SN Computer Science (2020) 1:229
https://doi.org/10.1007/s42979-020-00238-4

ORIGINAL RESEARCH

AACR: Feature Fusion Effects of Algebraic Amalgamation Composed Representation on (De)Compositional Network for Caption Generation for Images

Chiranjib Sur[1]

Received: 26 January 2020 / Accepted: 19 June 2020 / Published online: 8 July 2020

Abstract
Progress in image captioning is gradually getting complex as researchers try to generalize the model and define the representation between visual features and natural language processing. In the absence of any established relationship, every time a new dividend is added, it produced very little improvement, not considerable enough to make it general. This work tried to define such kind of relationship in the form of representation called Algebraic Amalgamation-based Composed Representation (AACR) which generalized the scheme of language modeling and structuring the linguistic attributes (related to grammar and parts of speech of language) which will provide a much better structure and grammatically correct sentence. AACR enables better and more unique representation and structuring of the feature space and enables transfer learning like infrastructure for all machines to interact with the external world (both human and machine) with these representations. A large part of the different ways of defining and improving these AACR are discussed and their performance concerning the traditional procedures and feature representations are evaluated for image captioning application. The new models achieved considerable improvement than the corresponding previous architectures.

Keywords Language modeling · Representation learning · Tensor product representation · Image description · Sequence generation · Image understanding · Automated textual feature extraction

Introduction

Image captioning [62] forms the lifeline of large number applications that require transferability between visual features, related to images and videos, and textual contents. While object detection was made possible through transfer learning from highly capable image classification models, there is a need to understand the happenings (events) in the images for better inference and help in the analysis of context, the recommendation of contextual similarity and decision making. This gradually drove the planning and modeling of image captioning inevitable for researchers in academia and industries, which will drive the next generation innovation in applications and services. Thus will enhance the reachability of these services to the unprivileged people in the form of assistance, guidance, and support. Apart from the inputs traditionally fed into the machines, the machine also needs to understand and classify the behavior of the end-users to make modern artificial successful. To understand means to detect the differences and to classify means to the characterization of these. This understands will help in the better deliverable and can provide better-personalized experience than the one-size-fits-all approaches, which hardly work for many people. Image captioning architectures [62] operated on feature generation and combinations for effective sentence generation from visual features like Vgg [29], ResNet [13, 21], Inception [79] etc, mostly relying on object and attribute detectors to describe images [29, 8, 14] and later focused on attention-based model [3, 91, 58, 79, 86, 13, 47] and semantic factorization [21], video captioning using Self-Aware Multi-Space Feature Composition Transformer [64], Bengali captioning for images [67], multi-role crossover [68], aiTPR [69], Coupled-Recurrent Unit [70], Tpsgtr [71], Semantic Tensor Product strategies [72, 73]. Recent works with top-down objects from image regions [46, 1, 71] have used hierarchical models.

✉ Chiranjib Sur
chiranjib@ufl.edu

[1] Computer and Information Science and Engineering Department, University of Florida, Gainesville, USA

Visual analysis and context understanding require more than the probability of co-occurrence of the artifacts and assembling all these combinations and make the machine understand them is an unfeasible task. When it comes to language modeling, Frederick Jelinek made the statement, "Every time I fire a linguist, the performance of the speech recognizer goes up." This supports the fact that manual patterns and rule-based approaches are challenging to drive systems towards artificial intelligence. Also, at the same time, the representation must be dynamic to adopt all possible combinations, whenever required, just like human beings. The focus emphasizes the development of representation to incorporate and reciprocate the intricacies of languages in the form of grammar, parts of speech and semantic leveling that computation can leverage. This can only be achieved through deterministic functional approximation and is a requirement for scalable systems with topological significant and structural data representation. However, the deterministic functional approximation is not sufficient as stochastic processes like adaptive gradient descent learning cannot be relied on to produce structurally correct and reasonable representations, instead most of the time they produce some useful ones, as mentioned by British statistician George Box, "All models are wrong, some are useful." Unless we put provable boundaries, theories, constraints, and ways to evaluate them, representation learning can never achieve what it is capable of.

As models are getting complex and their ability to gather and learn to assemble a large number of distinct features and their combinations, the number of network parameters is also increasing exponentially and reaches a few million. Training of such a model not only requires a considerable amount of data and computational resources but also involves lots of money, energy and human monitoring. In fact, selective training of a model ends up in the generation of discriminative features instead of generative ones [65] and the prospect to scale up fails and ineffective to handle unidentified or unseen objectives. In disguise, transfer learning

The rest of the document is arranged with problem description of language in "Existing Problem of Language Understanding", tensor product theory and representation capability in "Theory of Tensor Product", architectural details in "Architecture Description", description of the methodologies in "Methodology", results and analysis in "Results and Analysis", revisit of the existing works in the literature in "Literature Review", in "Discussion".

Our main contribution consists of the followings: 1) novel architecture and representations for sentence representations 2) the influence and effect of AACRs for different sentences and their constructions 3) the notion of feature decomposition and their interaction for sentences 4) ensemble of features fusion and how to effectively utilize the features and use them diversely to generate the different styles and context of sentences, a mode of naturalism for machines 5) traditional evaluation criteria is not always correct and need to have a better way of feedback evaluation of what is being learned by models and machines. Hence reinforcement learning is being used to performance enhancement 6) hierarchical stochastic decomposition of image features through learning to decompose.

Existing Problem of Language Understanding

The problem, that persists in images, language, and interchangeability, is the confusion created by the linguists through defining different rule-based language intricacies and baseless complexities. Present-day language generation model works on the probability of occurrence of the next word based on the previous word and is biased to the short term memory of the model. This lacks the proper appearance of the actors and the actions in the sentence that are relevant to the contexts (image here). Hence, we need some kind of generalized and robust structure that can make the machine sensitive to the variations it produces and, in reaction, can generate a very near correct sequence of objects and events detected in the image. However, the objects and events can not be perceived individually by the model but can be conveyed through a representation, which is an epitome of information that helps the machine generate the correct sequence.

However, a similar pattern of images also converges to similar kind of representation and machines getting adapted to such kind of invariant distribution makes the model inert and ineffective. Hence, the representation must be made robust so that a large number of different contexts can be represented in the framework without convergence and generalized, which allows non-trained contexts to have equally interpretable and distinct representation. Also, the model must have the capability to detect variations and differences. The representation is crucial, and a robust representation can never be generated directly as a transformation, but can only be generated when the original context features are decomposed and then recomposed to create the representation. This work is mainly focused on these kinds of heuristic schemes for decomposition and has illustrated these theoretical prospects through experimentation and statistical evaluations.

Literature Review

Image captioning had been solved in many different ways, including [45] from CNN features of images and hash-tags from users as input [46], template based approach where a sentence is generated with 'template' with slot locations. You et al. [93] discussed sentiment-conveying image

descriptions. Melnek et al. [50] reported the comparison of context-aware LSTM captioner and co-attentive discriminator for image captioning with conditional GAN training, enforcing semantic alignment between images and captions using two reinforcement learning training procedures known as self-critical sequence training (SCST) and Gumbel straight through (ST). This paper demonstrated that SCST behave in more stable gradient behavior and improved the effectiveness of captioning generation than Gumbel ST. Wu et al. [85] used question features and image features to generate question-related captions for visual question answer (VQA) dataset and the generated caption creates new knowledge for the VQA system. Here, a joint training occurred where the representation was learnt for both caption generation and VQA application and the joint training procedure helped in much better fit and generalization of the machine interpret-able representations. Kilickaya et al. [30] proposed a data-driven approach where the caption was generated based on the comparison made between the image and the other relevant training set images through the selection of a relevant image. The generated caption is derived out of deep learning framework. Here, object-based semantic image representation was used in a deep network as features to retrieve and select the relevant image(s). Chen et al. [9] introduced StructCap, where they used an extra set of features derived out the parsing tree that was created from the knowledge of the objects gathered from the visual features. The model parsed an image into key entities, derived their relations and organized them into a visual parsing tree. This visual parsing tree was transformed into an embedding using an sequence-to-sequence framework and visual attention. Jiang et al. [26] used a sequence-to-sequence framework by adding an extra set of component called guiding network, whose work was to introduce a feature space consisting of the different attributes from the images. However, the paper did not specify explicitly what attributes were used for their experiments, but it was clearly a multi-layer non-linear transformation from the images and constant training of this multi-layer non-linear parameters helped it fit the data in proper shape. Wu et al. [83] introduced a dual temporal modal which created a word-conditional semantic attention from word embedding for image caption generation. Word-conditional semantic attention was generated from object and attribute words from the images and a combination of these attributes word embedding was used for attention. Fu et al. [19] discussed image-text surgery for image description generation. Here, the model synthesized pseudo image-sentence pairs which were generated under the guidance of a knowledge base, with syntax from a MSCOCO data set and visual information from an existing large-scale ImageNet image base. Pseudo data helped in learning the novel concepts of the captioning model without any human-labeled pairs. This was far more autonomous than the crowd sourced data driven techniques. Chen et al. [3] introduced another attribute-driven attention for image captioning, where the attributes used were the objects detected in the images. However, a separate RNN network was used for detection of these good objects from the images in a sequence that can be favorable for better caption generation. Here, the model leveraged on co-occurrence dependencies among object attributes and used an inference representation based on it. Cornia et al. [12] reported image captioning approach in which a generative recurrent neural network was used to focus different sectors of the image during the generation of the caption, by exploiting the conditioning provided through a salient prediction model which was capable of distinguishing and segregating different parts of the image as salient and contextual. Zhao et al. [98] introduced an architecture named MLAIC which consisted of several components and cooperated to generate better representation that can be exploited for image caption generation. These components included a multi-objective (word and syntax classification) classification model that learned rich category-aware image representations using a CNN image encoder, a syntax generation model capable of learning better through syntax aware LSTM based decoder and lastly an image captioning model that generated image descriptions in text, sharing its CNN encoder and LSTM decoder with the object classification task and the syntax generation task. Here, the image captioning model was benefited from the additional object categorization and syntax knowledge and joint training for better representation. Li et al. [38] proposed a text-guided attention model for image caption where the attention was derived using the associated captions for training. A dataset associated with MS-COCO with Chinese sentences and tags was introduced. A recommendation-assisted collective annotation system was introduced which automatically correlate several tags and sentences as relevant with respect to the visual content. Chen et al. [4] introduced reference based long short term memory (R-LSTM) model which operated on references from images and solved the difficult problem of determination of which part of the images were essential and correlate with the sentences and this lead to mistraining during the training phase and during caption generation phase, it leads to misgeneration of caption. The reference scheme would gather information in prioritizing and characterizing the relevant information that can be related to sentence generation instead of just depending on transformation heuristics for everything to happen. Tavakoliy et al. [76] studied the difference between the bottom-up saliency-based visual attention and manual object referrals in scene description construction as image description is generated from them. Bottom-up saliency-based visual attention was generated from RCNN model, while manual description came from external involvements. Chen et al. [5] introduced Show-and-Fool, where they used crafted adversarial

examples for neural image captioning and studied the effect of adversarial conditions for the models and the robustness of language to adversarial deformations for machine perception based on its vision. This work was marked by the attempt whether adversarial training could help in effective caption generation. Ye et al. [92] discussed ALT, which worked on attentions based on the high-dimensional transformation matrix from the image feature space to the context vector space and used that processed matrix for caption generation, while the traditional models worked on learning spatial or channel-wise attention from the images, which were generated as part of the region based object detection. ALT was claimed to learn various relevant feature abstractions, including spatial attention, channel-wise attention and visual dependence. It combined global and local context vector along with attention probabilities for this purpose. Wang et al. [81] introduced a coarse-to-fine method where the image was used to generate series of skeleton sentence and its attributes and then use these skeleton sentence and attribute phrases to construct the caption for the image. All the skeleton sentence and attribute phrases came from decomposition of the image where the attributes were associated with skeleton sentences and when these attributes were incorporated into the caption, they generated much better captions. Chen et al. [6] discussed a caption generating system that could generate based on different specific styles including humorous, romantic, positive, and negative. The model was trained using a special set of data where the sentences, related to images, were categorized with specific categories. This kind of applications would help in describing the image content semantically accurately. Chen et al. [7] introduced a phenomenon where the model could incorporate information like structural relevance and structural diversity and accordingly produced image captions based what had been perceived. This helped in producing captions that contained diverse or relevant information into the sentences and thus moved towards an optimal collaborative captioning. Liu et al. [41] reported a model that utilized the multimodal attention model that was used as state-of-the-art sequence-to-sequence generator in machine translation scheme and the attention was composed of several sequence of detected objects feed in place of the original visual features to the encoder. Harzig et al. [23] introduced amodel that can generated captions and can also detect the popular brands in the images. This was mainly motivated by the fact that the caption must be able to generate certain descriptions of the different brands in the image. This kind of specific and customized captioning had very high impact in many businesses. Liu et al. [42] reported an image captioning model where a self-retrieval module was used as training guidance. The self-retrieval module helped in generation of discriminative sentences and generated gradient for additional learning session for the model. In comparison to reinforcement learning, this concept is similar but with a separate set of unlabeled images, whose diversification was utilized and also made to involve and incorporated into the training session of the model. Chunseong et al. [10] discussed a scheme for generation of descriptions for images through the use and aware of different user vocabularies accounting for prior vocabulary knowledge of such user through the usage of their previous documents. This was highly personalized scheme of image captioning being introduced and can be described a mimicry of a person and his/her style of writing. Sharma et al. [59] introduced a new dataset of image caption annotations and called it as Conceptual Captions with wider variety of captions for the images and contained enormous amount of images compared to the MS-COCO dataset, and also represented different specific varieties of both images and image caption styles. This data is capable of more specific identification of the happening and is more specific of the subcategories of the objects and even characterization of humans and celebrities. Yao et al. [90] experimented convolutional neural networks with recurrent neural networks image captioning framework for detection of describing novel objects in captions. This was another deviation effort being made from the traditional generalization towards personalization and specialization and construction of sentences with unique objects. Zhang et al. [97] studied actor-critic reinforcement learning based image captioning training where the optimization was achieved with non-differentiable quality metrics of interest like CIDEr, BLEU_N etc. The actor critic was achieved through a separate set of instrumental optimizer that acted on the model through a validation set other than the loss validation set. Fu et al. [18] introduced visual captioning with region-based image features as attention and with scene-specific contexts that could relate different specific places as context instead of general statements. This was also one kind of personalization where the caption generator will be able to definitely specify and recognize entities. Ren et al. [57] used a combination of policy network and a value network coordinate to generate sentence as description for images. Here, policy network served as a local embedding as a confidence of predicting the next word based on the current state, while value network provides the necessary global embedding or a look-ahead guidance, evaluating possibilities of extensions from the current state. Liu et al. [40] enhanced performance with prior MIXER approach as a reinforcement learning based training, that was mixing maximum likelihood estimation training with policy gradient, for image captioning through the use of a linear weighted combination of SPICE and CIDEr known as SPIDEr. Cohn-Gordon et al. [11] introduced a new concept that can provide image captions that can distinguish between similar kind of images and thus created the scope for diversification of the caption quality through attention representations and high end sensitivity of the models. This attention

