applied sciences

MDPI

Article

Generalized Image Captioning for Multilingual Support

Suhyun Cho [1] **and Hayoung Oh** [2,*]

[1] Applied Artificial Intelligence Convergence Department, Sungkyunkwan University, Seoul 03063, Republic of Korea
[2] College of Computing & Informatics, Sungkyunkwan University, Seoul 03063, Republic of Korea
* Correspondence: hyoh79@skku.edu

Abstract: Image captioning is a problem of viewing images and describing images in language. This is an important problem that can be solved by understanding the image, and combining two fields of image processing and natural language processing into one. The purpose of image captioning research so far has been to create general explanatory captions in the learning data. However, various environments in reality must be considered for practical use, as well as image descriptions that suit the purpose of use. Image caption research requires processing new learning data to generate descriptive captions for specific purposes, but it takes a lot of time and effort to create learnable data. In this study, we propose a method to solve this problem. Popular image captioning can help visually impaired people understand their surroundings by automatically recognizing and describing images into text and then into voice and is an important issue that can be applied to many places such as image search, art therapy, sports commentary, and real-time traffic information commentary. Through the domain object dictionary method proposed in this study, we propose a method to generate image captions without the need to process new learning data by adjusting the object dictionary for each domain application. The method proposed in the study is to change the dictionary of the object to focus on the domain object dictionary rather than processing the learning data, leading to the creation of various image captions by intensively explaining the objects required for each domain. In this work, we propose a filter captioning model that induces generation of image captions from various domains while maintaining the performance of existing models.

Keywords: image captioning; multimodal; vision; NLP

check for updates

Citation: Cho, S.; Oh, H. Generalized Image Captioning for Multilingual Support. *Appl. Sci.* **2023**, *13*, 2446. https://doi.org/10.3390/app13042446

Academic Editor: Valentino Santucci

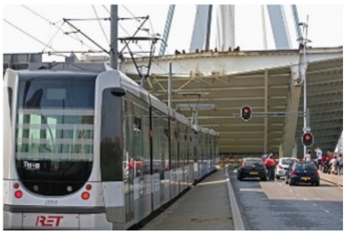
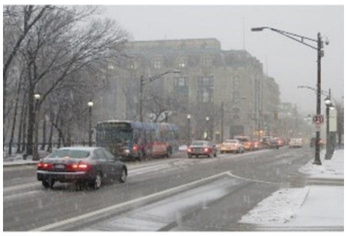
Received: 31 December 2022
Revised: 31 January 2023
Accepted: 8 February 2023
Published: 14 February 2023

1. Introduction

Image captioning is the process of converting an image into recognizable text. A general image caption study extracts information from an image based on the relationship between an object and its location, derives a semantic relationship based on the extracted information, and explains the information of the image in text. Image captioning problems include classifying sentences that best describe images in a fixed format in which words at a specific location are boxed to describe which words in the view fit the image and, recently, creating explanations or subtitles by generating text to explain images and inputs. To solve these problems, image caption research has developed at a rapid pace over the past few years with various methodologies.

BLEU [1], METEOR [2], ROUGE [3], SPICE [4], and CIDEr [5] image caption methods have been proposed for evaluating image captions generated from models. There is an approach using a CNN (convolution neural network), namely, a methodology for extracting semantic information from an image through a CNN and understanding sequential information through a recurrent neural network (RNN) to generate text. As a result, a model of a graph structure has been proposed. Recently, a BERT (bidirectional encoder representations from transformers) [6] model based on a transformer [7] has been proposed, which achieves remarkable performance in understanding images and generating text. Image captioning studies so far have focused on how general text can be generated for images of various domains. However, depending on the purpose and perspective, different captions may be required for the same image. As shown in Table 1, multiple captions may occur in one image depending on the goal. To solve this problem, this paper presents a method of generating Korean and English text captions and creating image captions according to the purpose. As a result, various image captions can be generated from one image using our proposal.

Table 1. Example of a contextual caption in a picture.

		
traffic	There are a few cars waiting for the signal.	A street with a lot of traffic during a snowstorm
blind	Railways and roads are left and right, so be careful.	There is a road on the left, and it is snowing a lot
weather	It is a bright day.	It is snowing a lot and piling up.
general	An electric train at an intersection with cars.	Cars drive down the street on a snowy day.

The contributions of this study are as follows:

1. Image caption data generation requires a lot of manual work, but there is no need to process new data through the domain object dictionary presented in this study.
2. Various image captions can be created from one image.
3. There is no need to learn new models when creating domain image captions.
4. Proposed filter captioning model that can generate various image captions.

2. Related Work

2.1. Image Caption Dataset

In order to solve the problem of image caption, many studies have been conducted to process datasets for image caption learning. The TextCaps [8] dataset study presents a problem with the existing image caption learning data. When a person sees and describes an image, he or she often uses text that is explained or translated to be understandable, which is often understood only by looking at the text in the image. In addition, there are cases where it is difficult to explain the image in a short sentence while also offering adequate understanding of the image. The existing image captioning datasets [9]—VQA data [10], TextVQA, and OCR-VQA data—consist of simple answers such as “yes” and “two”, indicating that most of the answers are fewer than five words. Because these short image captions cannot fully describe the image, TextCaps’ data consist of an average of 12.4 words, creating appropriate image captions for an image. In addition, in the problem of detecting text generated from an image using optical character recognition (OCR) and generating captions using this information, the existing dataset uses words as they are to create captions according to the description flow. In addition, data analysis and construction methods are being studied to apply learning data to various real-life situations. The Hateful Memes Challenge [11] studied how image captions could be used online to discriminate hateful data on photographs or in descriptions of photos and turn them into positive texts and build datasets.

