Multimedia Tools and Applications (2019) 78:32187–32237
https://doi.org/10.1007/s11042-019-08021-1

Survey of deep learning and architectures for visual captioning—transitioning between media and natural languages

Chiranjib Sur[1]

Received: 14 August 2018 / Revised: 18 May 2019 / Accepted: 17 July 2019 /
Published online: 31 July 2019

Abstract
Deep Learning Architectures has been researched the most in this decade because of its capability to scale up and solve problems that couldn't be solved before. Mean while many NLP applications cropped up and there is a requirement to understand how the concepts gradually evolved till date after perceptron was introduced in 1959. This document will provide a detailed description of the computational neuroscience starting from artificial neural network and how researchers retrospected the drawbacks faced by the previous architectures and paved the way for modern deep learning. Modern deep learning is more than what it had been perceived decades ago and had been extended to architectures, with exceptional intelligence, scalability and precision, beyond imagination. This document will provide an overview of the continuation of work and will also specifically deal with applications of various domains related to natural language processing and visual and media contents.

Keywords Neural network · Deep learning · Natural language processing · Visual features · Representation learning · Sequential memory network

1 Introduction

Deep Learning Architectures had emerged as the algorithm of the century because of its capability to generalize, learn and its ability to scale up to the expectation of the applications. These architectures had recently broken several records in large scale learning and have been viewed to have much more capability than that it has unleashed. Expertise in this domain will make many things more smart, sensitive to variation, proactive, automated and easy. It can bring closer the human dream of intelligent systems all around and doing all the work for them. There are applications, where the perceptions of deep learning is far more accurate than human perception and capability. But what made deep learning work

✉ Chiranjib Sur
chiranjib@ufl.edu

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

is very simple, but different. Decades of research enticed people to see the world in a linear way because of their inability to visualize non-linearity and correlate them with the nature. In fact, all natural phenomenons are highly complex and can easily be approximated with non-linear relationships. Deep learning is further ahead involving both linearity and non-linearity. Mathematically, deep learning is a complex model of linear and non-linear functions $[f(\mathbf{x}) = \sigma_{non-linear}(g_{linear}(\mathbf{x}))]$ which fits as a non-linear hyper plane for any dimension of data. Large number of variables in a series of $\sigma_{non-linear}(.)$ and $g_{linear}(.)$ functions help in taking any shape. However, it starts as a very simple model and then takes the form of the data based on the different training parameters and other considerations like the linear and non-linear functions chosen and also on how the different updation of the parameters are allowed to happen for the model. However there are high chances that sometimes the model gets over-fitted, generating high training accuracy and thus spoiling the testing accuracy. The variation of testing error and training error revealed that both decreases simultaneously upto a certain extent and then the testing error starts increasing. This point is the over-fitting point, and training must stop to avoid inaccuracy. However, present day large scale applications and NLP applications deals more with representation generation and learning than defining separation planes and hence it is important to review deep learning from the prospect of representation learning and NLP applications. Back-propagation training is a complex trade off of bias and variance and regularization function for error must be chosen accordingly. High variance (l2 norm or least square errors) over-fit the data producing large expected generalization error while low variance (l1 norm or least absolute errors) makes the bias very large and the expected error becomes large again. Technically, when the variance is very large, the separation curve takes all the separation data points on its curve and is a perfect true separator, while if the variance is zero, then it produces a curve which may not include any data point and thus may not act a true separator. Thus a proper and stochastic trade off is inevitable. This phenomenon is highly stochastic as it is both data driven and also experiment driven and till date there is no rule on how the separator should look like and also what should be the ideal measurement for bias and variance.

1.1 Computational & Cognitive Neuroscience

Cognitive Neuroscience is the study of underlying biological process of the brain which contributes to its intelligence and capability to store and retrieve information. This core field is the building block of all kinds of biological mimicry that are happening in the Computational Neuroscience and is the link between Cognitive Neuroscience and Deep Learning. While Cognitive Neuroscience is understanding the brain to fight back brain related diseases and understand and monitor human psychology, Computational Neuroscience deals with the underlying phenomenon and working principles and properties of the structures, information transferring, retaining and efficient retrieval. Artificial neural network (along with modern deep learning) is such a primitive prototype of these concepts. With the aim of mimicry the principle of brain in the form of information processing, learning and retrieval, the computational counterpart also performs the same with the exception that its capability and generalization is limited in terms of input it can be fed and output it can generate and handle. Herculano-Houzel [48] produces the facts that human brains contains 100 billion neurons and 10 times more glial cells than any other animals and is highly dense. While there is an overdeveloped cerebral cortex, which occupies 80% brain mass, and plays the important role in consciousness, it also reveals superior cognitive abilities and why human brain is so complex and so efficient than any other animals. The networks, used in computation, are much smaller and primitive. It also lacks the complexity to generalize and also

to scale with respect to attributes, relationships and concepts. Another important concept in cognitive Neoroscience is the notion of spikes where information are captured in the neuron cells only when there are spikes in them, which are nothing but some kind of information processing in the form of chemical changes. In [59] different kinds of spiking and bursting phenomenons for neural networks are being described. Gradually researchers are describing spiking neural network where the information processing is accompanied by spikes (equivalent to electronic cliches) in time domain and retention of information is accompanied by it. These classes of computational neural networks have very high correlation with the biological neural cells, while artificial neural network is much of mathematically facilitated equivalent of the brain cells.

In recent years numerous architectural modifications and developments have been introduced to use this robust concept appropriately, differently and it has delivered more than up to the mark, unlike its predecessors. The main purpose of this document is to bridge the gap in understanding the different architectures of neural network and deep learning and also includes some of the key break throughs in applications like in computer vision, natural language processing, bioinformatics, time series, speech, recommendation system and drug discovery.

The rest of the document has skimmed through the details of artificial neural network description in Section 2, state of the art description of deep learning specific to training and understanding of the principles in Section 3, different network architectures with separate memory components and their utilization to applications in Section 4, image caption state-of-the-art review of the existing literature in Section 6, more diverse domain specific applications and comparison of architecture along with different word embedding techniques in Section 5 and conclusion in Section 7.

2 Artificial Neural Network

Artificial Neural Network is the most primitive version of layered architecture of neurons where each individual perceptron is denoted as $f(\mathbf{x}) = \sigma(\sum(\mathbf{x})) = \frac{1}{1+e^{-\mathbf{ax}}}$ where $\sum$ is the linear function and non-linear functions like sigmoid functions $\{\sigma(\mathbf{x}) = sigmoid(\mathbf{x})\}$ and at a certain later stage, tanh function $\{\sigma(\mathbf{x}) = tanh(\mathbf{x})\}$ were mostly used. As Artificial Neural Network has evolved with the concept of learning and testing, it was regarded as one of the core bio-inspired computational framework derived from the structure of neural cells and inspired by the information flow in cognitive neuroscience. In this section we will mostly deal with the different classes of Artificial Neural Network and the way they were trained.

2.1 First learning networks of 1960s

The advent of deep learning dates back to the early 1950 when people were interested in understanding on how the brain works and what are the cause and effect relationships of different parts of the brain. The search is yet not complete as the problems are far too complex to be solved in one generation and the brain have highly complex structure and functioning. Research in neuroscience started primarily with the monitoring of neurones through receptors on cats as they were made to view certain changes in images where there were certain geometrical structures with different orientations. The experiments were prolong and the details of the phenomenon are described in [55]. However the conclusion was very interesting. The experiment claimed that the brain cell, they were monitoring, showed considerable excitation only when the image shown to them had a certain change

in orientation. In all other cases, there were no excitation. This phenomenon is quite similar to the activation functions in deep neural network where the nonzero gradient being transferred to the weights only when there is something to be learnt. In other cases, it remained stagnant. This is regarded as the most primitive research in vision and neuroscience, the two inevitable combo in modern computer science and artificial intelligence. Another contemporary achievement was made in the form of Perceptron by Frank Rosenblatt in the period 1957–1962. Perceptrons are the building blocks of the modern artificial neural network models which came much later and was superseded by Deep Learning in 2006.

2.2 Feed-Forward Neural Network

Feed-Forward Neural Network [4] marked the first layered neural network with hidden layer(s). Single layer Perceptron and Multi layer Perceptron were introduced where mostly sigmoid function was used and the weights of the layers were updated using $w_k(t+1) = w_k(t) + y_k(t) * \delta(t)$ where $y_k(t) \in \{0, 1\}$ dependent on the threshold θ and $\delta(t) \in \{-1, 1\}$. The whole network undergoes through a forward transfer of training instances and then accordingly the weights of the hidden layers are updated. This is yet another primitive way of back propagation based update of the weights where only incorrect prediction for a certain class introduced changes. The change values are quite discrete and is however a very special case of the Back Propagation strategy with no proper error functions and no differentiation involved to propagate the change. Later the Feed-Forward Neural Network transformed to Back-Propagation Neural Network with the generalization of the delta rule to chain rule based gradient descent based weight update technique for each layer.

2.3 Back-Propagation Neural Network

Back-Propagation Neural Network [46] works on the basic principle of Feed-Forward Neural Network, but for the first time it stressed on importance of the weight updation rules through $w_k(t+1) = w_k(t) + \eta * \frac{\partial E}{\partial w_k}$, exploiting the differentiable error function E or the loss function gradient and thus can propagate back a very small amount of differences for learning and can even scale up with the learning rate η. Gradually a number of adaptive learning rates were introduced like momentum based learning rate and were more intelligent and convergent for the situations. Also the differentiable Back-Propagation function based update for the intermediate weights of the Neural Network encouraged multiple hidden layers and with the increasing computation power, it was feasible to train up the network with a descent large dataset. But gradually with the intricacies of dataset feature space and limited spike capability of activation function, Back-Propagation Neural Network started facing problems like vanishing gradient problem and this gave rise to scalability issues for the network. Deep Learning overcame this problem and with tremendous computation power in modern days GPUs, it is possible to scale up to any level based on the requirement. So overall the most important breakthrough Back-Propagation Neural Network brought in was that the activation function for the artificial neurons should be differentiable.

2.4 Radial Basis Neural Network

Radial Basis Neural Network [11] was another modified Neural Network where radial basis functions were used as highly nonlinear transform function. Radial basis functions are known to have strict interpolation property in multidimensional space. This means that radial basis functions have better adaptibility to align itself with a highly non-linear set of

points and this property later was exploited as kernelization which can provide much better fit for non-linear data. Mathematically, radial basis functions for weights $\mathbf{x}$ of each neuron can be defined as $f(\mathbf{x}) = b_0 + \sum_{i \in S} b_i \rho(||\mathbf{x} - \mathbf{c}_i||)$ where $\rho(||\mathbf{x} - \mathbf{c}_i||)$ can be any radial basis functions like Gaussian radial basis function $\rho(||\mathbf{x} - \mathbf{c}_i||) = exp(-\beta||\mathbf{x} - \mathbf{c}_i||^2)$ or thin-plate-spline function $||\mathbf{x} - \mathbf{c}_i||^2 \log(||\mathbf{x} - \mathbf{c}_i||)$ or multiquadric function $(||\mathbf{x} - \mathbf{c}_i|| + \beta^2)^{1/2}$ or inverse multiquadric function $(||\mathbf{x} - \mathbf{c}_i|| + \beta^2)^{-1/2}$. The choice of activation function will not affect the solution as the Orthogonal Least Squares algorithm was used for learning for the different weights $\mathbf{x}$. It is claimed that Singular Value Decomposition Learning failed to provide effective learning for the weights. The orthogonal least squares method is used as a forward regression procedure to select a set of suitable centers or regressors from a large training data points.

2.5 Hopfield Network

Hopfield Network [53] is the primitive form of Recurrent Neural Network and a special case of Cohen-Grossberg Neural Network. It possesses a number of feedback weights to the components of the same layer and thus training can help in learning certain sequences and hence can also be regarded as one type of memory network. It was described by Little in 1974 and then reworked by John Hopfield in 1982. In Hopfield Network, there are lot of interconnections among the neurons and consists of a single hidden layer. The most important feature of the Hopfield Network is that the weights of the hidden layer can take only two values like [0, 1] or [−1, 1]. This restricts the model of the neural network fixed to certain state spaces which can also act as content based addressable memory based network and thus the mapped new feature space is also restricted. The binary value of the weights actually guarantee convergence for the network. The updation rule in Hopfield Network occurs for threshold of θ, where the weight is updated with either of the new state. In [53], Hopfield Network has been used for optimization of dynamical systems.