was regarded as pragmatically information and its objective was far more realistic than just truth. Liu et al. [43] discussed an approach with the purpose of evaluating and improving the correctness of attention in neural image captioning models. Here, the correctness and evaluation was made on the selection of regional visual features of the image through network transformation while generating the caption based on the manually prescribed selection. Yao et al. [91] pioneered long short-term memory with attributes (LSTM-A) where there was successful hybridization of the convolutional neural networks and recurrent neural networks for image captioning and the whole process operated as a sequence-to-sequence or end-to-end manner. Lu et al. [44] introduced an adaptive attention model with a visual sentinel where the adaption was made on selection of the regions of the image through models and networks known as visual sentinel. Instead of providing rigid attention and unstructured attention, this architecture focused on adaptively changing the transformation function or selected function based on the progress of the generated caption. Vinyals et al. [78] studied generative model based on a deep recurrent architecture, where it used combination of the recent advancement in computer vision and machine translation, connecting computer vision with natural language processing through generation of natural sentences and describing images. Anderson et al. [1] introduced bottom-up mechanism for image regions based feature tensor, to be selected through Faster R-CNN, to be used as weighted features in a top-down model for image caption generation. Here, the combinations of the regions to be used was determined heuristically through a model and was dependent on the training session, without paying much attention on the sequentiality and correctness of the arrival and combinations. Zhang et al. [96] discussed an adaptive re-weight scheme for the loss of different samples to be used as optimization of the weights of the network. These re-weighted loss function was based on online positive recall and used two-stage optimization strategy. Park et al. [55] introduced personalized image captioning through the generation of descriptive sentences with prior knowledge of a person's habit of using specific words known as active vocabulary or even writing styles through estimation of the likelihood of the person's active words from previous documents. Wang et al. [80] used an sequence-to-sequence model with deep bidirectional Long Short-Term Memory component for image captions, where the images were transformed using a deep convolutional neural network and two separate LSTMs predicted the next generated word for captions. Rennie et al. [58] introduced a new optimization approach called self-critical sequence training (SCST), through estimating a baseline to normalize the rewards and reduce variance through utilization of the output of its own test-time inference and normalization of the rewards. Wu et al. [82] experimented high-level concepts attention of a CNN-RNN model and achieved considerable improvement on the state-of-the-art performance in both image captioning and visual question answering [66]. Here attribute prediction layer was used for high level semantic concept layer. Vinyals et al. [79] introduced a generative model based on a deep recurrent units combining recent advances of computer vision based visual features and machine translation based attention combinations for generation of natural and grammatically correct sentences describing an image. Karpathy et al. [28] proposed a bidirectional retrieval model capable of retrieving description of images through the construction of sentences through a deep, multimodal embedding of visual and natural language data, where they used the inner product of image segments and sentence fragment to create fragment similariy or image-sentence similarity. Xu et al. [86] discussed CNN features based attention model to describe the content and relationships among contents in the image to construct the descriptive sentences. Fang et al. [16] introduced image descriptor capable of visual detectors, language models, and multimodal similarity models. Here no image features were used, no RNN network but only word from objects for sentence generation. This model used multiple instance learning to train visual detectors for words that commonly occur in captions, including many different parts of speech such as nouns, verbs, adjectives etc. Karpathy et al. [29] studied a model consisting of convolutional neural networks for image region selection and bidirectional recurrent neural networks for sentence construction, trained with a datasets of images and their sentence descriptions. This model learned the intermodal connection between language and visual data and aligned the two modalities through a multimodal embedding. Anne et al. [2] introduced deep compositional captioner for the task of generating descriptions of novel objects that were not present in the training set as paired image sentence in dataset. This approach leveragedlarge object recognition datasets and external text corpora and through transferring knowledge between semantically similar concepts. Chen et al. [8] discussed a recurrent neural network by dynamically building a visual representation of the scene as a caption automatically. Here, the model learned to remember long-term visual concepts and generalized well for all the images and was capable of generating novel captions from visual features, and also reconstruction of visual features from an image description. Devlin et al. [13] introduced a pipeline combining a set of candidate words generated by a convolutional neural network being trained on images and a maximum entropy language model used to arrange these words into a coherent sentence. The penultimate activation layer of the convolutional neural network was used as input for the network for sentence generation. Donahue et al. [15] experimented an end-to-end trainable recurrent convolutional network architecture for benchmark

video recognition tasks for activity recognition, image description, retrieval problems and video narration or video description challenges. Gan et al. [20] introduced StyleNet that did the task of generating attractive captions for images and videos with different styles and the styles were gathered through attention. Jin et al. [27] studied a model exploiting the parallel structures between images and sentences. Here the process of generating the next word based on previous, was aligned with visual perception experience with shifting attention among the visual regions creating a sense of visual ordering. Kiros et al. [31] introduced a framework for learning distributed representations of attributes like characteristics of text based representations and can be jointly learned with word embedding. Kiros et al. [32] proposed a framework for distributed representations generation for word embedding while keeping in mind that we can also jointly learn other language attributes including document indicators like sentence representation vector, language indicators like distributed language representations and other meta-data and side information like characteristic traits including age, gender of a blogger etc or even some kind of representations for authors. It is considered as a third-order model where the word context and attribute information representation collaborate through multiplication to predict the sequence. Mao et al. [47] discussed two sub-networks based model consisting of deep recurrent neural network for sentences and a deep convolutional network for images and this multimodal layer, capable of interaction with each other, is known as m-RNN model because of the multimodal combination of features for attention at different levels of the network. Memisevic et al. [51] introduced a probabilistic model for learning rich, distributed representations of image through transformations. This model was trained to learn a generalized transformations of its inputs using a factorial set of latent variables. Pu et al. [56] studied representation based on variational autoencoder for the image representation to associate labels or captions. This was regarded as deep generative deconvolutional network and worked on the generated latent image feature with the help of decoder while convolutional neural network acted as a encoder for the image feature extraction and to approximate the distribution for the latent deep generative deconvolutional network features. This latent code was directly linked to generative models for labels generation. Socher et al. [60] introduced DT-RNN model through the use of dependency trees embedding for sentence generation. Here, the dependency trees were converted into a vector space in order to retrieve images that were described by those sentences. Sutskever et al. [74] discussed RNN model, trained with the new Hessian-Free optimizer by applying them to character-level language modeling tasks. Here, a new character level embedding was introduced and was derived from the words of the objects. The character level embedding stack was converted as a tensor used for modeling instead of the traditional image features. Sutskever et al. [75] introduced multi-layered long short-term to map the input sequence to a fixed dimension representation, and then another deep LSTM to decode the target sequence from the representation. LTran et al. [37] proposed an approach for spatiotemporal feature learning using deep 3-dimensional convolutional networks called 3D ConvNets and were trained on a large scale supervised video database. Tran et al. [77] introduced image caption system that automatically described images, generating high quality caption with respect to human judgments, out-of-domain data handling and low latency required in many applications. This deep vision model helped in detection of a broad range of visual concepts, entity recognition (that identifies celebrities and landmarks), and caption outputs. Wu et al. [84] proposed a method of incorporating high-level semantic concepts into the CNN-RNN approach instead of the traditional image features to text approach for image captioning and visual question answering applications. High-level information was refereed to word level and object level feature spaces. Yang et al. [87] discussed RNN decoders with both CNN and RNN encoders, where thought vectors were used as the input of the attention mechanism in the decoder. The review network organized a number of review steps with attention on the encoder hidden states and outputted a thought vector after each review step. You et al. [94] introduced a model with semantic attention, that learned to select different attention based on semantic concepts and fused them into hidden states and outputs of recurrent neural networks for caption prediction. Young et al. [95] studied visual denotations of linguistic expressions to define novel denotational similarity metrics for comapring different images for captions and were as beneficial as distributional similarities for two tasks as semantic inference. Farhadi et al. [17] introduced a space of meanings as attention of the network model. This space of meanings resided in between the space of sentences and the space of images for generation of caption from images. Gan et al. [21] pioneered a semantic compositional network (SCN) for sentence generation from images, where series of semantic concepts were utilized from the image as attention for the SCN model. The probability of semantic layer was used for composition of the parameters of LSTM network. The SCN network extended each weight matrix of the LSTM to an ensemble of tag-dependent weight matrices. Girshick et al. [22] introduced a R-CNN based model containing high-capacity convolutional neural networks for bottom-up region selection in order to localize and segment objects and in scarcity of labeled training data, a supervised pre-training for an auxiliary task was used for the regions, followed by domain-specific fine-tuning, yielding a significant boost in performance. Hodosh et al. [24] discussed captioning as a frame for sentence-based image annotation as the task of ranking a given pool of

captions. This was done through the association of the images with natural languages based sentences and based on what had been detected as objects and attributes in the images. Jia et al. [25] introduced an extension of the long short term memory called gLSTM through the use of semantic information extracted from the image as an extra attention to each unit of the LSTM block, aiming to guide the model towards caption generation that was highly correlated and tightly coupled to image contents. Here, semantic representation was generated using normalized Canonical Correlation Analysis scheme. Krishna et al. [34] proposed the Visual Genome dataset that helped in proper modeling the relationships and interactions among different components and attributes of images through generation of graphs with dense annotations of objects, attributes, and relationships within different images. Kulkarni et al. [35] introduced model for generation of natural language descriptions from images, where two different components, namely content planning and recognition algorithms helped automatically generate captions. Content planning helped in smoothing the output of computer vision-based detection while recognition algorithms helped in determination of the best content words to use to describe an image with the help of statistics mined out of large pools of visually descriptive texts. Li et al. [39] studied an effective phenomenon for automatic composition of image descriptions from the visual features and using web-scale n-grams, unlike previous works where thetask was retrieval of related pre-existing text relevant to the image. It pioneered the task of generation of sentences from scratch with n-gram word sequence as feed. Kuznetsova et al. [36] introduced a new tree based approach to composing expressive image descriptions, making use of the naturally occurring web images with captions. Two related tasks, image caption generalization and generation were investigated where the former was an optional subtask of the latter. The high-level concept of this approach was to leverage the phrases expressive as tree fragments from existing image descriptions and then composing the new description by selectively combining the extracted tree fragments. Mao et al. [48] discussed a transposed weight sharing scheme which enhanced the caption generation capability of the m-RNN model and more suitable for the novel concept learning task. The transposed weight sharing scheme was generated using an auto-encoder and the objects of the images that were present in the sentence. Mathews et al. [49] introduced a model that can describe an image based on different emotions like positive or negative sentiments using switching recurrent neural network with word-level regularization with sentiments. It was able to produce emotional image captions using a training session of only 2000+ training captions tagged with different sentimental emotions. Mitchell et al. [52] proposed a model capable of human like description of the images through a computer vision detector system. This model leveraged syntactically informed word co-occurrence statistics, the generator filters and constrains the noisy detection output to generate syntactic trees that can summarize the vision and correlation of the computer vision system. Ordonez et al. [53] demonstrated automatic image description methods using a large captioned photo collection, where the Flicker was queried using captions and then the images were filtered to gather one million images with associated visually relevant captions. Yang et al. [88] introduced a sentence generation strategy that transferred an images into description consisting of the most likely nouns, verbs, scenes and prepositions that made up the core sentence structure from them. These descriptions in the form nouns, verbs, scenes and prepositions were derived using state of the art trained detectors and were very noisy estimates of the attributes of the images.

Theory of Tensor Product

Tensor Product is the systematic composition of a series of tensors that can be utilized for special representation with structured interpretation and has nice algebraic properties that can retrieve the nearest composite components. However, for our applications, we are dealing with some special situation of tensor products where one of them consists of orthogonal structures and thus creating the perfectly orthogonal segments of feature space to represent the data and the cumulative representation can be well utilized for inference, while the reverse multiplication of the orthogonal representation can retrieve the original space representations. While there can be different ways of generation of tensor products, we have concentrated on deterministic approaches with Hadamard matrix and due to its limitations, we switched to deterministic approximation techniques for tensor product generation.

Tensor Product Representation

Tensor Product Representation (TPR) [63] and [54] has tremendous potential to revolutionize the language problem and lack of structured representations problem. Apart from being able to help in inference, TPR is marked by reversibility and unification of representation and rules in separate forms that can be controlled, learned and utilized. Mathematically, TPR is generated by the product of the functional representation $\mathbf{r}$ and the component representation $\mathbf{f}$. While component representation is uncontrolled due to global interpretation, functional representation $\mathbf{r}$ must help in understanding the pattern of the topology and their significance, help in the interpretation of the language attributes, and can be controlled to diversify the sentence construction. This is accomplished through structuring the functional representation $\mathbf{r}$

as unique and uncombined (such as orthogonal or orthonormal). We provided a brief description of the deterministic Tensor Product Representation formulation and how it can be utilized for natural languages. Tensor Product Representation uses the concept of linear independence with transformation and inverse transformation, assuming that the inner product will help localization. However, the linearly independence criteria can be relaxed and represent the tensor product with other semi-independent vectors and rely on the assumption that tensors are far apart to interfere and the high dimension of $\mathbf{f}$ (or if needed $\mathbf{r}$) will provide adequate independence space for each of them. Generalized TPR can be represented as $\mathbf{s}(\mathbf{w})$ as,

$$\mathbf{s}(\mathbf{w}) = \sum f_i \otimes r_i^T, \tag{1}$$

where $\mathbf{w}$ is the feature vector, and $\{\mathbf{w} \rightarrow \mathbf{f} : \mathbf{w} \in \mathbf{W}_e\}$ is the transformation, $\mathbf{W}_e$ is the raw features or the embedding vectors for features which minimizes the context function such as Word2Vec for k contexts as $W2V\_Fn = \min \sum_i \sum_{j=1}^{k} ||\mathbf{w}_i - \mathbf{w}_j||^2$, $\mathbf{r}$ is the independence imposer for the TPR. This kind of tensor product creates a combined representations of the whole feature space and is unique, reduced in dimension and with the following decoupling equation,

$$\mathbf{f} = \mathbf{s}(\mathbf{w}) \otimes \mathbf{r} = \mathbf{s} \otimes \mathbf{r}, \tag{2}$$

where we can generate back the complete original vector $\mathbf{w}$ from $\{\mathbf{f} \rightarrow \mathbf{w} : \min_{\forall i \in N} \arg(\{\mathbf{f}_i\} - \mathbf{f})\}$ without any error. We have N sample instances and $\min_{\forall i \in N} \arg(\{\mathbf{f}_i\} - \mathbf{f})$ points to the closest possible sample i. So what we can conclude that this $\mathbf{s}(\mathbf{w})$ instantiate a much better comprehensive and compressed state of the samples than the whole feature space and can be help many learning algorithms to create models that can understand and differentiate the representations without explicitly supervising it to learn that these are different and need to be differentiated. At the same time, the feature representation can be migrated to its original form in constant time. Previous approaches for transferring $\min_{\forall i \in N} \arg(\{\mathbf{f}_i\} - \mathbf{f})$ point to the closest possible sample $i \in N$ were cosine distances or nearest neighbor with distance norm. However, the same task is possible in constant time as a transformation $\max_{\forall i \in N} \arg \mathbf{W}_f \mathbf{f} = \mathbf{f}_i$ through posing the problem as a probability distribution as we tune our model to gradient error rectification and learning schemes.