2.2. Image Caption Model

In the case of [12] studies, only the order of image objects is changed to generate various image captions, resulting in a significant decrease in accuracy, and thus the verification of accuracy is not possible. In this work, Fine-Tuning noise addition, conversion, and accuracy were verified during training to generate image universal captions. Visual Vocabulary Pre-Training for Novel Object Captioning (VIVO) [13] models involve COCO [9] and Flickr30k [14] data, which are existing data that presented problems in TextCaps [8] studies. Even though there is a dataset, it does not actually have a dramatic effect because it is trained by predetermined data. To solve this problem, a method for grasping the context of an image is presented, such as the pre-training method of the Bidirectional Encoder Representations from the Transformer (BERT) [6] model. First, by learning vision–language pre-training (VLP) using about 64,000 large-capacity photo data points with object detection information on the image and by improving the contextual understanding of the model's photos and fine-tuning for image captions using nocaps [15] (novel object captioning at scale), we present a method of focusing on fine-tuning and modeling the COCO [9], Conceptual Captions (CC) [16], SBU Captions [17], and Flickr30k [14] datasets for VL. OSCAR [18] learns using text, tags, and object information areas of images to pre-learn images and achieves SOTA scores in seven VL tasks. In addition, there is controlling length in image captioning [19], which adds length embedding of image captions for diversity in image captions. However, since only the length of the simple sentence changes, captions of various sentence lengths can be generated during inference. The Switchable Novel Object Captioner [20] (SNOC) study proposes a caption generation method for new objects, which is a disadvantage for existing image captioning tasks that rely on object tags. A disadvantage exists in that only the length of the caption changes and the content of the explanation cannot be changed or manipulated.

2.3. OCR (Optical Character Recognition)

Optical character recognition (OCR) is a technology for recognizing characters in an image. OCR plays an important role in image caption research. When visual information is recognized in an image, a caption is generated by understanding not only object information but also character information. There are sequence recognition methods [21,22] for understanding images in addition to PP-OCR [23] and Vinvl [24], which are learning methods using transformers that use various CNN-based embedding [25–27] methods. Unlike previous studies aimed at simply detecting and extracting text from an image, recent studies have accessed research that presents a semiconductor network (SRN) [28], a model that uses two transformer models, and text containing contextual meanings, making it difficult to compress into actual services.

3. Filter Captioning Model Algorithm

This paper proposes a filter captioning model that generates image captions suitable for a domain by filtering the image object tag, which is the input of the image caption model, rather than by creating new image caption data according to a new domain.

3.1. Image Captioning Model by Language

In this study, OSCAR [18] is used as the base model. The English model used the pre-trained model provided by OSCAR [18]. However, since there is no pre-training model in Korean, only the transformer model used the pre-trained KoElectra [29] model in Korean. The model is available on Huggingface [30].

3.2. Image and Natural Language Understanding Model

Image captioning requires understanding of images, which are input data, and explaining images with text data in a natural language. Therefore, a model is required that can simultaneously understand images and natural language. In this paper, the model is based on the OSCAR [18] model that simultaneously understands images and natural language. In this study, in order to understand the image and natural language at the same time, explanatory sentences for the image, object tags of the image extracted through the image object detection model, and regional vectors for the object are received as inputs. Two loss functions are used for input data.

$$LOSS_{MLT} = -E_{(v,h)-D} \log p(h_i | h_{/i}, v) \tag{1}$$

First, a method of predicting the token by covering the mask with a 15% probability of the image caption token is used. In Equation (1), D is randomly extracted data, v is an area vector for an image, h is a masked token, and $h_{/i}$ is a surrounding token of the masked token.

$$LOSS_C = -E_{(v,h)-D} \log p(y | f(h', w)) \tag{2}$$

The second loss function predicts whether a vector is changed by changing the vector to a probability of 50% of the contaminated image. In Equation (2), D is randomly extracted data, w is an image description token, and h is an image object tag and a region vector. The above two loss functions are combined to learn about 4.1 million images and 6.5 million texts, and image object tags and image data are using from COCO data [9], Conceptual Captions [16], SBU Captions [17], Flickr30k [15], and GQA [31] datasets.

3.3. Image Caption Fine-Tuning

In previous studies, as a fine-tuned learning Algorithm 1 for image captioning, image object tags and regional vectors for objects are received as inputs and learned. However, when learning in a conventional manner, the image caption model relies on image object tags. When the order of image object tags changes or some variations occur during inference after fine-tuned learning of image caption problems, normal captions for proper images may not be generated because they rely on image object tags.

Algorithm 1: Image Caption Fine Tuning

```
Input: Object Detection Tag T, Vector V, Random Caption Word W
Output: Image Caption
weights load Pretraining BERT Model;
for $v_i, t_i, w_i \in V, T, W$ do
    $r \leftarrow uniform(0, 1)$ if $r \geq 0.9$ then
        $t_i = random_choice(W)$
    end
    for i in generate_length do
        $w \leftarrow w_{i+1\ generate\_length} masking$
        $embedding\_words = Word\_Embedding(d_i)$
        embedding_words += position_embedding + type_embedding
        embedding_words = Dropout(Layer_norm(embedding_words))
        $embedding\_img = Dropout(Layer\_norm(linear(v_i)))$
        embedding = cat(embedding_words, embedding_img)
        for j in range(12)) do
            embedding = self_attention(embedding)
            embedding = Layer_norm(Dropout(dense(embedding + w)))
            embedding = gelu(dense(embedding))
        end
        output = tanh(dense(embedding[-1]))
        output = Layer_norm(Dropout(dense(output))) + bias
        $w_i = output$
    end
end
```

The filter caption model presented in this paper generates image captions by prioritizing the order of image object tags and area vectors by domain object dictionaries. For words in the domain object dictionary, it makes the model more focused on the domain object, leading to the creation of the desired image caption. At this time, when the model relies on the image object tag, a problem occurs in which normal tags cannot be generated. To address this problem, this paper proposes a filter captioning model, which is a method that can solve problems that depend on image caption tags of existing models and generate a variety of domain image caption information. For the implementation of image captioning during fine-tuned learning, as shown in Figure 1, if the image caption tag appears in the image caption, the order is changed with a 30% probability, and the segment embedding is changed to 2. To reduce image object tag dependence, we randomly extract image object tags with a probability of 5% and replace them in the dataset. By replacing the image object tag with one of the caption word data, the dependence on the image object tag is reduced, and the overall content of the image object tag can be described.

