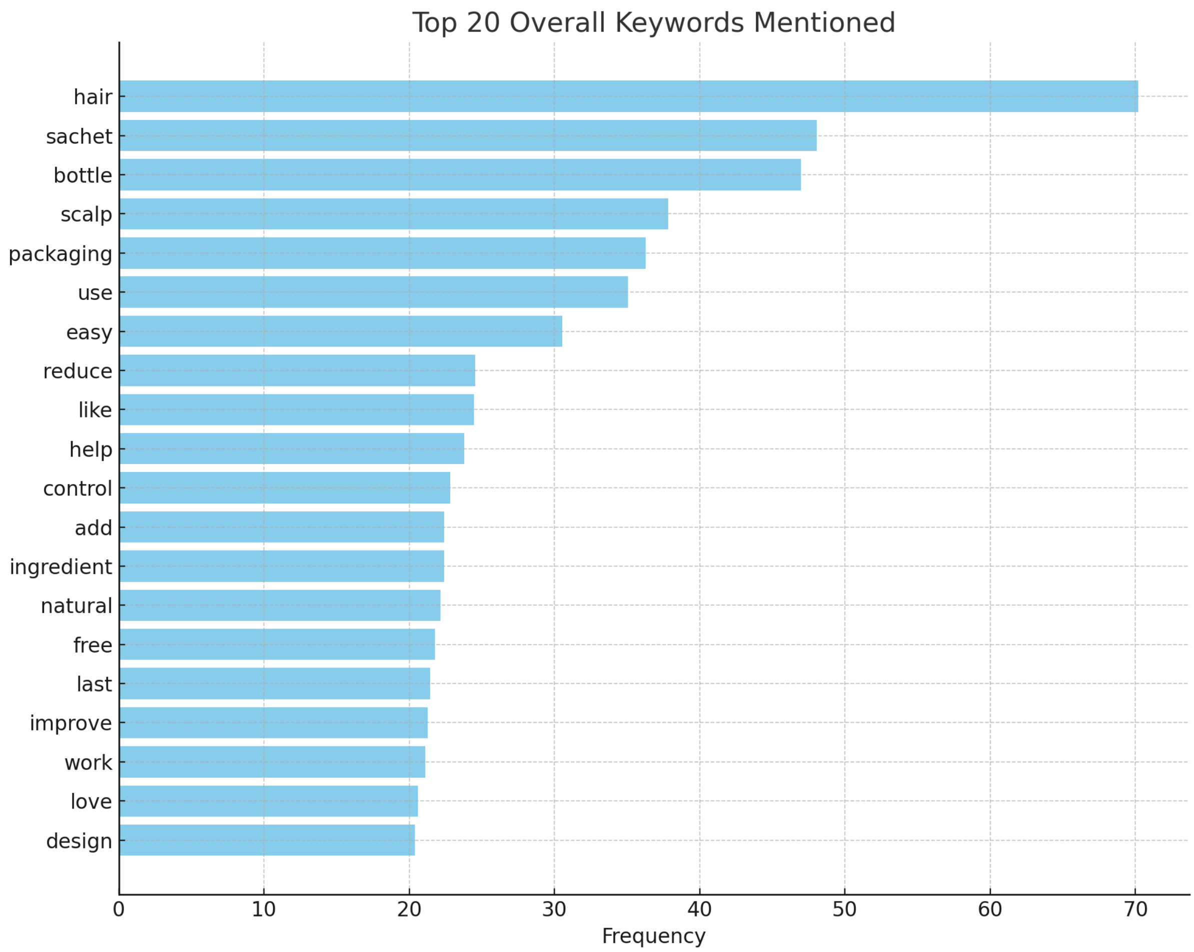
Section 15: Keyword & Phrase Extraction

Keyword Tables

- **Overall Top Keywords** (top 20)
- **Segment Top Keywords** (top 10 per demographic slice)



Insights from the Top 20 Keyword Frequencies Chart:

1. **Dominance of 'Hair'-Related Concerns**:
   - The term *"hair"* stands out as the most frequently mentioned keyword, indicating that consumers strongly associate the product with hair care benefits and outcomes.
2. **Packaging Formats are Key Considerations**:

- Keywords like *"sachet"* and *"bottle"* are highly ranked, highlighting that packaging format significantly influences user preferences and perceptions.

3. **Scalp and Ingredient-Focused Benefits**:

    - Words like *"scalp," "control," "natural," "ingredient,"* and *"reduce"* suggest that many consumers are focused on functional benefits such as dandruff control, scalp health, and natural ingredients.