Hadamard TPR

Hadamard TPR is generated with the assistance of orthogonal row vectors of Hadamard matrix as $\mathbf{r}$. As $\mathbf{r}$ is the predictable tensor, it helps in coding the continuous valued features into orthogonal forms and thus prevent mixing up the features and assists in re-generation and reciprocity. Hadamard matrix is a $(2^n \times 2^n)$ square matrix, consisting of $\{-1, 1\}$ and each of the rows are orthogonal to all the others. The consequence is that, it can be used to generate mutually independent vectors for the TPRs. Hadamard Code TPR was build on top of Hadamard Coded matrix using the following equations:

$$H_{2^n} = \frac{1}{c_{k-1}} \begin{bmatrix} H_{2^{n-1}} & H_{2^{n-1}} \\ H_{2^{n-1}} & -H_{2^{n-1}} \end{bmatrix} = \frac{1}{c_{k-1}} H_2 \otimes H_{2^{n-1}}, \tag{3}$$

$H_{2^n} \in \mathbb{R}_{0/1}^{nnnn \times nnn}$ mostly consists of zeros and ones. The rows and columns of $H_2 \otimes H_{2^{n-1}}$ are symmetric and form bases of Hadamard matrix where we have $\otimes$ as the Kronecker product, $\frac{1}{c_{k-1}}$ is the normalization factor, where $c_{k-1} = (\sum |x_i|^2)^{\frac{1}{2}}$ with Frobenius norm or L^2-norm of any row as the normalizing coefficient. If we consider $(k-1) = 2$, then the most fundamental Hadamard matrix with $c_2 = c_{(k-1)=2} = (\sum |x_i|^2)^{\frac{1}{2}}$ is denoted as the following:

$$H_2 = \{\frac{1}{c_2}\} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 0.707 & 0.707 \\ 0.707 & -0.707 \end{bmatrix}. \tag{4}$$

This matix forms the starting matrix for all other high dimensional Hadamard matrix generation.

In Hadamard Coding, the filler consists of multiplication of the Hadamard matrix row (Eq. 3) $r_i^T = (H_{2^n})_i$, and the individual feature representations $f_i = (\mathbf{W})_i$ from feature space $\mathbf{W}$ like in case of natural languages, $f_i = (\mathbf{W}_e)_{w_k}$ is the word embedding vector for corresponding word w_k from $\mathbf{W}_e$.

The Hadamard Code TPR individual is generated as an inner product of the rows $\{(H_{2^n})_{r_i} : i \in \{1, \ldots, p\}\}$ of $p-$level Hadamard matrix (with $\{1, p\}$ dimension) and $\{F_j : j \in \{1, \ldots, \lceil \frac{d}{q} \rceil\}\}$, the corresponding segment vector (with $\{q, 1\}$ dimension) of the $d-$dimensional features of the samples as denoted by $\{F_j\}\{(H_{2^n})_{r_i}\}^T$ to generate a $\{q, p\}$ matrix. Essentially we have $p = 2^k$, $p \geq \lceil \frac{d}{q} \rceil$ and symbolically $F = f(\mathbf{w})$. Therefore, the overall Hadamard Code TPR is denoted as

$$\mathbf{s}_H(\mathbf{w}) = \sum_i \{F_i\} \otimes \{(H_{2^n})_{r_i}\}^T, \tag{5}$$

where we can generate back the features $\mathbf{w}$ as $\mathbf{F} \rightarrow \mathbf{w}$ and

$$\mathbf{F} = \mathbf{s}_H \otimes (H_{2^n}). \tag{6}$$

This procedure helps in the easiest and efficient way of generating and dealing with tensor product representation through linear transformation of the weighted representations of the original features to the mutual orthogonal spaces. In addition, TPRs $\mathbf{s}_H$, generated from this procedure, have

very distinct, non-overlapping, and unique feature space for the samples. This created a discrete learning phenomenon, which sometimes goes against the variation tolerance and regularization compatible network-based training models. In such models, connectedness and relatedness, how insignificant it may be, is an inevitable part of the learning. This is why directly dealing with Hadamard Code TPR may not help, and there is some additional procedural requirement for the system to work. Next, we will describe the procedural flow for image captioning applications in detail, mainly dealing with natural languages.

Let we have sentence with word $w_1, \ldots, w_n$ and word embedding $\mathbf{W}_e \in \mathbb{R}^{v \times e}$, we can transfer one hot vector for each word w_i as $(\mathbf{W}_e)_i \mathbb{R}^{1 \times e}$, we have

$$\mathbf{s}_H = \sum (\mathbf{W}_e)_i * f_j, \tag{7}$$

for $w_j = i$ and $\mathbf{s}_H$ is the TPR.

Conversely, to retrieve the information from the TPR, for each $j \in N$, we have

$$(w_p)_j = \mathbf{s}_H * f_j, \tag{8}$$

and if we consider the nearest neighbor for $(w_p)_j$ in $\mathbf{W}_e$, we find that

$$\begin{aligned}(w_p)_j &= \arg\min_k \{(\mathbf{W}_e)_k \mid \min ||(\mathbf{W}_e)_k - (w_p)_j||\} \\ &= (\mathbf{W}_e)_{k=i} = (\mathbf{W}_e)_i = w_j.\end{aligned} \tag{9}$$

We have tested that the retrieval rate is 100% correct for word embedding like Word2Vec, GloVe for any dimension. The accuracy of the retrieval is not because of the dimension or the embedding, but due to the mutual orthogonal matrix which creates space for real f_i to be segregated when r_i is multiplied with $f_i r_i^T$ as $f_i r_i^T r_i$.

Properties and Characteristics of TPR

TPR can be generated in various ways and functional approximation has even created an enumerable number of opportunities for TPR generation that can facilitate specific applications and their needs. However, TPR is expected to satisfy a majority subset of these following properties as these will facilitate many computations privileges.

- Tensor product composition : $\mathbf{T} = \mathbf{f} \otimes \mathbf{r}$. Matrix multiplication based representation ensembles two identities into a common platform and helps the creation of an algorithm that abides by the utility and constraints. For example, in the case of language, there is grammar, contexts of thoughts, and words. Each of these reorganizes itself to form a sentence that is grammatically correct and makes sense. However, there are hardly any concrete and scalable rules, but the model must be able to produce and combine very near approximations, whose variations can be suppressed mathematically. So, if there are n number of r vector and m number of f vectors, there are possibilities of $m \times n$ number of $(\mathbf{r} \otimes \mathbf{f})$ combinations possible.
- Relative tensor interpretation : $\mathbf{T} = \mathbf{f} \otimes \mathbf{r} \neq \mathbf{T}' \forall\, \mathbf{T}' \in \mathbf{T}$, where $\mathbf{T}' = \mathbf{f}' \otimes \mathbf{r}$. Now, each sentence consists of several elements of topologically significant sequence of $(\mathbf{r} \otimes \mathbf{f})$ as the contributing features for the representation. As each $(\mathbf{r} \otimes \mathbf{f})$ is different, any linear and non-linear combinations forming T will be distinct provided r is mutually orthogonal to all other r for every f. However, each T is a relative point in representation space and the algebraic property helps to decompose and extract them.
- Reversibility : $\mathbf{T} = \varphi(\mathbf{f}, \mathbf{r})$ then $\mathbf{f} = \phi(\mathbf{T}, \mathbf{r})$ for some function φ and ϕ, considering $\mathbf{r}$ as the orthogonal composer. Reversibility is inevitable for natural language applications to counter the differences in structural comprehension and gap in between human interpretation and machine interpretation. Reversibility will help machine deal with the most efficient representation T and at the same time can effectively transform between the sentence and T and can transfer the knowledge with the external world. TPR has this property of reversibility and can be widely used in many applications for better effectiveness.
- Accountability : $\mathbf{T} = \varphi(\mathbf{f}, \mathbf{r})$ then $\mathbf{f}' = \phi(\mathbf{T}, \mathbf{r})$ for some function φ and ϕ, minimization of closeness is mandatory as $\min \delta(\mathbf{f}, \mathbf{f}')$ where δ is the distance metric for measurement. This accountability criteria will help in reverse check of what has been learned by the procedure and what does the representations represent. A prominent approximation can suppress the variations in T, arising due to approximations, through non-linear transformation and help in better accountability and reversibility.
- Compressible : $\dim(T) <<< (\dim(f_1) + \dim(f_2) + \ldots)$. Compression is another important criteria that will help in better modeling and quick optimization. Both Hadamard matrix-based TPR and approximation TPR from Deep Networks help in producing an effective representation. However, there are fundamental differences between the representation learning and representation generation. In representation learning, there is an effort to converge several diverge samples to similar feature space or to a distribution, which turns out to be a distribution learning. While in representation generation, each of the combinations of features is provided such a feature space that is unique and can help in Reversibility and Accountability.
- Generalization : $\mathbf{T} = \varphi(f_1, f_2, \ldots, \mathbf{r}), \phi(\mathbf{T}, \mathbf{r}) = (f_1, f_2, \ldots)$, where $\mathbf{T} \in \mathbb{S}$ with $\mathbb{S}$ as representation space, any new test case $\mathbf{T}' \in \mathbb{S}$ should have $\min \delta(\mathbf{f}_{\text{pred}}, \mathbf{f}_{\text{real}})$ with δ as distance metric or $\max \delta(\mathbf{f}_{\text{pred}}, \mathbf{f}_{\text{real}})$ with δ as similarity. Generalization helps in providing the perfect functional approximation and high accuracy for Reversibility and

Accountability for the model. While most of the modeling application expects that the model will learn to behave for the distribution of the data, representation learning deals with large number of unique tensors gathered as combination of symbolic features and also provide the functional approximation that can effectively decode other unavailable representations that are available in the space.

- Non-discriminatory approach for representation : Non-discriminatory approach is a global approach that is not specific to limited application, but the approach must also hold important contributions to other applications. So, the same approach can be interpreted for all applications for the creation of global representation among all systems.
- Mapping : $\mathbf{T} = \varphi(f_1, f_2, \ldots, \mathbf{r})$, $\phi(\mathbf{T}, \mathbf{r}) = (f_1, f_2, \ldots)$ for single topologically significance one to one mapping or $\phi(\mathbf{T}, \mathbf{r}) = \{f_1, f_2, \ldots\}$ for multiple topologically significance retrieve. Mapping is an important criteria for distinguishing different representations and hence one to one mapping is ideal, but reverse may demand many to one and depend on application. Like, for bag-of-word approaches, several bag-of-word will converge to the same representation if the words are considered in alphabetical order, while the reverse will consider them to ${}^{n}\mathbb{P}_{k} = \frac{n!}{(n-k)!}$ possible observations with $k = n$.
- Mandatory global initialization : $\mathbf{T} = \varphi(\mathbf{f}, \mathbf{r})$ then $\mathbf{f} = \phi(\mathbf{T}, \mathbf{r})$ for some function φ and ϕ and $\forall\ f = f_i$, i is unique. Mandatory global initialization of many features is required to support Reversibility criteria and there is requirement to involve transfer learning based pipeline for involvement of better feature representation that has been derived with context and other real world datasets. Mandatory global initialization will ensure enhanced prediction for Reversibility and also can interact with other applications which rely on similar benchmarks.

Approximation of Tensor Product Representation as Algebraic Amalgamation Composed Representation (AACR)

In the mathematical field of model theory, the amalgamation property is a property of collections of structures that guarantees, under certain conditions, that two structures in the collection can be regarded as substructures of a larger one. - Wikipedia. Algebraic Amalgamation Composed Representation (AACR) is a special case of Tensor Product Representation (TPR). It is made scalable and functionally enhanced by approximating the variations and thus deviating the tensors from being completely orthogonal and using a series of non-linearity and memory network parameter estimations. However, the AACR's effectiveness comes from the uniqueness of the feature space and the AACR itself. Even, the potential of the AACR to be able to help in generalization is immense as new representations get generated from contexts (image features) and can be said to have the same representation that could have been generated by the corresponding caption of that image. Consider an image I, with caption $\mathbf{H}$. Assume that a caption consists of n words including the start of a sentence and stop of a sentence. We define $\mathbf{H}_x = [(\mathbf{h}_x)_1, (\mathbf{h}_x)_2, \ldots, (\mathbf{h}_x)_n]$, where $(\mathbf{h}_x)_t \in \mathbb{R}^V$ is a one-hot encoding vector of dimension V and V is the size of the vocabulary. The length n usually varies from a caption to another caption. $\mathbf{W}_e \in \mathbb{R}^{d \times V}$ is a word embedding matrix, the i-th column of which is the embedding vector of the i-th word in the vocabulary; it is obtained by the Stanford GLoVe algorithm with zero mean. We varies the value of $d = 32, 50, 100$. The unbinding vector or the first attention $\mathbf{u}_t \in \mathbb{R}^d$ is derived from the image features and is given by

$$\mathbf{u}_t = \sigma_g(\mathbf{W}_{au}\mathbf{h}_{t-1} + \mathbf{W}_{su}\tilde{\mathbf{S}}_{t-1})\mathbf{U}, \tag{10}$$

where $\mathbf{U} \in \mathbb{R}^{d \times d}$ is a normalized Hadamard matrix as in Eq. 3; where $\sigma_g(.)$ is the logistic sigmoid function; $\mathbf{W}_{au} \in \mathbb{R}^{d \times 512}$, $\mathbf{W}_{su} \in \mathbb{R}^{d \times 1024}$, $\tilde{\mathbf{S}}_{t-1} \in \mathbb{R}^{d \times d}$ is the representational AACR of all the decoded words up to time $t-1$ and is defined by

$$\tilde{\mathbf{S}}_{t-1} = \sum_{i=1}^{t-1} \mathbf{W}_e(\mathbf{h}_x)_i \mathbf{r}_i^T = \sum_{i=1}^{t-1} \mathbf{x}_i \mathbf{r}_i^T, \tag{11}$$

where $\mathbf{r}_i$ is the role vector for word $\mathbf{x}_i$, and $\mathbf{r}_i^T$ is transpose of $\mathbf{r}_i$. Since a Hadamard matrix is an orthogonal matrix, we have $\mathbf{r}_t = \mathbf{u}_t$ at time $(t = 1, \ldots, n)$. The feature vector $\mathbf{q}_t \in \mathbb{R}^{2048}$ is given by

$$\mathbf{q}_t = \mathbf{v} \odot \sigma_g(\mathbf{W}_{av}\mathbf{h}_{t-1} + \mathbf{W}_{sv}\tilde{\mathbf{S}}_{t-1}), \tag{12}$$

where the operator $\odot$ denotes the Hadamard product (elementwise product); where $\mathbf{W}_{av} \in \mathbb{R}^{2048 \times 512}$, $\mathbf{W}_{sv} \in \mathbb{R}^{2048 \times d^2}$. The second attention of the image is considered as $\mathbf{S}_t \in \mathbb{R}^{d \times d}$ of word $\mathbf{x}_t$ is obtained by

$$\mathbf{S}_t = \sigma_h(\mathbf{C}_s\mathbf{q}_t), \tag{13}$$

where $\sigma_h(.)$ is the hyperbolic tangent function; $\mathbf{C}_s \in \mathbb{R}^{d \times d \times 2048}$. Alternate σ_h and σ_h helps in abstraction of the propagated information in discrete form to be beneficial for language decoder. The "filler vector" $\mathbf{T}_t = \mathbf{f}_t \in \mathbb{R}^d \rightarrow$ "unbound" from the AACR representation $\mathbf{S}_t$ with the "unbinding vector" $\mathbf{u}_t \rightarrow$ obtained by Eq. 14:

$$\mathbf{T}_t = \mathbf{S}_t\mathbf{u}_t. \tag{14}$$

Use LSTM to decode $\mathbf{f}_t(t = 1, \ldots, n)$. The input of the LSTM is $\mathbf{v}$ at $(t = 0)$, $\mathbf{f}_t$ at time $t(t = 1, \ldots, n)$, decision feedback $\mathbf{x}_{t-1}$ at time t. This is the overall architecture of the AACR