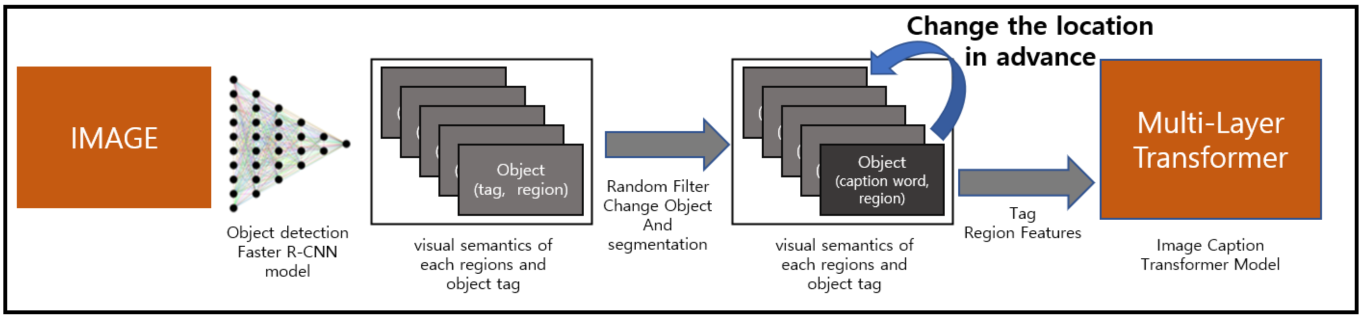


Figure 1. Image Caption It is an inference structure The Domain object dictionary re-aligns the location of the object to induce captions. for image captioning inference.

3.4. Image Caption Inference

This section explains how to build a domain object dictionary corresponding to the domain to be applied among image object tags and induce image captioning using the built domain dictionary In Algorithm 2 and Figure 2, rather than building and learning image caption data for each domain to be applied.

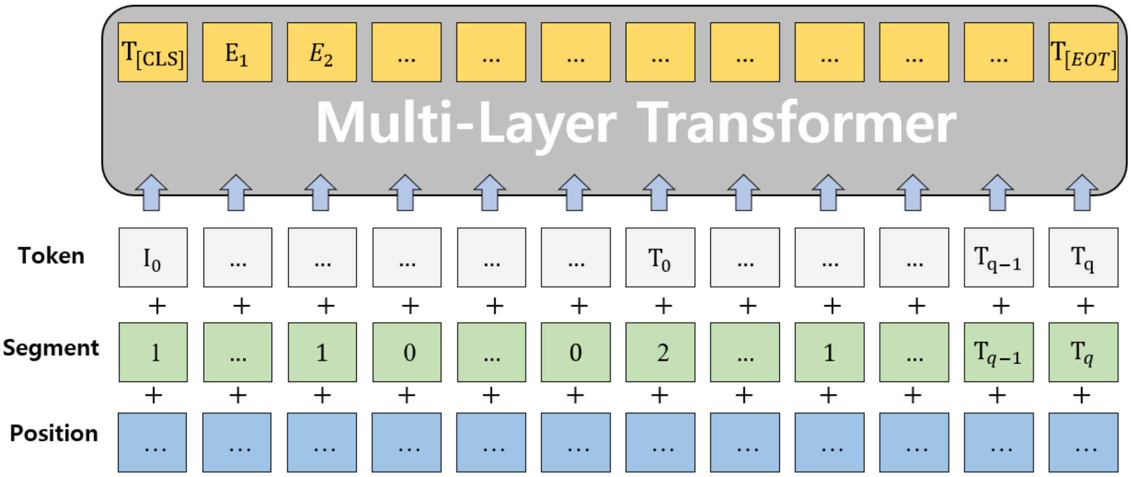


Figure 2. Image embedding I and image object tags through the CNN-based backbone network are input to the multi-layer transformer. To avoid overfitting, 30% of the tags generated and 5% of regular tags randomly change segment embedding to 2.

Algorithm 2: Image Caption Infernece

```
Output: Image Caption
weights load Pretraining Image Caption Model
for $v_i, t_i \in V, T$ do
    $target_{index} \leftarrow 0$
    for $i$ in $Len(v_i)$ do
        if $v_{i,i}$ IN caption_vocab then
            $v_{i,i}, v_{i,target\_index} \leftarrow v_{i,target\_index}, v_{i,i}$
            $t_{i,i}, t_{i,target\_index} \leftarrow t_{i,target\_index}, t_{i,i}$
            $target\_index \leftarrow target\_index + 1$
    for $i$ in $generate\_length$ do
        $w \leftarrow w_{i+1\ generate\_length} masking$
        $embedding\_words = Word\_Embedding(d_i)$
        embedding_words += position_embedding + type_embedding
        embedding_words = Dropout(Layer_norm(embedding_words))
        $embedding\_img = Dropout(Layer\_norm(linear(v_i)))$
        embedding = cat(embedding_words, embedding_img)
        for j in range(12)) do
            embedding = self_attention(embedding)
            embedding = Layer_norm(Dropout(dense(embedding + w)))
            embedding = gelu(dense(embedding))
        output = tanh(dense(embedding[-1]))
        output = Layer_norm(Dropout(dense(output))) + bias
        $w_i = output$
```

3.4.1. Domain Object Dictionary

The domain object dictionary induces the generation of captions by rearranging the order of the input data, image object tags, and local information about the image object, according to the domain, when creating an image caption. It is a method of selecting and applying tags related to the domain to be applied among image object tags without having to build all words related to the domain when creating a domain object in advance. It is easy to build a dictionary without professional domain knowledge by selecting a tag related to the desired domain among image object tags and adding it to the dictionary. When pre-built, it should select the minimum tags it wants to apply to create image captions that fit the domain. If there is no domain object dictionary and if all objects enter the domain object dictionary, there is no change in priority, which is the same as the basic model. Compared to new image captioning methods that require a lot of time and money, domain object dictionaries can be easily built.

3.4.2. Domain Object Pre-Filtering

The order of image objects and image region information entering the input of the filter captioning model is changed using the domain object dictionary built when inferring image captions. If there is a tag corresponding to the wholesale object dictionary among the ordered pairs of image object tags extracted from the image and regional information on the image object, the tag is sorted in order of its priority. At this time, in order to emphasize the objects shown in the domain object dictionary, the image object tag is copied and added to the number of repetitions using the number of repetitions parameter. By changing the order of tags rather than replacing and deleting, most of the image object tags enter the filter captioning model, and the main objects that are deleted due to objects added by the number of repetitions are deleted in order from the back. When data that do not overlap with the wholesale object dictionary occur, the general caption information is corrected rather than obtaining incorrect caption information or empty information.

