Using a Novel Capsule Network for an Innovative Approach to Image Captioning

Shima Javanmardi[1,*], Mehrdad Jahanbanifard[1], Marcello Bonsangue[1] and Fons J. Verbeek[1]

[1]Leiden Institute of Advanced Computer Science (LIACS), Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands

Abstract

Zebrafish is a popular model system for biomedical analysis, especially for compound screening in drug research. In this paper, we present a comprehensive investigation aimed at enhancing the processing pipeline for segmenting zebrafish larvae images. The emphasis is on the application of an unsupervised segmentation method for segmenting zebrafish in Optical Projection Tomography (OPT) images. We propose a novel pipeline that integrates the Transformer and U-Net, a convolutional neural network for bio-medical image segmentation, to achieve accurate segmentation of zebrafish larvae images. This accuracy is critical for precise 3D reconstruction. Leveraging transfer learning, we broaden the capabilities of our trained model to segment OPT images. This approach is intended to enhance the robustness and versatility of our pipeline, allowing it to cater to a broad range of imaging modalities beyond traditional microscopic images. The developed processing pipeline is then used for 3D reconstruction of the segmented areas, demonstrating its potential for advanced biomedical analysis. Our findings confirm the efficiency and accuracy of the proposed pipeline providing robust tools for future Zebrafish-based research, particularly in the domains of drug screening and cancer treatment.

Keywords

Machine Learning, Deep Learning, Image Segmentation, Transformer, Transfer Learning, 3D Reconstruction,

1. Introduction

Automatic image captioning is a challenging problem in computer vision, and it aims to generate rich content and human-understandable descriptions for given images [1]. The performance of image captioning models is closely related to the quality of extracted features from images. The power of the language model can help to generate accurate and meaningful descriptions related to image content. Considering the semantic relationships between the identified objects within the image is essential in the image caption generation task. However, identifying the objects (i.e., the nouns in the caption) within an image is still challenging. Moreover, finding their interaction (i.e., the verbs in the caption) is extremely difficult. In fact, ex-pressing object interaction by natural language as semantic knowledge, either as verbs or adverbial compositions, is the core issue in image captioning.

In this paper, we develop a novel method that (1) overcomes the limitations of CNNs, (2) generates descriptions with a non-restricted variety of words, and (3) is capable of describing the relationships between the objects. We use a novel encoder–decoder mechanism that addresses these challenges by using a capsule network (CapsNet) [2]. The result is a set of meaningful descriptions for the image via a language model. CapsNet can effectively compensate for the shortcomings of a CNN by detecting tissue overlap characteristics [3]. In CapsNet, more salient spatial features and geometrical attributes, such as direction, size, scale, and object attributions, can be represented for each input. This aspect of CapsNet contrasts with CNN since the lack of local invariance features produces excessive variations of global discriminating outputs [4]. In addition, our model employs an external knowledge base, i.e., Wikipedia, aiming to accomplish augmented textual training data to generate more meaningful and diverse captions.

The main contributions of our work are as follows:

- The development of a novel parallel structure for a capsule network can capture more comprehensive information about the objects within an image by considering their relationships.
- The use of Wikipedia as an external knowledge base for enrichment of all the textual training information and generating out-of-domain representation when describing the content of the image.
- The application of our framework on the MS-COCO large-scale dataset. Using large-scale datasets including RGB images requires a huge number of resources because of the architecture of capsule networks.
- We performed a bench-marking towards a list of existing state-of-the-art models.

The Third AAAI Workshop on Scientific Document Understanding, 2023, , DC, USA

*Corresponding author.

s.javanmardi@liacs.leidenuniv.nl (S. Javanmardi); m.jahanbanifard@liacs.leidenuniv.nl (M. Jahanbanifard); m.m.bonsangue@liacs.leidenuniv.nl (M. Bonsangue); f.j.verbeek@liacs.leidenuniv.nl (F. J. Verbeek)

[0000-0002-3027-5895] (S. Javanmardi); [0000-0002-2224-8387] (M. Jahanbanifard); [0000-0003-3746-3618] (M. Bonsangue); [0000-0003-2445-8158] (F. J. Verbeek)

In the next section, we will consider state-of-the-art approaches.