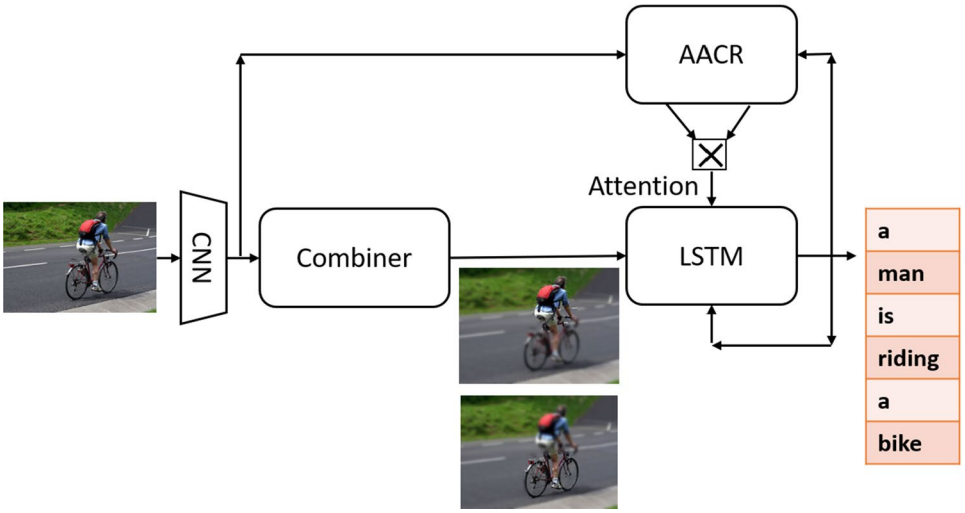

Fig. 1 Basic architectures with CNN features as prime source of feature generation

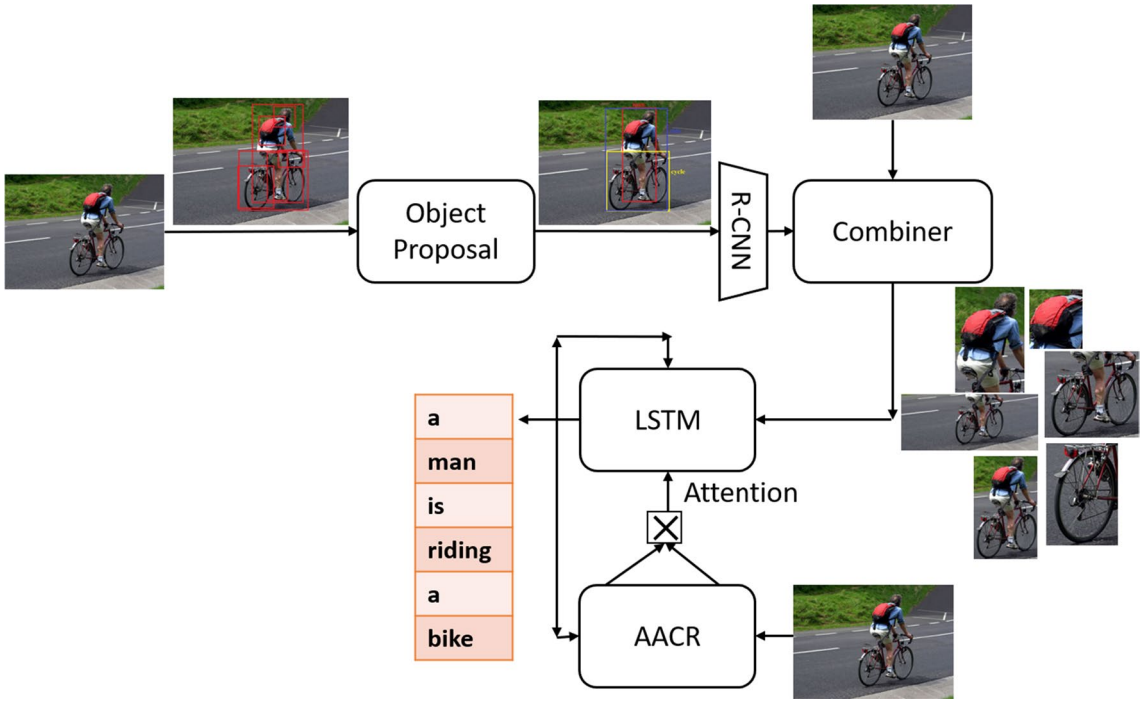

Fig. 2 Advanced architectures with RCNN features as prime source of feature generation and CNN as secondary sources

architectural details. Equation 14 produces the final Algebraic Amalgamation Composed Representation (AACR) for the model. The positional significance of AACR is demonstrated pictorially in Figs. 1 and 2. In Eqs. 10 and 12, the initial $\mathbf{h}$ comes with the initial states $\mathbf{c}_0$ and $\mathbf{h}_0$ are initialized as the followings and hence considered as a diversified image based attention, different from the regular one:

$$\mathbf{c}_0 = f_c(\mathbf{v}), \tag{15}$$

$$\mathbf{h}_0 = f_h(\mathbf{v}), \tag{16}$$

where functions $f_c(.)$ and $f_h(.)$ are realized by two separate multilayer perceptrons (MLPs) (say, 3 layers). Finally, LSTM produces a decoded word $(\mathbf{h}_x)_t \in \mathbb{R}^V$, $\mathbf{x}_t \in \mathbb{R}^d$ by

$$(\mathbf{h}_x)_t = \sigma_s(\mathbf{W}_x \mathbf{h}_t), \tag{17}$$

$$\mathbf{x}_t = \mathbf{W}_e(\mathbf{h}_x)_t, \tag{18}$$

where $\sigma_s(.)$ is a softmax function; $\mathbf{W}_x \in \mathbb{R}^{V \times 512}$.

In the end-to-end training, the objective function is a sum of the cross-entropy plus $Q(\mathbf{a}_t^{(u)})$, where function $Q(.)$ is defined by

$$Q(\mathbf{a}) = \sum (a_i)^2(1-a_i)^2 + (\sum (a_i)^2 - 1)^2. \tag{19}$$

Function Q generates a bias favoring attention vectors $\mathbf{a}_t^{(u)}$ that are 1-hot. The first term of Q is minimized when each component of $\mathbf{a}$ satisfies $a_i = [\mathbf{a}]_i \in \{0, 1\}$; the second term is minimized when $||a||_2^2 = 1$. The sum of these terms is minimized when $\mathbf{a}$ is 1-hot. Q drives learning to produce weights in the final network that generate $\mathbf{a}_t$ vectors that are approximately 1-hot, but there is no mechanism within the network for enforcing (even approximately) 1-hot vectors. Figure 3 has provided a diagrammatic overview of the AACR generator along with the other generator and decoder segments. Here, the whole set up works as a end-to-end network and helps in initialization of the decoder with proper direction related to the objects in the image. Whether, AACR can provide valuable inputs regarding the predicate of the sentence is yet to be estimated.

Architecture Description

Context from visual images consists of a diverse range of embedded objects, and when these are utilized for image captioning, a weighted combination of a regional or focused object gets highlighted as attention instead of the individuals. The main problem is a lack of activation for the memory network, which generates the likelihood of diverse activities and objects for the composition of the sentence. However, if the visual feature is decomposed through heuristics (transformation based focus) and then placed in the model as a generator or attention, it can help in better caption generation. In this section, we have featured different types of decomposition techniques, their utility, and their effectiveness and placement in the traditional memory networks. Our experimental approaches are also marked by analysis of the different decomposed feature fusion notions of the deep learning architecture and memory networks through utilization of these principles and provided a qualitative and quantitative comparison overview. Traditional LSTM, as a generator, is initialized with the visual features as $\mathbf{h}_0$ and $\mathbf{c}_0$ and represented by the following set of equitation:

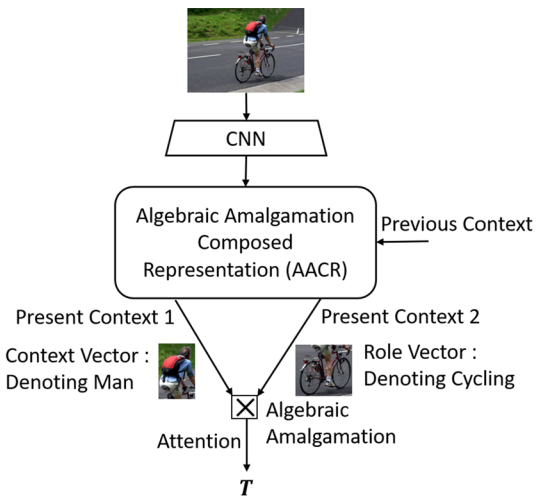

Fig. 3 Algebraic amalgamation composed representation (AACR) generator, here **T** is AACR

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i), \tag{20}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f), \tag{21}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o), \tag{22}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{xg}\mathbf{x}_t + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_g), \tag{23}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{24}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{25}$$

The estimation of the categorical likelihood is performed as

$$\mathbf{y}_t = \mathbf{h}_t\mathbf{W}_{hy}, \tag{26}$$

$\mathbf{y}_t$ is evaluated through convergence to the categorization distribution through softmax layer defined as

$$\mathbf{y}_t = \sigma(\mathbf{y}_t) = \frac{\exp(\mathbf{y}_t)}{\sum_{k=1}^{C} \exp((y_t)_k)}, \tag{27}$$

where we have $\sigma(\mathbf{y}_t) \in \mathbb{R}^C \in [0,1]^C$ with C as the set of categories. In addition, we have $\mathbf{x}_t \in \mathbb{R}^m$, $\mathbf{y}_t \in \mathbb{R}^C$, $\mathbf{W}_{hy} \in \mathbb{R}^{C\times d}$, $\mathbf{i}, \mathbf{f}, \mathbf{o}, \mathbf{g}, \mathbf{c} \in \mathbb{R}^d$, $\mathbf{A}_t \in \mathbb{R}^{m'}$, $\mathbf{h}_t \in \mathbb{R}^d$, $\mathbf{W}_{x*} \in \mathbb{R}^{d\times m}, \mathbf{W}_{A*} \in \mathbb{R}^{d\times m'}, \mathbf{W}_{h*}, \mathbf{W}_{c*} \in \mathbb{R}^{d\times d}, \mathbf{b}_* \in \mathbb{R}^d$. The objective minimization function is defined as $J(\mathbf{W}) = \arg\min \frac{1}{2s} \sum_{\forall s} \sum_i ||y_{t,i} - y'_{t,i}||^2 = \arg\min \frac{1}{2s} \sum_{\forall s} ||\mathbf{y}_t - \mathbf{y}'_t||^2$ and s number of training samples. The parameters are updated with $\alpha \frac{\partial J(\mathbf{W})}{\partial \mathbf{W}}$. Value of α determines the learning rate for adaption with the changing topology of the objective function space. Here, at any point of time to generate $\mathbf{h}_t$, attention is generated through previous sequential flow $\mathbf{h}_{t-1}$ and sequential evidence $\mathbf{x}_t$. The problem with $\mathbf{h}_{t-1}$ and $\mathbf{x}_t$ is that they are focused, unidirectional, biased and insensitive to variations. This is why we need additional focus features through decomposition and construct new context combinations for the generation of captions. LSTM network can be

made to behave as a decomposition and composition network architecture. However, here we are more interested in understanding the relative effects of the different external decomposed tensors and how they affect overall performance through statistical interpretation and qualitative evaluation of the difference in focus in the selection of words for sentences.

Hierarchical Stochastic Decomposition

Hierarchical Stochastic Decomposition decomposes the image features into a series of known vectors, which can be predicted to generate the sentence. However, this kind of heuristic decomposition requires the model to be able to correlate the image vector to the corresponding object representations. Hierarchical stochastic decomposition of image features can also be done through transfer learning like R-CNN. In this work, we have considered the decomposition of features using stochasticity of the learning of the models for better representation definition and gathering the representation for better decision making. Decomposition of features occurs when the image features are factorized and an approximate factorization occurs through weight multiplication, where the weights are learned. This kind of stochasticity and randomness in learning and selection helps generate many randomized and approximation algorithms to replace many deterministic algorithms for NP-hard problems. With the development of supervised learning, the models were upgraded to deterministic approximation algorithms through functional approximation and had a certain performance guarantee. Mathematically, the model consists of the following equations: $*= i/f/o/g$

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m}S \odot \mathbf{W}_{x,*n}\mathbf{x}_{t-1}, \tag{28}$$

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m}S \odot \mathbf{W}_{h,*n}\mathbf{h}_{t-1}, \tag{29}$$

where we have the LSTM equations as follows:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{b}_i), \tag{30}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{b}_f), \tag{31}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{b}_g), \tag{32}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{b}_o), \tag{33}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{34}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{35}$$

Figures 4 and 5 has provided an overview of the overall LSTM structure and the LSTM+AACR structure from a very high-level prospective.

