Sephora Product Recommendation System

Isabella Gonzales, Carine Wong, Libby Amir, Hajera Laique

About the Data

Products Data

About 8,000 products

- Product ID
- Brand name
- Rating
- Highlights (e.g. "cruelty-free, good for dry skin")
- Product category

Reviews Data

About 1,000,000 Reviews

- Author ID
- Product ID
- Review Title
- Review Text
- Author features
  - Skin type
  - Skin tone
  - Eye color

Source: Kaggle, Sephora Products and Skin

Inputs and Outputs

User Inputs

- Product name and brand
- Their features
  - Eye color
  - Skin type
  - Skin tone

Outputs

Table with:

- similar product recommendations
- A predicted rating from reviewers (i.e. "what other people with your features rate this product…")

Project Goal

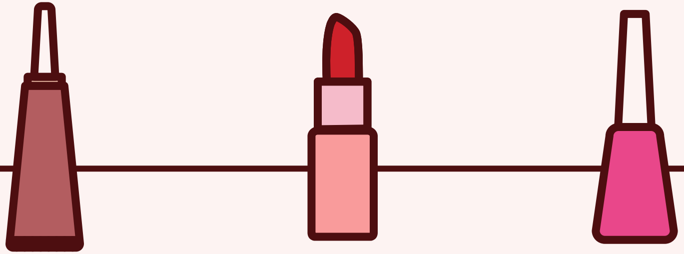
Create a product recommendation system

Help shoppers

- Find products similar to ones they already love
- Have a smooth browsing experience

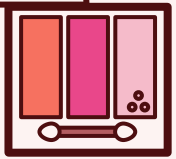
Help brands

- Search engine optimization (SEO)
- Increase revenue

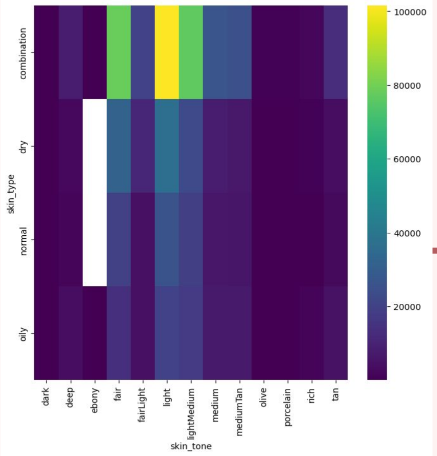


Data Cleaning

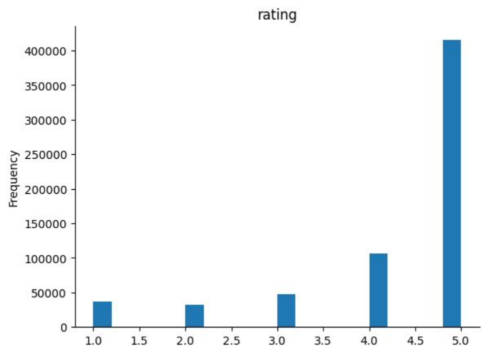
Removing NA values	We removed NA values to avoid unnecessary errors and promote consistency.
Filtering Reviews	We filtered reviews by removing those written by authors who have written less than 10 reviews
Review Text Preprocessing	Tokenization, removing stop words/non-words/emojis/numbers, lowercasing, remove punctuation, stemming
Filtering for product type	We decided on focusing only on makeup and skincare (no fragrances, etc.)
Removing Potential Duplicates	Minis, Travel Size versions, Items with same name, shades listed separately



Exploratory Data Analysis



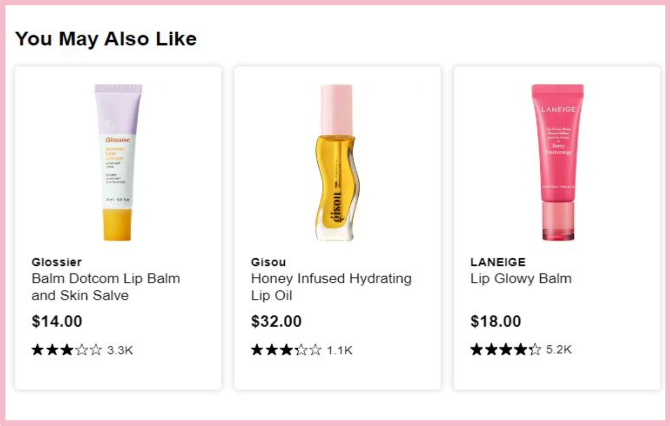
Heatmap comparing skin tones and skin types