2.6 Self-Organizing Maps

Self-Organizing Maps [70] can be regarded as one kind of single-layered artificial neural network and the primitive of multiple-layered Auto-Encoder or Encoder-Decoder Network and performs the functionality of unsupervised learning. It actually transform higher dimensional data to very low dimensional space, very similar to non-linear transformation of multidimensional scaling. The main difference of Self-Organizing Maps from traditional artificial neural network is that it used competitive learning instead of differentiable error gradient back propagation. Competitive learning is one kind of Hebbian learning, where neural network is initiated with some randomly distributed weights and a strength is associated with each neuron. Thus when the data is applied, they are operated differently and segregate to different clusters. Hebbian learning describes some rules on how the weights must be manipulated to adapt the neural network with the data. For any two neurons, whether they activate simultaneously or not determines whether the weight connecting them must be increased or decreased. In Self-Organizing Maps, each time a sample is taken and its distance is calculated from each of the neuron and the best matching unit is considered as Best Matching Unit (BMU). Then the weights are updated using the following mathematical equations $\triangle w_{ji} = \eta(t) T_{j,I(x)}(t)(x_i - w_{ji})$ where $\eta(t)$ is the learning rate, $I(x)$ is the Best Matching Unit and $T_{j,I(x)}(t)$ is the Gaussian neighborhood.

3 Deep Learning architectures

Deep Learning is the next generation of the Artificial Neural Network which failed to provide the necessary scalability. Scalability in Artificial Neural Network can be achieved through increasing the number of neurons of the layers which can create redundancy for the number of neurons in large numbers compared to the dimension of the features. Scalability in Artificial Neural Network can also revived if the number of hidden layers is increased. In that case there occurs the gradient vanishing problem if activation functions like sigmoid, tanh, etc are used or the gradient explosion problem if non-linearity is invoked through ReLU, etc functions. In both the cases, the learning will be affected and creates a hyperplane which hardly separates the classes. Another problem persisted in the Artificial Neural Network was that uniform and symmetric connection based network for all the neurons hardly created effective variability in feature selection and later was solved by [126]. Also selection and combination of all features characteristics hardly give rise to new combined feature instances. Instead if a subset is considered, then it may happen that better combined features characteristics are reproduced and this was the main concept behind convolutional neural network [86], where a window based sectional selection is considered to generate new combined feature characteristics.

3.1 Dynamic Neural Network

In Dynamic Neural Network [120], the structure is marked by the incorporation of dynamic elements with continuous feedback which brings the notion of linearity and timeliness into the system. Normally feed-forward neural networks have static modeling and fail to involve causality for sequence of events. In Dynamic Neural Network, a first or second order dynamic neuron is defined using differentiation as a non-linear function for gain and time components and is feedback into the network as a replacement for gradient based update. The model is designed using transformation of differential equation based causality for time and other parameters of the dynamic systems for incorporating non-linearity for the neural network. The main application of Dynamic Neural Network is to design control systems which can predict the future behavior. Such control systems are like process modeling or nonlinear model based controller which captures a rich range of nonlinear feature dynamics.

3.2 Learning Vector Quantization (LVQ) Network

In Self Organizing Map, all feature space is being transformed to another feature space or some discrete output space which represents the unsupervised classes of the data. Now this discrete space can be large and is dependent on the transformed feature. Vector Quantization limits the input features space to certain discrete levels and thus also limits the output feature space. Learning Vector Quantization (LVQ) Network provides an algorithm to find a good approximation of that input space to a certain discrete levels. This quantized input space is then used to train a neural network and thus helps in better classification. The analogy of quantization comes from k-means where all the data points of a certain cluster can be replaced by its cluster head. This kind of approach is highly useful in applications like data compression and reduction of feature dimension. In Learning Vector Quantization (LVQ) Network [69], the neural network have the feature vector $\mathbf{x}$ and the weights $\mathbf{w}$. Now if the output of the neural networks $\rho(\mathbf{xw})$ works perfectly aligned with the labels $\mathbf{y}$ of the data, then the neural network is perfectly trained, if not then change $\mathbf{w}$ with $\triangle w_{BMU} = \beta(y_i - \rho(\mathbf{x}_i\mathbf{w}))$ if y_i and $\rho(\mathbf{x}_i\mathbf{w})$ are in same class (this will move the data close to the region

of matching) and using $\triangle w_{BMU} = -\beta(y_i - \rho(\mathbf{x}_i\mathbf{w}))$ if y_i and $\rho(\mathbf{x}_i\mathbf{w})$ are in different class (this will move the data far from the region of matching).

3.3 Auto-Encoders

Auto-Encoders [2] are multi-layered artificial neural network models which are used to learn the representation of the data or rather transform the underlying encoding of the data to enriched form and very useful for unsupervised learning. It can be used for various functionalities like non-linear dimensional reduction, determination of the generative models of the data, non-linear feature separation, feature selection, etc. Fundamentally, Auto-Encoders aimed at learning discriminative representation for different contents, Variational Auto-Encoder aimed at learning the distribution and providing more generality and bounded by mathematical models of Bayesian statistics.

3.3.1 Traditional Auto-Encoders

Basically, it is a parametric neural network, where the output layer possesses same set of vectors as the input layer and some intermediate hidden layer possess the transformed or extracted feature-forms and can be utilized as different representation. The whole system is build on the condition that there is least distortion of the data at both the input and the output ends. Mathematically the objective function is represented as $\arg\min_{\phi,\varphi} ||\mathbf{X} - \phi(\varphi(\mathbf{X}))||^2$ where $\{\varphi = X' : X \rightarrow \varphi(X)\}$ and $\{\phi = X'' : \varphi(X) \rightarrow \phi(X')\}$. Auto-Encoders was introduced much earlier in the 1980s and had been trained up using Hebbian learning, until recently with the advent of deep learning architectures, people used better learning techniques to propose better Auto-Encoders models for the data. Restricted Boltzmann Machines is a stacked version of the Auto-Encoders model and are being trained bottom up in unsupervised way and then fine tuned with phases of supervised training sessions to train up the upper layers. This has gradually provided much better trained up representation for the architectures.

3.3.2 Variational Auto-Encoder

Variational Autoencoders [28] helps in generation of latent representation vectors that are devoid of variations through merging large part of the data with similar objective into similar distribution or summation of distributions as a functional approximation. This functional approximation is generated through convolution of features and series of linear and non-linear transformation. Constraints are imposed on encoding and decoding network to enforce latent representation follow a unit Gaussian distribution and scale down effects of different features. Other constraints include regularization penalty or KL-divergence cost penalty to enforce sparsity in the representation and fit in lots of useful information. This kind of penalty constrained sparsity helps in unsupervised features extraction or representation generation that is naturally favorable to clustering or unsupervised learning. This is one kind of structuring the data, when the original space is not favorable for some purposes, through inculcation of supervised labeled information through gradient based structuring of the learned representations. This kind of structuring helps in generation of very distinct, low variational and machine interpret-able representation that are bounded by distribution in some feature space.

3.4 Restricted Boltzmann Machine

Restricted Boltzmann Machine is a modifier and enhanced version of Auto-Encoders which has several applications starting from topic modeling, collaborative filtering, feature engineering, reduction in dimension to even regression or classification. These consists of Boltzmann Machines which are nothing but a particular form of log-linear Markov Random Field (MRF). The log-linear part provides the non-linearity, which being trained, can represent complicated distributions of the data. In Restricted Boltzmann Machine, there is restriction in connection between two neuron of the same layer and hence the name. Temporal Restricted Boltzmann Machine [131] is a probabilistic model based sequences of Restricted Boltzmann Machines which have been efficient in handling high dimensional sequences like events or motion capture in a video. Normal inference is exponentially expensive with the dimension of the data and that is the reason why heuristic inference procedures are used to get some near approximate learning for the neural network.

3.5 Recurrent Neural Network

In Recurrent Neural Network [36], the connection not only exists between layers but also between the individual neurons in the same level. So the output data not only dependent on present output but also correlated with topologically connected previous inputs and thus forming one kind of causal relation between individuals. It is very efficient in prediction output in case of series data where the output not only depends on the present features/element but on all or majority subset of the previous elements of the series. For example, a multi-dimensional dynamical system can be represented approximately by the internal state of a continuous time recurrent neural network with some trained hidden units, and an appropriate initial condition.

Mikolov et al. [97] described a language model based on simple recurrent neural network or Elman network for speech recognition. Feed-forward network had limited and fixed capacity of accepting reference into its structural model and it limits the capability of getting trained and thus can easily break with exceptions and extensions. But Recurrent Neural Network possesses the notion of temporal encoding of contextual sequence of information of arbitrary length though the selection of the size of hidden (context) layers.

In [7], recurrent network has been used to model system representations for learning long-term dependencies in sequences. It has been shown that gradient clipping above a given threshold can prevent explosion of gradient effects and spanning longer time ranges with leaky integration can reduce the effect of vanishing gradient problem. When the output vector for the prediction are multivariate, to model the high-order dependencies between variables, powerful output probability models like Restricted Boltzmann Machine should be used. Sparser gradient learning for deep architectures via Sparse Output Regularization and Rectified Outputs using an L1 penalty on outputs of hidden units is used to promote sparsity of activation. The underlying reason behind this is that if the gradient propagation is concentrated in a few paths of the network like in the unfolded computation graph of the structure, it can reduce the vanishing gradients effect. Also here Nesterov accelerated gradient is used as a first-order optimization method to improve stability and convergence of regular gradient descent. Lastly it also shows the difference in learning using vanilla SGD versus SGD plus some of the enhancements like that of gradient clipping, leaky-integration units, use of rectifier units with L1 penalty, Nesterov momentum and also just Hessian-Free optimization.

3.6 Neural History Compressor

Normal sequence based models for data performs very poorly and are computationally expensive when the time lags are exorbitantly large. Neural History Compressor [118] has proposed a principle where the description of a sequence of events can be reduced without loss of generality and information. The principle maps the sequences segments by recursively decomposing them to patterns where the unexpected are the most relevant. This kind of 'divide and conquer' can help in better representation of the sequence and will provide better insight for construction of neural architectures and training them. Out of the two architectures, the first functions acts as a self-organizing multi-level hierarchy of recurrent networks, while the second involves two recurrent networks, one trying to collapse a multi-level predictor hierarchy into a single recurrent net.

3.7 Recursive neural networks

Recursive neural networks a generalization of the recurrent neural network with specific type of skewed tree structure and operates on structured inputs like on directed acyclic graphs. Recursive Neural Networks [123] can merge image segments or natural language words based on deep learned semantic transformations of their original features. Briefly it "is a max-margin structure prediction architecture based on recursive neural networks that can successfully recover such structure both in complex scene images as well as sentences". This recursive structure helped to identify the components of image or sentence and the way they interact to form a lower level whole. In [57], a fine-grained sentiment classification methods have been deployed with recursive neural networks. Recursive neural networks has been known to operate on structured inputs and have applied previously "to model compositionality in natural language using parse-tree-based structural representations". Conventional deep feed-forward networks and deep recurrent neural networks are known for hierarchical representations. Deep recursive neural network (deep RNN) is another architecture constructed by stacking multiple recursive layers and has the same capability.

