Flexible Fashion Product Retrieval Using Multimodality-Based Deep Learning

Yeonsik Jo ¹, Jehyeon Wi ², Minseok Kim ³ and Jae Yeol Lee ²*  
¹ School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, 123, Cheomdangwagi-ro, Buk-gu, Gwangju, Korea  
² Department of Industrial Engineering, Chonnam National University, 77, Yongbong-ro, Buk-gu, Gwangju, 61186, Korea  
³ Korea Institute of Science and Technology Information (KISTI), 245 Daehak-ro, Yuseong-gu, Daejeon, 34141, Korea  
* Correspondence: jaeyeol@chonnam.ac.kr; Tel.: +82-62-530-1782  
Received: 14 January 2020; Accepted: 17 February 2020; Published: 25 February 2020

Abstract: Typically, fashion product searching in online shopping malls uses meta-information of the product. However, the use of meta-information is not guaranteed to ensure customer satisfaction, because of inherent limitations on the inaccuracy of input meta-information, imbalance of categories, and misclassification of apparel images. These limitations prevent the shopping mall from providing a user-desired product retrieval. This paper proposes a new fashion product search method using multimodality-based deep learning, which can support more flexible and efficient retrieval by combining faceted queries and fashion image-based features. A deep convolutional neural network (CNN) generates a unique feature vector of the image, and the query input by the user is vectorized through a long short-term memory (LSTM)-based recurrent neural network (RNN). Then, the semantic similarity between the query vector and the product image vector is calculated to obtain the best match. Three different forms of the faceted query are supported. We perform quantitative and qualitative analyses to prove the effectiveness and originality of the proposed approach.

Keywords: deep learning; fashion product retrieval; multimodality-based retrieval; sustainable online shopping mall

1. Introduction

One of the key objectives of online shopping malls is to maintain efficient interaction and communication between consumers and online vendors for enhancing consumer trust [1]. Many commodities are registered on e-commerce sites or online shopping malls, and consumers try to find desired products through various search methods. Usually, when customers search for products related to fashion and apparel, they are asked to enter or select a search query they want to find. Query information includes the product name, manufacturer name, product type, product category, and/or product image. The online shopping mall then provides product information corresponding to the request of the user. Typically, fashion product search methods use product meta-information that is either entered by sellers or extracted from product images or related information. Images of fashion products may have a higher priority rather than product categories in the search result. The principle of the meta-information-based search methods, including the category-based search, is robust and straightforward. However, there are inherent limitations due to the inaccuracy of input meta-information, imbalance of categories, and misclassification of product images [2]. Therefore, for an online shopping mall to be user-centric, it is necessary to provide correct product information,
including relevant images, while retrieving the results that match with the request of the consumer. For this reason, it is essential that the widely-used faceted search methods [3–5] and image-based search methods [6,7] must be combined effectively.

Figure 1 shows the search result of fashion products conducted in one of the major on-line shopping malls in South Korea. The accuracy of the category-based search, such as women's clothing → one-piece dress → red, was as high as 93.54%, as shown in the upper part of Figure 1. However, the shopping mall system often failed to distinguish between the color of the background and that of the clothing item. That is, the system recognized a specific color in the image but could not determine what it meant. For another test for the category-based search (male clothing → t-shirt → green), we identified not only the aforementioned problem but also another problem: Non-clothing items were included among the retrieved fashion products. In this case, the seller had incorrectly entered the product meta-information, or the automatic category classification or the image search algorithm might not work correctly. For example, a golf tee was mistakenly retrieved for clothes t-shirt (t-shirt is abbreviated as “T,” which is pronounced “tee”). The incorrect search result likely appeared because the same metadata were applied to different clothes or different items. In the online shopping mall, the seller modifies the metadata of the product slightly and then uploads the same product several times. This is one of the major impediments to the accuracy of the meta-information-based search. In the preliminary study, the accuracy of the category-based search in the case study was about 75%, as explained in Section 4.1.

| Women’s clothing → one-piece dress → red | Mismatched search results |
|----------------------------------------|---------------------------|
| # of images | Percent (%) |
| correct    | 159 | 93.52 |
| overlap    | 6   | 3.53 |
| mismatch   | 5   | 2.95 |
| total      | 170 | 100  |

| Men’s clothing → t-shirt → green | Mismatched search results |
|----------------------------------|---------------------------|
| # of images | Percent (%) |
| correct    | 78 | 45.88 |
| overlap    | 48 | 28.24 |
| mismatch   | 44 | 25.88 |
| total      | 170 | 100  |

**Figure 1.** Preliminary study on fashion product retrieval. We searched for the products semi-automatically using the function provided by a major shopping mall.

This paper proposes a new fashion product search method using multimodality-based deep learning which can provide flexible and user-friendly retrieval in an online shopping mall.
Multimodal learning is a technique for simultaneously processing information with different modalities, such as images and text-based queries. To utilize both metadata-based queries and product image-based features, the proposed approach combines a convolutional neural network (CNN) model and a recurrent neural network (RNN) model. The deep CNN model learns the key features of the fashion product image and generates a unique feature vector of the image. The proposed method also vectorizes the user query through the RNN to match the CNN-based vectorized image. Finally, the semantic similarity between the query vector and the product image vector is calculated to select proper fashion product information. In particular, the proposed approach is flexible and user-centric, as three different query forms are supported: category-based, Boolean-based, and pseudo-Structured Query Language (SQL)-based ones. A quantitative analysis was conducted to evaluate based on these data by applying and comparing the concatenation, pointwise (or elementwise) sum, and pointwise product methods. A qualitative analysis was also conducted to through query results. The results indicate that the proposed method is promising for fashion product search. The contribution of this paper is as follows:

- We propose a flexible fashion product search method that employs multimodality-based deep learning while utilizing the most widely-used faceted query for an online shopping mall. In particular, the proposed approach supports category-, Boolean-, and pseudo-SQL-based queries, increasing the scalability of the search.
- By performing pre-training of the encoder, it is possible to improve the efficiency of the learning process and to support the flexibility of handling user queries.
- We collected thumbnail images from an actual online shopping mall and stored them in a database for the experiment. Using this database, we analyzed the proposed system, confirming its advantage and practicability.
- We defined various combining techniques for the multimodal classifier and evaluated them quantitatively and qualitatively.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we describe how to search for fashion products using the proposed multimodality-based deep learning method. In Section 4, we present the experimental results obtained using a commercial online shopping mall data set collected through online crawling. Additionally, the limitations of the proposed approach are discussed. Section 5 presents our conclusions and discusses future work.

