Vietnam Journal of Science and Technology ***6x*** *(202x)*

doi:10.15625/2525-2518/22381

RGTranCNet: Effective image captioning model using cross-attention and semantic knowledge

Nguyen Van Thinh[1, 2, 3], Tran Van Lang[4, *], Van The Thanh[3]

[1]*Institute of Machanics and Applied Informatics, Vietnam Academy of Science and Technology (VAST), 291 Dien Bien Phu Street, 3 District, Ho Chi Minh City, Viet Nam*

[2]*Graduate University of Science and Technology, Vietnam Academy of Science and Technology (VAST), 18 Hoang Quoc Viet Street, Cau Giay District, Ha Noi, Viet Nam*

[3]*Faculty of Information Technology, HCMC University of Education (HCMUE), 280 An Duong Vuong, 5 District, Ho Chi Minh City, Viet Nam*

[4]*Journal Editorial Department, HCMC University of Foreign Languages and Information Technology (HUFLIT), 828 Su Van Hanh, 10 District, Ho Chi Minh City, Viet Nam*

[*]Emails: *langtv@huflit.edu.vn*

Received: 6 February 2025; Accepted for publication: 14 July 2025

Abstract. Generating captions for images is a key endeavour that connects visual processing and linguistic analysis. However, techniques relying on long short-term memory (LSTM) units and conventional attention systems face restrictions in managing intricate interconnections and supporting effective parallel processing. Additionally, precisely depicting elements absent from the training data presents a significant challenge. To overcome these obstacles, the present research introduces an innovative framework for image description, employing a Transformer architecture augmented by cross-attention processes and semantic insights sourced from ConceptNet. This setup follows an encoder-decoder paradigm, where the encoder derives features from object areas and assembles a graph of associations to depict the visual scene. At the same time, the decoder merges visual and semantic aspects through cross-attention to produce captions that are both accurate and varied. The inclusion of ConceptNet-derived knowledge enhances precision, particularly when handling items not encountered during training. Tests conducted on the standard MS COCO dataset reveal that this approach outperforms recent state-of-the-art approaches. Moreover, the semantic integration strategy outlined here can be readily adapted to alternative image captioning systems.

Keywords: Image captioning, cross-attention mechanism, transformer, ConceptNet knowledge base, relationship graph.

Classification numbers: 4.7.4, 4.8.3

1. INTRODUCTION

Creating captions for images ranks among the crucial and demanding activities in artificial intelligence. This involves a multi-modal learning procedure that integrates visual computing with linguistic analysis. The goal is to produce significant text-based narratives derived from given images [1]. The process of automatically crafting precise and semantically abundant

descriptions from visuals necessitates a deep comprehension of the image's components, coupled with the model's proficiency in identifying semantic connections among entities, surroundings, and activities shown in the visual [2]. Furthermore, image captioning has numerous practical applications, such as image captioning systems for assisting visually impaired individuals in perceiving their surroundings [3], medical image captioning to aid doctors in diagnosing diseases [4], and human-robot interaction [5], image captioning for explainable visual question Answering [6]…

At present, most approaches to creating captions for images depend on an encoder-decoder structure that incorporates attention mechanisms [1]. In this setup, the encoder processes the input image to create vectors with a fixed length, which are later utilised by the decoder-usually a long short-term memory (LSTM) system-to craft descriptions based on those extracted elements [7 - 10]. Often, pre-trained Convolutional Neural Networks (CNNs) or tools for object identification, such as Faster R-CNN and YOLO, function as encoders to extract features at the region level from the imagery [7, 11]. That said, employing CNNs comes with drawbacks stemming from the bottleneck issue, in which data from the entire image is compressed into a vector of predetermined size. For object detection frameworks, the downside is that features from specific regions fail to encompass the whole visual content. As a result, it becomes essential to map out the interconnections among these elements. By doing so, the encoder can convey a holistic depiction of the image's details, serving as enhanced input for the decoder and thereby improving the precision of the resulting captions.

Building on the limitations mentioned above, several research groups have published image captioning works that leverage the relationships between objects in the image, demonstrating their effectiveness [12 - 15]. These works construct graphs to represent the image, which supplements the encoder with additional information to fully comprehend the image content, thereby serving as input to the decoder-an LSTM network with attention mechanisms—for caption generation. While decoders using LSTM networks with attention mechanisms have become popular in recent years for image captioning tasks, they still face challenges such as slow training times due to sequential computation and issues like vanishing or exploding gradients. Inspired by the success of the transformer model [16] in natural language processing, current image captioning models are gradually replacing LSTM with transformer in the decoding stage, owing to its parallelization capabilities and superior performance. Notably, the attention mechanisms in transformer (self-attention, cross-attention) have demonstrated superior ability in learning the context of related objects compared to traditional attention mechanisms.

Moreover, recent image captioning methods have been trained on datasets consisting of paired image-caption examples. However, these datasets typically contain only a tiny number of captions (usually between 1 to 5 captions per image). As a result, these models need more information to describe new objects not present in the training set or aspects not explicitly represented in the image. This issue can be addressed by integrating information from external data sources into the caption generation process. Several studies have leveraged external data beyond the training dataset, such as extracting object knowledge from object recognition datasets and external textual data to generate captions for novel object [17], using knowledge graphs to enhance image captioning performance [18], and employing semantic representations (e.g., using the ConceptNet knowledge base) along with attention mechanisms for image captioning [19]. Therefore, utilizing external knowledge beyond the training dataset, specifically the ConceptNet knowledge base, to improve image captioning models' performance is necessary, feasible, and effective.

Therefore, this study proposes a novel approach named **RGTranCNet**, which is a combination of three key components: **RG** stands for *Relationship Graph*, **Tran** refers to the *transformer decoder*, and **CNet** represents the integration of semantic knowledge from *ConceptNet*. In this model, (i) the object-region features and the image's relational graph are unified within a single cross-attention block rather than employing separate dual attention, thereby enhancing attention efficiency and reducing complexity; (ii) external knowledge from ConceptNet is seamlessly integrated into the decoding process to refine the generated captions, allowing more effective handling of objects absent from the training set-a standard limitation in current image captioning models-and thus improving generalizability and enriching the semantic content of the captions; and (iii) only the decoder is trained, while the modules responsible for extracting object-region features, creating and representing the relational graph, and retrieving semantic knowledge from ConceptNet remain fixed. This strategy significantly reduces training costs yet maintains accuracy and scalability.

The main contributions of this paper include:

- Improving image captioning performance using a transformer decoder as the language model in place of LSTM networks and employing cross-attention mechanisms instead of traditional attention mechanisms to integrate multimodal information between the encoder and decoder.
- Integrating semantic knowledge from the ConceptNet knowledge base into the decoder to leverage external knowledge beyond the training dataset, thereby enhancing the accuracy of the generated captions, particularly for novel objects. This approach can be easily applied to other image captioning models.
- Extensive experiments on the benchmark MS COCO dataset demonstrate that the proposed model achieves higher accuracy than previous methods (including LSTM-based ones) across most evaluation metrics while maintaining low training costs by training only the decoder.