2. Related Work

Image captioning is a popular research topic in computer vision and natural language processing. Generating an accurate textual explanation that describes the content of an image is accomplished by understanding the visual content of the image. Recently, the interest in image captioning has broadened with the development of benchmark datasets such as MS-COCO [5], Flickr 8K [6], and Flickr 30K [7].

Current image captioning models can be categorized into template-based, retrieval-based, and neural network-based models. The template-based models [8], first detect all the image attributes using image classification and object detection methods. These methods generate captions by filling in predefined templates from the identified objects. This approach produces too flexible captions that cannot correctly describe the relationships between attributes [9]. Retrieval-based models [10] create a pool of similar images in an image database rank the retrieved images by measuring their similarities, and then change the found image descriptions to create a new description for the queried image. The usefulness of this strategy is severely constrained when dealing with images that are not in the dataset and thus not classified, i.e., unseen.

The neural network-based models are inspired by the success of deep neural networks in machine learning tasks and used in an encoder–decoder architecture [11]. An encoder extracts image contents by a CNN, a module associates contents to words, and a decoder by an RNN is used for language modeling and creating image captions. Liu et al. [12] proposed an ontology to describe the scene construction of images. Their constructed ontology can specify the object types and the special information for the objects (e.g., location, velocity). This visual and special information can be transformed into meaningful project information for generating captions using integrated computer vision and linguistic models. In [13], authors demonstrated that a large amount of data could lead to lower estimation variance and hence lower error with better prediction performance. However, data quality plays an important role in the performance of the model. The hypothesis is that more data may contain useful information. To this aim, Hossain et al. [14], proposed a method that leverages a combination of real and synthetic data generated by the Generative Adversarial Network (GAN). It is an efficient alternative for the techniques requiring human-annotated images, as they are labor-intensive to generate and time-consuming.

Various improvements are made to captioning models to make the network more inventive and effective by considering visual and semantic attention to the image. For example, Yang and Liu [12], introduced a method called ATT-BM-SOM to increase the readability of the syntax and optimize the syntactic structure of captions. This framework operates based on the attention balance mechanism and the syntax optimization module and effectively fuses image information. Their model generates high-quality captions, compensating for the lack of image information selection and syntax readability. In the next section, the structure of the image caption generation models and the employed networks in our experiments will be discussed in more detail.

3. Materials and Image Captioning Methods

Following the trend of current work, we use an encoder–decoder framework to create the captions of images. Understanding the image requires recognizing the objects, properties, and interactions in the encoder part. Moreover, producing sentences to describe images in the decoder requires understanding language syntax and semantics.

Figure 1 illustrates the employed Knowledge Discovery Database (KDD) of our model: images and descriptions proceed separately in the data processing phase. Then in the transformation phase, all the image and text data are processed to create feature vectors for the language model. A CNN is employed for predicting the labels from the given image.

In the text enrichment phase, we used Wikipedia to extract relevant information based on the predicted labels of images. Then, all the data sequences are fed to the language model in the NLP phase for tokenizing, embedding, and making word vectors from the image captions in the dataset and extracted knowledge from Wikipedia. After which, all the information is fed into the caption predictor in the evaluation section to produce a caption given the input image.

The novelty of our work consists of a new variant of the capsule network, parallelizing its basic structure to capture more comprehensive information about the objects within the image, thus leading to a more accurate description of the input image. The primary structure of capsule network works well on a simple dataset such as MNIST, which includes images with a single object and only one channel. However, the network efficiency significantly decreases when applied to images with large special dimensions and complex datasets such as MS-COCO and Flicker.

The presence of multiple channels and objects in the images increases the training time of the network and leads to weak results compared to state-of-the-art [15]. This problem happens due to inefficiency in capturing

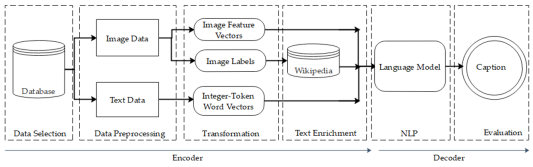

Figure 1: KDD methodology of the proposed model.

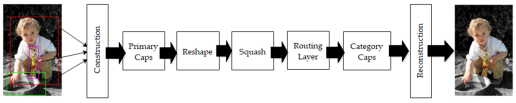

Figure 2: Capsule Network Architecture.

the underlying information of the image. To handle this issue, we extended the baseline network by parallelizing the convolutional layers and the primary capsules of the original CapsNet, followed by a concatenation approach to extract more complex and qualified features from the images. On the other hand, parallelizing the convolution layers reduces the dimensions of the fed features to the primary capsules and accelerates the learning process.