4. Research Method

The method of the model proposed in this paper uses a domain object dictionary to filter the identified image objects to change the sort order of the objects. Since we have learned to look more intensively at the objects in front of us according to the order of the objects, we can create image captions suitable for the desired domain. At this time, we aim to generate image captions without processing image caption data suitable for each domain and creating new ones, making it difficult to use evaluation methods such as BLEU [1], METEOR [2], ROUGE [3], SPICE [4], and CIDEr [5] because there are no image caption data for each domain. In this paper, we learn based on the COCO captioning dataset [9] and present a review of datasets and a method of pre-creating image caption domain objects.

4.1. Dataset for Research Subjects

This paper learned using COCO captioning data [9]. COCO captioning data [9] have been widely used to solve the image captioning problem in recent years. The COCO dataset has five image captions in one image and consists of 413,915 captions and 82,783 images' learning data; 202,520 captions and 40,504 images' evaluation data; and 40,775 captions and 179,189 test data. As shown in Figure 3, the average number of words generated per caption sentence is about 10.5. Based on the space, it is viewed as a single word, and the number of words generated from the image caption learning data is 43,360. In addition, there are five image caption data in one image, and each datum is a sentence unit. Among the image caption sentences generated in one image, there are sentences in which the meaning to be described overlaps. It is difficult to show quantitative values because it is sometimes ambiguous to check each sentence with overlapping meanings in the data to determine whether it is viewed or spoken with the same meaning. In this paper, by visualizing how many overlapping words there are in five sentences that explain an image (Figure 4), we intended to use a lot of limited words and indirectly show sentences with the same meaning.

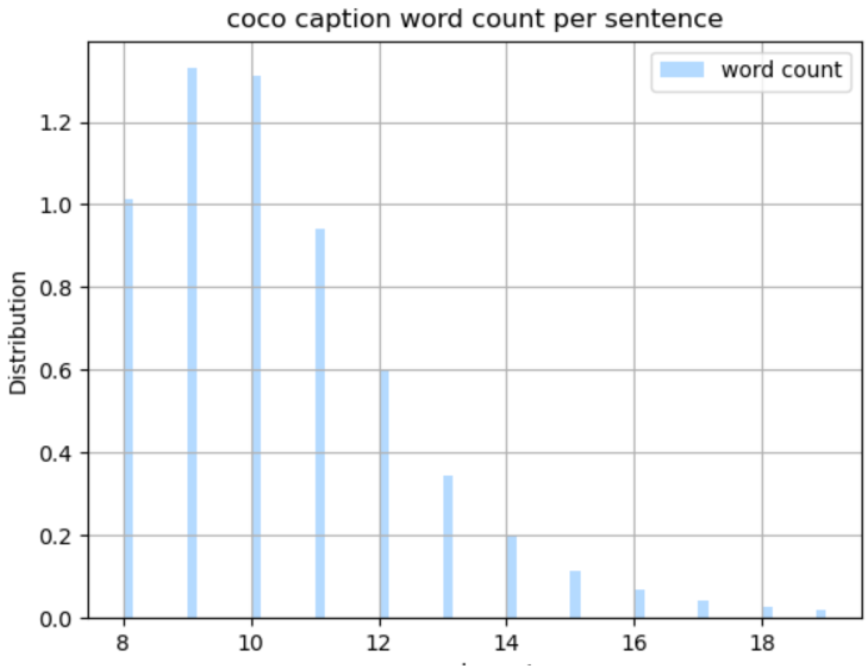


Figure 3. Chart of the number of words per sentence for the training data of the COCO captioning dataset.

4.2. Filter Captioning to Create a Fine Coordinated Learning Object Tag

Each of the five sentences describing the image in Figure 4 uses many words and various meanings when learning is fine-tuned. The filter captioning model attempts to generate image captions related to the domain while replacing image object information through a dictionary built to fit the domain. This is because there must be explanatory sentences with various meanings for one picture in the original learning data in order to create a caption suitable for the domain when inferring the caption for the image of the filter captioning model. The filter captioning model uses input data that is replaced with a 10% probability by cutting image caption data into word units during fine-tuned learning. In order to naturally generate data cut into word units, we tried to create a dictionary using the distribution of image caption words generated from the learning data. In this case, the distribution of occurrence of words generated in the image caption is very biased. For example, articles and prepositions such as 'a', 'on', 'the', 'of', 'in', 'to', and 'at' occurred 606,842, 204,585, 163,087, 147,008, 174,776, 65,083, and 40,535 times, respectively, accounting for a considerable frequency. In addition to articles and prepositions, nouns and adjectives frequently used in image captions, such as 'people', 'white', 'woman', 'table', 'street', 'person', 'top', and 'group', account for 34,704, 33,697, 30,862, 28,112, 27,870, 22,559, 21,101 and 19,687 occurrences, respectively. However, words such as 'care', 'brickwall', 'slide', 'visits', 'deal', 'life', 'ai', and 'wrapping', which are commonly used in everyday life, occur only once each in the captions of the learning data images. As shown in Figure 5, many words often occur only once, and several words occur very often. To solve this problem, fine-tuned learning of the filter captioning model uses a method of randomly extracting and selecting one of them by equalizing the probability of replacement of all words. In addition, it should be noted that among the words generated in the COCO captioning data [9], there are typos, such as when the word 'baseball' is written 'baaeball'.

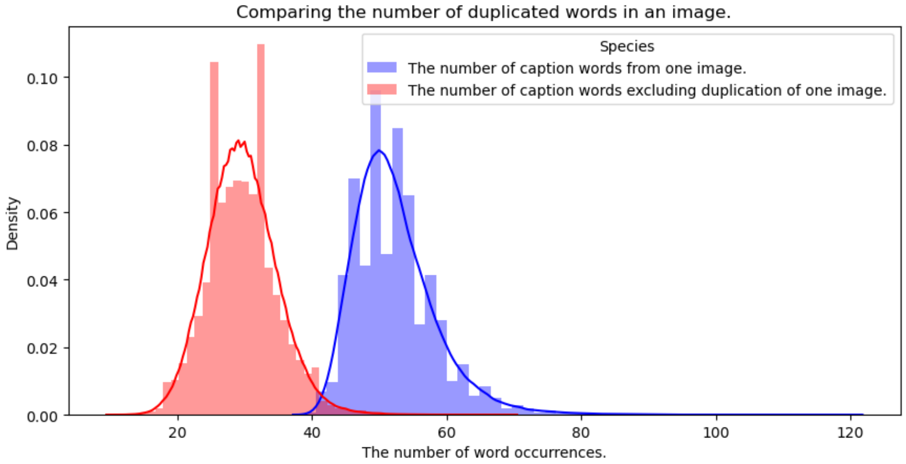


Figure 4. Comparing the de-duplication and removed word occurrence distribution of five explanatory sentences for one image.