The previous portion examines significant challenges in producing captions for images by incorporating cross-attention approaches and semantic components. The remaining parts of the article are organised in the manner described here: Section 2 examines pertinent research and underscores ongoing obstacles within the domain. Section 3 offers an in-depth overview of the proposed technique. Section 4 details the experimental setups and presents the results obtained from evaluations. Lastly, Section 5 concludes the research and proposes potential avenues for future exploration.

2. RELATED WORKS

Many recent image captioning works have been published based on the encoder-decoder framework with attention mechanisms, employing pre-trained CNN networks, object detection models, and models that predict relationships between objects in the image, as well as utilizing external data sources beyond the training dataset, such as:

Patwari *et al*. [20] introduced a method for describing images that relies on an encoder-decoder structure, employing a pre-trained Inception-v3 convolutional neural network to derive visual features. A GRU-equipped decoder, augmented by an attention system, then produces the descriptions. This approach yielded encouraging results on the MS COCO dataset, as reflected in its BLEU-1 through BLEU-4 metrics. However, it is still hampered by its heavy reliance on pre-trained CNNs for extracting visual elements, which leads to challenges in identifying fine-

grained object specifics and the connections between them, ultimately hindering a more profound grasp of the image's semantic essence.

Xie *et al*. [21] presented a framework designed to enhance the effectiveness of image description generation by combining bidirectional LSTM architectures and attention systems. Their methodology involves deriving features from object areas in the input visuals via Faster R-CNN, followed by their handling in a Bi-LSTM setup to produce explanatory text. This system underwent experimental testing on the Flickr30k and MS COCO benchmarks, where it exhibited better outcomes than standard references and various contemporary works. That said, a significant shortcoming lies in its narrow focus on isolating object regions, overlooking the interconnections between them—a factor that could enhance the image's semantic depiction and improve the precision of the generated descriptions.

Chen *et al*. [22] proposed a technique that generates an abstract scene graph from authentic captions to guide the production of image captions, offering increased variety and better alignment with user preferences. Drawing from this foundation, Yan and their team [23] advanced the system by incorporating transformer elements in conjunction with a two-level LSTM design to enhance smoothness and consistency. Within this setup, the primary LSTM layer integrates inputs across modalities, merging visual and linguistic signals, while the secondary layer constructs the captions. The transformer component oversees the weighting of diverse feature types during the decoding stage. Although these works adeptly utilise abstract scene depictions obtained from labelled captions to elevate the quality of descriptions, they both suffer from a shared limitation: the inadequate exploitation of the image's intrinsic elements, especially the semantic interconnections among objects. This deficiency leads to their somewhat inferior outcomes on multiple conventional evaluation measures.

Ramos *et al.* [24] integrated the ConvNeXt architecture with a Long Short-Term Memory (LSTM) system, augmented by a visual attention component, to boost the effectiveness of generating image descriptions. This framework underwent testing on the MS COCO dataset, where its performance was measured via the BLEU score, revealing greater precision relative to approaches that employ pre-trained CNNs as encoders. While ConvNeXt demonstrated advantages over established pre-trained CNN frameworks, it nonetheless encounters difficulties in thoroughly grasping the interconnections among elements within the images. The LSTM element also presents shortcomings, as noted in prior discussions. Moreover, assessing outcomes based exclusively on the BLEU metric falls short of providing a thorough analysis of the model's strengths, as supplementary evaluation criteria are necessary to encompass the diverse dimensions of the image description generation task.

Thinh *et al*. [12] introduced an image captioning framework that incorporates both object detection and relationship prediction to capture the semantic structure of an image. Initially, objects are identified using a detection model enhanced with a graph convolutional network, followed by the inference of inter-object relationships informed by contextual cues and predefined relational knowledge. These relationships are then categorised to form a structured graph that semantically represents the image. To support the caption generation process, a dual-attention mechanism is employed, allowing the model to selectively attend to both visual object regions and corresponding nodes in the relationship graph. Caption generation is carried out by an LSTM network equipped with this dual-attention design, utilising both the extracted image features and reference captions. Experimental evaluation on the MS COCO dataset confirms the model’s effectiveness. Nonetheless, the approach encounters limitations: the two independent attention modules within the dual-attention design do not sufficiently integrate visual and semantic features, and the reliance on LSTM networks introduces sequential processing

constraints. These limitations suggest the need to replace the LSTM component with a transformer architecture and adopt a cross-attention mechanism to align heterogeneous features better and improve captioning performance.

Wang *et al*. [25] developed a system for generating image descriptions using a transformer architecture, aiming to address the shortcomings inherent in CNN-LSTM configurations. Yang *et al.* [26] presented a transformer-oriented framework dedicated to context detection, aimed at boosting the precision of generated captions. Li *et al*. [27] suggested integrating a transformer with supplementary external data to capitalise on inter-object connections, thereby elevating the quality of image captioning results. These techniques were subjected to empirical testing on the MS COCO dataset and demonstrated their efficacy across typical performance indicators for image description tasks, including BLEU, METEOR, ROUGE, and CIDEr.

Zhou *et al.* [18] suggested an approach to boost the precision of generating image descriptions by utilising data from the ConceptNet repository. They incorporated semantic insights associated with image elements into the encoder section of the NIC framework for captioning [7], achieving improved outcomes over methods that rely solely on visual characteristics. That said, a notable drawback of this technique is that incorporating an overload of input data might create interference in the training phase, ultimately diminishing the model's overall efficiency. Hafeth and their team [19] presented a semantic attention-oriented network designed to embed supplementary knowledge (derived from ConceptNet) into the transformer's attention components, which in turn enhances the performance of image description generation.

From the survey and analysis of related works, it is evident that the image captioning problem, mainly using deep learning networks such as transformers, has garnered significant attention from the research community and has proven effective. Moreover, integrating semantic knowledge from external data sources beyond the training dataset is also feasible and practical. Building on the foundation of existing research and addressing the limitations of previously published methods, the proposed image captioning approach utilizes a relationship graph, a transformer decoder with cross-attention mechanisms, and the integration of semantic knowledge from ConceptNet into the decoder, aiming to improve accuracy and enhance the model's generalization capability.

3. PROPOSED METHOD

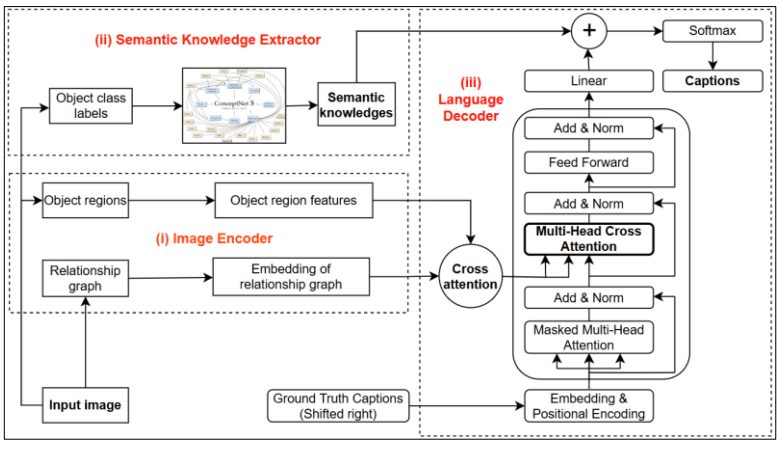

Figure 1. Architecture of the image captioning model integrating semantic knowledge and cross-attention mechanism.