2. Related Work

2.1. Typical Product Information Search

The faceted search, also known as guided navigation, is a popular and intuitive interaction paradigm for discovery and mining applications that allows users to digest, analyze, and navigate through multidimensional data. Faceted search applications are implemented in many online shopping malls [3–5]. The faceted search is mainly based on meta-information.

In particular, product images play an important role in online shopping. Thus, image retrieval techniques can be further divided into metadata (or meta-information)-based image retrieval and content-based image retrieval [6,7]. Meta-based image retrieval mainly depends on metadata related to products. Because metadata are primarily written in texts or words, meta-based image retrieval is considered a faceted classification problem. Support vector machines, decision trees, or Naive Bayes classifiers are used to solve the classification problem effectively. The metadata-based image retrieval method is simple, and its principle is straightforward. However, the search results mainly depend on the metadata of the pre-registered product, not image. As the metadata of the e-commerce site are primarily recorded by the seller, the consistency between the metadata and the related merchandise is not guaranteed, and metadata may be omitted frequently owing to human mistakes [2].

Content-based image retrieval uses generic features of the images, such as colors, shapes, and textures. The search results are likely to be consistent because the search method utilizes mathematically well-defined algorithms for related product images. Because of this unique
characteristic, the content-based image retrieval is widely used in databases, medicine, and archaeology [7]. However, semantic information cannot be obtained by only calculating image features to determine the difference between images. For example, if an image contains many areas with yellow colors, it may be difficult to determine whether the product or item is yellow or the background is yellow.

To enhance the accuracy and speed of the search, considerable attention has been paid to deep learning-based image retrieval. Since the advent of AlexNet in 2012 [8], deep learning has increasingly been used for image search and classification.

2.2. Deep Learning-Based Product Information Search

Image retrieval using deep learning is similar to the content-based retrieval. Deep learning-based methods have been successfully applied to classify images [8], to find objects in photographs, and generate textual descriptions of images [9]. Image retrieval methods using deep learning can be divided into two approaches.

One approach using deep learning is to classify the images according to predefined classification criteria. In this case, the input is an image, and the output is the probability of the image belonging to each classification criterion. CNN-based methods are widely used [10]. Well-known CNN structures include AlexNet [8], VGGNet [11], GoogleNet [12], and ResNet [13]. Some of these CNNs have been investigated for the retrieval of fashion products. Zalando Research [14] published the Fashion-MNIST DB, which made it possible to classify clothing into 10 different categories. Another research work was DeepFashion, in which various features and rich annotations were added to e-commerce apparel product images to adequately classify the mood and attributes of the products [15].

The other approach is to take input images and find or recommend similar images [16]. Lin et al. [17], Liong et al. [18], and Chen et al. [19] generated hash codes of images using CNN models. Among the store images, images with similar hash codes are presented as search results. This allowed finding almost identical images. Kiapour et al. [20] suggested a system that searched for items on shopping malls for street fashion. However, input images were needed for the product search, whereas in online shopping malls, keyword- or category-based searches are more commonplace. There are many situations where it is difficult to have an input image to search in advance.

2.3. Multimodality-Based Deep Learning

The problem to be addressed in this study is multimodal, because it involves both query data and image data. Multimodal learning is a learning method for simultaneously handling data with several different modalities. Thus, many multimodal-learning methods have been employed for audio-visual speech recognition, human-computer interaction, and action recognition [21–23].

Recent advances in deep learning have led to significant progress in multimodal learning. One such example is visual question answering (VQA) [24–26]. One of the basic VQA methods is iBOWING [24]. Ren et al. [25] proposed the VIS + LSTM model. The images were converted into vectors using a CNN, and each query word was also converted into a vector via a word embedding technique. Both vectors were used as inputs of the LSTM model, and the final output of the LSTM model was transferred to the softmax layer to generate an appropriate answer. Noh et al. [26] proposed a dynamic parameter prediction (DPP) network, called DPPNet, in which DPP is employed to determine parameters dynamically based on questions, and parameters can be shared randomly using a hash function. Fukui et al. [27] proposed a structure called multimodal compact bilinear (MCB) pooling. This method can be efficiently process data with different modalities and is based on the assumption that the cross product of the two vectors is efficient.

The purpose of VQA is to analyze an image and answer questions from the image. Therefore, VQA requires natural language-level understanding of the image scene. On the other hand, the proposed approach is based on the formal, faceted queries consisting of Boolean-, categorical-, or pseudo-SQL forms, which can retrieve fashion products efficiently and flexibly. Although there is a significant difference in the output, the proposed approach and VQA have to deal with images and
text or queries simultaneously. It is common to use a CNN to process image data and an RNN to process text data, although there are various ways to employ different networks together.

Both the keyword-based query search and the image-based product search are essential for sustainable online shopping malls. However, there is considerable room for the improvement with regard to enhancing the scalability of the query and simultaneously exploiting the image-based search.

3. Proposed Multimodality-Based Deep Learning for Fashion Product Retrieval

The proposed multimodality-based deep learning method consists of an image encoder, a query encoder, and a multimodal classifier as shown in Figure 2.

![Diagram](image)

**Figure 2.** Proposed deep learning network for efficient fashion product retrieval. Feature vectors of product images created by the deep CNN and query vectors generated by LSTM-based one-hot encoding are then combined together to semantically train the model with fashion products and classify them.