Embedding + AACR

In this scheme, we decomposed the word Embedding only and analyzed its interaction with the AACR. The decomposition is done using an approximate factorization method where the weights decide which part of the image features to be selected for caption analysis. Mathematically, the model possesses the following set of equations: $*= i/f/o/g$

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m}S \odot \mathbf{W}_{x,*n}\mathbf{x}_{t-1}. \tag{36}$$

The notable portion is the decomposition of word embedding and AACR interaction:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{W}_{Ti}\mathbf{T} + \mathbf{b}_i), \tag{37}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{W}_{Tf}\mathbf{T} + \mathbf{b}_f), \tag{38}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{W}_{Tg}\mathbf{T} + \mathbf{b}_g), \tag{39}$$

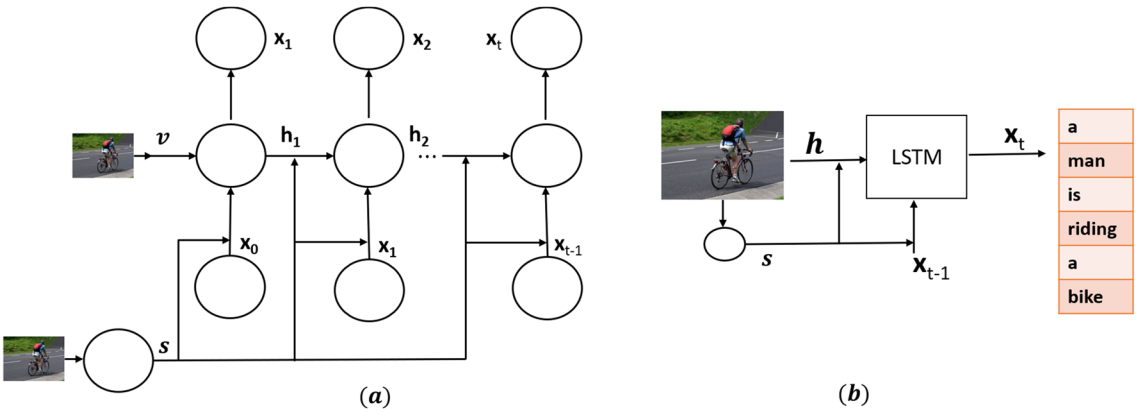

Fig. 4 Overall architecture with LSTM

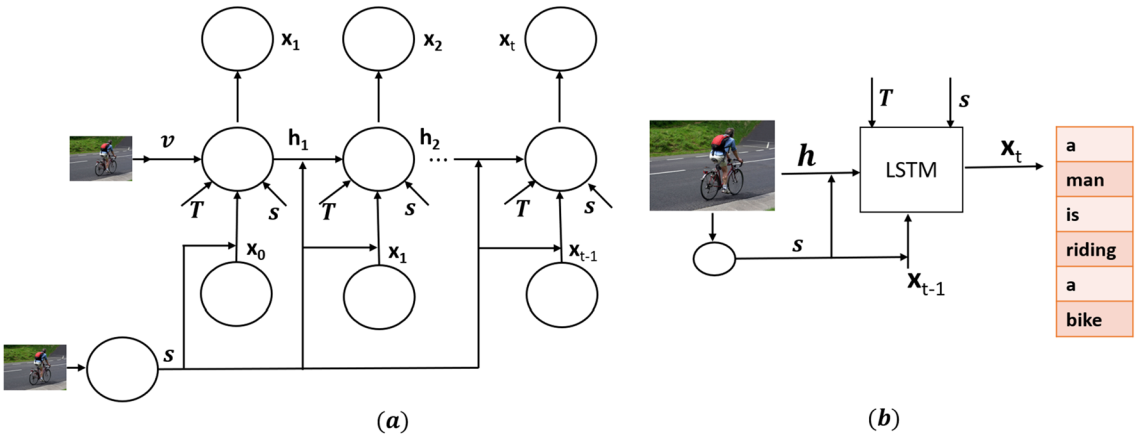

Fig. 5 Overall architecture with LSTM+AACR

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{To}\mathbf{T} + \mathbf{b}_o), \tag{40}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{41}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{42}$$

Decomposition of embedding and its interaction with the image features is something new and has never been tried before, and it will be interesting to observe the interactions and it outperformed some of the existing works in image captioning.

Hidden + AACR

This architecture is dedicated to the study of the interaction between the decomposed hidden layer and the AACR. We have denoted this scheme as (Hidden and AACR) in Table 1. Mathematically, the model equations are as follows: $*= i/f/o/g$:

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m}S \odot \mathbf{W}_{h,*n}\mathbf{h}_{t-1}. \tag{43}$$

Here, only $\mathbf{h}_t$ undergoes the factorization and interacts with the AACR for caption generation:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{W}_{Ti}\mathbf{T} + \mathbf{b}_i), \tag{44}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{W}_{Tf}\mathbf{T} + \mathbf{b}_f), \tag{45}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{W}_{Tg}\mathbf{T} + \mathbf{b}_g), \tag{46}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{W}_{To}\mathbf{T} + \mathbf{b}_o), \tag{47}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{48}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{49}$$

Theoretically, embedding decomposition may not help, also embedding helps in the determination of the next characteristics for captions and keeps the continuity of the grammatical topology for the sentences. While, the perception is that the decomposition of the hidden layer helps in the composition of fusion of the image features and the decomposed tag features weighted with hidden tensors. However, it is the embedding decomposition that performed well in terms of evaluation metrics. There is no way to determine which portion is helping the most for internal composition. However, since the performance metrics improved, we can claim that the analysis of the representation's structural composition is going on the right track.

Embedding + Hidden + AACR

If we combine Embedding and Hidden decomposition with the AACR, the performance of the LSTM gets enhanced as both the representation creates vibration in the model and with time, the LSTM model has learned to be sensitive and react to the variations of the parameters, while the decomposed Embedding and Hidden embedding and estimated weights have learned to help up meaningful information in-front of the model networks. The main important part of the Embedding and Hidden decomposition is the constantly changing or shifting mode of operation, which unsaturated the long short term model memory and helps prevent the appearance of similar kinds of word sequences as sentences. Mathematically, the equations for all decomposed models consist of the following: $*= i/f/o/g$:

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m}S \odot \mathbf{W}_{x,*n}\mathbf{x}_{t-1}, \tag{50}$$

Table 1 Performance evaluation for different LSTM architectures

Algorithm	CIDEr-D	Bleu_4	Bleu_3	Bleu_2	Bleu_1	ROUGE_L	METEOR	SPICE
Human [82]	0.85	0.22	0.32	0.47	0.66	0.48	0.2	–
Neural Talk [29]	0.66	0.23	0.32	0.45	0.63	–	0.20	–
Mind'sEye [8]	–	0.19	–	–	–	–	0.20	–
Google [79]	0.94	0.31	0.41	0.54	0.71	0.53	0.25	–
LRCN [15]	0.87	0.28	0.38	0.53	0.70	0.52	0.24	–
Montreal [86]	0.87	0.28	0.38	0.53	0.71	0.52	0.24	–
m-RNN [47]	0.79	0.27	0.37	0.51	0.68	0.50	0.23	–
[25]	0.81	0.26	0.36	0.49	0.67	–	0.23	–
MSR [16]	0.91	0.29	0.39	0.53	0.70	0.52	0.25	–
[27]	0.84	0.28	0.38	0.52	0.70	–	0.24	–
bi-LSTM [80]	–	0.244	0.352	0.492	0.672	–	–	–
MSR Captivator [13]	0.93	0.31	0.41	0.54	0.72	0.53	0.25	–
Nearest Neighbor [14]	0.89	0.28	0.38	0.52	0.70	0.51	0.24	–
MLBL [33]	0.74	0.26	0.36	0.50	0.67	0.50	0.22	–
ATT [94]	0.94	0.32	0.42	0.57	0.73	0.54	0.25	–
[82]	0.92	0.31	0.41	0.56	0.73	0.53	0.25	–
Adaptive [44]	1.085	0.332	0.439	0.580	0.742	–	0.266	–
MSM [91]	0.986	0.325	0.429	0.565	0.730	–	0.251	–
ERD [89]	0.895	0.298	–	–	–	–	0.240	–
Att2in [58]	1.01	0.313	–	–	–	–	0.260	–
Top-Down† [1]	1.054	0.334	–	–	0.745	0.544	0.261	0.192
[3]	1.044	0.338	0.443	0.579	0.743	0.549	–	–
LSTM [21]	0.889	0.292	0.390	0.525	0.698	–	0.238	–
SCN [21]	1.012	0.330	0.433	0.566	0.728	–	0.257	–
LSTM + S as Attention	0.910	0.317	0.424	0.557	0.716	0.532	0.240	0.172
LSTM + (Embedding + AACR)	1.001	0.332	0.437	0.573	0.736	0.543	0.256	0.184
LSTM + (Hidden + AACR)	1.018	0.334	0.437	0.573	0.736	0.545	0.257	0.186
LSTM + (Hidden + Embedding + AACR)	1.014	0.3352	0.439	0.575	0.737	0.546	0.257	0.1888
LSTM + (Embedding + dAACR)	1.022	0.338	0.443	0.578	0.737	0.546	0.256	0.1862
LSTM + (Hidden + dAACR)	1.0158	0.333	0.437	0.572	0.735	0.544	0.258	0.1864
LSTM + (Hidden + Embedding + dAACR)	1.006	0.331	0.435	0.570	0.734	0.542	0.256	0.186

† Ensemble and reinforcement learning used

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m} S \odot \mathbf{W}_{h,*n} \mathbf{h}_{t-1}. \tag{51}$$

This architecture is the most primitive LSTM and AACR attention architecture:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{W}_{Ti}\mathbf{T} + \mathbf{b}_i), \tag{52}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{W}_{Tf}\mathbf{T} + \mathbf{b}_f), \tag{53}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{W}_{Tg}\mathbf{T} + \mathbf{b}_g), \tag{54}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{W}_{To}\mathbf{T} + \mathbf{b}_o), \tag{55}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{56}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{57}$$

This architecture can be regarded as a AACR attention-based LSTM and it had already out-performed the basic architecture for the fusion of image features and the semantic tag features defined as SCN in [21]. Their model failed to achieve this level of performance because of the expansion of the model dimension. The increase of the model dimension diminishes the flow of knowledge in the network. In our model, we chained the knowledge and removed bottlenecks. These are the points, where structural composition and proper fusion of knowledge occur, which can be utilized and interpreted later. The AACR attention is a generalized tensor product representation and their interpretation is described in (Fig. 6).

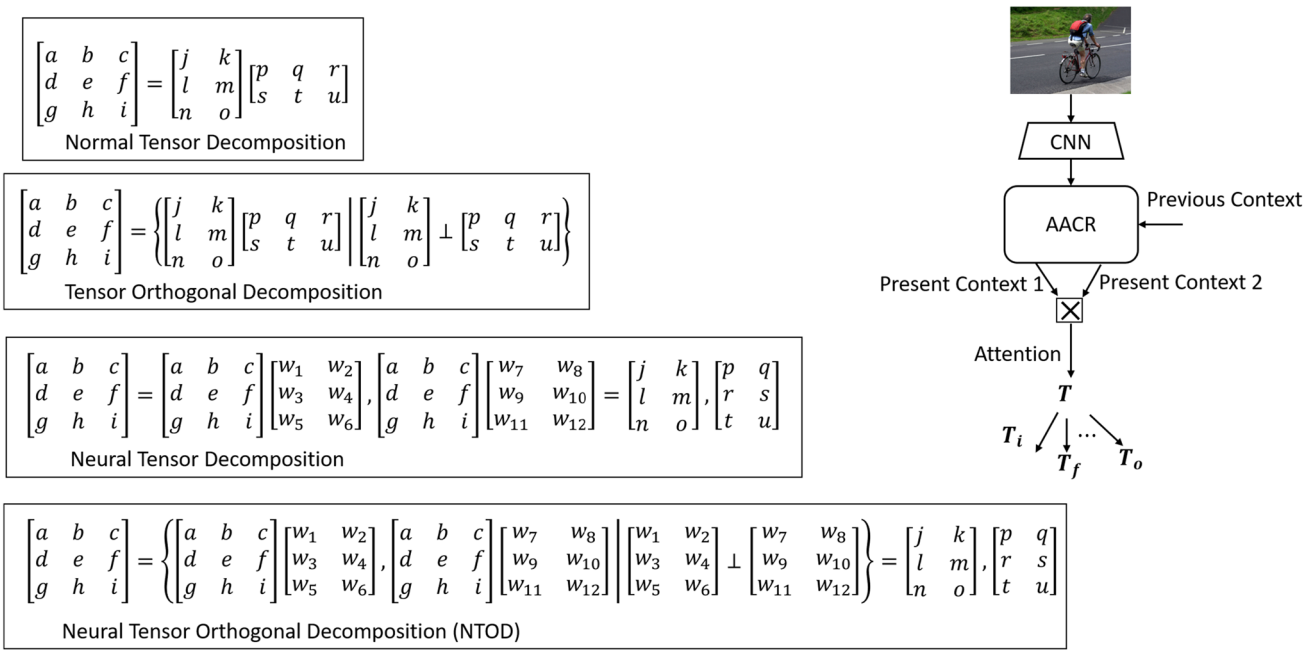

Fig. 6 Tensor decomposition concept illustration

Decomposed AACR

The next set of subsections includes decomposition of the AACR (dAACR) and its interaction with different other decomposed factors related to image features weighted hidden tensor and the word embedding. Decomposed AACR (dAACR) is concentrated with the decomposition of the AACR with the help of heuristic matrices like learned weights, which are dependent on the training and do not change with each iteration. Thus this kind of approximate decomposition tends to absorb similar topological regions again for interpretation and caption generation. While AACR is a holistic representation, a decomposed AACR will focus on different aspects and will provide better information for the memory network to dig up: $*= i/f/o/g$:

$$\mathbf{T}_* = \mathbf{P}_{*m} S \odot \mathbf{P}_{*n} \mathbf{T}. \tag{58}$$

Figure 7 provided an diagrammatic overview of the decomposed AACR based architecture, which is denoted by the following extra equation to replace $\mathbf{T}$ in LSTM equations.