In this study, we introduce an image captioning model built upon the encoder-decoder architecture, as illustrated in Figure 1. The model comprises three core modules: (i) an image encoder responsible for learning visual representations from the input image; (ii) a semantic knowledge extractor that retrieves relevant object-level semantics from the ConceptNet knowledge base; and (iii) a Transformer-based decoder that generates image captions by utilizing the visual features from (i) in conjunction with the semantic information from (ii), thereby improving caption quality and semantic alignment.

3.1. Image Encoder

The image encoder comprises two main processes: (3.1.1) identifying object regions within the image and extracting their visual features, and (3.1.2) generating and embedding a relationship graph that captures interactions among the detected objects. These two types of features are integrated through a cross-attention mechanism, which provides the contextual input for the decoder to generate descriptive captions.

3.1.1. Detecting and extracting features from object regions

Pre-trained object detection frameworks, including SSD, Faster R-CNN, and YOLO, trained on extensive datasets, have achieved impressive results in computer vision applications like image captioning. However, they often face challenges with visuals featuring multiple elements or complicated layouts. Additionally, these systems emphasise individual object attributes while overlooking connections between them, potentially causing detection errors in specific scenarios. To mitigate these issues, our previous study [12] introduced ODwGCN, a two-phase enhancement method. The initial phase applies a Graph Convolutional Network (GCN) to identify co-occurrence patterns among image objects. The subsequent phase adjusts outputs from pre-trained detectors using factors based on those patterns. Testing on the MS COCO dataset revealed gains of 1.2 to 2.2 points in mAP and 0.9 to 3.2 points in mAP@0.5 compared to standard benchmarks.

In the current study, ODwGCN is utilised to identify object regions. The visual features of these regions are extracted using ResNet101, yielding a set of feature vectors denoted as F_I for a given input image I. These features are later integrated with the embedding of the relationship graph and provided as input to the decoder for caption generation.

3.1.2. Constructing and representing the relationship graph of the image

A relational graph provides a robust framework for depicting complex interdependencies between objects, portraying them as vertices and their mutual connections as oriented links. According to the definition in [12], the image's relational graph—termed R-Graph—is officially represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, in which:

- Vertex set $\mathcal{V} = \{v_i \in \mathcal{B}, \forall i = \overline{1, N_B}\}$ corresponds to the detected object regions in the image.
- Edge set $\mathcal{E} = \{e_{ij} = (v_i, v_j, r_{ij}) \in \mathcal{R}, \forall i, j = \overline{1, N_B}, i \neq j\}$ represents the directed relationships between object pairs.

Here, $\mathcal{B}$ denotes the set of identified object regions, N_B is the number of such regions, and $\mathcal{R}$ is the predefined set of relationships derived from the training dataset.

3.1.2.1. Creating the relationship graph

In our prior research [12], we developed the VRP+RK model for predicting various relationship types, leveraging relational insights from entity pairs in the training data. Framed as

a multi-class classification challenge, it takes as input a duo of object regions plus their semantic tags, outputting one of $N_{\mathcal{R}}+1$ options—encompassing $N_{\mathcal{R}}$ specified relations plus a "$one\ relation$" class.

Once the VRP+RK model is trained on the Visual Genome dataset, it is used in conjunction with ODwGCN to construct the relationship graph for a given input image. Initially, the N_B detected object regions are combined to form $N_B(N_B-1)$ object pairs. For each pair, the relationship classifier produces a probability distribution over the $N_{\mathcal{R}}+1$ categories. If the predicted probability for the "*none relation*" is below a specified threshold γ, an edge is established between the corresponding nodes v_i and v_j, labelled with the most probable relationship. An illustrative example of the resulting relationship graph $\mathcal{G}$ is depicted in Figure 2(c), where the vertex set is $\mathcal{V}=\{child, tie, water\ bowl\}$, and the edge set is $\mathcal{E}=\{<child, tie, has>, <child,\ water\ bowl, playing>\}$.

After training VRP+RK on the Visual Genome dataset, it collaborates with ODwGCN to build the relational graph for an input image. This starts by forming $N_B(N_B-1)$ pairs from the N_B detected regions. The classifier then generates a probability spread across the $N_{\mathcal{R}}+1$ classes for each pair; if the "$one\ relation$" likelihood falls below threshold γ, a directed edge links nodes v_i and v_j, tagged with the top-probability relation. For instance, Figure 2(c) shows graph G with vertices $\mathcal{V}=\{child, tie, water\ bowl\}$ and edges $\mathcal{E}=\{<child, tie, has>, <child, water\ bowl, playing>\}$.

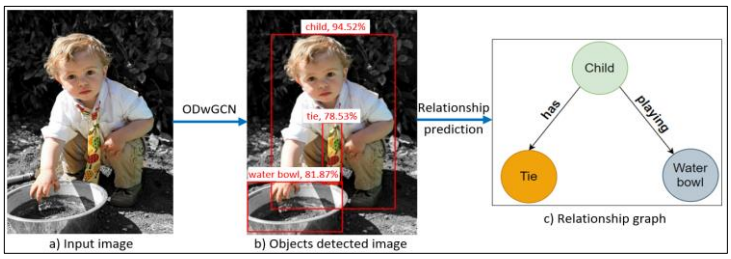

Figure 2. Creating a relationship graph from an input image: a) the input image, b) the result after applying the improved object detection model ODwGCN, and c) the relationship graph obtained after predicting the relationships between the objects.

3.1.2.2. Representation the relationship graph

Although the relationship graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ provides a comprehensive and accurate representation of image content, its heterogeneous structure poses challenges for integration into most learning algorithms that rely on semantic information [26]. To address this, it becomes necessary to convert the graph into a linearised format that retains the original semantic structure while making it compatible with deep learning models, particularly as input to the encoder in a language generation framework. To further exploit the semantic richness of object class labels and their interrelations-rather than relying solely on region-level features and raw graph representations as done in prior work-we proposed in [12] a transformation of the relationship graph into an extended form, referred to as the enriched relationship graph $\mathcal{G}^*=(\mathcal{V}^*,\mathcal{E}^*)$, or R-Graph*. This transformed graph is defined as follows:

– Vertex set $\mathcal{V}^*=\{v_i^*\in\mathcal{L}, \forall i=\overline{1,N_{\mathcal{L}}}\}$ where $\mathcal{L}$ includes two types of nodes: object class labels and relationship (predicate) labels.