The image encoder is used to extract the features of apparel images for effective searching. We added an elementwise (or pointwise) convolution of the mobile-net to the VGG-16 structure to improve the computational efficiency [28]. The CNN model is modified to remove the fully connected (FC) layer and to store the vectorized image data in the database before the FC layer is reached.

The query encoder inputs the words of the query into the LSTM model [29] one-by-one. Generally, words are expressed via vector embedding. However, in the proposed method, the number of words is small, because a set of keywords is used as a query. Therefore, a one-hot vector is used rather than a word embedding. After all the words of the query are vectorized, the vector stored in the cell is used as the input to the multimodal classifier for training and testing. The structure of the query form and its learning are presented in Section 3.1. Three different types of queries are supported for flexible and user-centric search: Boolean, category, and pseudo-SQL.

The multimodal classifier is a decision maker that determines whether a specific image in the DB is the image desired by the user (corresponding to the user query) by combining vectors obtained from the image encoder and the query encoder. The result is either “true” or “false”. As a result, the output of the multimodal classifier is a two-dimensional vector obtained via the softmax function or
logistic regression that outputs the matching probability for a pair of the encoded query and the encoded image.

Table 1 explains more details on the deep learning architecture of the proposed approach. It shows the input and output dimensions, the number of batch size (B), and deep learning methods. A customized VGG16 is mainly used for the image encoder, and the LSTM is used for the query encoder.

**Table 1.** Deep learning architecture of the proposed approach.

|                         | Input dimension | Output dimension | Deep learning models     |
|-------------------------|-----------------|------------------|--------------------------|
| Image encoder           | [B, 64, 64, 3]  | [B, 512]         | Customized-VGG16 (CNN)   |
| Query encoder           | [B, 10, 35]     | [B, 512]         | LSTM                     |
| Multimodal classifier   | [B, 512], [B, 512]: Output of image encoder and query encoder | [B, 2] | Logistic or softmax |

3.1. **Query Forms**

The query should reflect the intentions of the user and describe the features of the desired image to be retrieved. The proposed method supports three types of query forms. The category form is a popular and simple way to classify products in most online shopping malls. The Boolean form is a logical and accurate expression of relations. Boolean expressions use the operators such as AND and OR to compare values and return “true” or “false”. In comparison, the pseudo-SQL (i.e., simplified SQL) is more complex and is employed to determine whether the proposed method can handle this type of query.

As shown in Table 2, the category form has a simple tree structure in which metadata are hierarchically represented by the main category, intermediate category, sub-category, and so on. However, using the category form, it is difficult or even impossible to search for products with complex conditions, such as unisex clothes that both men and women can wear. Thus, if we wish to consider all shirts regardless of gender, we cannot use this type of query. Almost all e-commerce sites organize their fashion products in category format.

**Table 2.** Query examples using the category form. The query scope is confined to the order of gender, clothes, color, etc. “→” represents the sub-category.

| Query            | Meaning                       |
|------------------|-------------------------------|
| “men”            | Fashion product related to men|
| “men → coat”     | Men’s coat                    |
| “men → coat → black” | Men’s black coat             |

The Boolean form can be represented by combining queries or individual attributes with Boolean expressions (Table 3). Here, attributes can be defined as keywords used in the category form. For example, “men” represents for men’s fashion products, “men AND coat” represents all men’s coats, and “coat” represents all coats regardless of gender. Therefore, the Boolean form-based search is more flexible than the category form-based search. The expressions used in the Boolean format are AND, OR, and NOT, etc. Up to two Boolean expressions were used for training. Although queries form with more than two expressions were not used for training, we confirmed the scalability of the
proposed approach by applying a query with three Boolean expressions to the test dataset. The results are discussed in Section 4.

| Queries          | Meaning                                      |
|------------------|----------------------------------------------|
| “coat”           | coat (regardless of gender)                  |
| “coat OR jacket” | coat or jacket (regardless of gender)        |
| “NOT blue”       | fashion products without the blue color      |
| “men NOT women”  | men’s fashion products, not women’s ones     |
| “coat AND blue”  | blue coat (regardless of gender)             |

The pseudo-SQL (i.e., simplified SQL) form can express more flexible queries through clauses and expressions. In the “from” clause, each DB for fashion products can be defined. In the “where” clause, a condition for the search can be defined, as shown in Table 4. The primary purpose of supporting and testing the pseudo-SQL form is to verify that the proposed approach can understand and handle more complex queries correctly and flexibly. Thus, it is possible to determine whether the proposed approach is successful for various query forms.

| Queries                                      | Meaning                                      |
|----------------------------------------------|----------------------------------------------|
| “select * from women”                        | women fashion products                        |
| “select * from women where sweater”          | women’s sweater                              |
| “select * from women where red”              | women’s red clothes                          |
| “select * from women where not red”          | women’s clothes without the red color         |

### 3.2. Pre-Training of the Query Encoder

When the deep learning-based training was conducted, the learning sometimes did not proceed properly. We assumed that the deep learning model had difficulty in learning the relationship between keywords in the queries. To alleviate this problem, we conducted pre-training of the query encoder, as shown in Figure 3, to make the encoder respond appropriately to the queries.

![Figure 3. Pre-training of the query encoder.](image)

The pre-training process is the same as the primary training process, except that the generated vector is sent to the query decoder instead of the multimodal classifier. The query decoder produces
a semantic vector corresponding to the meaning of the user query. An example of a semantic vector is shown in Table 5. The semantic vector is created based on the category query; its element is 1 if the corresponding fashion product matches the query and 0 otherwise. Table 6 shows the pairs of query words used and their corresponding semantic vectors generated by the query decoder. Because different queries have the same meanings, they can be associated with the same semantic vector. The purpose of pre-training of the encoder is to have good initial weights before main training. In this research, when the accuracy of pre-training reached about 90%, pre-training was stopped. After this pre-training process, the query encoder could generate the query word vectors by reflecting the meaning of the queries.