In the proposed image captioning model, we use CNN and CapsNet architectures to incorporate visual context from an image, which is then used as the input of a machine translation, such as an RNN architecture, to generate objective sentences in the decoder part of the framework. We applied cross-entropy loss to adjust the model weights during the sequential model training. In this section, the entire model flow is described in more detail. We have applied both the Inception-V3 or VGG16 as image feature extractors. These networks are trained on the ImageNet dataset with more than one million images of 1000 classes. Training the CapsNet is done from scratch and based on 80 categories of objects in Category Caps. The details of these networks are shown in Table1.

Table 1
Specific parameters of the models in the evaluation

Parameters	VGG-16	Inc-v3	CapsNet
Depth	16	48	8
Image size (px)	224 × 224	299 × 299	299 × 299
Solver (opt.)	SGD+M	RMSProp	ADAM
Loss func.	cross-ent.	cross-ent.	MSE
Batch size	32	64	128
Learn rate	0.001	0.0001	0.001
Learn rate drop factor	0.1	0.1	0.5
Learn rate drop period	10	10	10
Momentum	0.9	0.9	0.9
Gradient thresh.	L2-norm	L2-norm	L2-norm

3.1. Capsule Network

A capsule is a set of neurons whose activity vectors indicate the posture characteristics of an entity and the length of the vector denotes the chance of that entity existing. Unlike a convolutional network, capsules save comprehensive information about the location and pose of an entity.

Sabour et al. [2], claimed that regardless of the high capability of CNNs, this network has two main disadvantages: 1) lack of rotation invariant and 2- using a pooling layer. The former causes failure in recognizing spatial relations between the objects, and the latter causes information loss due to the maximum value selection of each region. Therefore Sabour et al. [2], proposed a capsule network to address the issues mentioned above.

There are different concrete components in a capsule net-work for learning the semantic representations within the image (see Figure 2) These components map construction by reconstructing the discrepancy map from the input image.

The major components of the capsule network involve the following:

- Primary capsules combine the features extracted by convolutional layers in the construction phase.
- Reshaping the extracted feature maps from the primary capsules.
- Squashing is a non-linear activation function that squashes the weighted input vector of a particular capsule. This function distributes the length of the output vector be-tween 0 and 1.
- The dynamic routing layer produces output capsules with high agreements by automatically grouping input capsules. The pooling layers in the capsule network are re-placed by a mechanism called “routing by agreement” in the rooting layer: the output of each capsule in the lower level is sent to the parent capsules in the higher level only if their features have a dependency.
- Category capsules with a marginal classification loss and a reconstruction sub-network with a reconstruction loss for recovering the original image from capsule representations.

The operation of all these components is explained in this section in more detail. One important aspect of capsule networks is their ability to identify individual parts of objects in a single image and then represent spatial relation-ships between those parts. For example, in figure 2, the CapsNet has identified three different parts

of objects with-in the input image (tie, child, bin). The output image on the right side of the figure 2 shows the result of the reconstruction sub-network in the employed capsule network. Figure 3 shows the construction of a capsule and how data is routed between lower-level and higher-level capsules.

In Figure 3a, each capsule finds the appropriate parent in the next layer during the dynamic routing procedure to send its output to those capsules in the above layer. The input and output of capsules are vectors. Given u_i as the prediction vector of capsule i and $u_{j|i}$ as the output of parent capsule j in higher level will be computed by multiplying u_i with a weighted matrix W_{ij}:

$$\hat{u}_{j|i} = W_{ij} \cdot u_i \tag{1}$$

The length of u_i, indicates the probability of predicting a component in the image even after changing the viewing angle. The direction of u_i represents several properties of that component, such as size and position. A weighted sum over all $u_{j|i}$, and an intermediate coupling coefficient c_{ij}, is calculated as the total input vector to capsule j by the following function:

$$s_j = \sum_i c_{ij} u_{j|i} \tag{2}$$

Here, the coupling coefficient c_{ij}, are the class-specific likelihood calculated after flattening the vectors and is computed by a routing Soft-Max function as follows:

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \tag{3}$$

where b_{ij} represents the log probability of connection between capsules i and j. As shown in Figure 3b, the value of c_{ij} increases when the lower-level and higher-level capsules are consistent with their predictions and decreases when they are inconsistent. Based on the original paper, this parameter is initialized at 0 in the routing by agreement procedure. Instead of applying the ReLU activation function as in VGG16 and Inception-v3, the following non-linear squashing function will be calculated over the input vector in this network:

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|} \tag{4}$$

where s_j is the input vector and v_j is the normalized output between 0 and 1. The log probability is updated along with the routing mechanism by calculating the agreement between v_i as the output of capsule j in the above layer and $u_{j|i}$, as a prediction vector.

