**DSO 560 Group Project Part II**

**Due 11:59pm PST, Monday May 11th**

All available data will be uploaded to the **threadtogether** PostgreSQL database, available at

Host: **threadtogether.ychennay.com**
User**: dso560_student**
Password: **(in class)**
Port: **5432**

**If your group would like to use the database to create temporary tables, please message me and I will create a group-specific user/password for your team to log in as.**

**Deliverables:**

- **A notebook** that allows a user to enter inputs (defined below) – either product IDs or product descriptions and details, and returns recommended outfits.
- **Submit this code to the same Github repository** (with instructor ychennay added as collaborator containing your group's product attribution tool/app code base) you uploaded for Part I.

**Database Tables:**

- **outfit_combinations_view:** a view lists outfits curated by the subject matter expert (SME) stylist:
  - **outfit_id:** identifies the outfit – a combination of product IDs
  - **product_id:** the product ID of the piece of clothing
  - **outfit_item_type**: the part of the outfit the recommendation is for

The view is also available as a CSV file here: **https://dso-560-nlp-text-analytics.s3.amazonaws.com/outfit_combinations.csv**

**Objectives:**

**Outfit recommendations:** Based upon the **outfit_combinations** view, build a model/algorithm that **will accept 1 to 3 the following inputs for outfit type**:

- a shoe
- a bottom
- a top
- an accessory
- a one-piece

And **return a recommendation for an outfit for a "similar" aesthetic or fashion based upon the outfit collections generated by the subject matter expert (SME) stylist.**

The inputs can be either

A. **product IDs**

B. **brand, brand category, details, and description (**similar to the input for Part I).

**Example A**

| |
|---|
| **INPUT (Product IDs – what I pass in to the model):**<br><br>• shoe: **Penelope Mid Cap Toe Pump** (01DMBRYVA2ZFDYRYY5TRQZJTBD)<br>• bottom: **Slim Knit Skirt** (01DMBRYVA2P5H24WK0HTK4R0A1) |
| **OUTPUT (Recommended Outfit Combination – what is returned from the model (highlighted yellow are its recommendations for the outfit):**<br><br>• shoe: **Penelope Mid Cap Toe Pump** (01DMBRYVA2ZFDYRYY5TRQZJTBD)<br>• bottom: **Slim Knit Skirt** (01DMBRYVA2P5H24WK0HTK4R0A1)<br>• accessory: **medium margaux leather satchel** (01DMBRYVA2S5T9W793F4CY41HE)<br>• top: **Rib Mock Neck Tank (**01DMBRYVA2PEPWFTT7RMP5AA1T**)** |

**Example B**

| |
|---|
| **INPUT (Product Descriptions):**<br><br>• bottom: ***DESCRIPTION***: ***slim fitting, straight leg pant with a center back zipper and slightly cropped leg – BRAND: Reformation*** |
| **OUTPUT (Recommended Outfit Combination):**<br><br>• bottom: **Marlon Pant** (01DPKMH0D252JKMAA27MFCT5GM)<br>• shoe: **Doey Suede Ankle Boots** (01DTATDENPZ2G048Q6YTM51C91)<br>• accessory: **Cassi Belt Bag** (01DPEHS0XH9PDD1GH5ZE4P43A2) |

The model should give reasonable recommendations for similar products. For instance, if I provide in as input the following product description:

| |
|---|
| *Sexy silky, a-line mini skirt zipper Benson skirt* |

The model should be able to identify this as a close match against **Product ID 01DPKMGJ33SDFXM7XHGPQJWQ12**, the **Benson skirt**, which has an actual product description of

| |
|---|
| *Sexy silky. This is an a-line mini skirt with a center back zipper. The Benson pairs well with the Hailee Top.* |

and recommend this product ID as the outfit's **bottom**, and from there, return recommendations for a **top**, or an **accessory**, or **shoe** (at least two recommendations). *Note: If you match a product that already exists in the **outfit_combinations**, you can simply return one of its outfit recommendations.*

My test queries will be similar to these – I will pick a few products, change up their wording a bit, use some synonyms, use similar words, but expect that your model should be able to find a similar product.

| **Scoring Rubric** | **Points Available** | | | |
|---|---|---|---|---|
| | **1pt** | **2pts** | **3pts** | **4pts** |
| **The model will be tested by running 4 test queries, and returns expected matches (like the example above).** | No results are returned | Basic results are returned but they do actually make sense and are not similar in any way to the product query. | 3/4 test queries submitted with a free-form product description are returned with products that are actually "similar". | 4/4 test queries submitted with a free-form product description are returned with products that are actually "similar". |
| **The model handles both product IDs and also production descriptions + brand names (free-form text) as inputs** | No (0 pts) | | Yes (2 pts) | |
| **Code is documented with thought process easily visible to instructor/client to review** | No (0 pts) | | Yes (2 pts) | |
| **Code is published to a Github repository with team members added as collaborators** | No (0 pt) | | Yes (1 pt) | |
| **All group members have submitted 360 feedback reviews by deadline** | No (0 pt) | | Yes (1 pt) | |