3.8 Long Short Term Memory

The Long Short-Term Memory or LSTM network [51] is basically a modified version of recurrent neural network and is being trained up using back-propagation algorithm through time as levels and also has the capability to overcome the vanishing gradient problem. In recurrent back-propagation, insufficient, decaying error back-flow makes it longer time to store information for over extended time intervals. Recurrent networks store representations in activation forms (like short-term memory) slowly varying the weights through their feedback connections. This is as opposed to long-term memory where the stored representation is held for a longer span of time through other controlling features. Thus it cannot be used to create larger recurrent networks which are significantly suitable to address difficult sequence problems and applications in machine learning like speech processing, non-Markovian control, music composition etc. LSTM networks have memory blocks that are connected into layers but these memory blocks are different from the classical neuron structures with more smartness in form of control. There are gates that controls the state and output of the block. There are Forget Gate (decides what information to throw away from the block), Input Gate (decides the inputs to update the memory state and is introduced to protect the memory contents stored through removal of irrelevant inputs) and Output Gate

(decides what to output based on input and the memory of the block). The gates of the units have weights that are trained during the training phases. During training, the gates are being operated by the activated sigmoid functions and there is change of state and also information flow though the block. So the memory cell is a connected graph structure of several differentiable computational units like sigmoid function, tanh function, and the gating function and based on the structure of the data they either learns to get activated based on the context of the data or remain deactivated. The concept of the internal state of LSTM gradually evolved to the present stage with guarded input and output gates with a time delay self loop with unit weight to ensure retention of the value, unless the input gate open. Forget gate modulates the state's self-connection and helps in precise timing abilities. Peepholes were devised for direct connections from the state to all gates. LSTM has the capability to learn the sensitive language of the underlying contextual structure for thousands of time stamps in very small amount of time.

3.9 Deep Belief Networks

Deep Belief Network [6, 50] is a stacked Restricted Boltzmann Machine and thus can be trained layer by layer using unsupervised learning techniques. Unsupervised learning will help in constructing a generative model out of the neural network and can probabilistically learn to reconstruct the data. Each hidden layer of the Deep Belief Networks, being a visible layer of the next, can help in better learning the data through unsupervised techniques. Deep belief network can be seen as a multilayer generative model where each layer encodes statistical dependencies among the components of the hidden layer below it. The model is trained to (approximately) maximize the likelihood of its training data. For the first time [50] introduced the concept of training the network using greedy technique. The learning helps in generating a better generating model (low-level features) and performs much better than the discriminative model (high-level features). Thus the whole system acts as an associative memory for the unsupervised training data. Initially the whole network is trained using a fast greedy algorithm and then fine tuned using a slower wake-sleep algorithm.

3.10 Convolutional Deep Belief Networks

Deep Belief Networks failed to scale to full-sized, high-dimensional images. Convolutional Deep Belief Networks [77] is also an unsupervisely learnt hierarchical generative model like deep belief networks, which scales to realistic image. It is translation-invariant and supports efficient bottom-up and top-down probabilistic inference of the data. The most important component is probabilistic max-pooling which shrinks the representations of higher layers of the data in a probabilistically way. Another difference between Deep Belief Networks and Convolutional Deep Belief Networks is that in the latter, the weights between the hidden and visible layers are shared among all locations in an image.

3.11 Large Memory Storage and Retrieval Neural Networks

LAMSTAR or Large Memory Storage and Retrieval Neural Networks is characterized by three main characteristics: capable of forgetting the data it is fed or trained with, interpolation/extrapolation and capable of changing its working resolution based on the data. LAMSTAR [41] is a computationally efficient way for storing and retrieving patterns in neural networks like Self Organizing Map. This is achieved with the help of statistical decision tools. In LAMSTAR network, expert information is continuously being processed

to rank the representations for each case of the intelligent expert system through learning and correlation. LAMSTAR network has the capability to forget, interpolate and extrapolate features which can help in handling non-analytics (both exact and fuzzy) even when a large part of the attributes of the data are missing. Through extrapolation or interpolation, it is possible for the LAMSTAR network to zoom out of stored information via forgetting and even after that it can approximate forgotten information. LAMSTAR is mostly developed for applications which involve data categories which can be both exact and fuzzy and requires huge amount of memory for information storage and efficient retrieval. However the most spectracular feature of LAMSTAR network is that it can expand and shrink based on the feature dimension of the data without the requirement of re-engineering the network and the policies to train it.

3.12 Deep Boltzmann Machines

Deep Boltzmann Machines [116] are stacks of Boltzmann Machines where each of the Boltzmann Machines have been trained with the principle of the unsupervised learning and then supervised learning is used to fine tune the generative models network. It is trained based on data-dependent expectations and the performance is estimated with variational approximation. There are also data independent expectations and are used for approximation using persistent Markov chains. This kind of approximations helps each layer build a very complicated and useful notion of the data in the form of higher-order correlations between the data feed into the hidden features and the layer below. Two different techniques are used for estimation of expectation and is used in the gradient of the log-likelihood that helps to train up the millions of variable parameters in the multiple hidden layers of Deep Boltzmann Machines. The learning has been made more accurate and efficient by using the layer by layer training and thus prevention of dying gradient is possible. Also this kind of layer by layer learning is the pre-train phase while the final refinement occurs when the model is trained up using supervised techniques. However the pre-train phase training allows more variational inference that can be possble for the initialization of the parameters with a single bottomup pass.

In [125], Deep Boltzmann Machine is used as a generative model to learn and extract the unified representation of the data consisting of multiple and diverse input modalities. Such unified representation is derived by the model through the fusion of the modalities together in different forms from the data. These unified representations are very efficient and rich set of feature vectors for classification and information retrieval. The working principle of the model is based on learning the probability density over all the feature space of multimodal inputs of the data. In this paper, they have mainly concentrated on the application of Multimodal Deep Boltzmann Machine as a generative model for the joint represesntation space of data (like for images and text) for both unimodal and multimodal queries based information retrieval.

3.13 Stacked Denoising Auto-Encoders

Auto-Encoders are trained with unsupervised training samples through parameteric error minimization between the input and the output layers. To encourage better and error-invariant parametric learning for the weight variables and explore robust feature variables, it is necessary to make it learn in presence of noise. This kind of noise is inevitable for applications in computer vision and natural language processing. This also prevent the model from getting prone to error while operating and will also prevent it from only learning the identity

of the data. Learning only the data will prevent the model from generalization and will not be able to learn the distributions. Learning distributions will even help the autoencoder to reconstruct the input data from a corrupted version of it. Hence in denoising auto-encoder, the noisy version of the data is used to construct the data and can be regarded as stochastic version of the auto-encoder. Denoising auto-encoder performs two operations: First it tries to encode the fed noisy data and preserve the information of the original data. Secondly it tries to nullify the effects of a corruption process that has been added to the data stochastically. Mathematically the objective function is represented as $\arg\min_{\phi,\varphi} ||\mathbf{X} - \phi(\varphi(\mathbf{X} + \mathbf{N}))||^2$ where $\{\varphi = X' : X \rightarrow \varphi(X)\}$ and $\{\phi = X'' : \varphi(X) \rightarrow \phi(X')\}$, where $\mathbf{N}$ is the stochastic noise component. Stacked Denoising Auto-Encoder is the multi-layered version of the Denoising Auto-Encoder [142]. It is one kind of deep networks based on simple stacking of the components and the number of hidden layers and the number of components in each is subjected to the dimension of the data and the intensity of the problem. Denoising Auto-Encoder helps in training and also denoising the local corrupted versions of their input data. In [142], denoising autoencoder has been able to learn Gabor-like edge detectors from natural image patches and larger stroke detectors from digit images which was not possible for traditional edge detection algorithms.

Stacked Convolutional Denoising Auto-Encoders [30] was introduced for high dimensional feature mapping from the data to the lower level usable features. Here the features are generated through convolving features of the lower hidden layers with the trained kernels. These trained kernels are nothing but denoising auto-encoder stacks and trained in the same way as described before. This also prevented the model to be dependent on large volume of labeled data which can be inaccurate and also very expensive to obtain. Hence stacked convolutional denoising auto-encoder was used and the whole network was trained through unsupervised means and mapped images to lower hierarchical representations without any label information.

3.14 Tensor Deep Stacking Networks

Tensor Deep Stacking Network (T-DSN) [164] is a deep network architecture consisting of multiple stacked components and each layer contains a bi-linear mapping from two hidden layers to the output layer. This mapping consists higher order feature vectors via two parallel hidden layers and a hidden-to-output weight tensor. It is found that the model performance of Tensor Deep Stacking Network steadily gets better when it is trained with larger batch sizes for each epoch. However full batch based training is feasible for large datasets but using a parallel training procedure. Deep Stacking Network is based on the philosophy of "stacked generalization" which means that the lower layers are associated with learning with better effectiveness for feature extraction and classification. This also means that weight learning problem for the deep architectures is not convex anymore, but needs classical heuristics and other domain specific initialization information. Tensor Deep Stacking Network is linked with its predecessor in many ways. In Tensor Deep Stacking Network, there is information about higher-order and co-variance statistics of the data. This is done using a "bilinear mapping from two hidden representations to predictions using a third-order tensor". Another link is that Tensor Deep Stacking Network is based on the linear-nonlinear interleaving layering structure, but "it shifts the major learning problem from the lower-layer, non-convex optimization component to the upper-layer, convex sub-problem with a

closed-form solution". Tensor Deep Stacking Networks [56] has been used as Deep Tensor Neural Network for large vocabulary speech recognition application. Tensor Deep Stacking Network has also been used extensively for other speech and vision applications.

3.15 Spike-and-Slab Restricted Boltzmann Machines

Spike-and-Slab Restricted Boltzmann Machines (ssRBMs) [25] is a unique variant of the Restricted Boltzmann Machine series where there are real-valued vector called the slab and a binary variable called the spike associated with each unit component of the hidden layer. The slab variables allow the model to capture covariance information of the data and at the same time maintain very simple and efficient inference via a Gibbs sampling scheme. There are many fundamental differences between Spike-and-Slab Restricted Boltzmann Machines and normal (covariance) Restricted Boltzmann Machines (cRBMs). ssRBM is associated with Gibbs sampling phenomenon while cRBM takes the parameters from hybrid Monte Carlo (HMC) methods. This makes ssRBM much simplier, easy to deal with and implement because of the simplicity of Gibbs sampling method and more suitable mainly for deep learning architectures like Deep Boltzmann Machines and applications like predicting time series. Another important difference between two is that the ssRBM induces sparse real-valued representations of the data which is a fundamental situation for a lots of real valued data representations.

3.16 Compound Hierarchical-Deep Models

Compound Hierarchical-Deep Models or Hierarchical-Deep [117, 137] is a compositional learning architecture of deep learning models and structured hierarchical Bayesian (HB) models like hierarchical Dirichlet process. In [117, 137], the authors have shown how to learn a hierarchical Dirichlet process (HDP) prior though the activities of the top-level features of a Deep Boltzmann Machine and also found that the duo can actually learn notable features and concepts which were not possible independently and also from a few training samples. Training from few data is possible as low level features is more generic and the high-level features captured correlations among the low-level features. Thus a category of hierarchical sharing priors is generated and is far better than the high-level different kinds of conceptual features.

3.17 Deep Coding Networks

In Deep Coding Networks [81], high-dimensional sparse coding followed by linear classifier can help in image classification. Sparse coding can act as one kind of feature selection technique and has been found very useful for image compression, though it has limitations in sparse sensing. There are theoretical justifications for the working principle of sparse coding with local coordinate coding (LCC) for effective high dimensional non-linear function approximation methods. "LCC learns a nonlinear function in high dimension by forming an adaptive set of basis functions on the data manifold, and it has nonlinear approximation power". Deep Coding Networks implement that concept through the inclusion of sparse coding and has been proven to have better performance than single-layer approach and can be used for multi-layer hierarchical systems as well, pretty much like what can be seen in deep belief networks.

3.18 Deep Predictive Coding Networks

Deep Predictive Coding Networks [87] is a predictive neural network ('PredNet') architecture based on the concept of predictive coding, like learning to predict future frames in a video sequence. Here the network makes predictions based on local informatics and then the deviation is being forwarded to subsequent network layers for prediction of the next sequences. These have very robust learning ability where they learn the movement of the synthetic (rendered) objects. The movement of the object is being extracted as the internal representation and is used for decoding the latent parameters that helps in object recognition based on training views. The network consists of several stacked layers to make local predictions of the input to the module. There are in general four basic layers of the Deep Predictive Coding Networks. They are a representation layer, a recurrent convolutional network, a prediction layer and an error representation layer, which gets split into separate rectified positive and negative error populations. The error is then transmitted to the next convolutional layer as input for the next prediction. The recurrent prediction layer also receives a copy of the error.