Embedding + dAACR

Similar to the above, this Embedding and dAACR architecture studied the interaction of the decomposed AACR and decomposed word embedding with the hidden states. Mathematically,

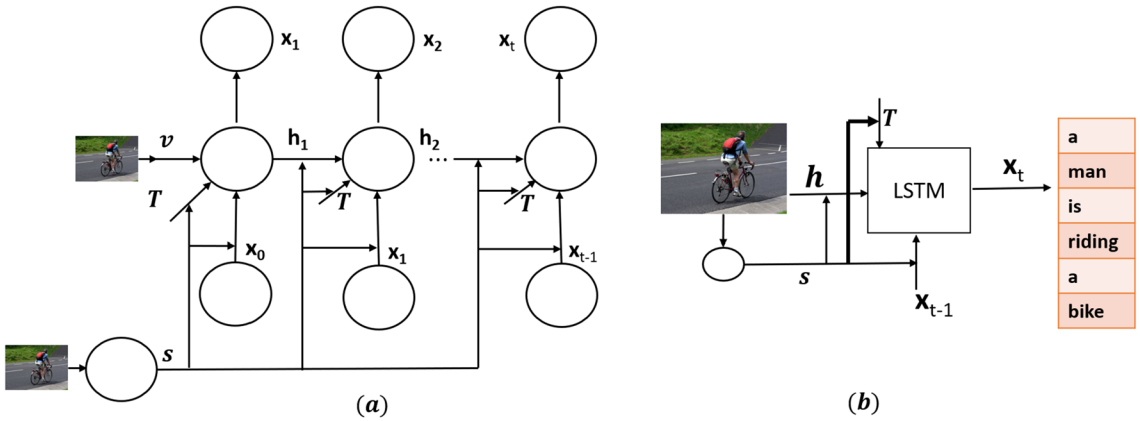

Fig. 7 Overall architecture with decomposed AACR

the equations for this model possesses the following set of equations: $*= i/f/o/g$

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m}S \odot \mathbf{W}_{x,*n}\mathbf{x}_{t-1}, \tag{59}$$

$$\mathbf{T}_{*} = \mathbf{P}_{*m}S \odot \mathbf{P}_{*n}\mathbf{T}, \tag{60}$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{W}_{Ti}\mathbf{T}_i + \mathbf{b}_i), \tag{61}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{W}_{Tf}\mathbf{T}_f + \mathbf{b}_f), \tag{62}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{W}_{Tg}\mathbf{T}_g + \mathbf{b}_g), \tag{63}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{To}\mathbf{T}_o + \mathbf{b}_o), \tag{64}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{65}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{66}$$

This architecture performed the best and generated BLEU_4 value of 33.8% and clearly established the fact that the decomposition of the embedding helped. The main reason is that most of the embedding is getting trained and updated frequently during training. A decomposition will actually help in refining the structural integrity for the topological dependency of the words in the sentence.

Hidden + dAACR

(Hidden and dAACR) has its hidden states decomposed along with AACR for its interaction embedding for generation of meaningful representation for captions. Mathematically, we can describe the model with the following series of equations: $*= i/f/o/g$

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m}S \odot \mathbf{W}_{h,*n}\mathbf{h}_{t-1}, \tag{67}$$

$$\mathbf{T}_{*} = \mathbf{P}_{*m}S \odot \mathbf{P}_{*n}\mathbf{T}, \tag{68}$$

where we have the LSTM equations as follows:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{W}_{Ti}\mathbf{T}_i + \mathbf{b}_i), \tag{69}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{W}_{Tf}\mathbf{T}_f + \mathbf{b}_f), \tag{70}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{W}_{Tg}\mathbf{T}_g + \mathbf{b}_g), \tag{71}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{W}_{To}\mathbf{T}_o + \mathbf{b}_o), \tag{72}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{73}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{74}$$

This architecture studies the interaction of the decomposed hidden layer with the decomposed tensors. This architecture, with decomposed hidden and dAACR, established itself better than the decomposed hidden and only AACR.

Embedding + Hidden + dAACR

Lastly, we have Embedding and Hidden and dAACR, where all the components factorized for structure combination generation for captions. This architecture is a very weighted architecture than the previous ones. Mathematically, the model equations are as follows: $*= i/f/o/g$:

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m}S \odot \mathbf{W}_{x,*n}\mathbf{x}_{t-1}, \tag{75}$$

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m}S \odot \mathbf{W}_{h,*n}\mathbf{h}_{t-1}, \tag{76}$$

$$\mathbf{T}_{*} = \mathbf{P}_{*m}S \odot \mathbf{P}_{*n}\mathbf{T}. \tag{77}$$

These weighted parts are plugged into the memory network for enhancement of captions as follows:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{W}_{Ti}\mathbf{T}_i + \mathbf{b}_i), \tag{78}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{W}_{Tf}\mathbf{T}_f + \mathbf{b}_f), \tag{79}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{W}_{Tg}\mathbf{T}_g + \mathbf{b}_g), \tag{80}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{W}_{To}\mathbf{T}_o + \mathbf{b}_o), \tag{81}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{82}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{83}$$

This architecture sees all the decomposition of hidden, embedding, and AACR. Though it diversified the signatures of the sentences, it is not optimal with respect to the evaluated metrics.

Extra Image Attention Effect

It would be worthwhile to add extra image attention to understand the effect on the decomposed structures, and hence, we also experimented with this. Image feature is used as attention to keep a whole image overview for the model all the time and to decide upon instead of dealing with a particular subset region or decomposed region representation. Mathematically, the model is established with the following set of equations: $*= i/f/o/g$

$$\mathbf{x}_{*,t-1} = \mathbf{W}_{x,*m} S \odot \mathbf{W}_{x,*n} \mathbf{x}_{t-1}, \tag{84}$$

$$\mathbf{h}_{*,t-1} = \mathbf{W}_{h,*m} S \odot \mathbf{W}_{h,*n} \mathbf{h}_{t-1}, \tag{85}$$

$$\mathbf{T}_{*} = \mathbf{P}_{*m} S \odot \mathbf{P}_{*n} \mathbf{T}, \tag{86}$$

$$\tilde{S} = \mathbf{W}_{S} S. \tag{87}$$

The rest of the equations are similar. $\mathbf{W}_S$ is used to match the hidden dimension of the memory network and may not be require. Non-transformed image features had been found to be beneficial than a weighted transformed one:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_{i,t-1} + \mathbf{W}_{hi}\mathbf{h}_{i,t-1} + \mathbf{W}_{Ti}\mathbf{T}_i + \tilde{S} + \mathbf{b}_i), \tag{88}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_{f,t-1} + \mathbf{W}_{hf}\mathbf{h}_{f,t-1} + \mathbf{W}_{Tf}\mathbf{T}_f + \tilde{S} + \mathbf{b}_f), \tag{89}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_{g,t-1} + \mathbf{W}_{hg}\mathbf{h}_{g,t-1} + \mathbf{W}_{Tg}\mathbf{T}_g + \tilde{S} + \mathbf{b}_g), \tag{90}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_{o,t-1} + \mathbf{W}_{ho}\mathbf{h}_{o,t-1} + \mathbf{W}_{To}\mathbf{T}_o + \tilde{S} + \mathbf{b}_o), \tag{91}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{92}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \tag{93}$$

We have compared these models and demonstrated that this modification on the LSTM along with the AACR attention and factorization of the components helped in the better structural enhancement of the tensors that can be transformed into sentences and these sentences can reflect diverse contexts, the interaction of objects in images and series of image description.

Architecture Description with RCNN Features

While we concentrated our work on the fundamental approach of image feature-based captions, we used regional CNN (RCNN) features extensively. RCNN provided an effective way of securing better combinations of contexts. Hence, we propose another approach, where we involved the RCNN features and performed similar experiments as [46] with the same vocabulary (8791 words) and surpassed their performance. Work in [1] used a vocabulary of 10,010 words, where the model is trained with data from both VQA, MSCOCO and Visual Genome and hence direct comparison will be totally unfair and we reached very close to that performance. They generalized the model with multiple-source data training and then fine-tuned the weights for different applications and hence the performance is much better. We initialized the training with scratch. In this section, we will discuss the AACR with RCNN architecture, while the language decoder can be replaced by each of the architectures, described in "Architecture Description" for both AACR and dAACR. Figure 2 provided a detailed pictorial description of the AACR+RCNN model, where different levels and variations of attention fuse in the language decoder. The novelty of this work is that we engage different sources of features, starting from the whole image features, which provide a global overview of the situation, while the variations and the lower level details and combinations are generated through the help of the weighted summation of the regional features and also through the AACR decomposition of the sparse CNN features, considered as a semantic probability distribution of the situation of the image. The equations for this model can be subdivided as the following components: Combination Selection Through Global Overview, Weighted Combination of the Regions, AACR and Language Decoder.

Combination Selection Through Global Overview

The first component aims at identifying the global overview of the image features and helps in identification of the weighted sum and conversion of the hidden information into topological selection for the recurrent unit. Though this component is not directly related to the context of the language decoder, it helps preserve the sequentiality to a large extent. Combination Selection Through Global Overview can be denoted as the following equations,

$$\overline{v} = \frac{1}{k} \sum_{i=1}^{i=k} v_i, \tag{94}$$

$$\overline{v} = \mathbf{v}. \tag{95}$$

The initial parameters are initialized as the followings:

$$\mathbf{h}_0, \mathbf{c}_0 = \mathbf{W}_{h_0}\overline{v}, \mathbf{W}_{c_0}\overline{v}, \tag{96}$$

$\mathbf{a}_t \in \mathbb{R}^{b \times d}$, $\mathbf{a}_t \in \mathbb{R}^{b \times d}$ However, we used the traditional mean of the objects RCNN features for more comprehensive understanding of the global view instead of the whole image based view, which is reduced through transformation.

Weighted Combination of the Regions

Weighted Combination of the Regions is the intermediate transfer layer and can be denoted as the followings:

$$\mathbf{a}_t = \mathbf{W}_a \tanh(\mathbf{W}_h \mathbf{h}_{t-1}), \tag{97}$$

where $\mathbf{a}_t \in \mathbb{R}^{b \times d}$, $\mathbf{a}_t \in \mathbb{R}^{b \times d}$, $\mathbf{a}_t \in \mathbb{R}^{b \times d}$. The value of $\mathbf{a}_t$is transferred into probability distribution through softmax

operation as the following to prevent over-bulging of the image features:

$$\alpha_t = \text{softmax}(\mathbf{a}_t), \tag{98}$$

where we have $\mathbf{a}_t \in \mathbb{R}^{\times} \in \{a_{1,t}, \ldots, a_{k,t}\}$ and $\sum \alpha_t = 1$. Finally, the attention context in place of hidden layer context for the language decoder is denoted as the followings:

$$\widehat{v}_t = \sum_{i=1}^{i=k} v_i \alpha_{i,t}. \tag{99}$$

Here, we have $\widehat{v}_t \in \mathbb{R}^{b \times d}$ where b is the batch size and d is the hidden layer dimension.

Language Decoder with AACR

Overall, we derived the following set of contexts: $\mathbf{q}_t$ as detailed lower level overview, $\mathbf{p}_t$ as previously generated context, and $\mathbf{T}_t$ as the bounded feature variation:

$$\mathbf{q}_t = \widehat{v}_t, \tag{100}$$

$$\mathbf{p}_t = \mathbf{W}_e \mathbf{x}_{t-1}. \tag{101}$$

Equation 14 provides the final version of the AACR $\mathbf{T}_t$ as

$$\mathbf{T}_t = f_1(\mathbf{U}, \mathbf{h}_{t-1}, \sum_{i=0}^{t-1} \mathbf{W}_e \mathbf{x}_i,) \otimes f_2(\mathbf{v}, \mathbf{h}_{t-1}, \sum_{i=0}^{t-1} \mathbf{W}_e \mathbf{x}_i), \tag{102}$$

where $\otimes$ is an algebraic operation. Here, we considered $\otimes = \odot$ as we try to rectify one context with the other context. Finally, we have the Assembled Selector Layer with Language Decoder with AACR attention component. The equations for Language Decoder and AACR can be denoted as the followings: $* = i/f/o/g$

$$\mathbf{p}_{*,t} = \mathbf{W}_{p,*m} S \odot \mathbf{W}_{p,*n} \mathbf{p}_t, \tag{103}$$

$$\mathbf{q}_{*,t} = \mathbf{W}_{q,*m} S \odot \mathbf{W}_{q,*n} \mathbf{q}_t, \tag{104}$$

$$\mathbf{T}_{*,t} = \mathbf{P}_{*m} S \odot \mathbf{P}_{*n} \mathbf{T}_t. \tag{105}$$

We operated different combination of the above three equations to generate the variations, like in "Architecture Description" to find out which tensor factorization provides maximum benefit.

$$\mathbf{i}_t = \sigma(\mathbf{W}_{pi} \mathbf{p}_{i,t} + \mathbf{W}_{qi} \mathbf{q}_{i,t} + \mathbf{W}_{Ti} \mathbf{T}_{i,t} + \mathbf{b}_i), \tag{106}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{pf} \mathbf{p}_{f,t} + \mathbf{W}_{qf} \mathbf{q}_{f,t} + \mathbf{W}_{Tf} \mathbf{T}_{f,t} + \mathbf{b}_f), \tag{107}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{po} \mathbf{p}_{o,t} + \mathbf{W}_{qo} \mathbf{q}_{o,t} + \mathbf{W}_{To} \mathbf{T}_{o,t} + \mathbf{b}_o), \tag{108}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{pg} \mathbf{p}_{g,t} + \mathbf{W}_{qg} \mathbf{q}_{g,t} + \mathbf{W}_{Tg} \mathbf{T}_{g,t} + \mathbf{b}_g), \tag{109}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{110}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{111}$$

$$\mathbf{x}_t = \max \arg \text{softmax}(\mathbf{W}_{hx} \mathbf{h}_t). \tag{112}$$

where we define $\mathbf{x}_t$ as the decoded words of the sentences at time t and the qualitative evaluation of MSCOCO data is provided in Table 2.

Methodology

Apart from defining the best of the features and the top-notch learning capable models, there are training criteria that need to be fulfilled to get the best out of the models. This section mainly deals with several such issues and the influence of each of them on the overall performance enhancement, achieved through their incorporation. These are several minor tweaks that can remove the shackle of over-fitting for classification problems and enhance the ability to detect diverse variations in representation structures for longer sentences with descriptive attributes.