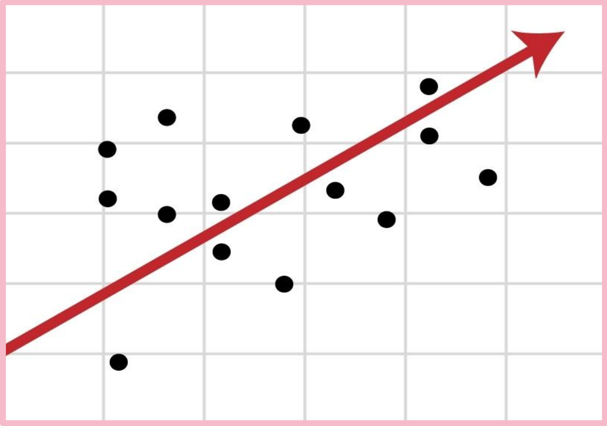
Barplot of the user ratings of products

Word cloud of the highest frequency of words used in the product reviews

Model Selection



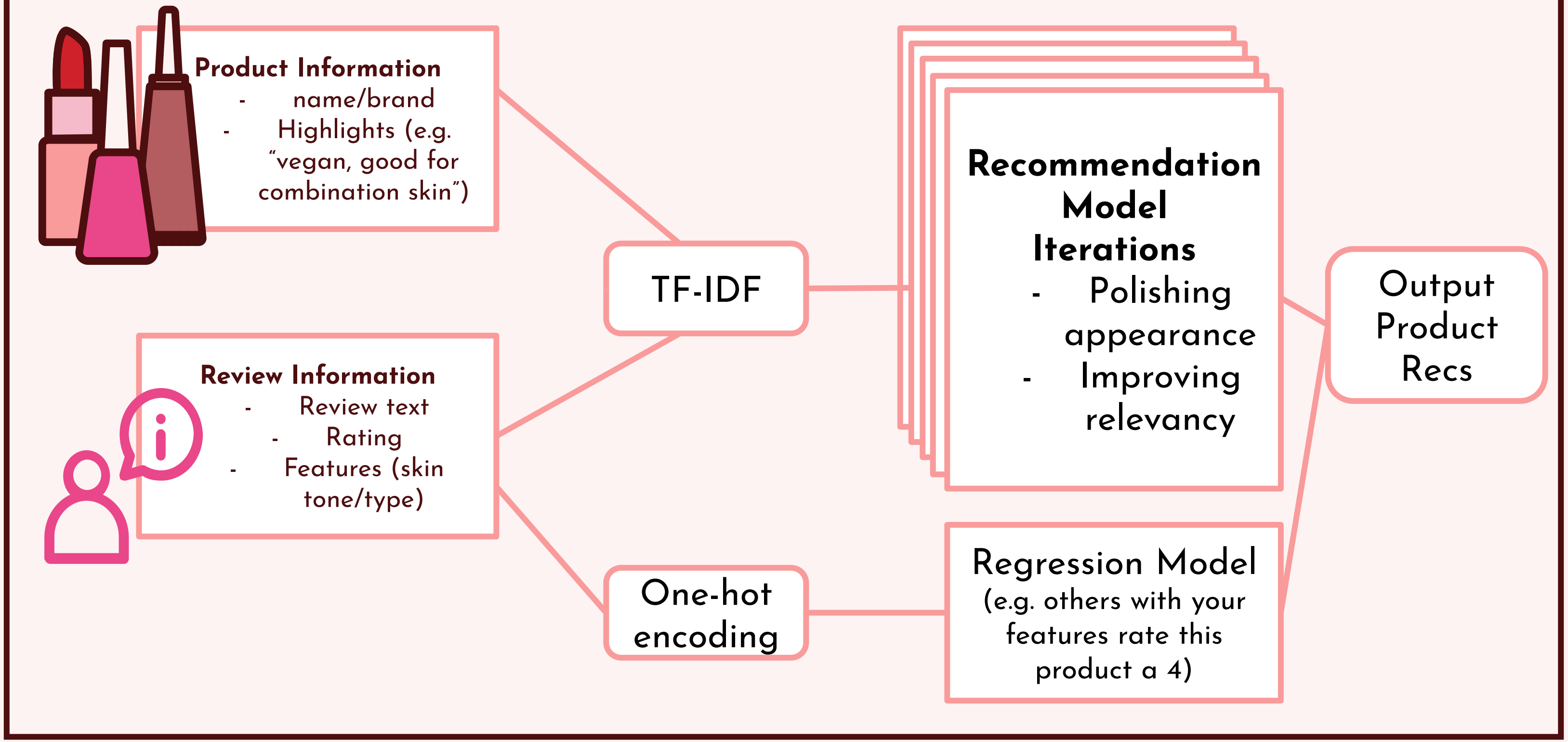
+



Recommendation

Regression

Model Flowchart



Measure used: TF-IDF

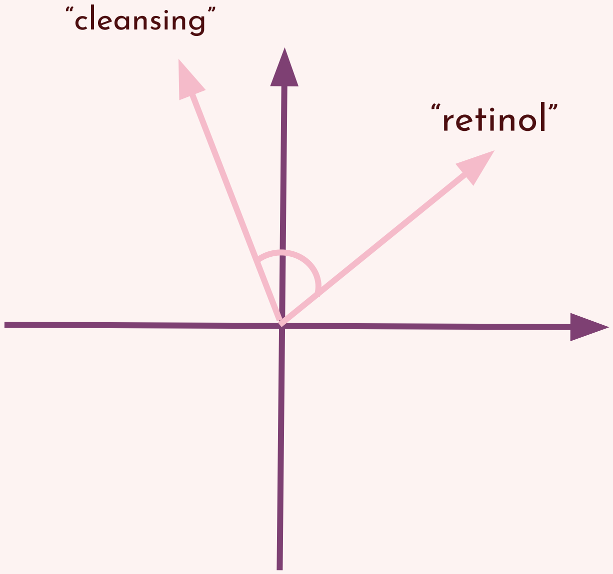
A statistical measure that evaluates how relevant a word is to a document in a collection of documents

Term Frequency (Of a word in a review)	×	Inverse document frequency (In other words, how common a word is in the entire reviews data set)

Higher TF-IDF score = higher relevance of word

Source: https://monkeylearn.com/blog/what-is-tf-idf/

Measure used: Cosine Similarity



Measure of similarity between two vectors

$$= \frac{\text{Dot product of the vectors}}{\text{Product of their lengths}}$$

Larger cosine similarity = larger similarity
Range = [-1, 1]

Improvements we made to our Recommendation System

Our example

Iteration 1 | First Version

```
# when calling for recommendations, acknowledge that spelling and grammar are crucial
get_recommendations("Lip Sleeping Mask Intense Hydration with Vitamin C", cosine_similarity)
```

```
297             Clinique iD Custom-Blend Hydrator Collection
1230                      Hydro Grip Hydrating Makeup Primer
2008                     Hyaluronic Acid 2% + B5 Hydrating Serum
2206                   Mini Superberry Hydrate + Glow Dream Mask
779                          Vanish Flash Highlighting Stick
840                     Mini Limitless Lash Lengthening Mascara
1014                            Tinted Face Oil Comfy Skin Tint
1244                      Sunshine Vitamin C + Squalane Face Oil
2205     Superberry Hydrate + Glow Dream Night Mask wit...
1716          Synchro Skin Self-Refreshing Foundation SPF 30
Name: product_name, dtype: object
```

Takeaways:

- Things accounted for: 'highlights', (characteristics, e.g. hydrating, good for sensitive skin)
- Suggestions aren't in the same category
- Next step: filtering for product type

Iteration 2

Category-Specific Recommendations

```
# when calling for recommendations2, we can compare the results with recommendations
get_recommendations2("Lip Sleeping Mask Intense Hydration with Vitamin C", cosine_similarity)
```

```
1101                                       Lip Glowy Balm
1956                                      The Kissu Lip Mask
160                              Squalane+ Rose Vegan Lip Balm
616                        Sugar Recovery Lip Mask Advanced Therapy
617                        Sugar Mint Rush Freshening Lip Treatment
1394                            Pout Preserve Peptide Lip Treatment
1105                                      Lip Treatment Balm
607                           Sugar Lip Balm Hydrating Treatment
596                 Sugar Advanced Lip Balm Intense Hydration Trea...
1667                                  Clean Lip Balm & Scrub
Name: product_name, dtype: object
```

Takeaways:

- Improvements made: recommendations are in same product type category
- Much better!
- Next step: adding brand names, as “clean lip balm” doesn’t tell me precisely what product it is referring to

Iteration 3 | Display Brand Name

```
# when calling for recommendations3, we can now see the brand of every product
get_recommendations3("Lip Sleeping Mask Intense Hydration with Vitamin C", "LANEIGE", cosine_similarity)
```

```
Recommendations for Lip Sleeping Mask Intense Hydration with Vitamin C:
```

	product_name	brand_name
1101	Lip Glowy Balm	LANEIGE
1956	The Kissu Lip Mask	Tatcha
160	Squalane+ Rose Vegan Lip Balm	Biossance
616	Sugar Recovery Lip Mask Advanced Therapy	fresh
617	Sugar Mint Rush Freshening Lip Treatment	fresh
1394	Pout Preserve Peptide Lip Treatment	OLEHENRIKSEN
1105	Lip Treatment Balm	LANEIGE
607	Sugar Lip Balm Hydrating Treatment	fresh
596	Sugar Advanced Lip Balm Intense Hydration Trea...	fresh
1667	Clean Lip Balm & Scrub	SEPHORA COLLECTION

Takeaways:
- Prettier formatting
- Can now see brand name
- New argument added (brand name)
- Next step: the suggestions themselves are still the same; can we improve recommendations by considering review text?

Iteration 4 | Factor in Reviews

```
# when calling for recommendations4, we get recommendations based on highlights and their reviews
get_recommendations4("Lip Sleeping Mask Intense Hydration with Vitamin C", "LANEIGE")
```

```
Recommendations for Lip Sleeping Mask Intense Hydration with Vitamin C:
```

	product_name	brand_name
486	Lip Glowy Balm	LANEIGE
489	Lip Treatment Balm	LANEIGE
814	The Kissu Lip Mask	Tatcha
295	Sugar Recovery Lip Mask Advanced Therapy	fresh
43	Squalane+ Rose Vegan Lip Balm	Biossance
203	Lippe Balm	Drunk Elephant
719	Brazilian Kiss Cupuaçu Lip Butter	Sol de Janeiro
666	Clean Lip Balm & Scrub	SEPHORA COLLECTION
234	Honey Butter Beeswax Lip Balm	Farmacy
18	Willow & Sweet Agave Plumping Lip Mask	alpyn beauty

Takeaways:

- Used TF-IDF to process review text with highlights
- Some recommendations are the same
- Improved reviews based on our judgement
- Next steps: can we improve relevance of recommendations based on user features?