4. **Ease of Use and Design Matter**:

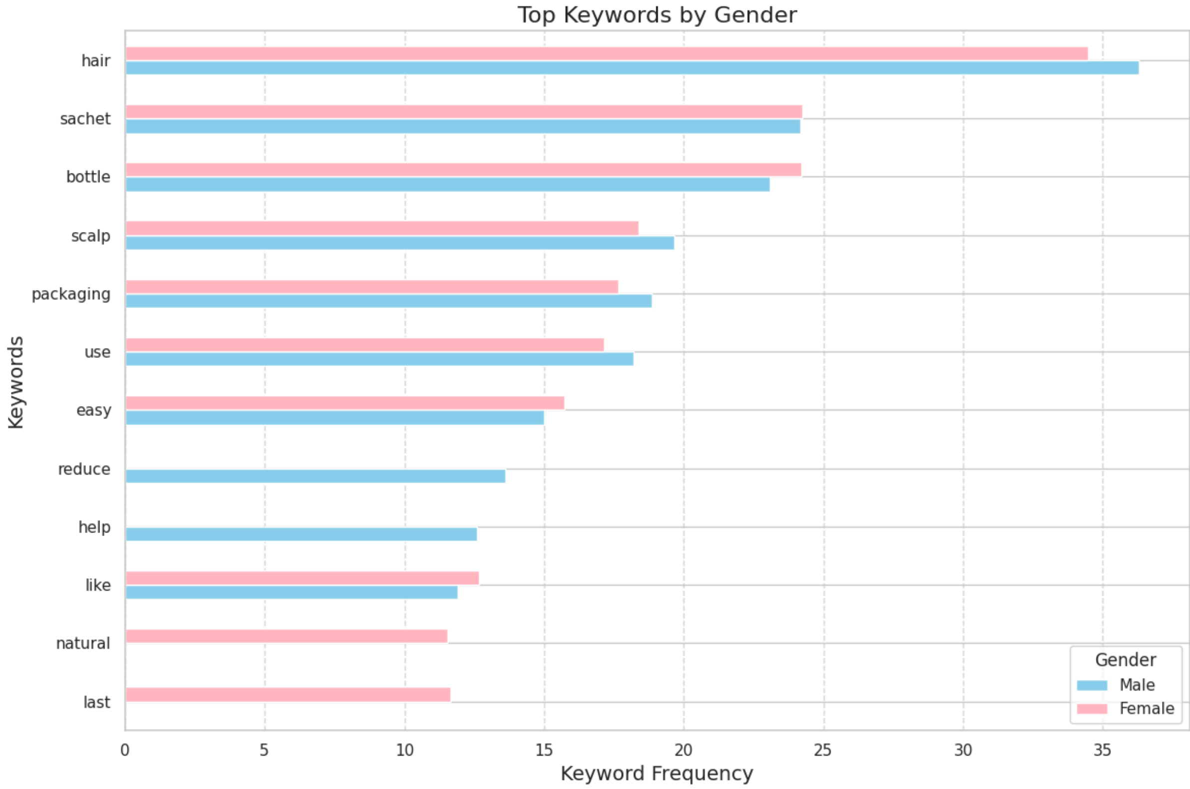
    - Terms such as *"easy," "use," "design,"* and *"packaging"* imply that usability and aesthetic aspects of the product and packaging are important to consumers.

5. **Positive Sentiment and Efficacy**:

    - Keywords like *"love," "help," "work,"* and *"improve"* reflect overall satisfaction and the perceived effectiveness of the product.

Key Phrase Tables

- **Overall Key Phrases** (top 10)



Insights from Key Phrase Diagram:

1. High Commonality Between Genders

- **"Hair"**, **"sachet"**, **"bottle"**, **"scalp"**, **"packaging"**, **"use"**, **"easy"**, and **"like"** are **common top keywords** for both genders, highlighting shared perceptions and priorities.

2. Top Priority: Hair

- Both **male (36.30%)** and **female (34.48%)** consumers most frequently mentioned "**hair**", clearly signaling that product communication and formulation must strongly resonate with hair care benefits.

3. Format Preference: Sachet vs Bottle

- **Sachet** and **bottle** keywords score nearly equally for both genders, though sachets have a **slightly higher association** with **female consumers**, which could imply **convenience preference** or **trial behavior**.

4. Functional Benefits Matter

- Keywords like "**use**", "**easy**", "**reduce**", and "**help**" indicate a focus on **functional value** (ease of use, helpfulness, problem-solving).

5. Emotional and Natural Aspects for Women

- Women associate terms like "**natural**" and "**last**" more frequently, suggesting **durability** and **natural ingredients** play a **greater role in female preference**.

6. Male Consumers Show Functional Orientation

- Males highlight "**reduce**" and "**help**", implying a stronger **utility-based outlook** (e.g., reduce dandruff, help scalp health).

2. **Analysis Notebook**
    - Documented code for TF-IDF and KeyBERT extraction
    - Parameter settings (ngram ranges, stop-word list)

https://colab.research.google.com/drive/18H6BxSEtdJiG19SZo6dnLsH2uL0OXTDE#scrollTo=PUIourInjttI

Description

Section 15: Keyword & Phrase Extraction

Scope: Extract meaningful words and short phrases from open-ended responses to support summarization, topic modeling, and segment-specific insights.

