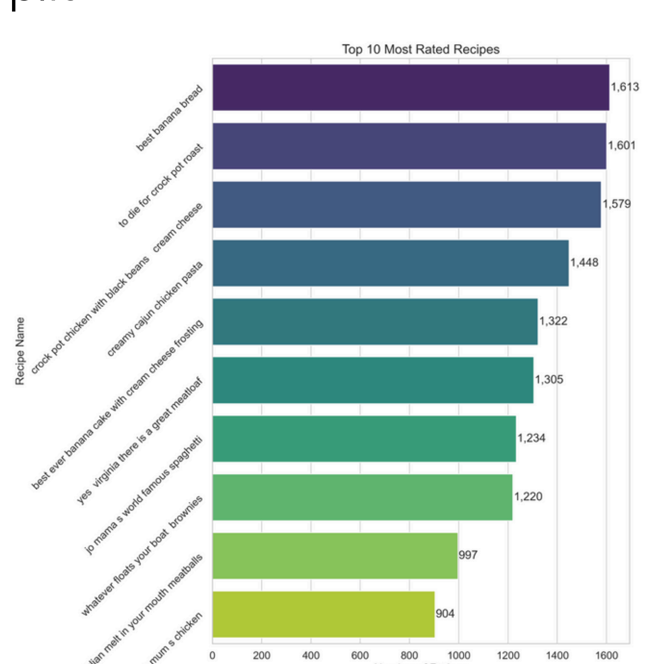
DATA BACKGROUND

Our data was sourced from Kaggle and has 180K+ recipes and 700K+ recipe reviews covering 18 years of user interactions and uploads on Food.com, totaling to ~1.2 M entries of user interaction data. Some features include minutes, n_steps, n_ingredients, tags, and ingredients. Ratings are heavily skewed right with the mean rating being 4.41 out of 5.

We cleaned the data by doing the following:

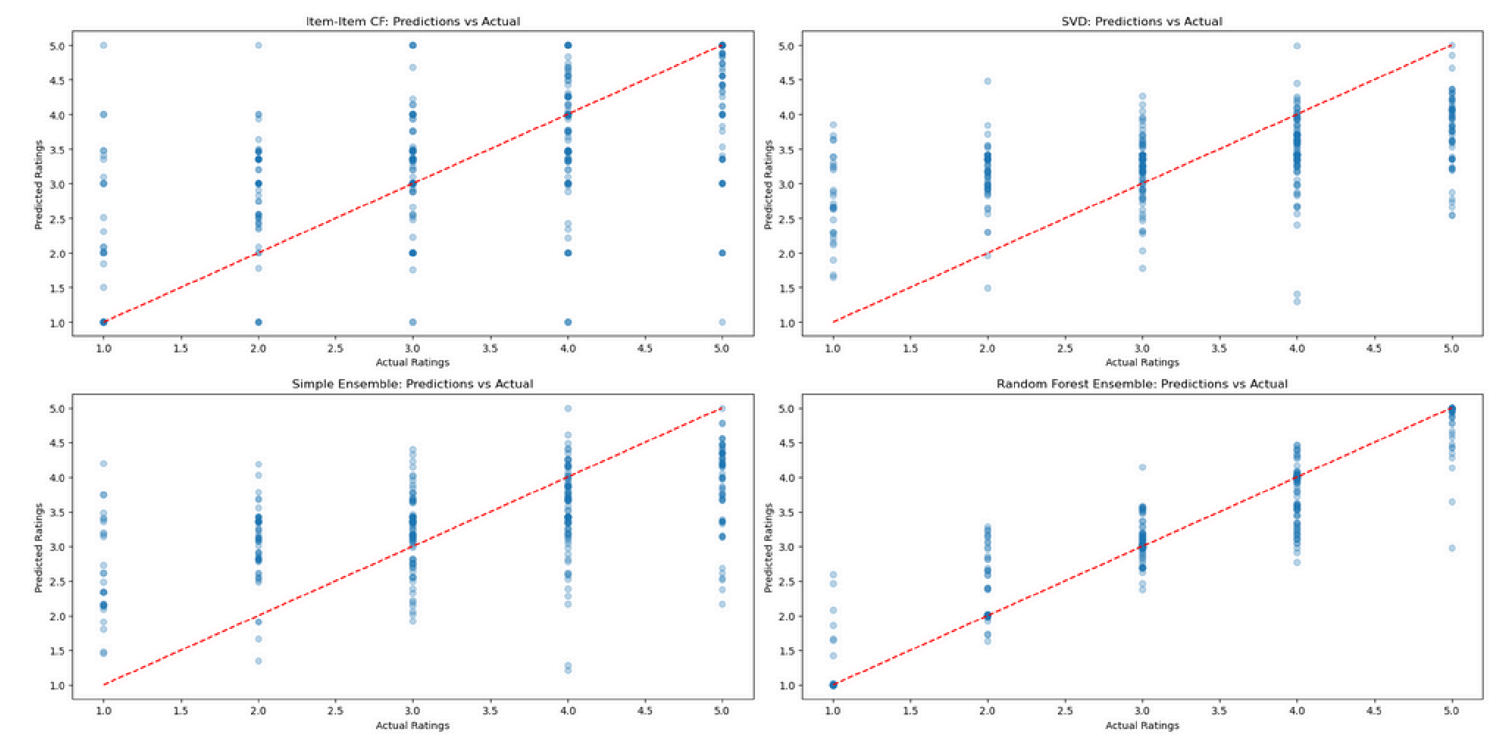
- Unnested and one hot encoded common tags (shown in 25% of the data or 50K recipes)
- Filtered out users with <2 ratings
- Took stratified sample from the last 3 years to balance out ratings for equal representation
- 80-20 Train Test Split