Iteration 5 | Regression

```
get_recommendations5('tan', 'brown', 'dry', "Lip Sleeping Mask Intense Hydration with Vitamin C", "LANEIGE")
```

```
Recommendations for Lip Sleeping Mask Intense Hydration with Vitamin C:
```

	product_name	brand_name	predicted_rating
486	Lip Glowy Balm	LANEIGE	4.347574
489	Lip Treatment Balm	LANEIGE	4.148530
814	The Kissu Lip Mask	Tatcha	3.741183
295	Sugar Recovery Lip Mask Advanced Therapy	fresh	4.691435
43	Squalane+ Rose Vegan Lip Balm	Biossance	3.829175
203	Lippe Balm	Drunk Elephant	3.716643
719	Brazilian Kiss Cupuaçu Lip Butter	Sol de Janeiro	4.096331
666	Clean Lip Balm & Scrub	SEPHORA COLLECTION	3.216920
234	Honey Butter Beeswax Lip Balm	Farmacy	4.218792
18	Willow & Sweet Agave Plumping Lip Mask	alpyn beauty	4.516091

Takeaways:

- Added a column of what other people with your features rate the product
- New arguments: features (skin tone, eye color, skin type)

Regression Accuracy

We created a regression function that is called upon within our get_recommendation() function

```
Mean Squared Error: 0.9294080006092733
```

Not amazing, but nor is human rating accuracy

Example Regression Usage

Serum containing a large amount of of oil for hydration that is very suitable for dry skin

```
# testing the model with specific values
new_data = {
    'skin_tone': ['fair'],
    'eye_color': ['brown'],
    'skin_type': ['dry'],
    'product_name': ['GENIUS Liquid Collagen Serum']
}

# creating df
new_data_df = pd.DataFrame(new_data)
new_data_encoded = encoder.transform(new_data_df)

# predict the rating for the new data
predicted_rating = model.predict(new_data_encoded)

print("Predicted Rating:", predicted_rating)
```

```
Predicted Rating: [4.50844373]
```

GENIUS Liquid Collagen Serum	6018	Algenist	67870	4.0259

Overall rating

Example Regression Comparison

For a product known to leave a slight white cast on darker skin:
Fair skin vs deep skin

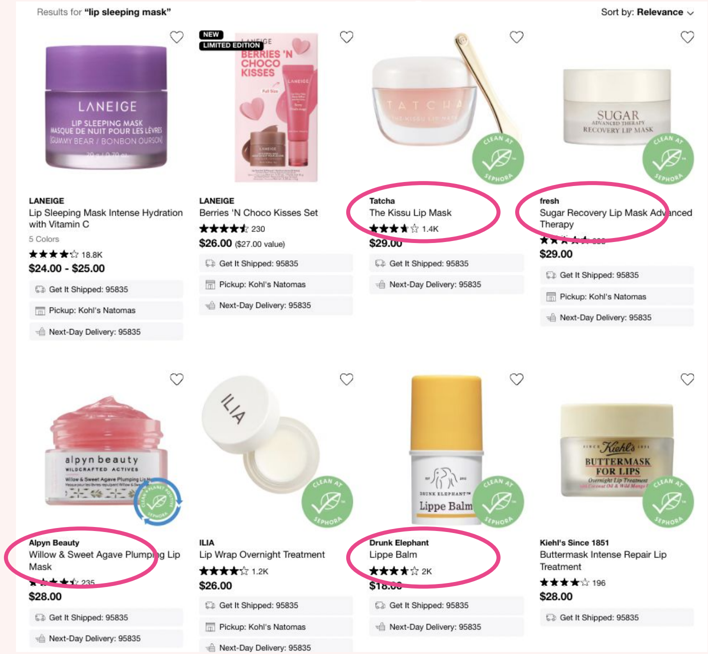
```
# testing the model with specific values
new_data = {
    'skin_tone': ['fair'],
    'eye_color': ['brown'],
    'skin_type': ['dry'],
    'product_name': ['Cicapair Tiger Grass Color Correcting Treatment SPF 30']
}

# creating df
new_data_df = pd.DataFrame(new_data)
new_data_encoded = encoder.transform(new_data_df)

# predict the rating for the new data
predicted_rating = model.predict(new_data_encoded)

print("Predicted Rating:", predicted_rating)
```

```
Predicted Rating: [3.94241924]
```

```
# testing the model with specific values
new_data = {
    'skin_tone': ['deep'],
    'eye_color': ['brown'],
    'skin_type': ['dry'],
    'product_name': ['Cicapair Tiger Grass Color Correcting Treatment SPF 30']
}

# creating df
new_data_df = pd.DataFrame(new_data)
new_data_encoded = encoder.transform(new_data_df)

# predict the rating for the new data
predicted_rating = model.predict(new_data_encoded)

print("Predicted Rating:", predicted_rating)
```

```
Predicted Rating: [3.87342615]
```