Table 2 Performance evaluation for different LSTM architectures with RCNN features

Algorithm	CIDEr-D	Bleu_4	Bleu_3	Bleu_2	Bleu_1	ROUGE_L	METEOR	SPICE
NBT† [46]	1.07	0.347	–	–	0.755	–	0.271	0.201
LSTM + (Embedding + AACR) + RCNN	1.069	0.349	0.455	0.590	0.747	0.554	0.264	0.196
LSTM + (Hidden + AACR) + RCNN	1.068	0.349	0.456	0.592	0.748	0.554	0.264	0.195
LSTM + (Hidden + Embedding + AACR) + RCNN	1.067	0.348	0.453	0.588	0.746	0.554	0.264	0.195
LSTM + (Embedding + dAACR) + RCNN	1.059	0.349	0.457	0.593	0.748	0.554	0.261	0.193
LSTM + (Hidden + dAACR) + RCNN	1.052	0.347	0.456	0.593	0.747	0.556	0.261	0.192
LSTM + (Hidden + Embedding + dAACR) + RCNN	1.071	0.3502	0.459	0.595	0.751	0.555	0.264	0.195
LSTM + (Hidden + Embedding + dAACR) + RCNN + RL	1.075	0.353	0.458	0.593	0.752	0.557	0.267	0.197

Application Description

Image captioning not about generating sentences from the detected, but being able to derive different attributes and their interrelated interactions with each other and the environment. While some of the applications related to image captioning is mainly focused with clustering similar kinds of images and these image captions roughly define such characteristics, we analyze on defining captions with attributes like adjectives (hairy dog), descriptions (standing in rain), color (blue candy) and precision (like police instead of person). All these will elaborate on different narration and precise descriptions of the images than mere object detection. While the tag features correlate the objects and limited activities, the AACR structure (or even the decomposed tensors) should be able to capture other information for the model and express in the captions.

Dataset Preparation

MS COCO is being used for the experiments and analysis and this data is perhaps the most comprehensive data available. MSCOCO consists of 123,287 train images and 566,747 train sentence, where each image is associated with at least five sentences from a vocabulary of 8791 words. There are 5000 images (with 25,010 sentences) for validation and 5000 images (with 25,010 sentences) for testing. Two sets of image feature being used: one is ResNet features with 2048 dimension feature vector and another is Tag features with feature vector of 999 dimension. Tag contributes more when used with image features, and the correlation-based fusion has been the turning point for these image captioning application. This is evident from the fact that, (Tag + Tag) fusion in LSTM achieved 32.7% BLEU_4 without AACR (even 32.8% BLEU_4 with AACR), (Img + Img) fusion in LSTM achieved 26.0% BLEU_4 without AACR, (Tag + Img) fusion in LSTM achieved 33.0% BLEU_4 without AACR (even 33.5% BLEU_4 with AACR). The maximum is achieved in (Tag + Img) fusion combination, and the rest of the results is provided in Table 1.

Different Tensor Regularization

Tensor Regularization through dropout is an essential part for extensive variation identification and learning for the models. Memory network directly assimilates different features to create representation, but without dropout it is equivalent to weighted average without estimation of importance. Hence, a significant amount of dropout is required for each of them instead of involving a common framework for the entry point of the network. Several dropouts independently feature out different combinations and estimation of generalization for different sentences. The optimum amount is kept at 0.5 because of the fact that 0.5 helps in protecting at least more than 50% of the feature participation for the tensors and thus, the scarcity of essential contents of the features can be avoided while at the same time opportunity is created for changes in parameters. Another important contribution of the regularization is enhancement of the sensitivity of the model to the different variations of the features and the capability to lock these variations.

Normalization of Images Features and Word Vector

To mitigate the effect of the diverse ranges of different image features, transfer learning model features are useful and have diverse effects due to the topological and positional differences of the objects of the images. Normalization of images features through mean-shifting helps positioning the features to zero mean scaling and neutralizes the variational effects, thus helping the models in effective learning. It achieved an improvement of $(100 \times 1/27 = 3.7\%)$. Similarly, we can introduce a global vector for all machines, just like vocabulary is the same for all persons, and communication will be feasible for machines in terms of interpretation, storage, and retrieval. While, sometimes, locally trained embedding may be better for specific models. Normalization of the Word Vector helped in 1.5% improvement in BLEU_4 accuracy which is an improvement of $(100 \times 1.5/27 = 5.56\%)$ improvement.

Beam Search

Beam Search helped in improvement in BLEU_4 performance which is an improvement of $(100 \times 2.5/28.0 = 8.93\%)$. We observed that any improvement in BLEU_4 metric could also increase other metrics substantially and hence our approaches and experiments targeted this. Beam Search helped in the exploration of other feasible options that can be considered for generating the sentence. Beam search mainly helps in the detection of the attributes and interaction parameters of the images, which would have been suppressed due to the appearance of other objects and characteristics. However, Beam Search is an exhaustive process and is dependent on the selection of the spreading (beam) parameter. We mostly used a 5 beam for our experiments, but increasing the beam size can provide better results for natural language applications, where the generated sequence is of variable length, and longer sentence generation has been observed to be more descriptive and with more details.

Results and Analysis

Several experiments were performed on the MSCOCO dataset to establish the effects of different feature fusions generated through stochastic decomposition and the

recombination of the memory network. We evaluated with a variety of assessment techniques to see whether the enhancement in performance is overall in a different dimension, as quality evaluation of language parameters is still undefined and is in perception with a wide variety of differences among humans. The metrics used for such assessment are mainly Bleu_n ($n = 1, 2, 3, 4$), METEOR, ROUGE_L, CIDEr-D, and SPICE to measure the overall sentence fluency. All these evaluation metrics reflect some kind of stature of language, while none of them actually reflect the complete significance and grammatical correctness. We also provided some qualitative evaluations with instances of the generated captions from different models.

Quantitative Analysis

In this subsection, we will be discussing the performance evaluation based on the metrics Bleu_n ($n = 1, 2, 3, 4$), METEOR, ROUGE_L, CIDEr-D, and SPICE as these are standardized in the research community for image captioning research. However, none of the evaluation is complete and reflects a very limited perspective of the generated captions. Table 1 provided a comparative study of our models with some of the existing works in this domain using these features. It would also be an injustice to compare other enhanced models that have used other feature vectors and provided the improvements. With this set of features, this work can be regarded as the state-of-the-art performance with an effort to structure and characterize data features and generate longer and more descriptive captions. The main functional characteristics of our work are the decomposition of the generated/trained structural features and then composed of new representation through different circumstances of weights and combinations, thus being able to derive a better strategy for decoding and generating the sentences. Most of our new architectures performed very well and either outperformed or at least the same with the existing architectures, which do not have concrete reasoning behind their working principles. However, LSTM + (Embedding + dAACR) is the clear winner (with respect to BLEU_4) without Reinforcement Learning based training enhancement. With SCST reinforcement learning, LSTM + (Hidden + Embedding + AACR) emerged as the best (with respect to BLEU_4), establishing that AACR-based solutions are useful and can derive beneficial structural qualities for the system.

Reinforcement Learning Effects

Self-critical sequence training (SCST) [58] utilizes reinforcement learning [61] for gathering improvements in the performance. SCST utilized the gradient of the difference in performance between the generated and the referenced or baseline captions. In our case, SCST reinforcement learning is based on the CIDEr-D of the reference sentence and the generated caption. The batch-normalized CIDEr-D score is used to update the gradient for the updation of the prior weights of the models. Since reinforcement learning is heuristic-based, it never guarantees improvement and less dependable. However, we can always improve these decomposition architectures with reinforcement learning but requires more experiments. It must be mentioned that SCST based reinforcement learning can promise improvement, but never guarantee improvement. However, it has been found that more experiments can sometimes provide improvement, which is not feasible without high-end GPUs and time. SCST based reinforcement learning can be denoted with these equations,

$$\frac{\delta L(\mathbf{w})}{\delta \mathbf{w}} = -\frac{1}{2b}\gamma \sum_{i} \Phi(\mathbf{y}, \mathbf{y}'), \tag{113}$$

$$\frac{\delta L(\mathbf{w})}{\delta \mathbf{w}} = -\frac{1}{2b}\gamma \sum_{i} \Phi(\{y_1, \ldots, y_c\}, \{y'_1, \ldots, y'_c\}), \tag{114}$$

where $\Phi(.)$ is the evaluation function or the reward function that evaluates certain aspects of the generated captions $\{y_1, \ldots, y_c\} \in \mathbf{y}$ and the baseline captions $\{y'_1, \ldots, y'_c\} \in \mathbf{y}'$ and b is the mini-batch size considered. Table 3 provided some evaluations where SCST based reinforcement helped in the improvement of the performance and achieved better BLEU_4 for architectures.

Qualitative Analysis

The statistical evaluation metrics used hardly reflect many aspects of languages, including meaning, grammar, correct part-of-speech, etc. These can only be evaluated through reading, and hence, this subsection is inevitable and the essential part of this research. This will also provide a comparison in terms of diversity and descriptive attributes. There are no models that can evaluate the quality of writing of any language and whether the language denotes what it should have conveyed. Figures 8 and 9 have provided some examples and comparative instances that are generated by different models and how they reflect some true form of what is being reflected in the images and can be little far from what is there as baseline or reference.

Effects of Topic Attention

Well-trained visual captioning models can produce contextual effectiveness and evaluations based on BLEU_4 etc never justify the structural learning and word importance. While the style of writing and special influences are

Table 3 Performance evaluation for different LSTM architectures with reinforcement learning

Algorithm	CIDEr-D	Bleu_4	Bleu_3	Bleu_2	Bleu_1	ROUGE_L	METEOR	SPICE
LSTM + S as Attention	0.989	0.331	0.436	0.572	0.733	0.542	0.253	0.182
LSTM + (Embedding + AACR)	0.991	0.334	0.440	0.576	0.738	0.544	0.253	0.184
LSTM + (Hidden + AACR)	1.001	0.335	0.440	0.576	0.738	0.547	0.255	0.184
LSTM + (Hidden + Embedding + AACR)	1.002	0.338	0.442	0.578	0.738	0.546	0.255	0.185
LSTM + (Embedding + dAACR)	1.003	0.336	0.441	0.577	0.738	0.545	0.255	0.184
LSTM + (Hidden + dAACR)	0.998	0.336	0.439	0.573	0.734	0.543	0.254	0.184
LSTM + (Hidden + Embedding + dAACR)	0.997	0.336	0.443	0.580	0.741	0.545	0.254	0.184

Fig. 8 Qualitative analysis. Part 1

Fig. 9 Qualitative analysis. Part 2

equally important for the model, it involves in the removal of language bias for sentence construction and creates scope for better representation variation learning. Hence, in this section, we will investigate topic modeling related to natural languages and understand whether it is possible to influence different topics for these sentences effectively. Different attentions can create different ways of interpretation of the images and this is whether the machine must develop distinction capability and leverage on it. In humans, this comes naturally due to feeling and emotion, as they describe the same, differently, based on the situations. The main aim of this section is to understand the diverse topic modeling attention and the corresponding selection of vocabulary for sentence.

Effects of Topic Attention Based on Blog Authorship

Authorship influence, in sentence and caption generation, is helpful to reduce the machine biases for sentence patterns. This establishes that different authors have different structure and vocabulary choices, and the machine can easily learn them. We considered 40 different topics related to different topics. We used the Blog Authorship Corpus dataset for this purpose. This dataset consists of blog posts of thousands of bloggers gathered from blogger.com. Each blog contains a minimum of 200 occurrences of commonly used English words.

Fig. 10 Diversity effects of label propagation algorithm for visual captioning

Effects of Topic Attention Based on Newsgroups

Another important breakdown of the natural languages is in the form of the Newsgroups categories. The categorical significance of the news articles lies in the different distinct topic models of the society and will play an important role in the future in the strategic generation of specialized articles that serve the purpose of a special group of people and won't sound like the other. We used the Twenty Newsgroups dataset, which contains information about newsgroups categories, each with 1000 articles, taken from 20 different newsgroups.

Label Propagation Algorithm for Visual Captioning

Label Propagation Algorithm (LPA) describes the procedure for the generation of the different topic labels for visual captioning training data and later use the topic labels for the guided generation of captions from test images. The Label Propagation Algorithm procedures are organized in the following sequence:

- Select Labeled Language Data.
- Generate the Vocabulary from Language Data.
- Select Top X% most frequently appearing Words from Vocabulary for each Topic. [Should include the 8.7K Words from the MS COCO vocabulary]
- Generate 0/1 Topic-Vector for the classes based on the Top X% most frequently appearing Words from Vocabulary for a topic.
- Generation 0/1 Topic-Vector: From 8.7K Word Vocabulary, consider 0 for all those disjoint words that represent other Topics, rest considered 1.
- Analysis of the generated captions with 0/1 Topic-Vector.

The algorithm, provided above, detailed the topic modeling and label-propagation algorithm for caption generation instead of gathering the topic-based training captions. This label propagation is based on existing trained models and existing topic definition and detection techniques. The label propagation accounts on the fact that the distribution of the most frequently appearing words and their complex representations are generated from combinations and will help in the propagation of the knowledge of topics for the testing data. Since there is no reference for these kinds of data, we limit our analysis of qualitative analysis. Figure 10 provided some of the instances where this kind of topic models generated totally out of the box captions and can be regarded as very close to the truth for the image, while the influence is

gathered from the topic models propagated from other data. We used the best model, which is LSTM + (Embedding + dAACR), for the caption generation using the topic attention vocabulary selection.

Discussion

In this work, we discussed some improvements to the existing structures of memory networks and feature decomposition and demonstrated that our approaches are better than previous actions in all the possible metrics. We nurtured AACR and its interaction with other informative structures of the memory network and leveraged for variation generation and for identification of the attributes and interaction in images to appear in the sentences. Our mission is for better representation and can be generalized for media features and help machines understand what is happening in the images. While AACR succeeded in gathering improvement, we utilized different decomposition techniques for the composition of structures, which are as good as introducing reinforcement learning to some architectures. The future works can be concentrated on introducing more sophistication of the features and introduction of other useful components and structuring the data that can be differentiated by the models and generalize the representation to its unique sentence counterparts.

Acknowledgements The author has used University of Florida HiperGator, equipped with NVIDIA Tesla K80 GPU, extensively for the experiments. The author acknowledges University of Florida Research Computing for providing computational resources and support that have contributed to the research results reported in this publication. URL: eduhttp://researchcomputing.ufl.