3.19 Deep Q-networks

To get a better trained and generalized deep neural network, a new algorithm is described to learn the policies and to control the behavior of the network during training phases. These policies are associated with rewards that accelerate the training by enhancing the training factors like the learning rate, batch selection etc. Other methods, which have provisions for tuning the parameters, have features hand-constructed of specific tasks and also failed to control stochastic rates. Deep Q-networks [33] provided specific algorithms to handle these kind of training sensing methods for the deep neural network and have shown to enhance the performance.

4 Networks Architectures with Separate Memory Structures

Deep Learning Architectures have lot of limitations. First of all there are limitations in the representation of data and in many cases the topological information remains hidden or not learned. This makes it difficult to make the deep learning structures learn complex stuffs and thus there is a great barrier in generalization. The mathematical model of the neural network hinders such provision. Another possible drawback of deep learning is scalability not only with respect to the number of classes, but also with respect to the variety of classes and this is the reason why many researchers and organization have emerged out of the traditional layered architectures and have introduced hierarchical based deep learning, parallel deep learning and other cooperative techniques where there are other numerical, computational and transformation structures which back or hide up many other dereliction of the deep learning networks and provide the prediction and intelligence from back-end. A number of such Networks Architectures with Separate Memory Structures are being discussed in this section.

4.1 LSTM-related Differentiable Memory Structures

LSTM-related Differentiable Memory Structures [3] is the way of analysis of the topological properties of the memory cell that can optimize or maximize the learning process

through the use of multi-objective optimization algorithms like NSGA-II etc. based on error and learning capacity, trading off performance and generalization with each individual component. The optimized structure of the memory network will help in understanding the usefulness of the structural aspects which facilitate the learning process and will also help in understanding the underlying dynamics of learning. This kind of dynamics of the memory network is very important in understanding the inter-dependency and topological importance of the elements and the learning process for sequence prediction and sequence classification problem. This kind of optimization will help in tackling problem related to sliding window based solution which is limited in space and also has vanishing gradient problem for the network.

4.2 Semantic Hashing

Semantic Hashing [115] is a parameteric deep graphical model (with Poisson Restricted Boltzmann Machines) based on transformation of vectors of words count from a document to a space that can be used for a better representation of each document and similar documents gets mapped to very close regions. Assuming counts of different words feature evidence of similarity, it will be better than Latent Semantic Analysis where two words having close meaning or concept is mapped together. "When the deepest layer is forced to use a small number of binary variables (e.g. 32), the graphical model performs semantic hashing". The documents are being mapped to address space and the semantically similar documents remain close to each other and thus prediction of other documents becomes easier. This is an efficient way of hash-coding based approximate matching based prediction for the documents. Even TF-IDF can be applied and has better accuracy of prediction through hash-coding based match.

4.3 Neural Turing Machines

Neural Turing Machines [42] extended the capacity of neural network to external structures and memory resources which can replace the learning experience and the variable vector structure or rather distribute the learning to external specific memory locations with addressing and can be interacted by attention processes. Thus provide better prediction, disjoint information processing and very less chance of getting drown in information. Neural Turing Machines have successfully being used for many fundamental problems like copying, sorting, and associative recall. It is analogous to Turing Machine as recurrent neural networks was proven to be Turing-Complete which means it can be used to simulate any single-taped Turing machine and also it can operate on a very large feature space and any combinations parallel which normal systems fail to scale up in polynomial time. Neural Turing Machine works on the principle of real brain with separate regions for different organs (memory bank) and a "central executive focuses attention and performs operations on data" (neural network controller). Here 'blurry' read and write operations makes sure that all the memory elements are interacted with. However the interaction with the different locations of the memory is highly sparse phenomenon and the specialized focus is concentrated on specific memory locations.

4.4 Memory Networks

Memory Networks [149] are learning systems with enormous memory component that can read and written efficiently with focus on cases base relationship of input with output.

These learning models also have inference components and a relatively long-term memory component which can be easy to read and write with respect to question answering application context. Here the memory network can act as a dynamic knowledge base with textual response as output for the network. Memory Networks broadly consists of four components namely input feature map (converts the incoming input to the internal feature representation), generalization (updates old memories given the new input and is called generalization), output feature map (produces a output from the feature representation space based on input and memory state) and response (converts the output into interpretable response).

End-To-End Memory Networks [128] is an extension of Memory Network with recurrent neural network architecture capable of intercepting long term dependencies in sequential data and the recurrence reads a large external memory through multiple locations through multiple computational steps before responding. It is trained end-to-end with much less supervision during training, suitable for general and realistic scenario. It can easily trained up with back-propagation and don't require layer wise supervision. It can be trained end-to-end from input-output pairs with no external intervene or feature transformations and is applicable to realistic tasks like language modeling or question answering systems.

A large-scale simple question answering system has been designed with Memory Network [9], where the emphasis is on multitask and transfer learning for simple question answering system. It is an effort to integrate the existing systems and all the other possibilities to train a system jointly for different data sources. It also integrates the reasoning to answer a question through retrieval of evidence and not on prediction mainly for large scale data processing.

Improved version of Memory Networks model to reasoning and natural language for building intelligent dialogue system has been described in [150]. It is a framework and a set of synthetic tasks for the goal of developing algorithms that can learn from text through understanding and reasoning. It is relatively difficult to evaluate the performance of a general dialogue agent and that is the reason of the term-goal which is relatively easy to evaluate responses to input questions. A wide ranging set of different domain specific tasks has been established to test the capabilities of learning algorithms differently for a common framework.

Dynamic Memory Networks for Natural Language Processing [74] has been introduced where dynamic memory network consists of a neural network architecture which processes input sequences and questions and generates episodic memories and relevant answers. The questions trigger an iterative attention process to condition the attention parameters with the inputs and delivers the result of the iteration process. The main components of the Dynamic Memory Networks consists of the following: Input Module (encodes raw text from the tasks into distributed vector representations), Question Module (encodes the question of the task into a distributed vector representation), Episodic Memory Module (given a collection of input representations, the episodic memory module chooses which parts of the inputs to focus on through the attention mechanism) and Answer Module (generates an answer from the final memory vector of the memory module).

Goal-Oriented dialog [8] uses End-to-end Memory Networks for the dialog generation from the memory network and is based on the learnt knowledge of a neural network. The capabilities of the systems is based on goal-oriented dialog. The primary goal being completion and ensurance of well-defined series of texts based on defined measure of performance. The memory components are directly trained on past dialog which are generalized that is no assumption on the domain or on the structure of the dialog state had ever being made.

4.5 Pointer Networks

Pointer Networks [144] is such a neural network architecture which can learn conditional probability of an output sequence with certain topological importance. Existing architectures can only predict based on fixed input length which will not help problems like sorting variable sized sequences and various combinatorial optimization problems. To overcome this problem, a variable size output dictionary for the neural network is used to solve approximately challenging geometric problems like finding planar convex hulls, computing Delaunay triangulation, and the planar Traveling Salesman Problem. Here an attention pointer is used to select a member of the input sequence as the output instead of bothering to train up the hidden layers of the encoder and the decoder for the network.

4.6 Encoder-Decoder Networks

Recurrent Neural Networks Encoder-Decoder [21] consists of two gradient learnt recurrent neural networks, one to encode a variable length sequence of symbols into a fixed length vector representation and another decodes the fixed representation into another variable sequence of symbols to accomplish learning of conditional log-likelihood probability for a sequence based on a given sequence. It has been used for machine translation applications and have semantically and syntactically better translated and meaningful representation of linguistic phrases.

In [20], neural machine translation using encoder-decoder neural network and gated recursive convolutional neural network (grConv) have been described here. Gated recursive convolutional neural network is a binary convolutional neural network whose weights are recursively applied to the input sequence until the output converges to a single fixed-length vector. It has been explored that neural machine translation performs quite fruitfully for short sentences without any unknown words. But the moment the length of the sentence or the number of unknown words increase, there is a rapid decrease in performance. The gated recursive convolutional network can automatically and easily learn the grammatical structure of the sentence.

4.7 Multi-layer Kernel Machine

Multi-layer Kernel Machine [127] was introduced for better learning of the different hidden layers of the neural network through the use of kernel function which help in better feature engineering for the data driven learning approaches. Typical large deep neural network failed to provide a good way of generalization for several datasets because of their limitation in size and functional capacity. Here kernel function is used at each layer and then it is optimize with the help of evaluation of a tight upper bound of leave-one-out error of support vector machine rather instead of the traditional dual objective function. Kernel trick have gained tremendous success throughout the history of the machine learning because of its non-linear capability that can generate better separation between the classes for variety of samples and different sizes of samples and thus enable the classifier to learn very complex non-linear decision boundaries with very few physical parameters but projecting them to high dimensional space like in high-dimensional reproducing kernel Hilbert space and then performing the operation in the reproducing kernel Hilbert spaces. This feature particularly encouraged several users to analyze the hybrid kernel based systems feature learning and

here deep learning with kernel has been introduced with optimized multiple complete layers of kernels while increasing generalization and performance of the network for a variety of datasets.

In [19], deep kernel-based architectures called multiple layer kernel machines (MKMs) has been introduced which can generate better and rich set of representations for the data by learning the complex mapping of transformation of the input to reproducing kernel Hilbert spaces by transforming the inputs with the kernel based nonlinear multiple layers of the deep architectures. Other motivations for the wide acceptance of the kernelization of the deep learning are the range of functions that are available and can be parameterized by composing weakly nonlinear transformations, the appeal of hierarchical distributed representations of the data in kernel Hilbert spaces, and the potential for combining both unsupervised and supervised methods for training the classifier with the data.

In [172], another kernel machines based Multiple Kernel Learning (MKL) was introduced for the deep learning networks to solve real valued machine learning problems of classification and through the exploration of the non-linear feature spaces with the help of multiple kernel functions. Traditional Multiple Kernel Learning processes are shallow and the reproduced kernel is just a linear (or convex) combination of some base kernels, but this effort will help in learning very deep kernelization processes by combining the multiple kernels with the non-linear transformation of the multi-layer structure, a certainly out of bound of the conventional Multiple Kernel Learning approaches.

4.8 Compressed Deep Learning

Compressed Deep Learning is inevitable with the incresaing size of the deep learning architecture. Han et al. [43] proposed a deep compression series of steps like pruning, trained quantization and Huffman coding. All the three together reduce the storage requirement of neural network architecture. Neural networks consume high amount of memory and computation and it makes them very low suitable for limited hardware infrastructures. But compression can make them easily deploy-able in them. The first step comprises of pruning the network through identifying the important connections, than quantize the weights to quantized centroids to decrease the representation of weights and also involve sharing of weights. The last application of Huffman coding. In this way very complex and approximated neural network structures can be easily deployed in hand-held devices where the application size and the bandwidth of information are the major constraints.

In [103], there is another effort to reduce the size of the feed-forward deep neural networks through the use of rank-constrained topology where they factors the weights in the input layer of the deep neural networks in terms of a low-rank representation. It utilizes the natural two-dimensional time-frequency structure in the input and are more efficient in providing better performance for its proper dimentionality reduction. In other words, the compression of fully-trained network occurs with a low-rank approximation of the weights of each individual nodes in the first hidden layer of the network with the aid of a rank-constrained deep neural network layer topology. This produces significant reduction in the number of independent parameters and no loss of generality and performance in the model.

In [12], a neural network architecture called HashedNets is introduced exploiting the redundancy in neural networks to achieve considerable size reductions. HashedNets takes help of a low-cost hash function that randomly group up the connection weights into hash buckets, and all the connections within the same hash bucket share the single parametric values. The common parameters are then standardized to the HashedNets weight sharing architecture with standard training with back-propagation algorithm. This will help in the

increasing use of neural network for mobile devices and also grow the parameters of the models with the demanding increase in data capacity.

4.9 Deep Reinforcement Learning

Deep Reinforcement Learning [101, 102] is a new architecture which is self learning and exploratory architecture and can learn and update itself on its own with the help of other reward and penalty functional schemes. Deep learning model are used here to learn from the control policies directly from feedback raw input spaces using reinforcement learning schemes. The raw data input can be like high-dimensional sensory network data or images etc. For images, it can be a convolutional neural network getting trained with a variant of Q-learning scheme with the images being transformed to raw pixels and feed back into the system. The output of the feedback input is associated with a value function estimating future rewards. Deep Reinforcement Learning will suffer performance and halt the learning process if the reward functions fail to provide the effectiveness in terms that the scalar reward signal is not frequently sparse, noisy and delayed. If the effectiveness of the reinforcement lessen is prolonged in time, it will be unfeasible to learn from it. No doubt that Deep Reinforcement Learning requires a lot of labeled data, but the data also must not be of highly correlated states so that the diversity of learning can spread all over the feature space and the algorithm can learn the whole underlying distribution.