The loss function of the network for each capsule k is computed as follows:

$$L_k = T_k \max(0, l^+ - \|v_k\|)^2 + \lambda(1 - T_k) \max(0, \|v_k\| - l^-)^2 \tag{5}$$

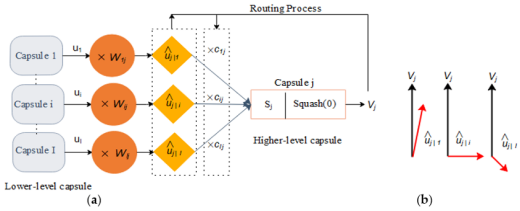

Figure 3: Transferring information among the capsules from [1...I] and high-level capsules (b) routing procedure.

where L_k is loss term for one prediction, T_k is a term equal to 1 when the class k is present; otherwise, it is 0. The upper and lower bounds of margin loss parameters, l^+ and l^-, are set to 0.9 and 0.1 [2]. It means that if an entity is present with a probability above 0.9, the loss is zero; otherwise, the loss is not zero. Regarding capsules that could not predict the correct label, if the predicted probability of all those labels is below 0.1, the margin loss is zero; otherwise, it is not zero. The parameter λ is set at 0.5 and is used for numerical stability to control the down weighting of the initial weights for the absent classes. $\|\cdot\|$ in all the equations denotes $L2$ norm.

3.2. Improved capsule network

In the improved version of the capsule network architecture, where we parallelized the convolution layers and primary capsules, the input image size is $229 \times 229 \times 3$. The different architecture of the capsule network distinguishes it compared to CNN. Except for the input and out-put layers, the capsule network consists of primary and category capsule layers. The output of the capsules is forward-ed to the decoder. The networks prevent overfitting by re-building the input image from the output capsules by minimizing the reconstruction loss as a regularization method in the decoder [16].

The original capsule network has been tested on the MNIST dataset with one color channel (grayscale). However, the color of objects is an important factor in object detection and image captioning tasks. Therefore, we propose a parallelized capsule network that generates the descriptions of the images by passing the RGB images with three color channels through the three blocks of parallel convolutional layers and parallel primary capsules. The three-color channels of RGB images can store information and intuitively visualize content. Therefore, color analysis is also addressed in this parallelized structure of the capsule net-work, which makes the model more informative and improves the descriptiveness of image captions by extracting more qualified features from the image [17]. Adding more convolutional layers was not logical due to the increasing model complexity computa-