**Table 5.** Category form-based semantic vector representation for the pre-training. r, g, b, w, and k represent red, green, blue, white and black colors, respectively.

| main category(gender) | sub-category(clothes) | shirt | ... | ... | sub-sub-category(color) | red dress for women | green shirt for men |
|-----------------------|-----------------------|-------|-----|-----|-------------------------|-------------------|-------------------|
| men                   | g b w k r g           |       | 0 0 | 0 0 | g b w k r g             | 0 1 0 0 0 0       | 0 1 0 0 0 0       |
| woman                 |                        |       |     |     |                         | 1 0 0 0 0 0       | 0 0 0 0 0 0       |

**Table 6.** Different user queries and the results obtained after the pre-training of the query encoder and decoder.

| Input queries       | Output of the query decoder through the pre-training of the query encoder |
|---------------------|--------------------------------------------------------------------------------|
|                     | men | shirt | coat | ... | women | shirt | coat | ... |
| "shirt"             | 1 1 | r g   | r g  | ... | 1 1   | r g   | r g  | ... |
| "red"               | 1 0 | 1 1   | 0 0  | ... | 0 0   | 0 0   | 1 0   | ... |
| "red AND coat"      | 0 0 | 0 0   | 0 0  | ... | 0 0   | 0 0   | 1 0   | ... |
| "NOT shirt"         | 0 0 | 0 0   | 1 1  | ... | 0 0   | 0 0   | 1 1   | ... |
| "shirt AND coat"    | 0 0 | 0 0   | 0 0  | ... | 0 0   | 0 0   | 0 0   | ... |
| "shirt OR coat"     | 1 1 | r 0   | r 1  | ... | 1 1   | r 0   | r 1   | ... |

In this study, the total number of queries for pre-training is 10,330: With 164 category-based, 3828 Boolean-based, and 6338 pseudo SQL-based queries. As the pseudo-SQL has the most complex representation, it requires more queries for pre-training compared with the category-based and Boolean-based queries.

Important reasons to perform the pre-training are to improve the efficiency of the learning process and to support the flexibility of handling user queries. For example, in contrast to fashion product images, the user query has many different keywords or forms with the same meaning. Therefore, to increase efficiency and flexibility of the multimodal learning, we first pre-trained the encoder with the decoder and then used the encoder with user queries and product images for training the dataset. Therefore, the meaning of the operators such as OR, AND, and NOT were learned in advance. Other query forms were similarly pre-trained. The suggested pre-training process is based on the seq2seq model proposed by Sutskever et al. [30]. This is a language generation model that is used for various applications, such as machine translation, speech recognition, and text summarization.

To interpret the results of the pre-training, we used the t-distributed Stochastic Neighbor Embedding(t-SNE) visualization technique [31], as shown in Figure 4. t-SNE is a two-dimensional
representation of a high-dimensional vector. Figure 4(a) shows all the query terms used in the pre-training through t-SNE. Figure 4(b) shows some query examples, which are defined as follows:

- **Example query 1**
  - "men → shirt → green",
  - "SELECT * FROM men WHERE shirt AND green"
- **Example query 2**
  - "women → dress → red"
  - "SELECT * FROM women WHERE dress AND red"
- **Example query 3**
  - "men → coat"
  - "men AND coat"
  - "SELECT * FROM men WHERE coat"

In the results of the pre-training with the foregoing three examples, the queries with similar meanings appear at similar positions, but the queries with different meanings appear at a distance. Thus, through the pre-training, the query encoder generates an appropriate vector according to the meaning of the query.

![Figure 4(a)](image)

**Figure 4.** Visualization of the results of the pre-training using t-SNE: (a) query distribution, (b) neighbor embedding.

The pre-training of the query encoder can have two effects. Firstly, the training is expedited as it is possible to acquire excellent initial weights in the early stage of the training in order to converge more quickly to the optimum solution. Second, the query encoder can develop the ability to understand the various forms of queries semantically. For example, in this study, even though only Boolean expressions with two keywords were used for the training, the capability of the semantic analysis of the query encoder was extended to three-term expressions. Because of the pre-training,
the query encoder understood not only the relationship between the product image and query text but also the meaning of the query.

To retrieve fashion product information using the proposed multimodal-based network, it is necessary to match all the product images one-by-one. However, the number of images is very large, and processing images using the CNN is memory- and time-consuming. Fortunately, our network can solve this problem by processing and storing the feature vectors of the product images into the database as shown Figure 2. Thus, when a user query is entered, the search can be expedited by using the feature vector of the image that has been calculated in advance.

4. System Implementation and Experimental Analysis

4.1. Fashion Product Dataset and Search

The training and test dataset for men’s and women’s fashion products were collected through the online crawling of the thumbnail images from one of the major online shopping malls in Korea. As shown in Figure 5, fashion product images are very useful for verifying the applicability and flexibility of the proposed approach because each image can have various attributes at the same time.

The apparel images were searched and collected with regard to gender, type, and color. In total, 12 categories of men’s clothes and 15 categories of women’s clothes were searched (shown in Tables 6 and 7, respectively). Additionally, each clothing category was crawled separately for five colors. Because there were numerous redundant and misclassified images, we first collected 2500 images for each sub-category. Then, duplicated images were deleted by a program. Among the total 67,500 obtained images, 28,035 images were identified as duplicates and removed. Furthermore, misclassified images were manually removed by human operators. In particular, 677 images were not related to clothes, and 8922 images were misclassified. After these were removed, 29,866 images remained. The accuracy was calculated after the elimination of misclassified and redundant images. We did not consider duplication as an incorrect result. Excluding redundancy, the accuracy of the crawled images from the shopping mall was approximately 75%, as indicated by Tables 7 and 8. This process is essential to prevent the proposed approach from training and testing the duplicate data at the same time or to prevent it from making a wrong decision, such as answering the false to the true or vice versa.