– Edge set $\mathcal{E}^* = \{e_{ij}^* \in \{0,1\}, \forall i,j = \overline{1,N_{\mathcal{L}}}, i \neq j\}$ is constructed using the rule: for each edge $e_{ij} = (v_i, v_j, r_{ij}) \in \mathcal{R}$, two directed edges are created—one from v_i to r_{ij}, and another from r_{ij} to v_j.

This formulation preserves the structural semantics of the original graph while making it more amenable to language-oriented neural architectures. For example, the relationship graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in Figure 2(c) has objects: $child, tie, water\ bowl$; relationships between the objects: $< child, tie, has >$, $< child, water\ bowl, playing >$ will be converted into the graph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$ consisting of 5 vertices and 4 edges as shown in Figure 3.

To learn representations for the vertices in the extended relationship graph (R-Graph*), we employ the GraphSAGE framework [28] in conjunction with an unsupervised training approach. The model is optimised using a contrastive loss function that encourages embeddings of neighbouring nodes to be more similar than those of non-neighbours. The process of generating vertex embeddings involves the following steps:

– **Step 1:** Utilize the word2vec (GloVe) technique to extract the feature vector of the vertex labels.
– **Step 2**: Divide the neighbouring vertices into two categories: the incoming set ($\mathcal{N}$-(v)), which includes nodes directing toward v, and the outgoing set ($\mathcal{N}$+(v)), which includes nodes that v directs toward.
– **Step 3**: Combine the information from ($\mathcal{N}$-(v)) and ($\mathcal{N}$+(v)) with the current information of node v, creating the vectors $h_{v-}^{(N_L)}$ and $h_{v+}^{(N_L)}$.
– **Step 4**: Concatenate $h_{v-}^{(N_L)}$ and $h_{v+}^{(N_L)}$ to create the final representation of vertex v, $z_v^* = [h_{v-}^{(N_L)}, h_{v+}^{(N_L)}], \forall v \in \mathcal{V}^*$..
– **Step 5:** Repeat steps 2 to 4 N_L times to obtain the final representation of vertex v.

Where, N_L is number of layers of GCN, $h_v^{(l)}$ is the hidden state of node v at layer l of the GCN.

Thus, for the relationship graph of an input image I, the resulting embedding set Z_I consists of the feature vectors of the vertices in the R-Graph*.

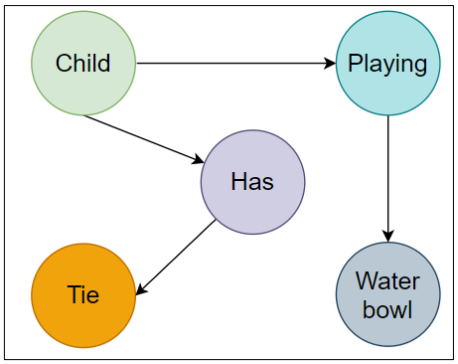

Figure 3. Result of converting the relationship graph R-Graph in Figure 2(c) into the extended relationship graph R-Graph*

3.2. Semantic knowledge extractor

ConceptNet is a multilingual knowledge base describing words and phrases commonly used by humans and their typical relationships. The knowledge in ConceptNet is collected from various sources, including community-contributed resources such as Wiktionary and Open Mind Common Sense, expert-curated resources like WordNet and JMDict, and many other open data sources [29]. This creates a knowledge base that links concepts through semantic relationships such as "IsA", "PartOf", "UsedFor", and many others. These relationships enable the model to understand better the context and semantic links between words and phrases. As a result, it aids artificial intelligence systems in understanding context, reasoning, and interacting with humans. In this study, the ConceptNet knowledge base $\mathcal{K}$ is defined as a graph as follows:

Definition 1. Graph **CK-Graph** $\mathcal{K} = (V, E, W)$ is a directed graph consisting of:

- Vertex set $V = \{v_i \in C, \forall i = \overline{1, N_C}\}$, N_C is the number of concepts in ConceptNet,
- Edge set $E = \{e_{ij} = (v_i, v_j, w), \forall i, j = \overline{1, N_C}, i \neq j\}$,
- Weight set $W = \{w_i \in \mathbb{R}^+, \forall i = \overline{1, N_E}\}$, N_E is the number of edges in E.

To integrate semantic knowledge from ConceptNet into the caption generation process to improve accuracy, particularly in describing novel objects not present in the training dataset, we first represent ConceptNet as a knowledge graph $\mathcal{K} = (V, E, W)$, as defined in Definition 1. Then, object class labels in the image are used to query semantically similar knowledge from this graph. Figure 4 illustrates the result of querying information for the object "laptop" from $\mathcal{K}$. Notably, each object corresponds to a probability value, representing the degree of correlation with the queried term.

In this paper, the top-k related objects for each detected object in the image are used to enhance the information for the decoder during the caption generation process. This set, denoted as O, is utilized when generating the next word of the decoder. The process of extracting knowledge for the related objects is described in Algorithm 1.

Algorithm 1. **ExtractRelatedObjectCNet**$(L_I, \mathcal{K})$

Input: The object label set of the image I, $L_I = \{l_1, l_2, .., l_{N_I}\}$, ConceptNet knowledge base $\mathcal{K}$

Output: The list of related objects and their corresponding weights O

Begin

Initialize the empty set O

$O \leftarrow \emptyset$

foreach $l_i \in L_I$ **do**

Retrieve the edge set $E_i \in E$ from ConceptNet $\mathcal{K}$

$E_i = \{(v_s, v_t, w) | v_s = l_i\}$

The set of related objects of l_i

$O_i = \{(v_t, w) | (l_i, v_t, w) \in E_i\}$

Update O

$O \leftarrow O \cup O_i$

end

End

Algorithm 1 performs the extraction of related objects from the ConceptNet knowledge base based on the class labels of objects in the image. The algorithm begins by initializing an empty set to store the related objects along with their corresponding weights. For each object

class label in the image, the algorithm retrieves edges from ConceptNet that are sourced from that label, thus obtaining the corresponding set of edges. The related objects of the class label are identified from these edges, and the result set is updated by incorporating them into the related object set. The final output of the algorithm is a list of related objects along with their weights. Thus, given an input image I containing the set of class labels of the detected objects in the image L_I, Algorithm 1 produces the result, the semantic knowledge of the related objects O_I, comprising the set of objects and their corresponding weight values.

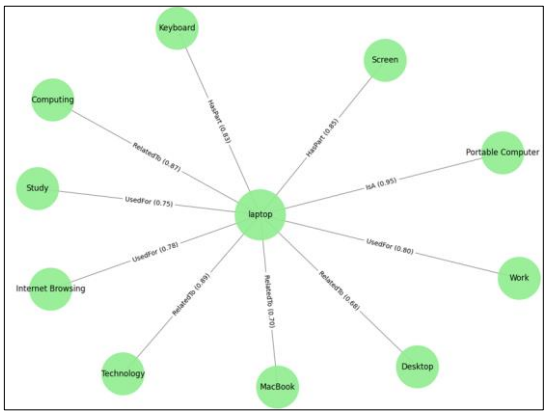

Figure 4. Illustration of knowledge extraction from ConceptNet for the object class label "laptop".