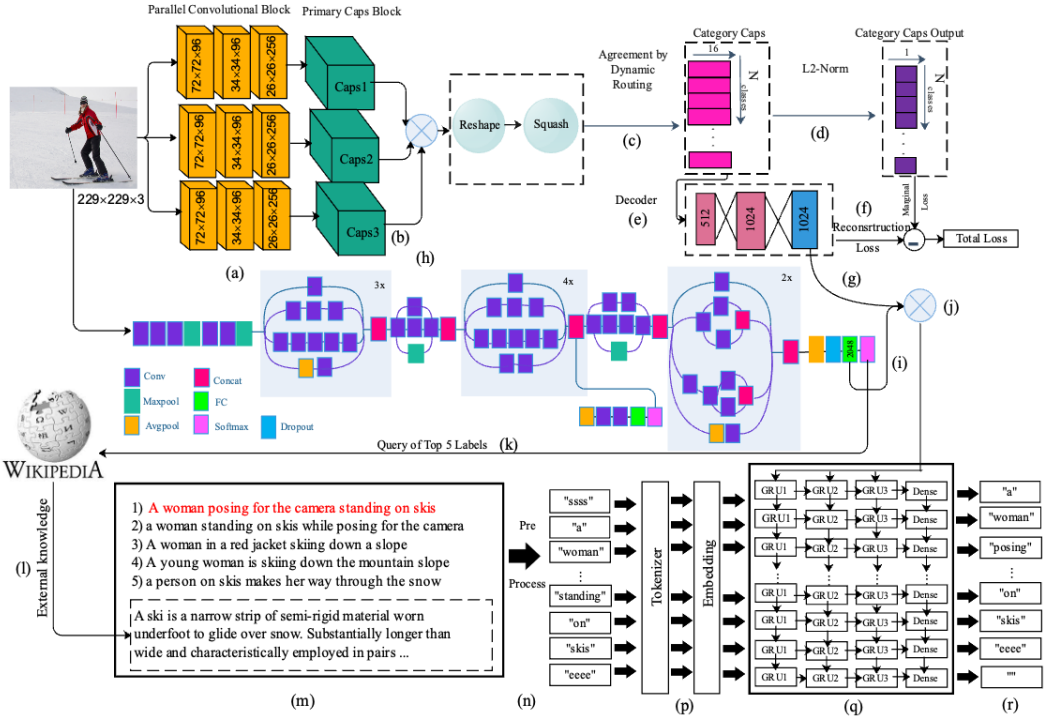

Figure 4: Our proposed model: a CNN and a CapsNet are applied to a given image to produce the visual features and predict the attributes of the image (a–k). The textual information of each sample comprises the descriptions of the image and the aggregated data from the external database, and a preprocessed method is applied to the text (l–n). After tokenizing and embedding process, the visual attention of the image is fed to a GRU with three levels to generate a caption to explain the content of the image (p–r) [9].

tional cost. The structure of the new network has been presented in figure 4.

The model steps in Figure 4 are summarized as follows:

1. Partitioning the image set into train, validation, and test subsets randomly
2. Applying image feature extractor models to extract visual features from the images (Figure 4a–j)
3. Extracting external knowledge for each image by searching the predicted labels from the previous step as a query in Wikipedia and adding it to the captions that already exist for the images in the dataset (Figure 4k–m)
4. Applying preprocessing methods to contextual data before feeding it to the RNN network, i.e., removing the punctuation numbers and wrapping each sentence around with “ssss” and “eeee” tokens to specify the beginning and end of sentences for the network (Figure 4n)
5. Transforming the textual features to the integers vector by tokenizing and embedding operations for training by the language model (Figure 4p)
6. Training language model for certain epochs based on its performance on validation data. During the training phase, the model predicts the next word of each word in the caption (Figure 4q,r)

After the training phase, the model is ready to evaluate test set images by extracting visual features and predicting the captions using a greedy search. Greedy search selects the word with the highest probability at each time step and uses it as the GRU input for the following time step until the end of the sentence is reached. In the next section, we will dis-cuss the details of the experiments and the obtained results by the analyzed methods.

3.3. Gated Recurrent Unit

Our image captioning framework used a three-layer RNN network with a Gated Recurrent Unit cell (Chung et al., 2014). This RNN is equipped with visual features in the feature maps of CNN and CapsNet. The proposed model generates a description for each image by maximizing the

probability of the current word predicted in the caption according to the following formula:

$$\theta^* = \arg\max_{\theta} \sum_{(I,M)} \log p(M|I;\theta) \tag{6}$$

where θ are the parameters of the proposed model and M is the correct description of image I. Suppose $\{m_0, \ldots, m_{N-1}\}$ is a sequence of words in transcription M of length N, then $\log p(M|I)$ as the probability of generating a word for an image I, is as follows:

$$\log p(M|I) = \sum_{t=0}^{N} \log p(m_t|I, m_0, \ldots, m_{N-1}, c_t) \tag{7}$$

where t is the time step and c_t is context vector. A two-step process feeds all the text data to the RNN network. The first step is tokenizing, and the second one is embedding. All the words in the sentences are converted into so-called integer token vectors during tokenizing. This process is based on 10,000 most frequent and unique words in the image captions.

4. Experiments

This section reports the details of implementations and the results of the experiments conducted by different variations of models.