---

**Figure 5.** Fashion products with various attributes.
Table 7. Accuracy of the collected images of men’s clothes after the removal of duplicates.

| Category of clothes | The number of original images | After removing duplicates (B) | After removing misclassified images (C) | Accuracy = C/B |
|---------------------|-------------------------------|-------------------------------|----------------------------------------|---------------|
| cardigan            | 2500 per each category        | 1592                          | 1222                                   | 0.7676        |
| coat                |                               | 1690                          | 1078                                   | 0.6379        |
| jacket              |                               | 1670                          | 1436                                   | 0.8599        |
| jeans               |                               | 1647                          | 995                                    | 0.6041        |
| jumper              |                               | 1025                          | 819                                    | 0.7990        |
| pants               |                               | 1392                          | 939                                    | 0.6746        |
| shirts              |                               | 1694                          | 1362                                   | 0.8040        |
| suit                |                               | 1380                          | 984                                    | 0.7130        |
| sweater             |                               | 1479                          | 1332                                   | 0.9006        |
| training-suit       |                               | 1252                          | 981                                    | 0.7835        |
| t-shirt             |                               | 1866                          | 1451                                   | 0.7776        |
| waistcoat           |                               | 1082                          | 860                                    | 0.7948        |
| total               | 30,000                        | 17,769                        | 13,459                                 | 0.7574        |

Table 8. Accuracy of the collected images of women’s clothes after the removal of duplicates.

| Category of clothes | The number of original images | After removing duplicates (B) | After removing misclassified images (C) | Accuracy = C/B |
|---------------------|-------------------------------|-------------------------------|----------------------------------------|---------------|
| cardigan            | 2500                          | 813                           | 671                                    | 0.8253        |
| coat                |                               | 1583                          | 1198                                   | 0.7568        |
| dress               |                               | 1918                          | 1482                                   | 0.7727        |
| jacket              |                               | 1705                          | 1478                                   | 0.8669        |
| jeans               |                               | 1535                          | 844                                    | 0.5498        |
| jumper              |                               | 1300                          | 1111                                   | 0.8546        |
| leggings            |                               | 1356                          | 831                                    | 0.6128        |
| pants               |                               | 1622                          | 1115                                   | 0.6874        |
| shirt               |                               | 1760                          | 1494                                   | 0.8489        |
| skirt               |                               | 1700                          | 1396                                   | 0.8212        |
| suit                |                               | 1118                          | 625                                    | 0.5590        |
| sweater             |                               | 499                           | 422                                    | 0.8457        |
| training-suit       |                               | 1469                          | 1101                                   | 0.7495        |
| t-shirt             |                               | 1836                          | 1420                                   | 0.7734        |
| waistcoat           |                               | 1482                          | 1219                                   | 0.8225        |
| total               | 37,500                        | 21,696                        | 16,407                                 | 0.7562        |

The dataset in Tables 7 and 8 were randomly partitioned into the ratio of 5:1 for training and testing. The number of training images is 24,885, and that of testing images is 4981. We also balanced the ratio among categories when partitioning. Each data used in training and testing consists of a pair of images and queries. For each image, several queries were randomly generated since there are many queries corresponding to the same image.
4.2. Experimental Analysis

The performance of the proposed approach was analyzed by using a confusion matrix as a performance evaluation index. The accuracy (ACC), precision (PPV), recall (TPR), and F₁ score were used as performance indices, as shown below (Equations 1–4) [32]. Additionally, the receiver operating characteristic (ROC) curve was used for performance visualization [33].

\[
ACC = \frac{TP + TN}{TP + TN + FP + FN}
\]

\[
PPV = \frac{TP}{TP + FP}
\]

\[
TPR = \frac{TP}{TP + FN}
\]

\[
F_1 = \frac{2 \times PPV \times TPR}{PPV + TPR} = \frac{2 \times TP}{2 \times TP + FP + FN}
\]

| Actual class | True | False |
|--------------|------|-------|
| True         | TP   | FN    |
| False        | FP   | TN    |

To evaluate the performance of the proposed method, various combinations of multimodal classifiers were applied, as shown in Table 9. The accuracy of the fashion product retrieval was analyzed by applying different combination methods: concat, ptwise sum, ptwise product, and multimodal compact bilinear (MCB). Additionally, we analyzed the performance of the classifier according to the number of hidden layers and the type of combination. The combination methods and their characteristics are described as follows:

- **concat**: This is a method of concatenating the image feature using the CNN and the query feature using the seq2seq-based RNN to obtain a single feature vector. The feature dimensions of the two vectors need not be the same. However, the two feature vectors may merge together, because they depend on either image data or text data. The simple concatenation is not good, but its performance improves as FC layers are added. This appears to be due to the improved ability to interpret the data with increasing FC layers, rather than the improved performance of the concat method.

- **ptwise sum, ptwise product**: In contrast to the concat method, these methods attempt to combine the image and query data properly (pointwise (or elementwise) sum and pointwise product). These methods are to calculate the sum or product of the elements of the same position with the same size of the image and query feature vectors. Thus, the two feature tensors processed by the CNN and RNN must have the same shape. In contrast to the ptwise sum method, the ptwise product can be interpreted as the inner product of two tensors or the statistical correlation of two tensors. Thus, the ptwise product method exhibits relatively good performance even without the FC layers.

- **MCB pooling**: This method is based on the assumption that the outer product of tensors yields better performance than the inner product. For example, assuming that the size of the image tensor and the text tensor is 256, the dimension of the outer product of the two tensors is 256 × 256 (2₁⁶). The dimension of 2₁⁶ is too large for the parameters to be learned. However, in the MCB pooling, the two tensors are transformed into tensors of the same dimension [24]. The transformed tensors are re-calculated to obtain a single tensor in the frequency domain using a fast Fourier transform (FTT) and are then returned as another tensor through an inverse fast Fourier transform (IFTT). Because the multiplication in the frequency domain has the same effect
as the convolution in the original state, it is possible to obtain a similar result while avoiding the calculation of the outer product.