3.3. Language Decoder

In this paper, the decoder component of the transformer is utilized as a language model for image caption generation. The features extracted from the image in the Encoder, including object region features and embeddings of the relational graph, are combined and input into the decoder through a cross-attention mechanism to train the caption generation model. Semantic knowledge extracted from ConceptNet is also integrated to enhance the model's performance. The training and caption generation process of the Encoder is described through two algorithms: Algorithm 2, which trains the transformer decoder by integrating semantic knowledge to create a model for image caption generation, and Algorithm 3, which generates captions for input images using the model trained in Algorithm 2.

To focus on two main improvements: (1) the multi-head cross-attention mechanism, which integrates information from object region features and relational graph embeddings, and (2) the integration of semantic knowledge from ConceptNet to adjust the predicted probabilities during generation, thereby aiding the model in producing more accurate, diverse, and meaningful captions, especially for objects not present in the training dataset. We omit the details of other layers in the transformer decoder, such as masked multi-head attention, feed forward layer, and add & norm, as these components are retained according to the original design.

Algorithm 2. **TrainingTransDecCNet**$(\mathcal{D}, \varphi)$

Input: Training dataset $\mathcal{D} = \{(F_i, Z_i, \mathcal{S}_i, O_i), \forall i = \overline{1, N_T}\}$.
Output: The parameters of the model φ have been optimized.
Begin
$\varphi \leftarrow$ Random Initialization
# Process each data sample in the training set
for $i = 1$ *to* N_T **do**

$Loss \leftarrow 0$
$H_{init} = Embedding(S_i)$
Masked Multi-Head attention
$Q_{masked} = H_{init}W_q^{masked}, K_{masked} = H_{init}W_k^{masked}, V_{masked} = H_{init}W_v^{masked}$
$Att_{masked} = Softmax\left(\frac{Q_{masked}K_{masked}^T}{\sqrt{d}}\right)V_{masked}$
$H_{masked} = Add\&Norm(H_{int} + Att_{masked})$
Multi-Head cross-attention between the outputs of the encoder and decoder.
Attention on the features of object regions
$Q_F = H_{masked}W_q^F, K_F = F_iW_k^F, V_F = F_iW_v^F$
$Att_F = Softmax\left(\frac{Q_FK_F^T}{\sqrt{d}}\right)V_F$
Attention on the embeddings of the graph nodes
$Q_Z = H_{masked}W_q^Z, K_F = ZW_k^F, V_Z = Z_iW_v^Z$
$Att_Z = Softmax\left(\frac{Q_ZK_Z^T}{\sqrt{d}}\right)V_Z$
Combining two sources of attention
$CombinedAtt = \alpha.Att_F + (1-\alpha).Att_Z$
$H_{cross} = Add\&Norm(H_{masked} + CombinedAtt)$
$H_{final} = FeedForward(H_{cross})$
$H_{final} = FeedForward(H_{final})$
Calculating the logits score
$Logits = W_oH_{final}$
Adjusting the logits with semantic knowledge from ConceptNet
foreach $(l, w) \in O_i$ **do**
$Logits'[l] = Logits[l] + \beta w$
endfor
$P = Softmax(Logits')$
Calculating the loss for the i[th] sample
$Loss_i = -\sum_{t=1}^{N_{S_i}} logP(s_t^i | S_{<t}^i, F_i, Z_i, O_i)$
Updating the model parameters
$\varphi = \varphi - \eta \frac{\partial Loss_i}{\partial \varphi}$
endfor
End

In Algorithm 2, N_T is the number of data samples in the training dataset, $F_i, Z_i, \mathcal{S}_i$ and O_i represent the object region features, embeddings of the nodes in the relational graph, ground truth captions, and the related knowledge object set for the i^{th} data sample (image), respectively. The ground truth captions for each image are denoted as $S = \{s_1, s_2, \dots, s_{N_S}\}$, where s_i represents the i^{th} word in the sentence, $\forall i = \overline{1, N_S}$, with N_S being the number of words in sentence S. H is the hidden state, and W_Q, W_K, W_V and W_O are the weight matrices of the transformer decoder. These weight matrices are randomly initialised and will be learned and updated during training.

Algorithm 2 trains the transformer decoder in producing captions for images by merging visual elements derived from object areas with semantic details from the image's relational graph embeddings, utilising a cross-attention system. Concurrently, it embeds insights from

ConceptNet to elevate the quality of the resulting captions. For every training instance, the procedure begins with masked multi-head self-attention, which restricts the model to relying solely on prior words for forecasting the subsequent one. Following this, the visual attributes of the image (pulled from its object zones) and semantic representations (via the relational graph) are fused within the multi-head cross-attention layer. The decoder's ultimate output feeds into a linear transformation to derive logits across the vocabulary. These logits receive modifications by incorporating pertinent ConceptNet values when the associated term appears in the relevant semantic knowledge collection. Afterwards, $softmax$ normalises the revised logits into probability distributions for each time step across the vocabulary. This sequence iterates across N_T training examples, employing cross-entropy loss to refine the caption generation.

Algorithm 3. **GenerateCaption**(F_I, Z_I, φ)

Input: F_I, Z_I, the trained transformer model φ

Output: the captions of image $\hat{S}_I$

Begin

$\hat{S}_I \leftarrow [< start >]$

$X = Embedding(\hat{S}_I)$

while $(last\ token\ \neq < end >)\ and\ \left(length\ of\ \hat{S}_I < \max length\right)$ ***do***

$H_{init} = Embedding(\hat{S}_I)$

$H_{masked} = MaskedMultiHeadAttention(H_{init})$

Combine features

$H_{cross} = MultiHeadCrossAttention(H_{masked}, F_I, Z_I)$

Predict the next word

$Logits(s_t) = W_o.FeedForward(H_{cross})$

$P(s_t) = Softmax(Logits(s_t))$

$s_t = argmax_{s \in Vocab} P(s_t)$

Update the caption

$\hat{S}_I = \hat{S}_I \cup s_t$

end

End

In Algorithm 3, captions for an input image are produced via a pre-trained transformer decoder (φ). Utilising the features from object regions and embeddings of the relationship graph, the process begins with the token. For every iteration, the ongoing sequence undergoes encoding through masked multi-head attention, followed by fusion with image features using multi-head cross-attention. The resulting output passes through a linear layer to yield logits across the vocabulary, which are then normalised by $softmax$ into probabilities. The term with the peak probability becomes the subsequent addition. This loop persists, adding each new word to the sequence, until it reaches the token or the predefined maximum length. Ultimately, this yields a precise and relevant caption drawn from the image's characteristics.