Wang et al. [146] introduced another Reinforcement Learning based Deep Learning called Deep Q-network, but this architecture is lead by better policy evaluation in the presence of many similar-valued actions. Deep Reinforcement Learning has many applications and they tend to use conventional architectures like convolutional neural networks, LSTMs, or auto-encoders and is based on requirement and data driven. But a model-free reinforcement learning based neural network architecture is presented with two separate estimators: one for the state value function and one for the state-dependent action advantage function. This Deep Q-network has been tested on different domains, such as Grid World, Mountain Car Problem, and Inverted Pendulum Problem to establish that the general architecture can learn the policy in these environments with the same architecture.

5 NLP & vision applications

There are various applications [129, 130] which have very high feature dimension space and many classes. Also the number of samples are so large that traditional machine learning based classifiers failed to provide considerable prediction accuracies, but Deep Learning have provided very promising performance because of its scalability both vertically and horizontally and because of its capability to deliver to the expectation of the problem. Also for deep learning, the hidden layers actually act as feature selectors and there are several regularizers and feature combination selection processes which can extract some very rich set of features and thus help in better prediction for any classifier at the extreme end. However there are difficulties in determining what is the best set of parameters for a dataset and feature space and what should be the optimum bias-variance trade-off. Also the learning rate and the optimization technique selection can be challenging or rather the best combination can prevent from getting the best possible prediction. NLP Applications & Vision Applications separately were most concentrated on classification tasks and some of the important applications with representation composition is discussion here, which gradually transformed into generative applications like language generation from contexts.

5.1 Convolutional Neural Network

Convolutional Neural Network [72] has provided the major breakthroughs in image classification and the main reason of huge success is all the low, middle and enriched level feature extraction and at the end using a classifier for classification. The primary idea behind Convolutional Neural Network is [55] where it was proven that certain change of visual effect has only limited effect on the receptor of the visual cortex. Convolutional Neural Network uses a filter like structure which replaces the lengthy generalized weight vector **w** and creates feature extraction at very local levels or regions. The dimension of the filter is an issue and have been proven that cost of generating these kind of local correlated features decreases if the filter size is kept low. This filter helps in generating weighted localized features through convolution followed by non-linear transformations. The convoluted features have strong spatial coorelation with the image and this is achieved through local connectivity enforcement between neuron layers. Convolutional Neural Network is a hierarchy of alternate convolution and pooling layers which helps in extracting and refining a rich series of spatial and temporal features of the large input space and is used for classification. Several pooling layers are defined like the max-pooling, min-pooling, average-pooling etc and are non-linear down-sampling as they "partition the input image into a set of non-overlapping rectangles", giving it a translation invariance through reduction of the spatial complexity of the feature space, which also prevents over-fitting due to reduced trainable parameters.

Karpathy et al. [65] used Convolutional Neural Networks for large-scale video classification. Convolutional Neural Network has provided state-of-the-art utility for image recognition, segmentation, detection and retrieval and effective model for understanding image content with high learning capability from very weakly-labeled data. the main reason is the high level of connectivity among the different portions of the architecture which integrates to extract several rich features out of the training data. Here slow fusion model for the Convolutional Neural Network has turned to be more effective than the early and late fusion alternatives.

In [23], a single convolutional neural network had been used as multitask learning engine for Natural Language Processing application. Multitask consists of the capacity to learn several labeling of the texts and then convolve these features to create new ones for better prediction and also extend it to semi-supervised learning for unlabeled texts. These labeling includes part-of-speech tags, chunks, named entity tags, semantic roles, semantically similar words and the likelihood sense of the sentence.

5.2 Convolutional Neural Network & Computer Vision

Deep Learning Architectures, mainly convolutional neural network, is perhaps the most widely applied in applications for Computer Graphics, Image Processing and Computer Vision. There are wide variety of applications in Computer Vision like image tagging, object detection, face recognition and biometrics, video processing and activity detection, etc and deep learning have been found very efficient in solving these problems.

In [112], faster version of region-based convolutional neural network (RCNN) had been proposed to solve the problem of object detection in images. Object detection in images is NP hard problem, and previously large amount of regions were actually used for object detection like in Selective Search, EdgeBoxes etc and was a time consuming affair. Faster R-CNN helped in reduction of time through the introduction of a region network for the images with the help of convolution features from the convolutional neural network. A

Region Proposal Network (RPN) is formed with the full-image convolutional features for detection network and detection of the objects can be done in near real time efficiency.

A 22 layers Deep Convolutional Neural Network architecture [134], named as GoogLeNet, was proposed and was applied on classification and detection of objects in images. It is based on the idea that to make a network learn a lot of things, there must be huge amount of data for learning and also the capacity of the network must scale up vertically and horizontally that is both in terms of levels and the number of hidden layers. One key principle of GoogleNet is "if the probability distribution of the data-set is representable by a large, very sparse deep neural network, then the optimal network topology can be constructed layer by layer by analyzing the correlation statistics of the activation of the last layer and clustering neurons with highly correlated outputs" which is another way of saying that learning is associated with triggering of certain part of the network and virtually they are the connected layers for certain learnt concepts and exploiting such concepts had been quite known as Hebbian Learning. They have also utilized the concept of Dropouts that is sparsely connected network structure for Deep Convolutional Neural Networks.

Another way of solving image recognition problem is done through Deep Residual Learning [45]. Deep Residual Learning was developed by Microsoft and is a modified version of very deep neural network and has a depth of up to 152 layers which is 8 times more than its previous contemporary VGG network. However it is not easy to train up such a big network, but Deep Residual Learning overcame this through the usage of skipping and forwarding the learnt coefficients structure before the propagation fades away. Gradient vanishing/exploding gradients problem was solved, but still efficient propagation of the mathematical model of learning requires special attention and also the size demands computational power. To overcome this problem, Deep Residual Learning utilized a strategy where one layer is connected with the next layer and also with the next to next layer. This actually helps in propagating the knowledge to two different layers in case it gets suppressed in one of them. Mathematically, it can be seen that the output of nth later is actually given by $F(\mathbf{x}_n) = \rho(\mathbf{x}_{n-1} + \rho(\mathbf{x}_{n-1}\mathbf{w}_{n-1})\mathbf{w}_n) = \rho(\mathbf{x}_{n-1} + \mathbf{x}_n\mathbf{w}_n)$. Till date this architecture is the winner of several image recognition competitions.

In [121], Deep Convolutional Networks (ConvNets) are trained to recognize actions in videos. The learning occurs from the motion of the object appearances in between two still images. Motion detection problem in videos is solved in three stages: one is the consideration of both spatial and temporal information of the videos and has separate Convolutional Networks to handle each of them. The spacial content comprises of individual objects and appearances like scene and state while timing information reveal the elements of motion like a car, hand of a person etc. Second is the multi-frame dense optical flow based learning for the networks and have used the motion flow of objects to learn what is happening in the sequence of images. Third is the task or action recognition for the data.

In [110], Deep Convolutional Neural Network has been used for upper human body pose estimation in videos using temporal information from multiple video frames. It is used as a generative model which is capable of tracking a human upper body movement through regression of the location of the human joints and produces much better constrained poses much faster.

In [122], Deep Fisher Network has been used for Large-Scale Image Classification. Deep Fisher Network, in addition to one deep convolutional neural networks, has an extra Fisher layer which acts as Fisher vectors based encoding for the images. This kind of Fisher vector encoding of feature based on parametric generative model (like Gaussian Mixture Model) can "captures the average first and second order differences between the features and each of the GMM centers". The main purpose of fisher layer is to "encode each local feature into

a high-dimensional representation, and then aggregates these encoding into a single vector by global sum-pooling over the whole image".

Apart from these, there are variety of classification problems in Computer Vision related to medical applications, satellite and spectral images, Synthetic Aperture Radar Images, fruit category classification [168], tea category classification [169], etc where deep learning can be very helpful in detecting categorical objects and event detection.

5.3 Natural Language Processing

Natural Language Processing like machine translation, question answering systems, text categorization, text summarization, ontology classification, sentiment analysis, etc are some of the most challenging NLP applications which lack feature topology and feature definition and yet believed to have very high dimension of sparse features. Traditional classifier using euclidean space cannot scale up for Natural Language Processing applications while non-linearity becomes inevitable. However, there are classifiers where the non-linearity is only applicable for small dimension and fails to scale up to applications like Natural Language Processing. Deep learning can always replace such inefficient traditional machine learning algorithms and followings are some Natural Language Processing applications that are solved by deep learning architectures.

In [165], Temporal Convolutional Network has been used to understand from various level of information that are conveyed in text starting from character level to abstract text concepts. Iyyer et al. [58] proposed a text classification method using Deep Unordered Composition Rivals Syntactic Methods. Compositionality is the key for text semantic and pragmatic deciphering of sequences and deep learning can be used for such computationally intensive task.

In [166], Character-level Convolutional Networks was used for Text Classification. This provided better results than bag of words, n-grams and their TFIDF variants while also defeated word-based ConvNets and recurrent neural networks in performance. Character-level Convolutional Networks is nothing but temporal Convolutional network or 1D Convolutional network.

In [104], a Health-Related Questions query model using Sparse Deep Learning has been proposed which can bridge the gap between what people seeks to maintain their health and what what the medical experts can provide them. This type of systems can help in developing "community-based health services" that can bridge the gap due to "vocabulary gap, incomplete information, correlated medical concepts, and limited high quality training samples". Deep Learning can help in inferring the possible diseases for the query made by the person and thus making it a text analysis based disease recognition problem.

In [141], a comparison has been made between Convolutional Neural Networks and Convolutional Tree Kernels, to solve the problem of answer sentence re-ranking for questions through feature engineering techniques like relational information. A very popular and promising break through in Natural Language Processing is the Word Embedding and has been dealt with details in Section 5.4 as it has provided some better ways of handling the Natural Language Processing problems.

5.4 Word Embedding

Large part of the success of NLP applications for large scale processing was achieved through word embedding and in image captioning, word embedding play a very important role. Word Embedding is another interesting application of deep learning network and

has emerged with very strong theoretical background and this is the reason why it is being dealt as a separate section instead of just an application. Word Embedding was originally introduced by Bengio [5] in 2001 and astonishingly it is much before Deep Learning was introduced in 2006. However distributed representations for symbols was originally introduced by Hinton in mid 1980s [49]. There are several entities, features and concepts which are equivalent but yet separated in space or rather they are mapped to a feature space that are much separated for the defined metric. In such cases, the learning gets affected as there occurs a contradiction for the concepts learnt. That is when Word Embedding concept can be helpful and can be extend to Concept Embedding, Symbol Embedding etc. Word Embedding can be defined as function which can take word to a definite set of numerical parameters which generally are in very high dimension and depends on the requirement and accordingly defined feature space. Mathematically it can be expressed as $f = \{W : \text{words} \rightarrow \mathbb{R}^n\}$. These mapping have also been defined and can vary from a lookup table, intelligent hash function, neural network etc. A number of such Word Embedding techniques are being described in this section. The main advantage of Word Embedding is that it connects the right sets of words to its equivalent or complementary counterparts through the mapping and thus reduces the chances of direct counteract during training sessions. Word embeddings, in disguise, helps in enabling better computational complexity for words through computing the word similarities through low-dimensional matrix operation.

Google introduced word2vec [40] where the main objective was to transform the words to a higher dimensional vector space with the use of a multi-layered neural network so that the vectors of word, that share common contexts in the corpus, have close vicinity in the transformed vector space. The number of layers and the level of the layers depends on the intention of length of the defined vector space where the word representation will reside. In word2vec, word pairing occurs based on context, relation and synonym and this is used to train the neural network and gradual learning will bring the similar contextual words clustered together. word2vec has found application in collaborative filtering and recommendation systems and also in bioinformatics as it uses several protein structure with different combination of ACGT protein component. The vector space of the resultant structure is generated automatically and is depends highly on initialization. Word Embedding like word2vec will prevent from the classifier from over-learning things and also bridge the gap for learning the same things in different ways.