Table 9. Performance analysis for different combinations of the image feature and query feature in the proposed approach.

| Method                   | Accuracy | Precision | Recall | F1 score |
|--------------------------|----------|-----------|--------|----------|
| concat                   | 79.298   | 74.711    | 62.221 | 67.896   |
| concat + FC              | 88.006   | 82.615    | 85.067 | 83.823   |
| concat + FC + FC         | 89.550   | 86.493    | 85.062 | 85.771   |
| ptwise sum               | 79.222   | 75.241    | 61.008 | 67.381   |
| ptwise sum + FC          | 88.220   | 83.943    | 83.843 | 83.893   |
| ptwise sum + FC + FC     | 89.864   | 86.185    | 86.238 | 86.486   |
| MCB + FC + FC            | 89.975   | 86.736    | 86.238 | 86.486   |
| ptwise product           | 86.296   | 81.957    | 79.578 | 80.750   |
| ptwise product + FC      | 88.192   | 83.943    | 83.843 | 83.893   |
| ptwise product + FC + FC | 90.000   | 86.348    | 86.853 | 86.600   |

As shown in Table 9, the ptwise product + FC + FC method exhibited the best performance. The performance of the methods varied in the F1 score range of 67.381-86.6. A larger number of FC layers yielded better results. However, adding FC layers increased the computation time. The performance of the proposed multimodal technique increased in the following order: concat, ptwise sum, MCB pooling, and ptwise product methods. The MCB pooling showed better performance than the ptwise product for the VQA problem, but it showed lower performance in the proposed approach [27].

Figure 6 shows the ROC curve for each combination presented in Table 9. The performance of each combination is indicated by the area under the curve (AUC). The x- and y-axes of the ROC curve have a range of [0,1]. An AUC closer to 1 (closer to the top-left corner) indicates better performance. An AUC of < 0.5 indicates a value-less test. The concat and ptwise sum combinations exhibited the smallest AUCs, and the ptwise product combination exhibited the largest AUC.

![Figure 6](image-url)  
*Figure 6. ROC curves for the different combinations of the CNN and RNN for fashion product research.*
We have compared the proposed approach with conventional methods indirectly. The proposed approach is unique with respect to multimodal learning of different types of queries and images for fashion product retrieval. Since it is impossible to compare it with conventional methods directly, two conventional methods and their variants without using multimodality learning were designed, as shown in Tables 10 and 11. Without the query encoder, they are similar to the proposed method. The input model of the conventional method is the image encoder in Table 1. As these methods cannot adopt query vectors, they need a new classification output with respect to the dataset. In this paper, there are 135 classes for fashion products (two genders, twelve categories for men, fifteen categories for women, and five colors). As there is a difference between the gender-related categories, we just merged them into 27 categories. As a result, $27 \times 5 = 135$ classes were defined for the single-label classification, and 75 and five classes were defined for the multi-label classification.

Table 10. Network architectures of conventional methods. $B$ represents the batch size.

| Input dimension | Output dimension | Deep learning models |
|-----------------|-----------------|---------------------|
| Single-label (SL) classification | [B, 512] | [B, 135] |
| Multi-label (ML) classification | [B, 512] | [B, 27], [B, 5] |

- Input model: Image encoder (customized VGG16)
- Output model: Single or multi-layer perceptron

Table 11. Indirect comparative analysis of conventional methods. The same dataset is used for comparison.

| Methods | Accuracy | Precision | Recall | F1 score |
|---------|----------|-----------|--------|----------|
| SL classification with one FC output layer | 45.035 | 65.079 | 33.874 | 44.556 |
| SL classification with three FC output layers | 45.090 | 65.194 | 33.858 | 44.569 |
| ML classification with one image encoder and one FC output layer | 45.018 | 65.119 | 33.752 | 44.460 |
| ML classification with one image encoder and three FC output layers | 45.092 | 65.187 | 33.877 | 44.584 |
| ML classification with two image encoders and one FC output layer | 44.953 | 65.047 | 33.657 | 44.361 |
| ML classification with two image encoders and three FC output layers | 44.929 | 65.031 | 33.604 | 44.311 |
As shown in Table 11, the proposed approach outperformed conventional methods regardless of the number of input models and that of FC out layers, as conventional methods did not utilize multimodal information simultaneously. If the number of type or factor increases, the number of classification increases drastically so that the conventional method requires a lot of resources, and the results deteriorates. In particular, the difference between precision and recall is due to the much higher rate of false negative decision of the models since there are only one true answer and 134 false answers. Nevertheless, a further study is still needed to evaluate the proposed approach compared with other methods with similar architectures directly.

4.3. Qualitative Results of Case Studies

In this subsection, we present the results of the proposed multimodality-based deep learning approach for fashion product retrieval. Figure 7 shows the template for the search results. The 10 apparel images with the highest probability corresponding to the user query were retrieved from the test dataset and displayed sequentially, and the 10 images with the lowest probability were also selected and displayed. The ptwise product method with two FC layers was used for the search, as it exhibited the best performance, as shown in Table 8. TensorFlow and Keras were used to implement the proposed approach [34,35].

| Query                                                                 |
|----------------------------------------------------------------------|
| - Ten recommended fashion product images with the highest probability (upper images) |
| - Ten images with the lowest probability (lower images)               |