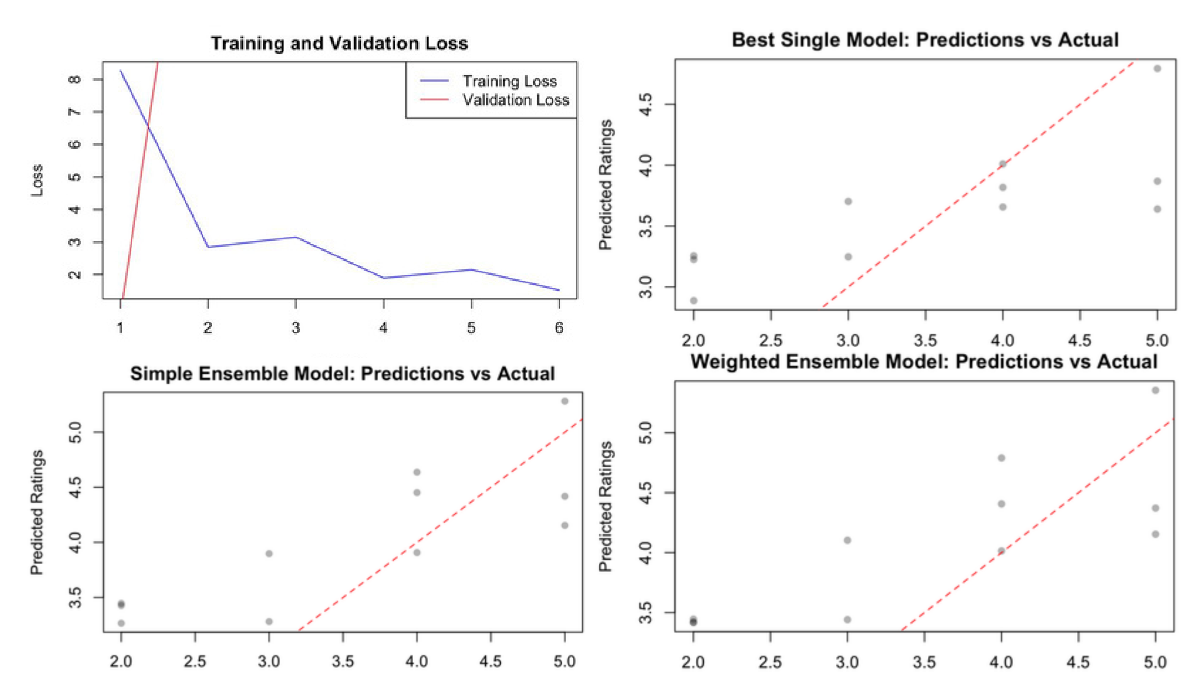
METHODOLOGY: CF

We chose an **Item-Item** over User-User CF because our dataset had more recipes (180K+) than users (25K). We also incorporated a time-decay with a 2-year half-life to prioritize recent ratings, knowing food preferences evolve over time. Our pure Item-Item CF (neighbors=10) struggled with cold start and our users complex preference patterns, and our Hybrid Item-Content model yielded negative R2 values, signaling us to move away from content features. We then tried to blend matrix factorization with **SVD** (25 factors) into our CF, and found significant improvement with **Random Forest Ensemble** with additional features: User Bias (mean rating deviation), Recency ('days ago'), Prediction Disagreement (SVD vs CF), & Other (consistency).



METHODOLOGY: MLP

Our MLP model combines embeds user and recipe IDs, and uses 2 hidden layers to make predictions. We used **Random Grid Search** tests a combination of hyperparameters (dimensions, hidden layers, dropout, learning rates, batch size) to hypertune parameters. We also utilized ensembles; simple ensemble is the average predictions from the top five models, weighted ensemble is the weighted average based on model performance. Our training loss decreased steadily but our validation loss ended up increasing, signaling overfitting.



SINGLE MLP MODEL OUTPERFORMED ENSEMBLE

Param	Val
Embed Dim	32
Hidden 1	128
Dropout 1	0.4
Hidden 2	128
Dropout 2	0.3
Learning Rate	.0005
Batch Size	**64**
Epochs	**10**

RESULTS

METHOD	R2	RMSE	RANK
Item-Item CF	0.1713	1.1072	7
Singular Value Decomposition (SVD)	0.2563	1.0489	6
Simple Ensemble (CF/SVD)	0.2776	1.0338	5