Objectives

- Extract the most informative keywords and phrases from Q19–Q21, Q25–Q26, Q28–Q29, Q33, and Q40–Q41.
- Enable targeted summarization and segmentation-based language insights.
- Surface unique terms used by demographic cohorts (Age, Gender, NCCS).

Analysis Tasks

Task	Details	Method
1. Text Collection	- Combine open-ended responses from questions Q19–Q21, Q25–Q26, Q28–Q29, Q33, Q40–Q41. - Include respondent metadata: Gender, Age, NCCS.	`python import pandas as pd df = pd.read_csv("open_ended_data.csv") # Filter and combine relevant questions columns_to_use = ['Q19', 'Q20', 'Q21', 'Q25', 'Q26', 'Q28', 'Q29', 'Q33', 'Q40', 'Q41'] df['Combined_Text'] = df[columns_to_use].fillna('').agg(' '.join, axis=1) df_text = df[['Respondent_ID', 'Combined_Text', 'Gender', 'Age', 'NCCS']]`
2. Preprocessing	- Lowercase, remove punctuation, stop words. - Optionally lemmatize.	`python import spacy nlp = spacy.load("en_core_web_sm", disable=["parser", "ner"]) def preprocess(text): doc = nlp(text.lower()) tokens = [token.lemma_ for token in doc if not token.is_stop and token.is_alpha] return " ".join(tokens) df_text["Clean_Text"] = df_text["Combined_Text"].apply(preprocess)`

3. TF-IDF Extraction	- Calculate TF-IDF scores for unigrams and bigrams. - Identify top keywords globally and by segment.	`python from sklearn.feature_extraction.text import TfidfVectorizer tfidf = TfidfVectorizer(ngram_range=(1, 2), max_features=5000) tfidf_matrix = tfidf.fit_transform(df_text["Clean_Text"]) tfidf_df = pd.DataFrame(tfidf_matrix.toarray(), columns=tfidf.get_feature_names_out()) top_keywords = tfidf_df.mean().sort_values(ascending=False).head(20)`
4. KeyBERT Phrase Extraction	- Use a Sentence-BERT model via KeyBERT. - Extract top key phrases (2–3 words) by segment.	`python from keybert import KeyBERT kw_model = KeyBERT(model="all-MiniLM-L6-v2") def extract_phrases(text): return kw_model.extract_keywords(text, keyphrase_ngram_range=(2, 3), stop_words="english", top_n=5) df_text["Key_Phrases"] = df_text["Clean_Text"].apply(lambda x: extract_phrases(x))`
5. Segment Comparison	- Slice by demographics: Age buckets, Gender, NCCS. - Compare TF-IDF ranks and KeyBERT scores for uniqueness.	`python def get_top_keywords_by_segment(df, segment): segment_text = df.groupby(segment)["Clean_Text"].apply(" ".join) tfidf_seg = tfidf.fit_transform(segment_text) tfidf_seg_df = pd.DataFrame(tfidf_seg.toarray(), index=segment_text.index, columns=tfidf.get_feature_names_out()) return tfidf_seg_df.T.apply(lambda x: x.sort_values(ascending=False).head(10)).stack().reset_index(name="TFIDF_Score") keywords_by_gender = get_top_keywords_by_segment(df_text, "Gender")`

6. Output Tables	- Export keyword and phrase tables for overall and segment-specific analyses.	`python keywords_by_gender.to_csv("segment_keywords_gender.csv", index=False) df_text[['Respondent_ID', 'Key_Phrases']].explode("Key_Phrases").to_csv("key_phrases_all.csv", index=False)`

Deliverables

Keyword Tables

- **Overall Top Keywords (Top 20):** Sorted by average TF-IDF score
- **Segment Keywords (Top 10):**
  File: segment_keywords_<segment>.csv
  Columns: `Segment`, `Keyword`, `TFIDF_Score`, `Rank`

Key Phrase Tables

- **Overall Phrases:**
  Top 10 from the full dataset using KeyBERT
- **Segment-Specific Phrases:**
  File: `key_phrases_all.csv`
  Columns: `Respondent_ID`, `Phrase`, `Relevance`

Analysis Notebook

- Documented Python code:
  - Preprocessing
  - TF-IDF extraction
  - KeyBERT phrase extraction
  - Segment comparisons
- Parameters: `ngram_range=(1,2)`, `top_n=5`, custom stopword list via spaCy

Summary Report

- Highlight most unique or distinctive words and phrases by:
    - Gender
    - Age Bucket (<30 vs. ≥30)
    - NCCS (A vs. B/C)
- Identify potential tags for visual dashboards and thematic filters