Comparison to Sephora Results



Potential Biases:

- Incomplete data set (new product releases, limited edition drops)
- Suggestion results could already be powered by AI/browsing history

Other Product Examples

Recommendations for Retinol Anti-Aging Serum:

	product_name	brand_name	predicted_rating
515	Retinol Youth Renewal Serum	Murad	4.667177
205	A-Passioni Retinol Cream	Drunk Elephant	4.092556
584	CLINICAL 1% Retinol Treatment	Paula's Choice	4.674557
677	Retinol Reform Treatment Serum	Shani Darden Skin Care	4.288439
242	1% Vitamin A Retinol Serum	Farmacy	4.737337
271	FAB Skin Lab Retinol Serum 0.25% Pure Concentrate	First Aid Beauty	3.681529
598	Retinol Face Stick	Peace Out	4.570142
431	Micro-Dose Anti-Aging Retinol Serum with Ceram...	Kiehl's Since 1851	4.451425
763	A+ High-Dose Retinol Serum	Sunday Riley	3.970229
406	Argan Beta Retinol Pink Algae Serum	Josie Maran	4.307496

Other Product Examples

```
Recommendations for Soy Hydrating Gentle Face Cleanser:
```

	product_name	brand_name	predicted_rating
394	Confidence in a Cleanser Hydrating Facial Clea...	IT Cosmetics	4.223640
40	Squalane + Amino Aloe Gentle Pore-Minimizing C...	Biossance	4.146985
908	Superfood Antioxidant Cleanser	Youth To The People	4.165284
846	Fulvic Acid Brightening Cleanser	The INKEY List	4.728240
847	Mini Fulvic Acid Brightening Cleanser	The INKEY List	4.824334
315	Blueberry Bounce Gentle Cleanser	Glow Recipe	3.887933
377	Keep It Clean Hydrating Gel Cleanser with Cera...	iNNBEAUTY PROJECT	4.568328
77	Vinoclean Gentle Foam Cleanser	Caudalie	4.333074
803	The Rice Wash Skin-Softening Cleanser	Tatcha	4.344725
903	Yo Glow AHA & BHA Facial Enzyme Scrub	Wishful	4.042546

```
get_recommendations5('tan', 'brown', 'dry', "Soy Hydrating Gentle Face Cleanser", "fresh")
```

If we were to continue from here...

We could add price customization

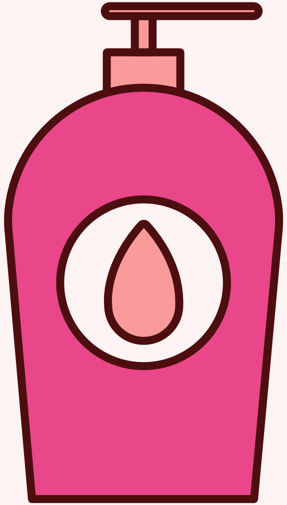
By including cost data, and creating a new parameter involving a customer's budget.

We could change product ranking

We could factor in the predicted product rating into the ranking of recommendations

And create a user Interface

But none of us are stellar at JavaScript (yet!)



Conclusion & Possible Implications

Recommendation Algorithms are Complex

Machine learning applications for recommendation systems tend to involve several metrics and models, and are particularly difficult to evaluate

Not Every Product is for Every Person

Our model reaffirmed the notion that recommended products differ from person-to-person, as we saw predicted ratings are dependent on personal features

More Personalized Recommendations

Websites like Sephora usually recommend products based on popularity, price, and category, but not personal attributes like our function

Customer Segmented Marketing

When brands release a new product, they can use our algorithm to anticipate which segment of the market is most likely to get use and satisfaction from their product

THANKS!

DO YOU HAVE ANY QUESTIONS?