4.3. Establishing a Domain Dictionary for Filter Captioning Inference

The filter captioning model requires a domain dictionary built for image caption inferences suitable for the domain. In general, to create an image caption suitable for a domain, image caption learning data suitable for that domain must be created. It takes a lot of time and money to make these learning data. In addition, even if data are constructed, data that are not constructed under an accurate plan may not be used for learning and may be discarded. The filter captioning model presents a solution to the above problems through the domain dictionary. Domain dictionary construction does not have to create a dictionary that includes all words related to the domain, but only extracts image object tags that are intended to be constructed among image object tags. Words with a large frequency, such as in Figure 5, are determined, and the number of object words is limited, so it is easily accessible because it does not take much time to build a domain dictionary. In this case, if the diversity of the image object tags is insufficient, difficulties may occur when constructing a domain dictionary. If necessary, more image object tag learning data can be used to newly learn the image object tag model to increase the usability of the domain dictionary.

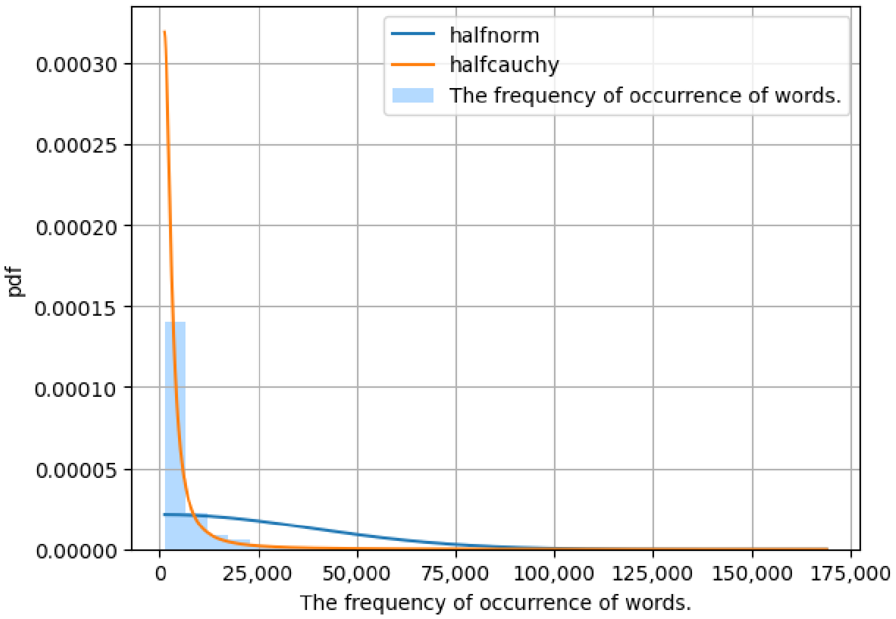


Figure 5. COCO captioning dataset: a chart comparing the frequency of word occurrence in image captions in training data.

5. Results

We propose a filter captioning model that generates image captions without new construction of learning data. Since there is no new construction of learning data, quantitative evaluation is impossible because there is no correct answer data when generating image captions built in advance from images input from the filter captioning model. When fine-tuning the filter captioning model, we evaluated the accuracy compared to the OSCAR [18] model, which is the baseline model, to determine whether the learning has been well performed. Compared to the noise-free general OSCAR [18] model of input data, we aimed to ensure that the evaluation metrics do not fall significantly.

5.1. Filter Captioning Learning and Brotherhood Indicators

In this paper, BLEU [1], METEOR [2], ROUGE [3], and CIDEr [5] are used as evaluation methods of image captions generated by the model during fine-tuned learning.

$$BLEU_k = min(1, \frac{predictsentence(wordlength)}{labelsentence(wordlength)}) * (\prod_{i=1}^{K} precision_i)^{\frac{1}{k}} \tag{3}$$

The Bilingual Evaluation Understudy Score (BLEU Score) is a measurement method created to evaluate language machine translation. This is a method of evaluating predictive sentences based on the precision regarding how many ordered pairs overlap based on n-gram. K stands for n in n-gram.

$$METEOR = \frac{10PR}{R+9P}(1-p), \qquad p = 0.5(\frac{c}{u_m})^2, \qquad P = \frac{m}{w_t}, \qquad R = \frac{m}{W_r} \tag{4}$$

Metric for Evaluation with Explicit Ordering (METEOR) is a measurement method designed to evaluate language machine translation like BLEU. Unlike BLEU, recall is also considered, and predictive sentences are evaluated based on uni-gram based on aligned text. In Equation (4), P refers to precision, m is the number of uni-grams of the correct answer sentences found in the uni-gram of the model's predicted sentence, w_t is the number of uni-grams in the predicted sentence, R is recall, and u_m is the number of uni-grams in the correct sentence. Since precision and recall describe only word matching, we use p to calculate the penalty for sorting. Finally, c is the number of adjacent mapping of sentences and correct answer sentences predicted by the number of chunks, and u_m is the number of mapped uni-grams.

$$ROUGE\_L = \frac{2PR}{P+R}, \qquad P = \frac{LCS(predict, label)}{n}, \qquad P = \frac{LCS(predict, label)}{m} \tag{5}$$

Recall-Originated Understudy for Giving Evaluation (ROUGE) is a measurement method designed to evaluate machine translation language and text summaries. This is a method of evaluating the predictive sentence based on the recall of how much the ordered pairs overlap based on n-gram. This paper describes the bi-gram-based ROUGE_L evaluation method. The ROUGE_L evaluation method measures the longest matching string using the longest common subsequence (LCS) technique. In the above formula, m and n are the length of the correct answer sentence and the length of the predicted sentence, respectively, and $prediction$ and $label$ are the predicted and correct sentences, respectively.

$$g_k(s_{ij}) = \frac{h_k(s_{ij})}{\sum_{w_j \in \Omega h_l(s_{ij})}} \log(\frac{|I|}{\sum_{I_p \in I} MIN(1, \sum_q h_k s_{pq})}) \tag{6}$$

$$CIDEr_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{g^n(c_i) \cdot g^n(s_{ij})}{\|g^n(c_i)\| \|g^n(s_{ij})\|} \tag{7}$$