4.1. Dataset and Implementation Details

We use the MS-COCO dataset [5], to evaluate the proposed model in our experiments. MS-COCO contains 123,287 k images with five captions and 80 object categories for each image annotated by Amazon Mechanical Turk (AMT) workers. Since there are no available annotations for the test set, in this work, we used publicly available splits provided by Karpathy et al. [18]. We use 5000 images for validation and testing and the rest for the training set. All the models are implemented in Python version 3.6 and using the capabilities provided by Keras version 2.2.5 and TensorFlow version 1.15.0 deep learning libraries. Table 1 shows the parameters set for each network. The training was done using a machine equipped with two GeForce RTX 2080 GPU cards with 8 GB memory. The machine was installed with two GPUs, but for the experiments, only one was necessary.

4.2. Metrics

To compare our results to other baseline models, we measure the performance of the implemented models by the commonly used metrics, BLEU 1–4 [19], ROUGE [20], and METEOR [21].

BLEU is one of the popular metrics to evaluate the correspondence between generated sentences by humans and machines. This metric measures the maximum number of co-occurrence n-grams between reference and candidate sentences. Here, 'n' takes the value of 1, 2, 3, and 4 depending on the length of sentences. Each BLEU-N metric averages the calculated accuracy from $n = 1$ to $n = N$. It means that BLEU-1 is the accuracy of the description created for the image with the reference description based on 1-gram, BLEU-2 is the geometric mean of the calculated accuracy based on 1-gram and 2-gram, BLEU-3 is the geometric mean of the calculated accuracy based on 1-gram, 2-gram, and 3-gram, and so on.

ROUGE evaluates the performance of generated sentences by a machine based on their similarity to the reference sentences. This metric finds the longest subsequence of tokens between candidate and reference sentences and calculates how many tokens from the human reference summaries were duplicated in the machine-generated summaries. Unlike BLEU, which prioritizes precision, ROUGE is recall-oriented and can estimate correlated n-grams better than BLEU.

METEOR is the last evaluation metric in this paper. In this metric and the exact word match, the stemmed and wordnet synonym tokens are taken into account between the alignment of the candidate and the reference sentence.

4.3. Baselines

We provide two baseline approaches to verify the effectiveness of the models. The framework for the baseline is almost the same as the model in [11] as a baseline method, except that GRU replaces the LSTM language model. We used inception-V3 and VGG16 as the feature extractor method for the encoder part.

4.4. Our approaches

We assess different variations of our approach. CN + IncV3 utilizes the extracted features from the capsule network and inception-V3 as image features extractors. CN + VGG16 uses a VGG16 network rather than inception-V3 in the en-coder. The Wikipedia knowledge base enriches the contextualized language model in this model. So, CN + IncV3 + EK and CN + VGG16 + EK are the models that use relevant external knowledge from Wikipedia. We also have performed additional experiments to check the importance of the capsule network in describing the content of images. To that end, we implemented IncV3 + EK and VGG16 + Ek methods to verify the effectiveness of the capsule net-work for image captioning models.

5. Results and Discussions

This section discusses the results from the different implementations of our framework and then compares them to state-of-the-art. Table 2 reports image captioning results for different implementations of our method on the MS-COCO dataset. The results demonstrate that the CN + IncV3 + EK model with capsule network and inception-V3 feature extractors can generate more human-like sentences by adding external knowledge to the language model. This model archives significantly better results in the overall metrics.

Table 2
The experimental results of implemented models. Bold text indicates the best overall performance.

Models	B1	B2	B3	B4	R	M
VGG 16 (Baseline)	0.33	0.24	0.18	0.16	0.21	0.24
IncV3 (Baseline)	0.36	0.26	0.21	0.17	0.23	0.28
CN + IncV3	0.77	0.54	0.43	0.35	0.47	0.35
CN + VGG 16	0.41	0.30	0.25	0.19	0.28	0.34
CN + IncV3 + EK	**0.89**	**0.74**	**0.61**	**0.54**	**0.66**	**0.45**
CN + VGG 16 + EK	0.59	0.44	0.37	0.29	0.31	0.38
IncV3 + EK	0.63	0.43	0.34	0.28	0.29	0.31
VGG 16 + EK	0.38	0.27	0.22	0.18	0.23	0.26

To prove the effectiveness of this model, we compare the result of the CN + IncV3 + EK method with state-of-the-art research. In Table 3, the bold numbers show that Table 3 shows that our best model outperforms previously published results on the MS-COCO "Karpathy" test split dataset.

Table 3
Comparison of the best result to state-of-the-art.