4. EXPERIMENTS AND RESULTS

Grounded in the theoretical framework and model architecture described earlier, this section presents the experimental setup and performance evaluation using standard metrics

commonly adopted in image captioning tasks. It also provides an analysis of the results, along with a comparative assessment against baseline methods and recent state-of-the-art models, in order to emphasise both the strengths and potential limitations of the proposed approach.

4.1. Data and Experimental setup

This section describes the experimental data, parameters, and configuration settings for implementing the proposed method. It also presents performance evaluation metrics.

4.1.1. Experimental Data

The proposed framework for generating image captions underwent evaluation on the MS COCO dataset [30], a standard reference widely utilized for object recognition, segmentation, and image captioning tasks. This collection features 82,783 training images and 40,504 validation images, each provided with a minimum of five captions authored by people. For uniformity in our analyses, we limited usage to the first five captions per image. To facilitate equitable comparisons with existing research, we adhered to the established partitioning scheme from [31], assigning 82,783 images to training, 5,000 to validation, and a further 5,000 to testing. In the preparation phase, words occurring fewer than five times were omitted from the lexicon, culminating in 10,010 unique terms and a maximum of 16 tokens per caption.

4.1.2. Implementation Details

The proposed model was developed using Python version 3.9 and implemented with the PyTorch deep learning framework version 2.0. All experiments were conducted on the Google Colab Pro platform, utilising the following computational settings and hyperparameters:

The process of creating and embedding the relationship graph was carried out according to the setup in our previous study [12].

Transformer Decoder: The decoder consists of N = 6 blocks with 8 heads. The vector dimension for word representation is 512. The Adam optimizer is used with a learning rate of 0.00004 and a batch size of 32.

ConceptNet: ConceptNet 5.7 is employed to retrieve related entity knowledge of objects in the image via the REST API at api.conceptnet.io. Five objects with the highest probability are selected for each image to extract semantic knowledge from ConceptNet. For each object, the top 10 most relevant semantic knowledge entries (with the highest probability) are selected and input into the decoder to enhance performance during caption generation.

The configuration of Google Colab Pro used: Tesla T4 GPU with 15 GB, 51 GB RAM. The image captioning model training time is approximately 12 hours (20 epoches), and the average inference time per image is approximately 1.5 seconds.

4.2. Evaluation metrics

The evaluation metrics used in this study are widely adopted measures for assessing the quality of generated image captions compared to the provided ground truth caption set, including BLUE [32], METEOR [33] và ROUGE [34] and CIDEr [35]. Each metric evaluates the captions from distinct perspectives and uses different calculation methods. However, a common characteristic of these metrics is that the higher the score, the better the model's performance.

4.3. Results and Discussion

The experimental results of the proposed image captioning method are presented in Table 1, with the BLUE1, BLUE4, METEOR, ROUGE, and CIDEr scores achieving 77.5, 34.9, 28.3, 55.3, and 98.4, respectively, for the RGTran model (without integrating semantic knowledge from ConceptNet), and 79.8, 36.3, 35.6, 57.2, and 107.8 for the RGTranCNet model (with semantic knowledge integration from ConceptNet). These results indicate that the RGTran model (using transformer and cross-attention mechanism) outperforms the OD-VR-Cap method [12] (which uses LSTM and dual-attention mechanism) across all evaluation metrics, particularly with a significant increase in CIDEr (+13.3 points). This improvement is mainly attributed to the cross-attention mechanism of the transformer, which is capable of integrating information from various sources into a shared space, producing more comprehensive and semantically rich features than independent attention mechanisms as in OD-VR-Cap. Additionally, the transformer decoder is more effective than LSTM in handling complex relationships, thanks to the self-attention mechanism that flexibly and robustly captures the dependencies between words in a sentence. Furthermore, integrating semantic knowledge from ConceptNet into the decoder of the RGTranCNet model leads to an overall improvement across all evaluation metrics, with notable increases in METEOR (up by 7.3 points) and CIDEr (up by 9.4 points). The improvement in METEOR is due to the ability to match synonyms. At the same time, CIDEr reflects the fluency and coherence of the generated captions, which aligns with the integration of semantic knowledge from ConceptNet, resulting in more accurate and meaningful captions.

Table 1. Image captioning performance of the proposed method on the experimental dataset's test set.

Methods	BLUE-1	BLUE-4	METEOR	ROUGE	CIDEr
RGTran (**without ConceptNet**)	77.5	34.9	28.3	55.3	98.4
RGTranCNet (**with ConceptNet**)	**79.8**	**36.3**	**35.6**	**57.2**	**107.8**

An example of the results from the proposed image captioning model is presented in Figure 5. In this figure, (a) shows the input image, and (b) shows the captions generated by the respective models. The results indicate that the OD-VR-Cap model [12] only captures the primary objects and relationships in the image (the object "man" and action "fixing" on the left, and the objects "person," "skateboard," and action "jumping" on the right), with details and context not fully represented. RGTran enhances the model's ability to identify specific locations and contexts; however, it still struggles with detailed contexts or objects and actions not present in the training dataset. In contrast, RGTranCNet incorporates additional semantic knowledge from ConceptNet into the decoder, allowing the model to handle unseen objects by substituting them with semantically relevant concepts or enriching the relationships between objects. As a result, the generated captions are more accurate and meaningful.

These qualitative improvements are a direct result of integrating structured semantic knowledge into the captioning process. Specifically, the relationship graph enables the model to encode pairwise spatial and functional relations between detected objects. At the same time, ConceptNet provides external semantic associations that help enrich the meaning of individual concepts. For instance, replacing "fixing" with "repairing" or "jumping" with "performing a skateboard trick" reflects a deeper understanding of the functional context of the scene, not just object labels. This combination allows RGTranCNet to generalize better to unseen situations and

produce captions that are not only accurate in terms of object recognition but also more human-like in their semantic expressiveness.

To further demonstrate the effectiveness of RGTranCNet, Table 2 presents a comparative evaluation against several baseline models [9] and recent state-of-the-art approaches on the MS COCO dataset. RGTranCNet achieves the highest performance across all reported metrics, including BLEU-1 (79.8), BLEU-4 (36.3), METEOR (35.6), ROUGE (57.2), and CIDEr (107.8). Compared to CNet-NIC—a model that also incorporates ConceptNet but relies on a conventional NIC architecture—RGTranCNet outperforms it significantly in BLEU-4 (+6.4), METEOR (+10.0), and CIDEr (+0.6), indicating substantial improvements in both expressiveness and semantic representation. Similarly, in comparison to ConvNeXt, a model built upon a modern visual encoder architecture, RGTranCNet also achieves superior results in BLEU-1 and BLEU-4.

Notably, the Caption TLSTMs model achieves a CIDEr score of 101.8, surpassing both OD-VR-Cap and RGTran, which demonstrates its ability to produce highly informative captions. This performance stems from its architectural design, which integrates an abstract scene graph and a Transformer block inserted between two LSTM layers. Such a configuration enables the model to generate captions that are diverse in content and coherent in structure, thereby contributing to a notable improvement in CIDEr. However, its CIDEr score remains lower than that of RGTranCNet, suggesting that while the architecture is effective, it does not reach the level of semantic and structural richness provided by the proposed model. Moreover, Caption TLSTMs underperforms on other metrics such as BLEU and METEOR, indicating that its semantic expressiveness is still less comprehensive compared to RGTranCNet, which achieves consistently high performance across both syntactic and semantic dimensions.