In [91], Recursive Neural Network has been used for representing the transformation of morphologically similar words. This kind of transformation will prevent diversity in learning the contextual information in the morphologically similar word representations. Normally rare and complex word representations are poorly determined whereas the unknown words are represented using very few vectors. Recursive Neural Networks helps in learning the model of morphological structures of words and transform it to syntactic information where the aim is to learn the morphemic compositionality. This solves the problem of better estimation of rare and complex words and also tackling the unseen words becomes easier and the representation can be constructed as vectors of known morphemes.

The neural language models [5] or NLMs, on the other hand, utilize surrounding word contexts to provide the extra learning context and further hidden natural language semantics to be learnt for the Recursive Neural Networks to generate the morphemic representations of the words. The result is that this context-sensitive morpho Recursive Neural Networks embeddings have easily gained the edge over the embeddings on word similarity tasks and have significantly outperform those simple word embeddings.

In [100], Latent Semantic Analysis was introduced which can frame the representation of the words semantics in the most efficient way. Normal word embedding or even their

word count vector transformation is highly sparse and of high dimension and this is a crucial problem in learning and prediction based on distance metric. Hence Latent Semantic Analysis introduced a norm which can reduce such vectors to lower dimensions real-valued word embeddings and that even in less time. This speed gain is achieved through the elimination of the normalization costs during training and replacing it with simple variants of the log-bilinear model (can also be combined with noise-contrastive estimation for better performance and enhance modeling) which reduced the parameter update complexity linear with the word embedding dimensionality. Continuous-valued word embeddings learned by neural language models have great capacity of unrevealing the semantic and syntactic information about the associated words and have been very efficient for handling systems which deals with word based tasks. The main advantage of this kind of unnormalized embedding is that it fits an unnormalized models and thus time to train effectively becomes independent of the vocabulary size.

In [5] introduces a Neural Probabilistic Language Model whose main advantage is to counter the curse of dimensionality by learning a distributed representation for words and transform each sentence to some semantically similar neighboring sentences to be used for training. This allows the model to get trained with the exponential number of semantically neighboring sentences and thus prevents the model from learning diversely for the similar contextuality. Here neural network is used for the probability function and also takes advantage of longer contexts.

Collobert and Weston [23] introduced multitask learning for a convolutional neural network architecture that can convert a sentence to a host of language processing predictions like part-of-speech tags, chunks, named entity tags, semantic roles, semantically similar words and the likelihood that the sentence makes sense (grammatically and semantically) using a language model. This model uses the divide and conquer rule to identify the subtasks carefully for analysis. The subtask consists of syntactic like part-of-speech tagging, chunking and parsing, and semantic like wordsense disambiguation, semantic-role labeling, named entity extraction and anaphora resolution. This model uses some form of semi-supervised learning which helps in learning the unlabeled text while all the tasks use labeled data for determination of the subtasks.

In [99], a fast hierarchical language model for automatic construction of word trees from the data has been proposed and has been far better than the n-gram language models. constructing a binary tree of words is done by using expert knowledge, data-driven methods, or with both. Neural probabilistic language models form a binary tree of words from the data and thus have two orders of magnitude faster than the non-hierarchical model. Also normal neural probabilistic language models have very long training and testing sessions. Thus hierarchical neural probabilistic language model is introduced which provides exponential reduction in time complexity for learning and testing by dealing the unstructured word vocabulary with binary tree and thus represents a hierarchical clustering of words. Each word is placed in a leaf of the tree and uniquely represented by the edges from the root to that leaf.

In [135], large network embedding method called Large-scale Information Network Embedding (LINE) has been introduced. LINE can be used for any kind of graphs like undirected, directed, and/or weighted. It converts very large information networks into low-dimensional vector spaces which can be very useful for applications like visualization, node classification, link prediction etc. Language Network embeddings is based on word analogy and document classification and have been very effective in learning graphs with with millions of vertices and billions of edges.

DeepWalk [109] is an approach learning latent representations of vertices in a network. This is a very important framework for encoding the relation or the interaction in a continuous vector space through the use of statistical models. DeepWalk also helps in generalization of language modeling with unsupervised feature learning, like in deep learning, from sequences of words to graphs. "DeepWalk uses local information obtained from truncated random walks to learn latent representations by treating walks as the equivalent of sentences". The goal of this kind of language modeling is to gather statistical estimation and calculate the likelihood of a specific sequence of words appearing in a corpus.

Vector space representations of words from fine-grained semantic and syntactic regularities using vector arithmetic has been studied with much emphasis on the origin and details of the working principle of these regularities. In [108], the model properties needed for such regularities has been analyzed to emerge words as word vectors. They also introduced a global log bilinear regression model combining the advantages of two major models of global matrix factorization and local context window methods. It is being trained with the nonzero elements in a word-word co-occurrence matrix instead of the entire sparse matrix or on individual context windows in a large corpus. They have introduced a specific weighted least squares model that trains on global word-word co-occurrence counts and makes perfect use of statistics.

In [140], another way of dealing with sparse and very rare words has been described based on unsupervised methods or may be semi-supervised learning for large unlabeled or partly labeled corpora. Words are sometimes simply convert into a symbolic ID and then transformed into a feature vector using a one-hot representation. But the one-hot representation of a word have the problem of sparsity mainly for rare words which will be poorly estimated and the representation cannot handle the unappeared or new words. However the unsupervised learning based representations suffers from not so high accuracy and semi-supervised based learning can perform better because of the joint supervised and unsupervised investigation based learning.

Dhillon et al. [27] introduces Eigenwords and also qualitatively and quantitatively estimates that simple linear transformation based approaches give competent performance with respect to non-linear deep learning based representations. Eigenwords is a fast and scalable spectral algorithms for learning word embeddings to a low dimensional real vectors (called Eigenwords) where the "meaning" of words from their context is utilized for processing. The meanings of the words are processed using the the multi-view nature of text data with left and right context of each word.

Huang et al. [54] is based on pulling in global context instead of the local ones as the problem lies in words being polysemous and still represented as a single vector context. Avoiding such global context can be misleading mainly for large text processing applications where it is not even possible to review the learning and correctness of the model's interpretations. So a neural network architecture is being described which concentrates on "learns word embeddings that better capture the semantics of words by incorporating both local and global document context" and "accounts for homonymy and polysemy by learning multiple embeddings per word".

In [78], a linear contexts based continuous word embeddings was described. experiments shows that this kind of contexts dependent embedding can help in providing a completely different word embedding. The added advantages are that the word representations are less topical and exhibit more functional similarity than original skip-gram embeddings. However it can be regarded as generalization of the skip-gram embeddings where the linear bag-of-words based contexts are replaced with arbitrary ones.

In Latent semantic analysis (LSA) [76], the main idea is that it assumes that words, similar in meaning, will occur in similar documents and to statistically explore the expected contextual usage of words. Then it represent the word counts as matrix for different passages of a document and applies Singular Value Decomposition to reduce the dimension of the representations of the words and then cosine of the angle between the two vectors can be used to determine the similarity in context of the two words.

6 Image captioning

Describing an image with captions and sentences has been a significant research topic due to its impact on many applications and large number of people are engaging with each other through media as language. This creates the immense need for diversification of architectures and definition of representations that can be transformed to sentences. As machines have limitations in comprehension of the content of these media, it becomes evident to develop the capacity that can read and interpret these representations. In this section, we will provide some broad overview of the architectures and the different features and principles on which these work.

6.1 Different architecture description

6.1.1 Global Architectural Principles

There are many different Architectural Principles of captioning and we have defined 7 broad types where the main component is defined through the selected features from images. They are image-based, object-based, semantic-based, image and semantic-based, RCNN-Object-based, and RCNN-Object-Relationship-based. The other sector contains various other types which are computationally expensive (like Nearest Neighbor¡ Query and Ranking) or are mostly used for enhancing the performance as post-training heuristics (like Reinforcement Learning and GAN contention based training). Figure 1 provided the different architecture types as a graph for image captioning based on the different features utilized.

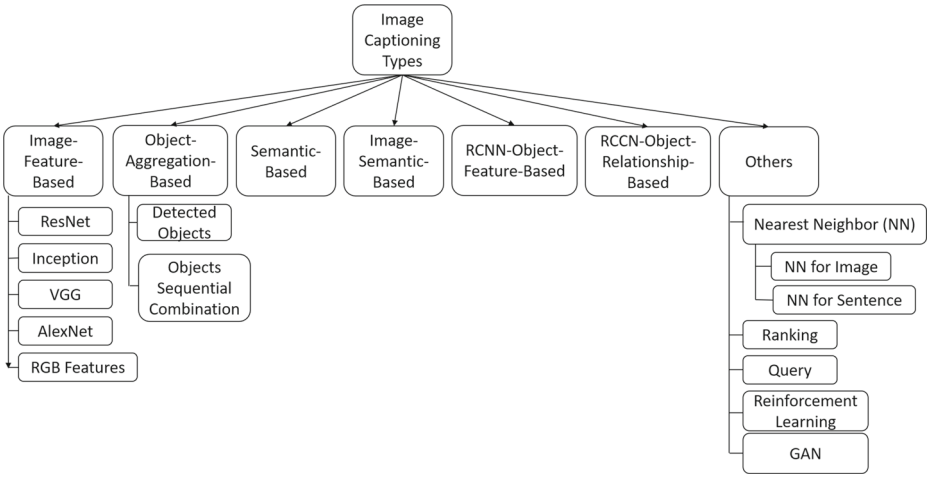

Fig. 1 Architectural Principles of captioning

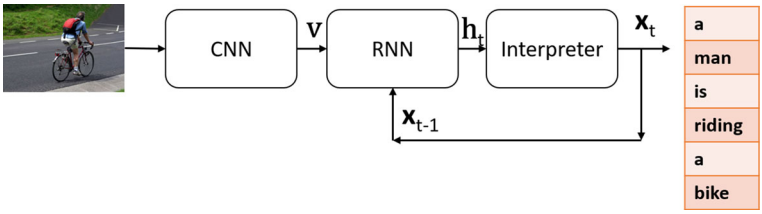

Fig. 2 Image Feature Based Captioning

6.1.2 Different convolution networks structural

Image features are important and the structural differences of the different convolution neural networks (CNN) provide diverse information. The most predominant CNN models are VGG, AlexNet, Inception (GoogleNet) and ResNet (from MircoSoft). While, most of the initial works were based on VGG extracted features and even on RGB features of images, recent works uses different trained architectures of ResNet (like ResNet50, ResNet101) for 2048 dimension feature extraction. ResNet uses a skip-through architecture where different layers of information are ensembled for propagation of information and thus prevent loss of information. Equation for each layer of ResNet can be denoted as $\mathbf{x}_t = \Phi(\mathbf{x}_{t-2}\mathbf{W}_{(t,t-2)}, \mathbf{x}_{t-1}\mathbf{W}_{(t,t-1)})$ where t is the tth layer and $\Phi(.)$ is the functional transformation.