Models	B1	B2	B3	B4	R	M
Ours	**0.89**	**0.74**	**0.61**	**0.54**	**0.66**	**0.45**
(Aneja et al., 2018 [22])	0.72	0.55	0.40	0.30	0.53	0.25
(Tan et al., 2019 [23])	0.73	0.57	0.43	0.33	0.54	0.25
(Wu et al., 2017 [11])	0.73	0.56	0.41	0.31	0.53	0.25
(Zhang et al., 2021 [24])	0.75	0.62	0.48	0.36	-	0.27
(J. Yu et al., 2019 [25])	0.81	0.67	0.52	0.40	0.59	0.29
(Lu et al., 2017 [26])	0.75	0.58	0.44	0.33	0.55	0.26
(Ande. et al., 2018 [27])	0.80	0.64	0.49	0.37	0.57	0.27
(Jiang et al., 2018 [28])	0.80	0.65	0.50	0.38	0.58	0.28
(Yang et al., 2020 [12])	0.73	0.53	0.39	0.28	0.56	0.25

Compared to our model, Aneja et al. [22], has proposed an attention mechanism to leverage spatial features of an image to find salient objects. Tan et al. [23], proposed a tuning model with a small number of parameters in the RNN. Their model can produce a very sparse decoder for generating a caption preserving the performance of the method compared to their baseline. Zhang et al. Zhang et al. [24], implemented a cooperative learning mechanism to combine two image caption and image retrieval modules while generating a caption. Then, during a multi-step refining process, they refined the image-level and object-level information to produce a meaningful caption. Instead of using GRU as RNN, Yu et al. [25], proposed a model which employed a multimodal transformer as a language model in the decoder to generate a caption. Contrary to our approach, Lu et al. [26], Anderson et al.[27], have focused on important image regions. Lu et al. [26]), proposed an adaptive attention framework that could decide whether to rely on special attention to the image and when to attend to the textual image information. In [27], Anderson et al. extracted a set of salient regions from the image by applying a bottom-up mechanism. They also implemented a top-down mechanism to determine the distribution of attention over the image to compute feature weightings in different regions. Jiang et al. [28], proposed a framework that includes a recurrent fusion network. This fusion procedure is implemented between the encoder and decoder to exploit interactions among the represented features from the encoder part for creating a new set of vectors from decoder outputs.

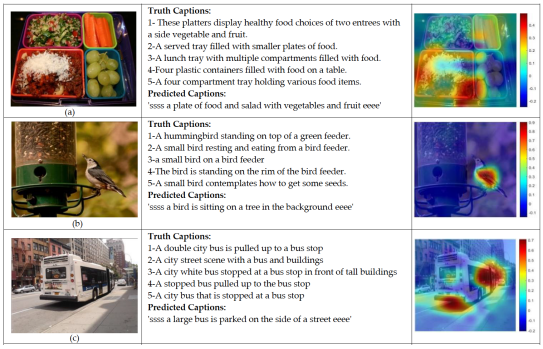

Figure 5: Generated examples by the best proposed model.

5.1. Qualitative Results

In this section, we present some examples to show the performance of the CN + IncV3 + EK method as our best model. We used the occlusion sensitivity function to visualize and localize the most important regions of the images for the network. The occlusion function computes sensitivity maps for CNNs. Figure 5 shows some examples from our results.

As demonstrated in Figure 5, using occlusion sensitivity helps us better understand features used by the network and provide insight into the reasons for the misclassified images. These examples show that CN + IncV3 + EK is the best descriptor model as it can generate more human-like sentences for each image.

6. Conclusions

In this paper, we developed an encoder–decoder framework employing a novel parallelized capsule network as a feature extractor and the Wikipedia database as an external knowledge provider to establish if this approach can out-perform state-of-the-art solutions. We implemented different architectures to produce contextual knowledge from images to achieve this. Our novel approach demonstrated that using a parallel capsule network as an encoder model provided a versatile image feature extractor. Moreover, we have demonstrated that the use of external knowledge further improved the results. Our best model was trained with the capsule network and inception-V3 as a feature extractor, with caption enrichment by an external contextual description.

References

[1] Y. Wei, L. Wang, H. Cao, M. Shao, C. Wu, Multi-attention generative adversarial network for image captioning, Neurocomputing 387 (2020) 91–99.