Table 2. Comparison of image captioning performance across methods on the experimental dataset.

Methods	BLUE-1	BLUE-4	METEOR	ROUGE	CIDEr
Show, attend and tell (Hard-ATT) [9]	71.8	25.0	23.0	-	-
Show, attend and tell (Soft-ATT) [9]	70.7	24.3	23.9	-	-
CNet-NIC [18]	73.1	29.9	25.6	53.9	107.2
En-De-Cap [20]	70.6	24.3	-	-	-
Caption TLSTMs [23]	-	22.9	25.2	50.9	101.8
Bi-LS-AttM [21]	68.8	25.2	21.5	-	41.2
ConvNeXt [24]	74.8	34.8	-	-	-
OD-VR-Cap [12]	72.6	28.1	24.8	53.4	85.1
RGTran (ours)	77.5	34.9	28.3	55.3	98.4
RGTranCNet (ours)	**79.8**	**36.3**	**35.6**	**57.2**	**107.8**

In summary, the experimental results validate that integrating relationship graphs, a Transformer decoder with cross-attention, and semantic knowledge from ConceptNet constitutes a practical approach to image captioning. RGTranCNet not only exploits visual and relational structural information but also demonstrates the ability to synthesise semantics and infer meaning from external knowledge sources. Consequently, it generates captions that are more accurate, fluent, and semantically enriched than those produced by prior methods.

4.4. Computational Complexity Analysis

This section presents a detailed analysis of the computational complexity of the proposed RGTranCNet framework, considering both training and inference phases. The framework comprises four major components: object detection, relationship graph construction, graph embedding, and transformer-based caption generation.

Importantly, in the proposed approach, only the transformer decoder is trained, while the remaining components - including the object detection module, the relationship graph construction module, and the graph embedding module - are reused from our previously published OD-VR-Cap [12]. These components are fixed and frozen during both training and inference. As a result, the training cost is significantly reduced.

4.4.1. Training Complexity

Since the transformer decoder is the only trainable component, the training complexity of the framework is dominated by the self-attention and feedforward layers. For a caption sequence of length N_A and embedding dimension N_H, the self-attention sublayer in each decoder layer has a time complexity of $\mathcal{O}(N_A^2.N_H)$, while the position-wise feedforward network adds $\mathcal{O}(N_A N_H^2)$. Assuming N_D decoder layers, the overall training-time complexity becomes:

$$\mathcal{O}(N_D.(N_A^2.N_H + N_A.N_H^2)).$$

This lightweight training setup makes the model computationally efficient and feasible for training on limited hardware.

4.4.2. Inference Complexity

During inference, all modules are activated to generate image captions. The overall inference complexity includes:

- The **object detection module**, derived from ODwGCN in OD-VR-Cap [12], enhances conventional detectors by incorporating co-occurrence relations via a GCN. However, during inference, this component functions as a fixed model and performs a single forward pass through the GCN over the detected objects. As such, the inference-time complexity is $\mathcal{O}(N_\mathcal{B})$, where $N_\mathcal{B}$ is the number of detected objects in the image.
- The **relationship graph construction module**, also reused from OD-VR-Cap [12], builds a scene-level relationship graph by applying deterministic rules over object pairs. This step involves pairwise evaluation and has worst-case time complexity $\mathcal{O}(N_\mathcal{B}^2)$.
- The **graph embedding module**, inherited from OD-VR-Cap [12] as well, uses the GraphSAGE architecture to embed the relationship graph. Since the GCN parameters are fixed, the complexity is determined by message passing operations over the graph. For $|\mathcal{V}|$ nodes and dimension $\mathrm{N_H}$, and N_G layers, the total cost is $\mathcal{O}(N_G.|\mathcal{V}|.N_H^2)$.
- The **transformer decoder**, used to generate the caption, has inference complexity $\mathrm{O}(N_D.N_A^2.N_H)$, where N_A is the caption length and N_D the number of decoder layers.

  Thus, the overall inference-time complexity of the framework can be approximated as:

$$\mathcal{O}(N_\mathcal{B}^2 + N_G.|\mathcal{V}|.N_H^2 + N_D.N_A^2.N_H)$$

Given that $N_\mathcal{B}$ is usually less than 20, N_A ranges from 10 to 15, and N_H is typically 512 or 1024, the framework remains computationally tractable and efficient in practical settings.

Given that $N_{\mathcal{B}}$ is usually less than 20, N_A ranges from 10 to 15, and N_H is typically 512 or 1024, and both N_G and N_D are commonly between 2 and 8, the framework remains computationally tractable and efficient in practical settings.

Regarding space complexity, the primary memory consumption stems from storing intermediate visual feature maps, static knowledge embeddings, and attention states maintained within the transformer decoder. Since the object detection, relationship graph construction, and graph embedding modules are all reused from previous work and remain frozen during both training and inference, the memory footprint is highly stable. Additionally, the relationship graph structure is compact and fixed, contributing to the model's efficient memory utilization. As a result, the overall space complexity of the framework remains low and well-suited for deployment in resource-constrained environments.

These complexity characteristics suggest that RGTranCNet is computationally efficient and scalable, with low memory requirements and a minimal training cost.

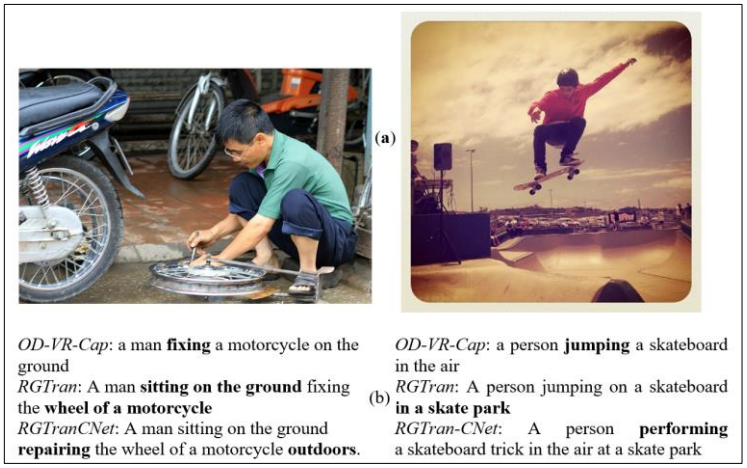

Figure 5. Example results from the test image set for the proposed method and OD-VR-Cap.