Random Forest Ensemble (CF/SVD)	0.8832	0.4158	**1**
Single Model MLP	0.4741	0.8391	3
Simple Ensemble MLP	0.4317	0.7463	4
Weighted Ensemble MLP	0.4928	0.8241	**2**

- Random Forest Ensemble of CF/SVD performed best (R2=0.88)
- SVD yielded higher performance than CF alone
- Underperformance by MLP showed limitations of neural networks when it comes to sparse data and non-linear relationships

FEATURE ANALYSIS

FEATURE	Bias	Time	Other	Disagreement	SVD	CF
IMPORTANCE	0.945	0.018	0.018	0.009	0.007	0.003

The Random Forest Ensemble approach was successful because it learned non-linear relationships between model outputs and made context-aware decisions on which model to trust per user.

- **User Bias** – showed that individual tendencies are crucial
- **Time** – showed that recency is a factor in rating prediction
- **Disagreement** – showed that model outputs have valuable signals
- **Other** – showed that rating consistency can influence accuracy
- **Base Models** – showed that the models still serve as indicators

CONCLUSION & FUTURE

Our study found that collaborative filtering methods, particularly the Random Forest Ensemble (CF/SVD), outperformed deep learning approaches in predicting recipe ratings. The MLP model's underperformance suggests that sparse datasets with limited metadata are not well-suited for neural networks. These results highlight the strength of hybrid CF models in capturing user behavior patterns for accurate recommendations. User bias had the greatest impact on model accuracy, reinforcing the idea that individual tendencies strongly shape rating behavior. Moving forward, we aim to explore features like seasonality to enhance predictions and uncover deeper user insights.