Consensus-based Image Description Evaluation (CIDEr) is a measurement method designed to evaluate sentences made for image descriptions. It uses a method of determining whether one sentence is similar to another in generating an absolute indicator for evaluating a sentence. For the comparison of sentences, all words in the sentence are mapped with a word root and a word stem, where each sentence is set with n-gram. Since the n-gram may have little information, the frequency inverse document frequency (TF-IDF) [32] weight is determined in Equation (6) to lower the weight. Here, Ω is an n-gram vocabulary and l is all image datasets. Additionally, w_k is the TF-IDF weight calculated by referring to the number of each image set $|l|$.

$$CIDEr_n(c_i, S_i) = \sum_{n=1}^{N} w_n CIDEr_n(c_i, S_i) \tag{8}$$

Here, the weight for the n-gram of n is calculated in Equation (7) using cosine similarity, and the vector denominator c_i of g^n corresponds to all g_k of length n in size. Using this value, the CIDEr value is calculated in Equation (8), where w_n is $1/N$ divided by the weight, and N is 4.

5.2. Filter Captioning Model Training

In this paper, we learned using RTX 3090 TI GPU and, in terms of case, it was replaced with lowercase letters. The learning batch size is 2, where the BERT model learns and compares the filter captioning model and the existing model up to 180,000 steps using freezing embedding. As described above, the filter captioning model does not submit to performance evaluation because there is no correct answer data suitable for the domain. The captioning model learns by including additional noise in the input data compared to the existing model during fine-tuned learning. Therefore, it aims to maintain similar scores compared to the accuracy of the existing model.

5.3. Filter Captioning Model Results

Table 2 shows the results of fine-tuning the filter captioning model, and Table 3 displays the results of fine-tuning the existing model. Each model was trained up to 180,000 steps and evaluated in units of 10,000 steps. In order to compare the results of the filter captioning model and the existing model, they are visualized and compared for each evaluation index. Figure 6 includes charts that visualize filter captioning models by score, and Tables 2 and 3 show the results of fine-tuning of existing models. Both the existing model and the filter captioning model showed unstable scores in BLEU [1], METEOR [2], ROUGE [3], and CIDEr [5] at the beginning of learning and similar scores after 60,000 steps. The filter captioning model is well learned.

Table 2. The base model is the result of evaluating Bleu_1, Bleu_2, Bleu_3, METEOR, ROUGE_L, and CIDEr scores every 10,000 steps up to 180,000 steps.

STEPS	BLEU_1	BLEU_2	BLEU_3	METEOR	ROUGE_L	CIDEr
10,000	0.42127	0.20266	0.11601	0.12247	0.33375	0.09163
20,000	0.71462	0.52953	0.36954	0.23072	0.50887	0.89150
30,000	0.70895	0.52509	0.36691	0.23589	0.50220	0.91953
40,000	0.73438	0.55464	0.39340	0.24312	0.51699	0.96477
50,000	0.73607	0.55556	0.39715	0.24187	0.52158	0.97054
60,000	0.72510	0.54554	0.39294	0.23722	0.51872	0.95560
70,000	0.73662	0.55446	0.39710	0.24398	0.52163	0.98510
80,000	0.74010	0.56100	0.39824	0.24772	0.52673	0.99428
90,000	0.74544	0.56322	0.40071	0.24576	0.52677	0.99567
100,000	0.74129	0.56459	0.40614	0.24620	0.52770	1.00714
110,000	0.73404	0.56014	0.40337	0.24641	0.52487	1.00562
120,000	0.73852	0.56181	0.40515	0.24865	0.52758	1.01481
130,000	0.74709	0.56888	0.40597	0.24676	0.52776	1.00650
140,000	0.74031	0.56349	0.40255	0.25084	0.52789	1.01821
150,000	0.74121	0.56514	0.40694	0.24896	0.52990	1.00385
160,000	0.74501	0.56633	0.4044	0.24766	0.52776	1.00549
170,000	0.74086	0.56254	0.39975	0.24765	0.52614	0.98849
180,000	0.70593	0.53553	0.38207	0.23373	0.49929	0.95181

Table 3. The filter captioning model is the result of evaluating Bleu_1, Bleu_2, Bleu_3, METEOR, ROUGE_L, and CIDEr scores every 10,000 steps up to 180,000 steps.

STEPS	BLEU_1	BLEU_2	BLEU_3	METEOR	ROUGE_L	CIDEr
10,000	0.46965	0.26674	0.13260	0.11563	0.35089	0.06661
20,000	0.00011	0.00000	0.00000	0.00008	0.00015	0.00001
30,000	0.00000	0.00000	0.00000	0.00001	0.00000	0.00000
40,000	0.70185	0.51810	0.35942	0.22918	0.50381	0.88718
50,000	0.72191	0.54263	0.38456	0.23680	0.51336	0.94791
60,000	0.72714	0.54356	0.38166	0.23899	0.51698	0.95379
70,000	0.73415	0.55288	0.39198	0.24205	0.51970	0.97153
80,000	0.73601	0.55712	0.39863	0.24615	0.52486	0.98178
90,000	0.73091	0.55250	0.39219	0.24363	0.52348	0.98480
100,000	0.72987	0.54703	0.38805	0.24623	0.52281	0.98792
110,000	0.74525	0.56238	0.39909	0.24368	0.52491	0.98655
120,000	0.73675	0.55613	0.39652	0.24493	0.52313	0.99187
130,000	0.74340	0.56317	0.39923	0.24049	0.52366	0.98576
140,000	0.73526	0.55514	0.39534	0.24632	0.52502	0.99858
150,000	0.73647	0.56033	0.39929	0.24671	0.52459	1.00759
160,000	0.73844	0.55914	0.39964	0.24705	0.52253	1.00374
170,000	0.74422	0.56506	0.40350	0.24360	0.52643	1.00244
180,000	0.73687	0.55853	0.39854	0.24690	0.52613	1.00990

Existing models and filter captioning models use self-critical sequence training (SCST) [22] techniques to learn models using errors in CIDEr-D [5] scores between the model inference results and correct answer data among model pre-learned image object tags and objects. Table 4 is an example of generating image captions for filter captioning models for people with impaired vision compared to the basic model. The figure on the left describes a relatively dangerous bus between people and buses. The figure in the center describes many motorcycles in the upper left, not the motorcycle in front passing by. The figure on the right comprehensively explains the situation for the person in front and the train. Table 5 is an example of image captioning in Korean. Simply, it can be seen that various texts are generated according to one object dictionary registration.