**Figure 7.** Template for the fashion product search results.

Figure 8 shows the results of fashion product retrieval in the case of a user query with a single keyword. The first example shows the search results for the query term “men”. The 10 best matches include shirts, suits, and hoodies that men usually wear. In the 10 matches with the lowest probability, there are clothes that only women wear (e.g., a dress). For the second example (i.e., the query term “skirt”), the top 10 matches include women’s skirts. On the other hand, the 10 worst matches include men’s apparel. Finally, in the case of the query term “green”, all the best matches are green fashion products, regardless of gender. Conversely, the worst matches are red or white fashion products. This is because the distance between the green and red in the image color space is large. The distances among green (0, 255, 0), red (255,0,0), and white (255,255,255) are larger than the distances of these colors to black (0,0,0). Even the color blue has a large distance from green, but the training dataset contained fewer blue products than red and white ones. We found that the blue ones were also far from the green ones.
Boolean operators include AND, OR, and NOT. As shown in Figure 9, these logical operators were employed to determine whether the proposed method could understand the queries properly. For the first and second queries in Figure 9, we checked whether meanings of AND, OR, and NOT were properly understood. First, in the case of “black AND NOT white” query, only fashion products with the white color were retrieved. Second, in the case of the OR operation, product images containing either blue or green were correctly retrieved. Similarly, Figure 10 shows the results of the apparel retrieval for category-based queries. These search results indicate that the proposed approach correctly retrieves fashion products according to the queries.
Pseudo-SQL-based queries were employed to determine how well the proposed deep learning method can interpret complicated queries and to thereby investigate the scalability and flexibility of the method (Figure 11). The first query — “SELECT * FROM men” — is similar in meaning to the query using the single keyword, “men”, shown in Figure 8, although the format is completely different. The search results for this pseudo-SQL query are consistent with the results for the single keyword query shown in Figure 8. The results with the highest probability mainly include men’s clothes, whereas those with the lowest probability are women’s clothes. For the second query — “SELECT * FROM men WHERE red” — the search focused on red apparel for men. The results with the highest probability were appropriate, and those with the lowest probability included women’s clothing without the red color. For the last query — “SELECT * FROM men where blue AND shirt” — the results with the highest probability were blue men’s shirts, as intended, and the 10 images with the lowest probability included women’s clothes without the blue color.

We compared and analyzed the results for the three aforementioned types of queries. As shown in Figure 12, when the queries were designed to find black pants for men, they retrieved similar or the same images. The 10 results with the highest probability included men’s black trousers, as intended. The 10 results with the lowest probability mainly included women’s clothing.
We also examined the result for a ternary Boolean operation. Previously, we did not use a Boolean operation with three query terms. A third term was arbitrarily added to evaluate the flexibility of the proposed approach. The results for the query “men AND pants AND black” were good, indicating that the proposed deep-learning network works correctly and flexibly even for user queries not included in the training.

We also investigated the fashion product retrieval results for queries that were not used during the training, such as the ternary Boolean query shown in Figure 12. We randomly mixed two query forms or nonexistent conditions (e.g., women’s leggings in men’s clothing). As shown in Figure 13, the query “men → coat AND shirt” is a mixture of Boolean and category queries. Nevertheless, the search worked well. In particular, the query “men → leggings” is unusual, as there is no such fashion product for men. According to the characteristics of leggings, the query results included pants, and skirts and dresses had the lowest probability.

| query : ‘men → pants → black’ |
|-------------------------------|
| ![Image of query results](image1.jpg) |

| query : ‘men AND pants AND black’ |
|-------------------------------|
| ![Image of query results](image2.jpg) |

| query : ‘SELECT * FROM men WHERE pants AND black’ |
|-------------------------------|
| ![Image of query results](image3.jpg) |

**Figure 12.** Search results for three different types of queries with similar or the same meaning.

| query : ‘men → coat AND shirt’ |
|-------------------------------|
| ![Image of query results](image4.jpg) |

| query : ‘men → leggings’ |
|-------------------------------|
| ![Image of query results](image5.jpg) |

**Figure 13.** Search results for undefined queries.
4.4. Limitation and Future Consideration

Although the proposed multimodality-based deep learning approach allows flexible and effective fashion product retrieval, it has limitations. In particular, clear criteria must be established for image classification and keywords. For example, Figure 14 shows the search results for cardigans and sweaters. The queries are simple, but the search results for “sweater” and “cardigan” are unsatisfactory. In the case of “men AND cardigan,” the gender is distinguished, but some of the results do not match the query. The results indicate that the proposed method did not understand the keywords “sweater” and “cardigan” properly. This is because the boundary between cardigans and sweaters in the training data was vague. Because clothes have various designs, they are often difficult to classify. Figure 15 shows some of the image data used for the training. The design patterns are similar. The large amount of similar images between the two classes appears to have adversely affected the learning.

![Figure 14. Unsatisfactory search results.](image)

![Figure 15. Training data set for cardigans and sweaters.](image)

5. Conclusion

This paper presented a new fashion product search method using multimodality-based deep learning, which can provide reliable and consistent retrieval results by combining a faceted query and a product image. An accurate and user-friendly search function is crucial for a sustainable online shopping mall. Our method combines a CNN and an RNN. The deep CNN extracts generic feature information of a product image, and the user query is also vectorized through the RNN. The semantic similarity between the query feature and the product image feature is calculated, whereby the closest matching fashion products are selected.
In this study, fashion product-related images from an actual online shopping mall were collected. Additionally, a wide range of query terms was automatically generated to test the flexibility of the search method. Three query forms were supported: category-based, Boolean-based, and pseudo-SQL-based forms. We presented a new seq2seq pre-training method for the RNN based query encoder. Furthermore, we compared various combination techniques for handling different modalities. The results indicated that the proposed approach was effective for queries of different forms. Although the filtering of the training data was insufficient, the model achieved higher accuracy and F1-score. Obviously, the proposed approach will provide better performance if well-refined data are used.

In future research, we plan to use natural language-type metadata for training. E-commerce metadata have large amount of noise but contains a wealth of information. We will also attempt to extend the proposed query forms such as the pseudo-SQL to handle more complicated questions. In addition, we will improve the efficiency of the query search by applying heuristic or top-k algorithms [36]. Furthermore, a further study is needed to evaluate the proposed approach by conducting comparative analysis with more conventional methods.

**Author Contributions:** Conceptualization, Y.J. and J.Y.L.; data curation, Y.J. and J.W.; methodology, Y.J., M.K. and J.Y.L.; formal analysis, M.K. and J.Y.L.; investigation, Y.J., J.W. and J.Y.L.; software, Y.J. and J.W.; supervision, J.Y.L.; writing-review and editing, J.Y.L. All authors contributed extensively to the work presented in this paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF-2019R1I1A3A01059082).