5. CONCLUSIONS

In this research, we present RGTranCNet, an innovative model for image captioning grounded in the encoder-decoder paradigm. It employs a transformer-based decoder equipped with cross-attention functionality, while integrating semantic insights from ConceptNet directly into the decoding phase. By harnessing the transformer's strength in modelling contextual dependencies and drawing on outside semantic resources, the system boosts both the precision and variety of the generated descriptions. Evaluations performed on the benchmark MS COCO dataset indicate that our framework surpasses contemporary efforts in terms of overall effectiveness. The addition of ConceptNet-derived semantics enhances the model's ability to craft more precise and contextually aligned descriptions, thereby improving the capabilities of automated captioning solutions. Importantly, this technique can be seamlessly incorporated into the decoding components of alternative encoder-decoder-based captioning systems to amplify their results. As such, the proposed solution proves both viable and applicable, laying the groundwork for advancing captioning technologies across diverse practical sectors. While our tests were confined to the MS COCO dataset, the methodology lends itself to adaptation for

other collections (such as Flickr8k or Flickr30k), given that training is limited to the decoder with other elements held constant.

Furthermore, the external knowledge infusion from ConceptNet operates independently of particular image traits, ensuring the method avoids over-reliance on the format of any given training set. Looking ahead, we intend to deploy and assess the model on these additional datasets to confirm its versatility. We also aim to incorporate inter-concept relationships from ConceptNet within the encoder to provide richer context and refine caption quality.

Acknowledgements. We acknowledge the support of the Institute of Mechanics and Applied Informatics - VAST, the Faculty of Information Technology and Telecommunications - Graduate University of Science and Technology - VAST, and the Faculty of Information Technology - Ho Chi Minh City University of Education (HCMUE) for this study.

CRediT authorship contribution statement. Nguyen Van Thinh: Methodology, Implementation, Investigation, and Writing. Tran Van Lang: Formal analysis, Supervision and Review. Van The Thanh: Supervision and Review.

Declaration of competing interest. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

1. Jamil A., *et al*. - Deep Learning Approaches for Image Captioning: Opportunities, Challenges and Future Potential, IEEE Access, 2024.
2. Verma A., *et al.* - Automatic image caption generation using deep learning, Multimedia Tools and Applications **83** (2) (2024) 5309-5325.
3. Kavitha R., *et al*. - Deep learning-based image captioning for visually impaired people, in E3S Web of Conferences, EDP Sciences, 2023.
4. Pavlopoulos J., Kougia V., and Androutsopoulos I. - A survey on biomedical image captioning, In: Proceedings of the second workshop on shortcomings in vision and language, 2019.
5. Szafir D. and Szafir D. A. - Connecting human-robot interaction and data visualization. in Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction, 2021.
6. Lin Y. J., Tseng C. S., and Hung-Yu K. - Relation-Aware Image Captioning with Hybrid-Attention for Explainable Visual Question Answering, Journal of Information Science & Engineering **40** (3) (2024).
7. Vinyals O., *et al.* - Show and tell: A neural image caption generator. in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.
8. Huang L., *et al*. - Attention on attention for image captioning, In: Proceedings of the IEEE/CVF international conference on computer vision, 2019.
9. Xu K., *et al*. - Show, attend and tell: Neural image caption generation with visual attention. in International conference on machine learning, 2015. PMLR.
10. Thinh N. V., Lang T. V., and Thanh V. T. - Automatic image captioning based on object detection and attention mechanism, In: The 16th National Conference on Fundamental and Applied IT Research, FAIR'2023, Natural Science and Technology Publishing House: Da Nang, 2023, pp. 395-404.

11. Anderson P., *et al*. - 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.
12. Thinh N. V., Lang T. V., and Thanh V. T., OD-VR-CAP: Image captioning based on detecting and predicting relationships between objects. Journal of Computer Science and Cybernetics **40** (4) (2024).
13. Xu N., *et al*., Scene graph captioner: Image captioning based on structural visual representation. Journal of Visual Communication and Image Representation, 2019. **58**: p. 477-485.
14. Thinh N. V., Lang T. V., and Thanh V. T. - A Method of Automatic Image Captioning Based on Scene Graph and LSTM Network, in The 15th National Conference on Fundamental and Applied IT Research (FAIR'2022), Natural Science and Technology Publishing House: Ha Noi, 2022, pp. 431-439.
15. Li Z., *et al.* - Modeling graph-structured contexts for image captioning, Image and Vision Computing **129 (**2023) 104591.
16. Vaswani A., *et al.* - Attention is all you need. Advances in neural information processing systems **30 (**2017).
17. Hendricks L. A., *et al*. - Deep compositional captioning: Describing novel object categories without paired training data, In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
18. Zhou Y., Sun Y., and Honavar V. G. - Improving Image Captioning by Leveraging Knowledge Graphs. IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, IEEE, January 7-11, 2019.
19. Hafeth D. A., Kollias S., and Ghafoor M. - Semantic representations with attention networks for boosting image captioning, IEEE Access. **11 (**2023) 40230-40239.
20. Patwari N. and Naik D. - En-de-cap: An encoder decoder model for image captioning, In: 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), IEEE, 2021.
21. Xie T., *et al.* - Bi-LS-AttM: A Bidirectional LSTM and Attention Mechanism Model for Improving Image Captioning, Applied Sciences **13** (13) (2023) 7916.
22. Chen S., *et al*. - Say as you wish: Fine-grained control of image caption generation with abstract scene graphs, In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020.
23. Yan J., *et al.* - Caption TLSTMs: combining transformer with LSTMs for image captioning, International Journal of Multimedia Information Retrieval **11** (2) (2022) 111-121.
24. Ramos L., *et al.* - A study of convnext architectures for enhanced image captioning, IEEE Access, 2024.
25. Wang Y., Xu J., and Sun Y. - End-to-end transformer based model for image captioning, In: Proceedings of the AAAI Conference on Artificial Intelligence, 2022.
26. Yang X., *et al*. - Context-aware transformer for image captioning, Neurocomputing **549 (**2023) 126440.

27. Li Z., Q. Su, and T. Chen - External knowledge-assisted Transformer for image captioning, Image and Vision Computing **140 (**2023) 104864.
28. Hamilton W., Z. Ying, and J. Leskovec - Inductive representation learning on large graphs, Advances in neural information processing systems **30 (**2017).
29. Speer R., J. Chin, and C. Havasi - Conceptnet 5.5: An open multilingual graph of general knowledge, In: Proceedings of the AAAI conference on artificial intelligence, 2017.
30. Lin T. Y., *et al*. - Microsoft coco: Common objects in context. in European conference on computer vision, Springer, 2014.
31. Karpathy A. and L. Fei-Fei - Deep visual-semantic alignments for generating image descriptions. in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.
32. Papineni K., *et al*. - Bleu: a method for automatic evaluation of machine translation. in Proceedings of the 40th annual meeting of the Association for Computational Linguistics, 2002.
33. Banerjee S. and 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.
34. Lin C. Y. - Rouge: A package for automatic evaluation of summaries, In: Text summarization branches out, 2004.
35. Vedantam R., C. Lawrence Zitnick, and D. Parikh - Cider: Consensus-based image description evaluation, In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.