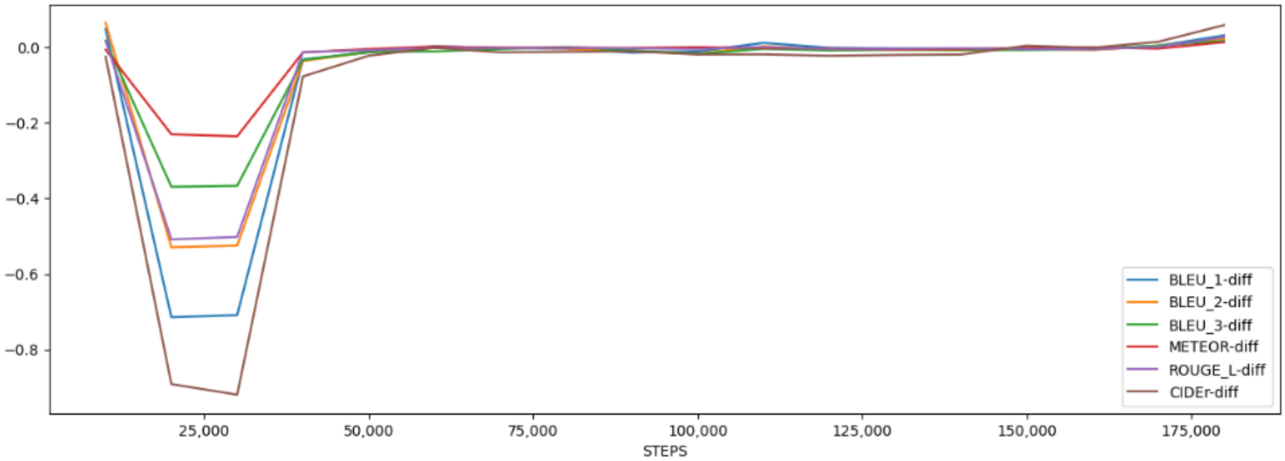
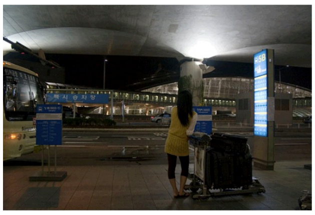
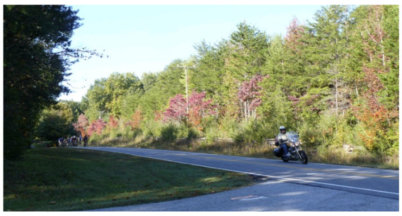
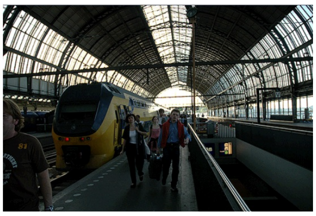


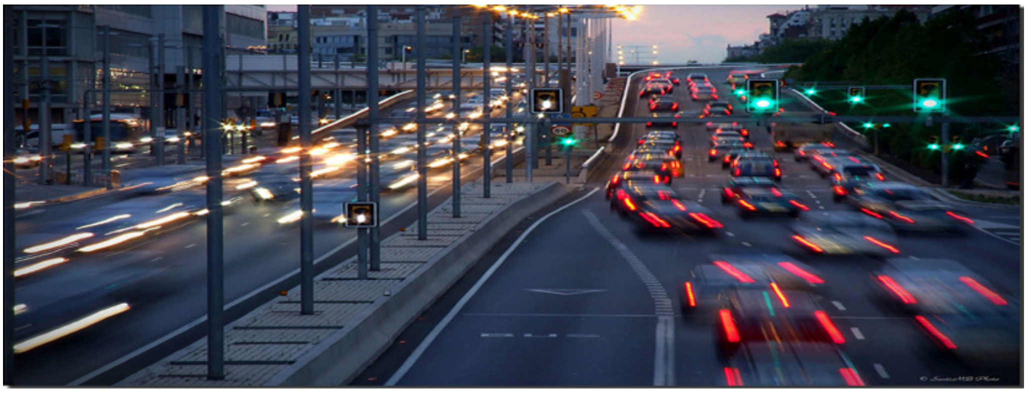
Figure 6. To show the difference in performance from the base model, visualization of BLEU, METEOR, ROUGE_L, and CIDEr score is shown by subtracting the filter captioning score from the default model score.

Table 4. This is an example of applying the filter captioning Model by constructing a domain dictionary for the visually impaired. The default model in the left picture depicts a person, but the filter captioning model depicts a relatively dangerous bus, and the central picture describes not only one motorcycle passing by, but also many motorcycles coming from the back of the road. The picture on the right comprehensively explains the situation of people and trains.

		Default Model	Filter Captioning Model
Image Caption	Left	a woman is standing in front of a bus	a bus is parked on the side of a street
	Middle	a motorcycle parked on the side of a street	a group of people walking down a street with a road.
	right	a group of people standing next to a train station	a group of people walking down a street with a train
Changed object tag	Left	sidewalk, bus, building, car, street, pole, sign	
	Middle	trees, tree, person, road, street, people, line, pole	
	right	street, people, sign, wall, man, train, pole, person, line	
Domain dictionary	bus, train, road, sidewalk, sign, person, building, pole, door, wall, man, people, cars, car, street, trees, line, tree		

Table 5. Korean image captioning examples of various image captions generated according to object dictionary in one picture. We show that image captions are generated differently for each dictionary change on the left.