**Conflicts of Interest:** The authors declare no conflicts of interest.

**References**

1. Sohaib, O.; Kang, K.; Nurunnabi, M. Gender-based iTrust in E-commerce: The moderating role of cognitive innovativeness. *Sustainability* 2019, 11, 175.
2. Ha, J.-W.; Pyo, H.; Kim, J. Large-scale item categorization in e-commerce using deep learning and distributed computing. *Commun. Korean Inst. Inf. Sci. Eng.* 2016, 34, 32–38.
3. Arenas, M.; Grau, B.C.; Kharlamov, E.; Marciuška, Š.; Zheleznyakov, D. Faceted search over RDF-based knowledge graphs. *Web Semant. Searchservices Agents World Wide Web* 2016, 37–38, 55–74.
4. Ben-Yitzhak, O.; Golbandi, N.; Har’El, N.; Lempel, R.; Neumann, A.; Ofek-Koifman, S.; Steinwald, D.; Shekita, E.; Sznaider, B.; Yogeve, S. Beyond basic faceted search. In Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM ’08), Palo Alto, CA, USA, 11–12 February 2008; pp. 33–44.
5. Hearst, M. Design recommendations for hierarchical faceted search interfaces. In Proceedings of the ACM Workshop on Faceted Search (SIGIR ’06), Seattle, USA, August 2006; pp. 26–30.
6. Image Retrieval. Available online: https://en.wikipedia.org/wiki/Image_retrieval. 2017.10 (accessed on 20 July 2018).
7. Juneja, K.; Verma, A.; Goel, S.; Goel, S. A survey on recent image indexing and retrieval techniques for low-level feature extraction in CBIR systems. In Proceedings of the 2015 IEEE International Conference on Computational Intelligence and Communication Technology, Ghaziabad, India, 13–14 February 2015.
8. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1106–1114.
9. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. *arXiv* 2015, arXiv:org/abs/1411.4555.
10. Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, G.; Cai, J.; et al. Recent advances in convolutional neural networks. *Pattern Recognit.* 2018, 77, 354–377.
11. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* 2015, arXiv:org/abs/1409.1556.
12. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. *arXiv* 2014, arXiv:org/abs/1409.4842.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv 2015*, arXiv.org/abs/1512.03385.
14. Xiao, H.; Rasul, K.; Vollgraf, V. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv 2015*, arXiv.org/abs/1708.07747.
15. Liu, Z.; Lou, P.; Qiu, S.; Wang, X.; Tang, X. DeepFashion: Powering robust clothes recognition and retrieval with rich annotations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Las Vegas, NV, USA, 27–30 June 2016.
16. Kang, W.-C.; Kim, E.; Leskovec, J.; Rosenberg, C.; McAuley, J. Complete the look: Scene-based complementary product recommendation. *arXiv 2018*, arXiv.org/abs/1812.01748.
17. Lin, K.; Yang, H.-F.; Hsiao, J.-H.; Chen, C.-S. Deep Learning of binary hash codes for fast image retrieval. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR'15), Boston, MA, USA, 7–12 June 2015.
18. Liong, V.E.; Lu, J.; Wang, G.; Moulin, P.; Zhou, J. Deep hashing for compact binary codes learning. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15), Boston, MA, USA, 7–12 June 2015.
19. Chen, W.; Wilson, J.T.; Tyree, S.; Weinberger, K.Q.; Chen, Y. Compressing neural networks with the hashing trick. *arXiv 2018*, arXiv.org/abs/1504.04788.
20. Hadi Kiapour, M.; Han, X.; Lazebnik, S.; Berg, A.C.; Berg, T.L. Where to buy it: Matching street clothing photos in online shops. In Proceedings of the 2015 IEEE International Conference on Computer Vision (CVPR'15), Santiago, Chile, 7–13 December 2015.
21. Potamianos, G.; Neti, C.; Gravier, G.; Garg, A.; Senior, A.W. Recent advances in the automatic recognition of audiovisual speech. *Proc. IEEE 2003*, 91, 1306–1326.
22. Fels, S.S.; Hinton, G.E. Glove-Talk: A neural network interface between a data-glove and a speech synthesizer. *IEEE Trans. Neural Netw. 1993*, 4, 2–8.
23. Ordoñez, F.J.; Roggen, D. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors 2016*, 16, 115.
24. Zhou, B.; Tian, Y.; Sukhbaatar, S.; Szlam, A.; Fergus, R. Simple baseline for visual question answering. *arXiv 2015*, arXiv.org/abs/1512.02167.
25. Ren, M.; Kiros, R.; Zemel, R.S. Exploring models and data for image question answering. *arXiv 2015*, arXiv.org/abs/1505.02074.
26. Noh, H.; Seo, P.H.; Han, B. Image question answering using convolutional neural network with dynamic parameter prediction. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Las Vegas, NV, USA, 27–30 June 2016.
27. Fukui, A.; Park, D.H.; Yang, D.; Rohrbach, A.; Darrell, T.; Rohrbach, M. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv 2016*, arXiv.org/abs/1606.01847.
28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv 2017*, arXiv.org/abs/1704.04861.
29. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst. 2017*, 28, 2222–2232.
30. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *arXiv 2014*, arXiv.org/abs/1409.3215.
31. Maaten, L.v.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res. 2008*, 9, 2579–2605.
32. Sasaki, Y. The Truth of the F-Measure. 2007. Available online: https://www.cs.odu.edu/~mukka/cs795sum10dm/Lecturenotes/Day3/F-measure-YS-26Oct07.pdf (accessed on 30 July 2018).
33. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett. 2006*, 27, 861–874.
34. TensorFlow. 2018. Available online: https://github.com/tensorflow/tensorflowpdf (accessed on 10 March 2018).
35. Keras. 2018. Available online: https://www.tensorflow.org/guide/keraspdf (accessed on 10 March 2018).
36. Bruno, N.; Wang, H. The threshold algorithm: From middleware systems to the relational engine. *IEEE Trans. Knowl. Data Eng.* 2007, 9, 523–536.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).