6.1.3 Different features assembling architectures

It is important to understand that the automated feature space definition is based on different convolution neural network training and detection capability, while human annotated tagging are sometime used for caption as well. Figures 2, 3, 4, 5, 6 and 7 provided pictorial overview of the different architectures and how the different features are assembled in the neural structure for caption generation. These figures will provide the different overview of the architectures and will project their differences from the prospect of what happens in

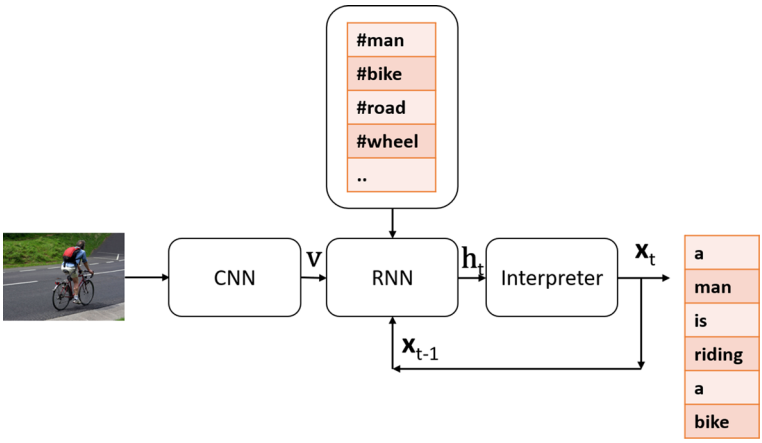

Fig. 3 Object Aggregation based captioning

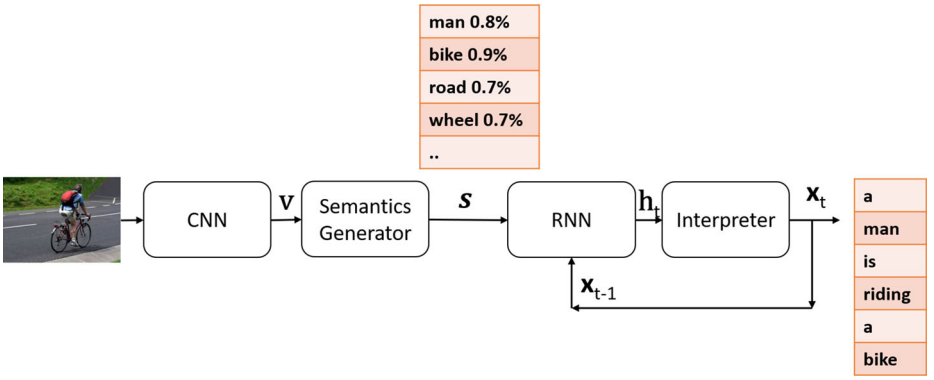

Fig. 4 Semantic based captioning

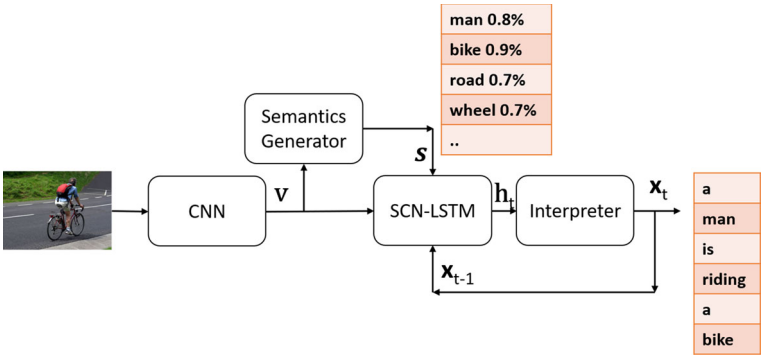

Fig. 5 Image-Semantic based captioning

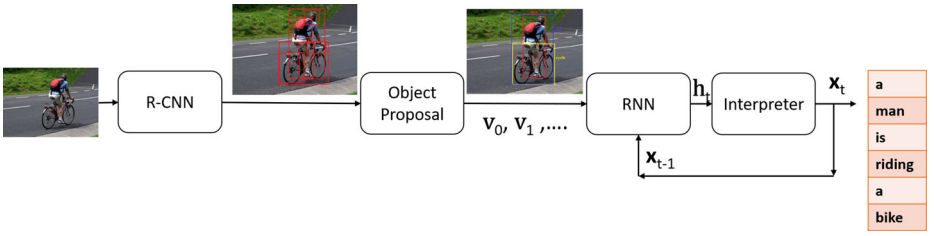

Fig. 6 RCNN-Object-Feature based captioning

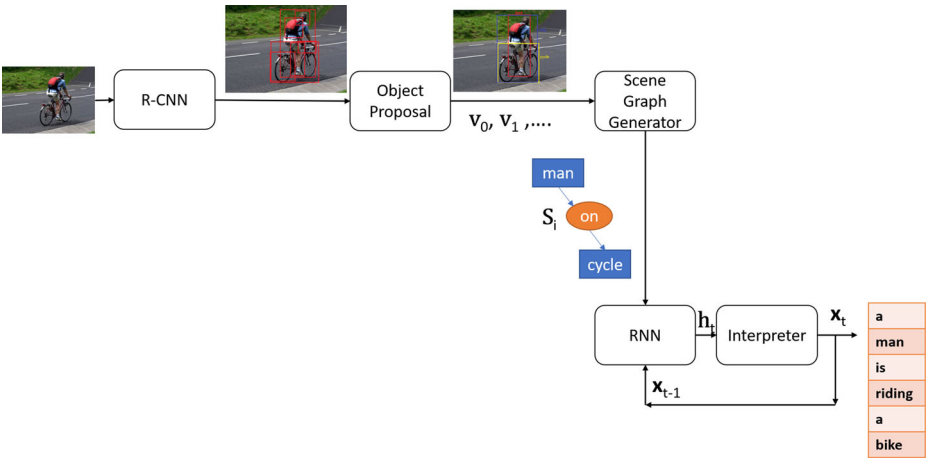

Fig. 7 RCNN-Object-Relationship based captioning

the algorithm. Figure 2 provided the Image Feature Based Captioning architecture, where deep convoluted features are extracted and used for providing the context overview for caption generation. Object-Aggregation-based architecture is shown in Fig. 3, where the object features are feed as embedding or extracted features for generation of context and the main aim of this architecture is to create the sequential combination of different objects through encoding for generation of captions. Unlike, semantic layer, where the layer combination of likelihood of the objects are used, object combination strategy advocates representation generation through combination and identification of similar contexts to have similar effects. Semantic-based architecture is shown in Fig. 4, where the semantic layer contributes as context for the caption. Initially, these semantic information were extracted from human annotated images and online tagged images, but later these were extracted from trained CNN architectures like ResNet. Image-Semantic-based architecture, demonstrated in Fig. 5, used the fusion of image and semantic layer for enhanced image captioning through the use of SCN-LSTM like units, shown in Fig. 10. The successful combination of two features is achieved though the guidance of the semantic layer to counter the variations in image features. RCNN-Object-Feature-based architecture is illustrated in Fig. 6, where RCNN object feature is derived from Faster RCNN, where combination of the region prediction and accurate object detection helping in detection of objects and their corresponding features for image captions. RCNN-Object-Relationship-based architecture of Fig. 7, used the multiple types of relationships among the different RCNN object features and helped in captioning of images.

6.1.4 Attention & no-attention models

Attention mechanism has significantly enhanced the image caption quality and provided a better overview of the contexts for the sequential recurrent units, where the weightage of the

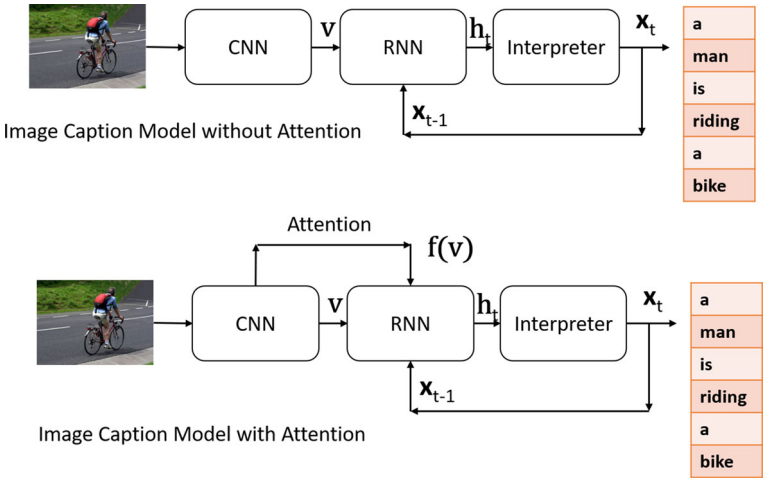

Fig. 8 Attention & no-attention models

context features gradually fades away and are hijacked by the language embedding features. Figure 8 provided the pictorial overview of attention and the no-attention model.

6.1.5 Examples of ground truth & generated captions

The training samples consist of five or more ground-truth sentences and some instances are provided in Fig. 9, where the initial sentences are some examples of the ground truth sentences and some generated captions from an attention based LSTM network is provided. MSCOCO (along with Flicker) dataset is mostly used.

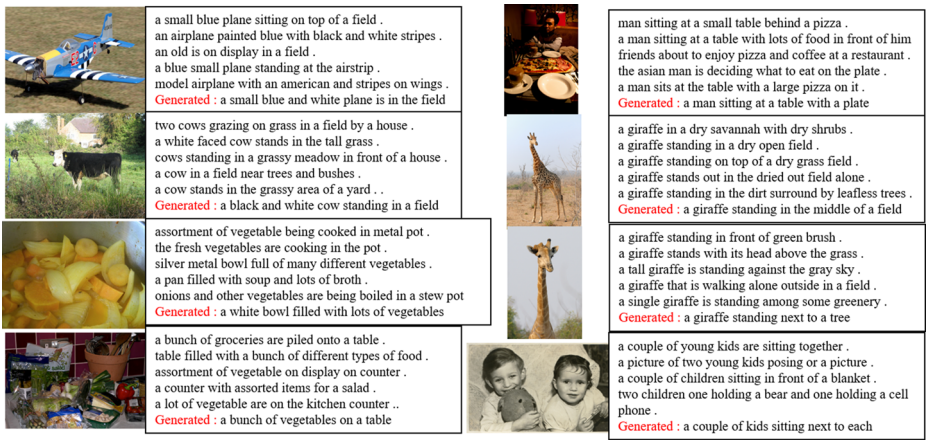

Fig. 9 Examples of ground truth & generated captions from attention based LSTM model using ResNet features

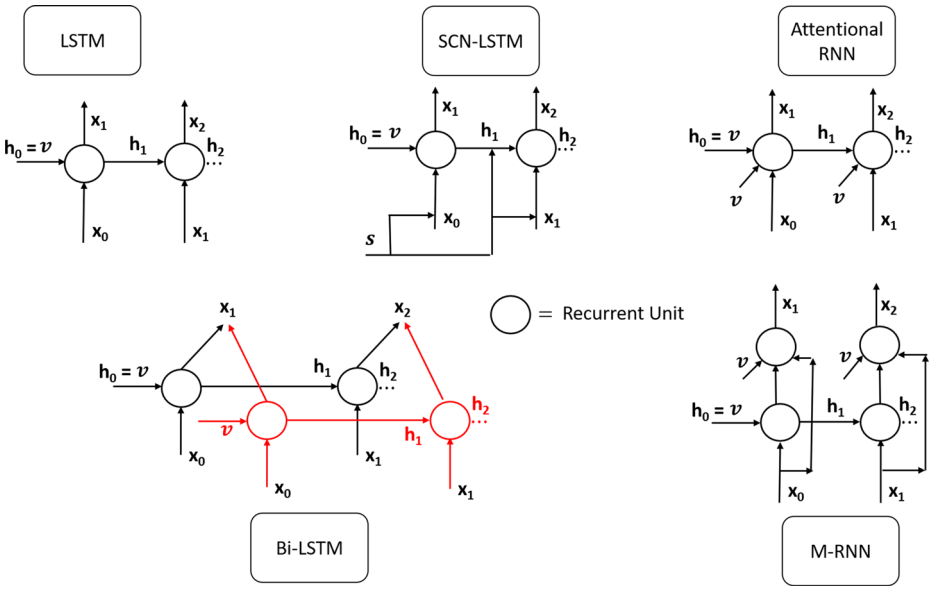

Fig. 10 Different Recurrent Neural Network Units

6.1.6 Different Recurrent Units Structures

Apart from the features, different Recurrent Units Structures are described in Fig. 10, which provided some description of feature utilization and fusion of features in the recurrent units and their structural differences in terms of physical weight-age for the variance and non-variance contexts.