Object Dictionary	Image Caption
-	밤에 길을 따라 운전하는 교통 신호등
보도 or 도시	밤에 도시 거리를 따라 운전하는 한 무리의 사람들
도로	자동차와 신호등이 있는 도시의 거리.
건물	고층 건물들로 가득 찬 거리.
Image Objects	폴, 빛, 도로, 교통, 불, 건물, 도시, 거리, 나무, 보도, 숫자, 하늘

6. Conclusions

In this paper, COCO captioning data [9] were used to solve the image caption problem. Based on the BERT [6] model, the OSCAR [18] model that learned the image COCO captioning data [9] studied is used as the default model. The filter captioning model was proposed using the OSCAR [18] model, and the filter captioning model was verified by comparing the BLEU [1], METEOR [2], ROUGE [3], and CIDEr [5] indicators during fine-tuned learning for image captions. In the field of image captioning, when we want to create an image caption suitable for the specific domain, that is, we want to use for real services, it must process image caption data suitable for the new domain for the input image every time. This paper proposes a filter captioning model that simply generates image captions suitable for the domain by processing the domain dictionary of the image object tag without processing new image caption data suitable for each domain. This can save a lot of time and money by processing image caption data suitable for the domain. In addition, in this paper, the flow of image caption research and the latest research were analyzed and presented. The domain dictionary method used in this paper relies on the object detection tag used for pre-training. If the object required by the domain to be created is not in the label of object detection, it cannot be added to the domain dictionary. Research such as expansion into an object detection model, including various object tags, is needed. In addition, when these various image captions are generated, there is a need for an indicator study to evaluate whether image captions have been created for the domain.

Author Contributions: Methodology, software, Writing—original draft preparation S.C.; Writing—review H.O.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. NRF-2022R1F1A1074696).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: "COCO captioning dataset" at https://cocodataset.org/#home, accessed on 1 April 2022.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–331.
2. Banerjee, S.; Lavie, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Michigan, MI, USA, 25–30 June 2005 ; pp. 65–72.
3. Lin, C. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*; Association for Computational Linguistics: Barcelona, Spain, 2004; pp. 74–81.
4. Anderson, P.; Fernando, B.; Johnson, M.; Gould, S. SPICE: Semantic Propositional Image Caption Evaluation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Amsterdam, The Netherlands, 2016.
5. Vedantam, R.; Zitnick, C.L.; Parikh, D. CIDEr: Consensus-Based Image Description Evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4566–4575.
6. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
7. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
8. Sidorov, O.; Hu, R.; Rohrbach, M.; Singh, A. TextCaps: A Dataset for Image Captioning with Reading Comprehension. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Amsterdam, The Netherlands, 2020; pp. 742–758.
9. Chen, X.; Fang, H.; Lin, T.Y.; Vedantam, R.; Gupta, S.; Dollár, P.; Zitnick, C.L. Microsoft COCO Captions: Data Collection and Evaluation Server. *arXiv* **2015**, arXiv:1504.00325.
10. Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; Rohrbach, M. Towards VQA Models That Can Read. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8317–8326.
11. Kiela, D.; Firooz, H.; Mohan, A.; Goswami, V.; Singh, A.; Ringshia, P.; Testuggine, D. The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 6–12 December 2020; Volume 33, pp. 2611–2624.
12. S. Cho; H. Oh. A general-purpose model capable of image captioning in Korean and English and a method to generate text suitable for the purpose VIVO: Visual Vocabulary Pre-Training for Novel Object Captioning. *J. Korea Inst. Inf. Commun. Eng.* **2022**, *26*, 1111–1120.
13. Hu, X.; Yin, X.; Lin, K.; Wang, L.; Zhang, L.; Gao, J.; Liu, Z. VIVO: Visual Vocabulary Pre-Training for Novel Object Captioning. In Proceedings of the AAAI Conference on Artificial Intelligence, Palo Alto, CA, USA, 2–9 February 2021; pp. 1575–1583
14. Young, P.; Lai, A.; Hodosh, M.; Hockenmaier, J. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. In *Transactions of the Association for Computational Linguistics*; MIT Press: Cambridge, MA, USA, 2014; Volume 2, pp. 67–78.
15. Agrawal, H.; Desai, K.; Wang, Y.; Chen, X.; Jain, R.; Johnson, M.; Batra, D.; Parikh, D.; Lee, S.; Anderson, P. nocaps: Novel object captioning at scale. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8948–8957.
16. Sharma, P.; Ding, N.; Goodman, S.; Soricut, R. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 2556–2565.
17. Ordonez, V.; Kulkarni, G.; Berg, T. Im2text: Describing Images Using 1 Million Captioned Photographs. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 12–15 December 2011; Volume 24, pp. 1143–1151.
18. Li, X.; Yin, X.; Li, C.; Zhang, P.; Hu, X.; Zhang, L.; Wang, L.; Hu, H.; Dong, L.; Wei, F.; Choi, Y.; Gao, J. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Amsterdam, The Netherlands, 2020; pp. 121–137
19. Luo, R.; Shakhnarovich, G. Controlling Length in Image Captioning. *arXiv* **2005**, arXiv:2005.14386.
20. Wu, Y.; Jiang, L.; Yang, Y. Switchable Novel Object Captioner. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 1162–1173. [CrossRef] [PubMed]
21. Shi, B.; Bai, X.; Yao, C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2298–2304. [CrossRef] [PubMed]
22. S. Rennie, J.; Marcheret, E.; Mroueh, Y.; Ross, J.; Goel, V. Self-Critical Sequence Training for Image Captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7008–7024.
23. Du, Y.; Li, C.; Guo, R.; Yin, X.; Liu, W.; Zhou, J.; Bai, Y.; Yu, Z.; Yang, Y.; Dang, Q.; Wang, H. PP-OCR: A Practical Ultra Lightweight OCR System. *arXiv* **2020**, arXiv:2009.09941.
24. Zhang, P.; Li, X.; Hu, X.; Yang, J.; Zhang, L.; Wang, L.; Choi, Y.; Gao, J. Vinvl: Revisiting Visual Representations in Vision-Language Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 20–25 June 2021; pp. 5579–5588.

25. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
26. Ma, N.; Zhang, X.; Zheng, H.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
27. Howard, A.; Sandler, M.; Chu, G.; Chen, L.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
28. Yu, D.; Li, X.; Zhang, C.; Liu, T.; Han, J.; Liu, J.; Ding, E. Towards Accurate Scene Text Recognition with Semantic Reasoning Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12113–12122.
29. Clark, K.; Luong, M.T.; Le Q.V.; Manning, C.D. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv* **2020**, arXiv:2003.10555.
30. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv* **2019**, arXiv:1910.03771.
31. Hudson, D.A.; Manning, C.D. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6700–6709.
32. Manning, C.D.; Raghavan, P.; Schütze, H. Scoring, term weighting and the vector space model. *Introd. Inf. Retr.* **2008**, *100*, 2–4.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.