[2] S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules, Advances in neural information processing systems 30 (2017).

[3] X. Ai, J. Zhuang, Y. Wang, P. Wan, Y. Fu, Rescaps: an improved capsule network and its application in ultrasonic image classification of thyroid papillary carcinoma, Complex & Intelligent Systems (2021) 1–9.

[4] G. E. Hinton, S. Sabour, N. Frosst, Matrix capsules with em routing, in: International conference on learning representations, 2018.

[5] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.

[6] M. Hodosh, P. Young, J. Hockenmaier, Framing image description as a ranking task: Data, models and evaluation metrics, Journal of Artificial Intelligence Research 47 (2013) 853–899.

[7] P. Young, A. Lai, M. Hodosh, J. Hockenmaier, From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, Transactions of the Association for Computational Linguistics 2 (2014) 67–78.

[8] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, D. Forsyth, Every picture tells a story: Generating sentences from images, in: Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11, Springer, 2010, pp. 15–29.

[9] S. Javanmardi, A. M. Latif, M. T. Sadeghi, M. Jahanbanifard, M. Bonsangue, F. J. Verbeek, Caps captioning: a modern image captioning approach based on improved capsule network, Sensors 22 (2022) 8376.

[10] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, Y. Choi, Generalizing image captions for image-text parallel corpus, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2013, pp. 790–796.

[11] Q. Wu, C. Shen, P. Wang, A. Dick, A. Van Den Hengel, Image captioning and visual question answering based on attributes and external knowledge, IEEE transactions on pattern analysis and machine intelligence 40 (2017) 1367–1381.

[12] Z. Yang, Q. Liu, Att-bm-som: A framework of effectively choosing image information and optimizing syntax for image captioning, IEEE Access 8 (2020) 50565–50573.

[13] D. Martens, F. Provost, Pseudo-social network targeting from consumer transaction data (2011).

[14] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, H. Laga, M. Bennamoun, Text to image synthesis for improved image captioning, IEEE Access 9 (2021) 64918–64928.

[15] M. K. Patrick, A. F. Adekoya, A. A. Mighty, B. Y. Edward, Capsule networks–a survey, Journal of King Saud University-computer and information sciences 34 (2022) 1295–1310.

[16] B. Mandal, S. Ghosh, R. Sarkhel, N. Das, M. Nasipuri, Using dynamic routing to extract intermediate features for developing scalable capsule networks, in: 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), IEEE, 2019, pp. 1–6.

[17] S. Albawi, T. A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in: 2017 international conference on engineering and technology (ICET), Ieee, 2017, pp. 1–6.

[18] A. Karpathy, L. Fei-Fei, Deep visual-semantic alignments for generating image descriptions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3128–3137.

[19] K. Papineni, S. Roukos, T. Ward, W. Zhu, A method for automatic evaluation of machine translation", the Proceedings of ACL-2002, ACL, Philadelphia, PA, July 2002 (2001).

[20] C.-Y. Lin, Rouge: A package for automatic evaluation of summaries, in: Text summarization branches out, 2004, pp. 74–81.

[21] S. Banerjee, A. Lavie, 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, 2005, pp. 65–72.

[22] J. Aneja, A. Deshpande, A. G. Schwing, Convolutional image captioning, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5561–5570.

[23] J. H. Tan, C. S. Chan, J. H. Chuah, Image captioning with sparse recurrent neural network, arXiv preprint arXiv:1908.10797 (2019).

[24] W. Zhang, S. Tang, J. Su, J. Xiao, Y. Zhuang, Tell and guess: cooperative learning for natural image caption generation with hierarchical refined attention, Multimedia Tools and Applications 80 (2021) 16267–16282.

[25] J. Yu, J. Li, Z. Yu, Q. Huang, Multimodal transformer with multi-view visual representation for image captioning, IEEE transactions on circuits and systems for video technology 30 (2019) 4467–4480.

[26] J. Lu, C. Xiong, D. Parikh, R. Socher, Knowing when to look: Adaptive attention via a visual sentinel for image captioning, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 375–383.

[27] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, L. Zhang, Bottom-up and top-down attention for image captioning and visual question answering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6077–6086.

[28] W. Jiang, L. Ma, Y.-G. Jiang, W. Liu, T. Zhang, Recurrent fusion network for image captioning, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 499–515.