Compliance with Ethical Standards

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Anderson P, He X, Buehler C, Teney D, Johnson M, Gould S, Zhang L. Bottom-up and top-down attention for image captioning and visual question answering. CVPR. 2018;3(5):6.
2. Anne HL, 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, pp. 1–10.
3. Chen H, Ding G, Lin Z, Zhao S, Ha J. Show, observe and tell: attribute-driven attention model for image captioning. In: IJCAI, 2018, pp. 606–12.
4. Chen M, Ding G, Zhao S, Chen H, Liu Q, Han J. Reference based LSTM for image captioning. In: AAAI, 2017, pp. 3981–87.
5. Chen H, Zhang H, Chen PY, Yi J, Hsieh CJ Show-and-fool: crafting adversarial examples for neural image captioning. arXiv preprint. 2017; arXiv:1712.02051.
6. Chen T, Zhang Z, You Q, Fang C, Wang Z, Jin H, Luo J. Factual or emotional: stylized image captioning with adaptive learning and attention. arXiv preprint. 2018; arXiv:1807.03871.
7. Chen F, Ji R, Sun X, Wu Y, Su J. GroupCap: group-based image captioning with structured relevance and diversity constraints. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1345–53.
8. Chen X, Lawrence Zitnick C. Mind's eye: a recurrent visual representation for image caption generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 2422–31.
9. Chen F, Ji R, Su J, Wu Y, Wu Y. Structcap: structured semantic embedding for image captioning. In: Proceedings of the 2017 ACM on multimedia conference, ACM, 2017, pp. 46–54.
10. Chunseong Park C, Kim B, Kim G. Attend to you: personalized image captioning with context sequence memory networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 895–903.
11. Cohn-Gordon R, Goodman N, Potts C. Pragmatically informative image captioning with character-level reference. arXiv preprint. 2018; arXiv:1804.05417.
12. Cornia M, Baraldi L, Serra G, Cucchiara R. Paying more attention to saliency: image captioning with saliency and context attention. ACM Trans Multimed Comput Commun Appl. 2018;14(2):48.
13. Devlin J, et al. Language models for image captioning: the quirks and what works. arXiv preprint. 2015; arXiv:1505.01809.
14. Devlin J, Gupta S, Girshick R, Mitchell M, Zitnick CL. Exploring nearest neighbor approaches for image captioning. arXiv preprint. 2015; arXiv:1505.04467.
15. Donahue J, et al. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 2625–34.
16. Fang H, et al. From captions to visual concepts and back. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1473–82.
17. Farhadi A, et al. Every picture tells a story: generating sentences from images. In: European conference on computer vision, Springer, Berlin, Heidelberg, 2010.
18. Fu K, Jin J, Cui R, Sha F, Zhang C. Aligning where to see and what to tell: Image captioning with region-based attention and scene-specific contexts. IEEE Trans Pattern Anal Mach Intell. 2017;39(12):2321–34.
19. Fu K, Li J, Jin J, Zhang C. Image-text surgery: efficient concept learning in image captioning by generating pseudopairs. IEEE Trans Neural Netw Learn Syst. 2018;99:1–12.
20. Gan C, et al. Stylenet: generating attractive visual captions with styles. In: Proceedings of the IEEEConference on Computer Vision and Pattern Recognition, 2017, pp. 3137–46.
21. Gan Z, et al. Semantic compositional networks for visual captioning. arXiv preprint. 2016; arXiv:1611.08002.
22. Girshick R, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014. pp. 580–87.
23. Harzig P, Brehm S, Lienhart R, Kaiser C, Schallner R. Multimodal image captioning for marketing analysis. arXiv preprint. 2018; arXiv:1802.01958.
24. Hodosh M, Young P, Hockenmaier J. Framing image description as a ranking task: data, models and evaluation metrics. J Artif Intell Res. 2013;47:853–99.

25. Jia X, et al. Guiding the long-short term memory model for image caption generation. In: Proceedings of the IEEE international conference on computer vision, 2015, pp. 2407–15.
26. Jiang W, Ma L, Chen X, Zhang H, Liu W. Learning to guide decoding for image captioning. arXiv preprint. 2018; arXiv :1804.00887.
27. Jin J, et al. Aligning where to see and what to tell: image caption with region-based attention and scene factorization. arXiv preprint. 2015; arXiv:1506.06272.
28. Karpathy A, Armand J, Fei Fei FL. Deep fragment embeddings for bidirectional image sentence mapping. In: Advances in neural information processing systems, 2014, pp. 1889–97.
29. Karpathy A, Fei-Fei L. Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015. pp. 3128–37.
30. Kilickaya M, Akkus BK, Cakici R, Erdem A, Erdem E, Ikizler-Cinbis N. Data-driven image captioning via salient region discovery. IET Comput Vis. 2017;11(6):398–406.
31. Kiros R, Ruslan S, Zemel RS. Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint. 2014; arXiv:1411.2539.
32. Kiros R, Zemel R, Salakhutdinov Ruslan R. A multiplicative model for learning distributed text-based attribute representations. Adv Neural Inf Process Syst. 2014.
33. Kiros R, Salakhutdinov R, Zemel R. Multimodal neural language models. In: International conference on machine learning, 2014, pp. 595–603.
34. Krishna R, et al. Visual genome: connecting language and vision using crowdsourced dense image annotations. Int J Comput Vis. 2017;123(1):32–73.
35. Kulkarni G, et al. Babytalk: understanding and generating simple image descriptions. IEEE Trans Pattern Anal Mach Intell. 2013;35(12):2891–903.
36. Kuznetsova P, et al. TREETALK: composition and compression of trees for image descriptions. TACL. 2014;2(10):351–62.
37. LTran D, et al. Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4489–97.
38. Li X, Wang X, Xu C, Lan W, Wei Q, Yang G, Xu J. COCO-CN for cross-lingual image tagging, captioning and retrieval. arXiv preprint. 2018; arXiv:1805.08661.
39. Li S, et al. Composing simple image descriptions using web-scale n-grams. In: Proceedings of the fifteenth conference on computational natural language learning. Association for computational linguistics, 2011.
40. Liu S, Zhu Z, Ye N, Guadarrama S, Murphy K. Improved image captioning via policy gradient optimization of spider. Proc IEEE Int Conf Comput Vis. 2017;3:3.
41. Liu C, Sun F, Wang C, Wang F, Yuille A. MAT: a multimodal attentive translator for image captioning. arXiv preprint. 2017; arXiv:1702.05658.
42. Liu X, Li H, Shao J, Chen D, Wang X. Show, tell and discriminate: image captioning by self-retrieval with partially labeled data. arXiv preprint. 2018); arXiv:1803.08314.
43. Liu C, Mao J, Sha F, Yuille AL. Attention correctness in neural image captioning. In: AAAI, 2017, pp. 4176–82.
44. Lu J, Xiong C, Parikh D, Socher R. Knowing when to look: adaptive attention via a visual sentinel for image captioning. Proc IEEE Conf Comput Vis Pattern Recognit. 2017;6:2.
45. Lu D, Whitehead S, Huang L, Ji H, Chang SF. Entity-aware image caption generation. arXiv preprint. 2018; arXiv:1804.07889.
46. Lu J, Yang J, Batra D, Parikh D. Neural baby talk. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7219–28.
47. Mao J, et al. Deep captioning with multimodal recurrent neural networks (m-rnn). arXiv preprint. 2014; arXiv:1412.6632.
48. Mao J, et al. Learning like a child: fast novel visual concept learning from sentence descriptions of images. In: Proceedings of the IEEE international conference on computer vision, 2015, pp. 2533–41.
49. Mathews AP, Lexing X, Xuming H. SentiCap: generating image descriptions with sentiments. In: Thirtieth AAAI conference onartificial intelligence. 2016.
50. Melnyk I, Sercu T, Dognin PL, Ross J, Mroueh Y. Improved image captioning with adversarial semantic alignment. arXiv preprint. 2018; arXiv:1805.00063.
51. Memisevic R, Geoffrey H. Unsupervised learning of image transformations. In: 2007 IEEE Conference on Computer Vision and PatternRecognition, IEEE, 2007, pp. 1–8.
52. Mitchell M, et al. Midge: generating image descriptions from computer vision detections. In: Proceedings of the 13th conference of the European chapter of the association for computational linguistics. Association for computational linguistics, 2012.
53. Ordonez V, Girish K, Berg TL. Im2text: describing images using 1 million captioned photographs. In: Advances in neural-information processing systems, 2011, pp. 1143–51.
54. Palangi H, Smolensky P, He X, Deng L. Question-answering with grammatically-interpretable representations. 2017. arXiv :1705.08432
55. Park CC, Kim B, Kim G. Towards personalized image captioning via multimodal memory networks. IEEE Trans Pattern Anal Mach. 2018;41(4):999–12.
56. Pu Y, et al. Variational autoencoder for deep learning of images, labels and captions. Adv Neural Inf Process Syst. 2016.
57. Ren Z, Wang X, Zhang N, Lv X, Li LJ. Deep reinforcement learning-based image captioning with embedding reward. arXiv preprint. 2017; arXiv:1704.03899.
58. Rennie SJ, Marcheret E, Mroueh Y, Ross J, Goel V. Self-critical sequence training for image captioning. CVPR. 2017;1(2):3.
59. Sharma P, Ding N, Goodman S, Soricut R. Conceptual captions: a cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: Proceedings of the 56th annual meeting of the association for computational linguistics, 2018, vol. 1, pp. 2556–65.
60. Socher R, et al. Grounded compositional semantics for finding and describing images with sentences. Trans Assoc Comput Linguist. 2014;2:207–18.
61. Sur C. UCRLF: unified constrained reinforcement learning framework for phase-aware architectures for autonomous vehicle signaling and trajectory optimization. Evol Intell. 2019;12(4):689–12.
62. Sur C. Survey of deep learning and architectures for visual captioning-transitioning between media and natural languages. Multimed Tools Appl. 2019;78(22):32187–237.
63. Sur C. Representation for language understanding. Gainesville: University of Florida; 2018. pp. 1–90. https://drive.google.com/file/d/15Fhmt5aM_b0J5jtE9mdWInQPfDS3TqVw.
64. Sur C. SACT: Self-aware multi-space feature composition transformer for multinomial attention for video captioning. 2020; arXiv:2006.14262.
65. Sur C. ReLGAN: generalization of consistency for GAN with disjoint constraints and relative learning of generative processes for multiple transformation learning. 2020; arXiv:2006.07809.
66. Sur C. Self-segregating and coordinated-segregating transformer for focused deep multi-modular network for visual question answering. 202; arXiv:2006.14264.
67. Sur C. Gaussian smoothen semantic features (GSSF)--exploring the linguistic aspects of visual captioning in Indian languages (Bengali) using MSCOCO framework. 2020; arXiv:2002.06701

68. Sur C. MRRC: Multiple role representation crossover interpretation for image captioning with R-CNN feature distribution composition (FDC). 2020;arXiv:2002.06436.
69. Sur C. aiTPR: Attribute Interaction-Tensor Product Representation for Image Caption. 2020;arXiv:2001.09545.
70. Sur C. CRUR: Coupled-Recurrent Unit for Unification, Conceptualization and Context Capture for Language Representation--A Generalization of Bi Directional LSTM. 2019;arXiv :1911.10132 .
71. Sur C. Tpsgtr: Neural-symbolic tensor product scene-graph-triplet representation for image captioning. 2019;arXiv :1911.10115.
72. Sur C, Pei L, Yingjie Z, Dapeng W. Semantic tensor product for image captioning. In: 2019 5th international conference on big data computing and communications (BIGCOM), pp. 33–37. IEEE, 2019.
73. Sur C. Feature Fusion Effects of Tensor Product Representation on (De) Compositional Network for Caption Generation for Images. 2018;arXiv:1812.06624.
74. Sutskever I, James M, Hinton GE. Generating text with recurrent neural networks. In: Proceedings of the 28th international conference on machine learning (ICML-11), 2011.
75. Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Advances in neural information processing systems, 2014, pp. 3104–12.
76. Tavakoliy HR, Shetty R, Borji A, Laaksonen J. Paying attention to descriptions generated by image captioning models. In: Computer vision (ICCV), 2017 IEEE international conference, IEEE, 2017, pp. 2506–15.
77. Tran K, et al. Rich image captioning in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 49–56, 2016.
78. Vinyals O, Toshev A, Bengio S, Erhan D. Show and tell: lessons learned from the 2015 mscoco image captioning challenge. IEEE Trans Pattern Anal Mach Intell. 2017;39(4):652–63.
79. 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, pp. 3156–3164.
80. Wang C, Haojin Y, Christoph M. Image captioning with deep bidirectional LSTMs and multi-task learning. ACM Trans Multimed Comput Commun Appl. 2018;14(2s):40.
81. Wang Y, Lin Z, Shen X, Cohen S, Cottrell GW. Skeleton key: image captioning by skeleton-attribute decomposition. arXiv preprint 2017;arXiv:1704.06972.
82. Wu Q, Shen C, Wang P, Dick A, van den Hengel A. Image captioning and visual question answering based on attributes and external knowledge. IEEE Trans Pattern Anal Mach. 2017;40(6):1367–81.
83. Wu C, Wei Y, Chu X, Su F, Wang L. Modeling visual and word-conditional semantic attention for image captioning. Signal Process Image Commun. 2018;67:100–7.
84. Wu Q, et al. What value do explicit high level concepts have in vision to language problems?. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 203–12.
85. Wu J, Hu Z, Mooney RJ. Joint image captioning and question answering. arXiv preprint. 2018;arXiv:1805.08389.
86. Xu K, et al. Show, attend and tell: neural image caption generation with visual attention. In: International conference on machine learning, 2015, pp. 2048–57.
87. Yang Z, et al. Review networks for caption generation. In: Advances in neuralinformation processing systems, 2016, pp. 2361–69.
88. Yang Y, et al. Corpus-guided sentence generation of natural images. In: Proceedings of the conference on empirical methods in natural language processing. Association for computational linguistics, 2011.
89. Yang Z, Yuan Y, Wu Y, Salakhutdinov R, Cohen WW. Encode, review, and decode: reviewer module for caption generation. arXiv preprint. 2016;arXiv:1605.07912.
90. Yao T, Pan Y, Li Y, Mei T. Incorporating copying mechanism in image captioning for learning novel objects. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), IEEE, 2017, pp. 5263–71.
91. Yao T, Pan Y, Li Y, Qiu Z, Mei T. Boosting image captioning with attributes. In: IEEE international conference on computer vision, ICCV, 2017, pp. 22–29.
92. Ye S, Liu N, Han J. Attentive linear transformation for image captioning. IEEE Trans Image Process. 2018.
93. You Q, Jin H, Luo J. Image captioning at Will: a versatile scheme for effectively injecting sentiments into image descriptions. arXiv preprint. 2018;arXiv:1801.10121.
94. You Q, et al. Image captioning with semantic attention. In: Proceedings of the IEEE conferenceon computer vision and pattern recognition, 2016, pp. 4651–59.
95. Young P, et al. From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. Trans Assoc Comput Linguist. 2014;2:67–78.
96. Zhang M, Yang Y, Zhang H, Ji Y, Shen HT, Chua TS. More is better: precise and detailed image captioning using online positive recall and missing concepts mining. IEEE Trans Image Process. 2018;28(1):32–44.
97. Zhang L, Sung F, Liu F, Xiang T, Gong S, Yang Y, Hospedales TM. Actor-critic sequence training for image captioning. arXiv preprint. 2017; arXiv:1706.09601.
98. Zhao W, Wang B, Ye J, Yang M, Zhao Z, Luo R, Qiao Y. A multi-task learning approach for image captioning. In: IJCAI, 2018, pp. 1205–11.