Deep Learning

Milih Skincare

A BERT-Based Multi-Label Skin Type Classifier and Ingredient-Safe Skincare Recommender

Ofelia Marvela Amin

Faradilla Alfaira Chandra

Warm greetings to all present. I am excited to introduce our research MilihSkincare.

Background of the Study

Skincare products contain rich textual information such as descriptions and ingredient lists. Users often face difficulty selecting products suitable for their skin type. Natural Language Processing (NLP) enables automatic understanding of product descriptions. Our study focuses on evaluating the skincare landscape, consumer trends, and competition pertinent to the introduction of a new product. This study proposes a transformer-based recommendation system for skincare products.

Problem Statement

Our study focuses on evaluating the skincare market, and consumer needs.

How to classify skincare products according to multiple skin types?

How to represent skincare products semantically for recommendation purposes?

How to generate relevant product recommendations without user interaction data?

Dataset

Skincare product dataset containing textual descriptions and ingredient information.

Text features include:

- Product name
- Brand
- Description
- Functional explanation
- Ingredient list

Target labels:

- Combination
- Dry
- Normal
- Oily
- Sensitive

Each product may belong to multiple skin types (multi-label).

Data Preprocessing

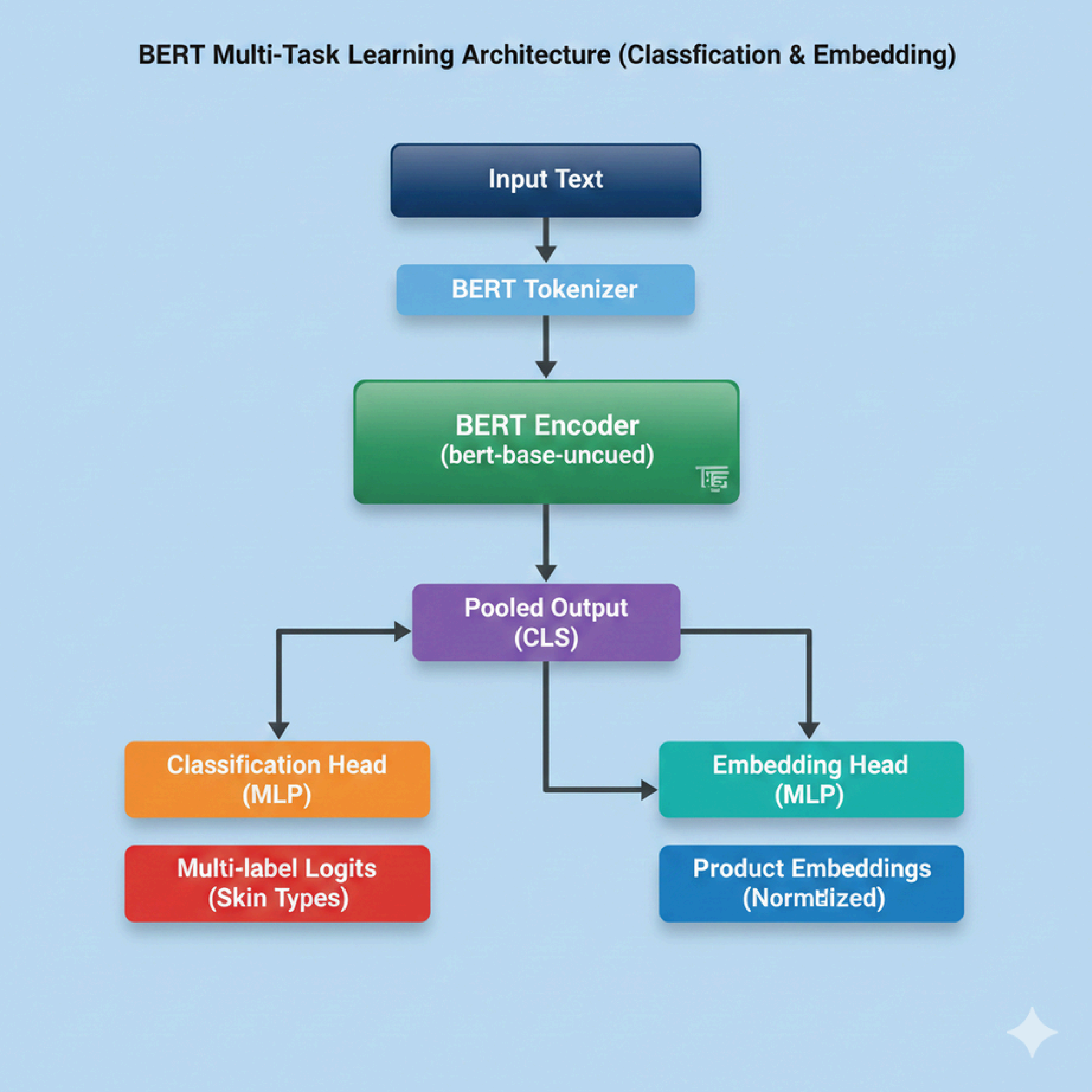
- Multiple text fields are concatenated into a single input text.
- Missing values are handled by replacing with empty strings.
- Text is tokenized using a BERT tokenizer.
- Labels are represented as binary vectors for multi-label learning.

Model Architecture

- The model uses a BERT-based hybrid architecture.
- A shared BERT encoder extracts contextual representations.
- Two parallel output heads are applied:
- Multi-label classification head
- Embedding head for product representation

Architecture Diagram

- Input text → BERT Tokenizer
- Tokenized input → BERT Encoder
- Pooled output ([CLS]) is shared by:
- Classification head → skin type prediction
- Embedding head → product embedding
- Embedding vectors are normalized for similarity computation.



Model Training Process

- The model is trained using supervised learning.
- Binary Cross Entropy with Logits is used as the loss function.
- AdamW optimizer is applied for parameter optimization.
- Training and validation sets are used to monitor learning progress.
- The model is trained end-to-end to learn both classification and embeddings.

Model Evaluation

- Evaluation is performed on the validation set.
- Sigmoid activation is applied to model outputs.
- Predictions are converted to binary labels using a fixed threshold.
- Macro-averaged F1-score is used as the evaluation metric.
- This evaluation protocol is standard for multi-label NLP tasks.

Product Embedding Generation

- After training, embeddings are extracted for all products.
- Each product is represented as a dense semantic vector.
- Embeddings are normalized to ensure stable similarity computation.
- These embeddings are stored for efficient retrieval.

Recommendation System Mechanism

- The system uses a content-based filtering approach.
- Recommendation process consists of two stages:
    a. Filtering products based on predicted skin type labels.
    b. Ranking products based on embedding similarity.
- Cosine similarity is used to measure product relevance.

Implementation

System Implementation

- Core model training and embedding generation are implemented in ModelDeepLearning.ipynb.
- The trained model is integrated into an application layer.
- No retraining occurs during system usage.

User Interface Implementation

- The user interface is implemented in skincare_recommender_system.py.
- The interface allows users to input preferences and receive recommendations.
- The UI handles inference and result visualization only.
- Model parameters remain fixed during user interaction.

System Output

The system outputs a ranked list of recommended skincare products.

Recommendations are based on semantic similarity and skin type compatibility.

The system operates without explicit user ratings or interaction history.

Conclusion

A BERT-based multi-label model was successfully implemented. The model learns both classification and semantic representations. Product embeddings enable content-based skincare recommendation. The system demonstrates the applicability of NLP in skincare recommendation.

Thank You

Deep Learning - Group 7