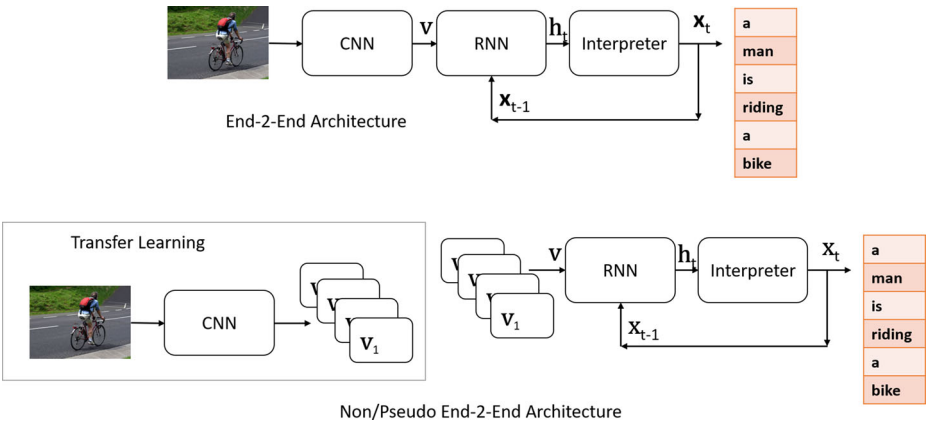

Fig. 11 End-2-end and non/pseudo end-2-end models

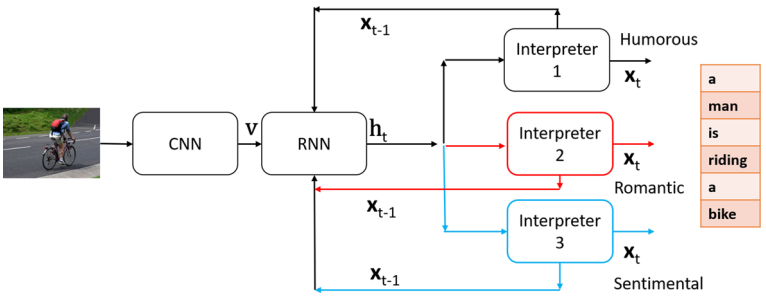

Fig. 12 Stylish Caption Generator models

6.1.7 Transfer learning and end-to-end

All decoding based generative models are end-to-end and there is no way to avoid it. However, since both the encoder and decoder parts are different and sometimes separate, people visualize them separately and make one of them constant to see the change in another. Figure 11 provided the difference between end-to-end and non-end-to-end models with transfer learning features.

In transfer learning, some parts of the model are trained/dealt separately and used as features, however, they were suppose to be integral part of the system. Transfer learning practices gained in as the training sessions were lengthy, costly and resource crunching and beyond the capability of many infrastructures. CNN, Word Embedding, discussed in Sections 5 and 5.4 provided some overview of the transfer learning technology individuals.

6.1.8 Stylish Caption Generator

Stylish Caption Generator is very important for many applications in modern day fancy applications and media based expression practices. Style (like Romantic, Humorous, Sentiments etc) can be extracted through definition of the vocabulary and combination of the contexts and such an architecture is shown Fig. 12.

6.1.9 Reinforcement Learning based caption

Normal sequential training generates the dependency among the different words in a vocabulary, however, the overall sentence remain neglected. Reinforcement Learning based captioning architecture, shown in Fig. 13, provides such a provision, where a heuristic difference (like evaluation equations) can be gradiented as training component for the

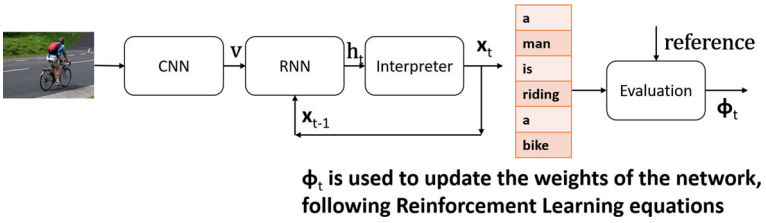

Fig. 13 Reinforcement Learning based captioning model

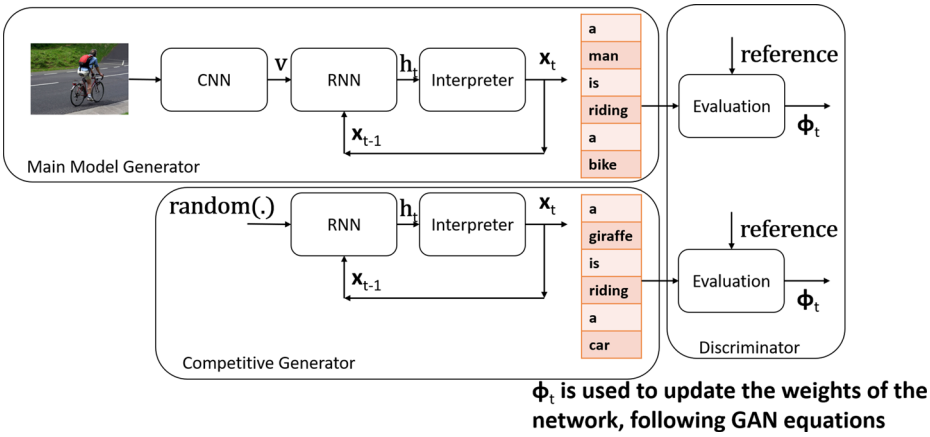

Fig. 14 GAN based captioning model

model. This is called Reinforcement Learning. Though, in many situations, Reinforcement Learning had been beneficial in enhancing the captions, it never guarantees such enhancement.

6.1.10 GAN based caption

Generative Adversarial Network (GAN) is a contention based training architecture used for training the architecture. A pictorial description of GAN based captioning architecture is shown in Fig. 14, where gradient is derived through the random generation and guided generation and thus helps in generating more opportunity for utilization of the unused weight space of the model.

6.1.11 Evaluation metric

The image caption uses different statistical formula for evaluation of the generated sentences. BLEU_n (as n-gram with $n = 1, 2, 3, 4$), CIDEr-D, ROUGE_L, METEOR and SPICE are the most prevalent evaluation techniques, used for image caption and language generation problems. Most of these techniques provide limited visibility of the quality of the generated sentences. However, there are hardly any complete evaluation technique and different statisticians have different opinions. However, BLEU_n (with $n = 3, 4$) provides estimation of combination based appearances compared with the ground-truths.

6.1.12 Beam Search

Beam Search is an important criteria in language model and generation of sentences. Figure 15 provided a diagram of Beam Search technique for beam size $= 3$. Beam Search helps in considerable improvement of the image captions and searches other longer and more viable combination of words as sentence. The deduction of any node is done through the calculation of the summation of the negative log scaled of the likelihood of the sentence segments.

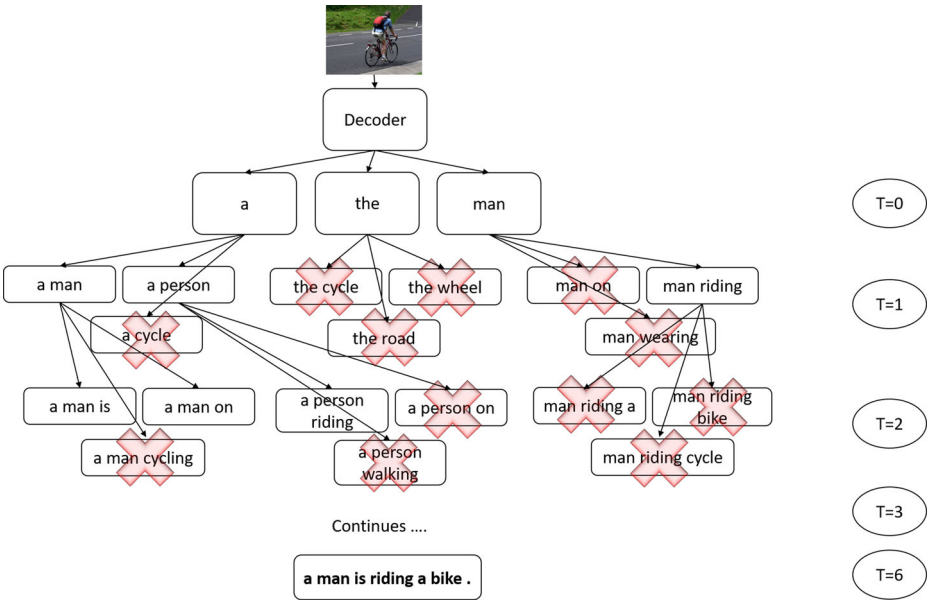

Fig. 15 Beam Search Example with Beam Size = 3, cancellation of any node is dependent on the log scaled of the individual likelihood of appearance of the words in a sentence for the corresponding likelihood vector

6.2 Progress in research

Table 1 provided selected description classification of some of the prominent and pioneering works in image captioning for different architectures.

Lu et al. [89] introduced a model that can generate informative image captions through features from CNN features of images and hash-tags from users as input, where extra set of information came from hash-tag embedding and provided more guidance for caption generation.

Lu et al. [90] approach first generated a sentence 'template' with slot locations explicitly related to different and specific image regions and then these identified slots were filled with prediction identified from visual concepts. The detectors were object predictors mainly regions based CNN models and the whole architecture for sentence template generation and slot filling with object detectors was operated as an end-to-end model.

You et al. [162] discussed sentiment-conveying image descriptions. Sentiment analysis and capability to generate based on sentiment attention is an effective way of describing many images. They mainly worked on effective ways of injecting the sentiment instincts without changing the semantic matching between visual features and generated sentences.

Melnyk et al. [95] 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. [153] used question features and image features to generate question-related captions for Visual Question Answer (VQA) dataset and the generated caption creates new

Table 1 Description details of some image caption papers highlighting the contributions and uniqueness

ID	Description (brief summary of the paper)	Categories (feature types)	Uniqueness (main contribution of the paper, which is unique)
Lu et al. [89]	Images and hash-tags from users as input	Image-semantics based	Hash-tags
Lu et al. [90]	Generated a sentence 'template' with slots for specific image regions	RCNN-Object-based	Sentence 'template' with slot locations
You et al. [162]	Sentiment without semantic matching between visual and sentences	Image based	Sentiment-conveying image descriptions
Melnyk et al. [95]	Enforcing semantic alignment between images and captions	Image based	Conditional GAN training
Wu et al. [153]	Question features and image features to generate question-related captions	Image based	Joint training
Kilickaya et al. [66]	Caption generated based on comparison between image and relevant images	Semantic based	Object-based semantic representation
Chen et al. [14]	Visual parsing tree transformed into an embedding	Image semantic based	Parsing tree features
Jiang et al. [61]	A feature space of different attributes from images	RCNN-Object-Relationship	Guiding network feature space
Wu et al. [154]	Word-conditional semantics from object and attribute words from images	Image-Semantic	Dual temporal modal
Fu et al. [35]	Synthesized pseudo image-sentence pairs, learning concepts of captioning	Image-based	Synthesized pseudo image-sentence pairs
Chen et al. [16]	Attributes used were the objects detected in the images	RCNN-Object-based	Attribute-driven attention
Cornia et al. [24]	Segregating different parts of the image as salient and contextual	Image-based	Generative recurrent neural network
Zhao et al. [171]	Rich category-aware representations and syntax aware LSTM decoder	Image Based	Added categorization, syntax knowledge
Li et al. [80]	Recommendation-assisted collective annotation system	Semantic-Image based	Text-guided attention
Ye et al. [160]	Spatial attention, channel-wise attention and visual dependence	Image based	High definition matrix from image
Chen et al. [18]	Styles including humorous, romantic, positive, and negative	Image based	Special set of sentences
Chen et al. [17]	Producing captions that contained diverse or relevant information	Image based	Structural relevance and structural diversity
Harzig et al. [44]	Detect the popular brands in the images and relevant captions	RCNN-Object	Specialized captions
Liu et al. [85]	Self-retrieval module for generation and gradient	Image based	GAN

Table 1 (continued)

ID	Description (brief summary of the paper)	Categories (feature types)	Uniqueness (main contribution of the paper, which is unique)
Sharma et al. [119]	Conceptual Captions with wider variety of captions for the images	Image based	Specific varieties and styles
Anderson et al. [1]	Bottom up top down model with RCNN, multiple utilizable training	RCNN-Object-based	Multiple level representation
Zhang et al. [170]	Adaptive re-weight scheme for the loss of different samples	Image based	Semantic loss functional gradient
Park et al. [107]	Personalized image captioning with prior knowledge of a person's habit	Image based	Personalized
Wang et al. [148]	Deep bidirectional LSTM for image captions	Image based	Bi-LSTM
Gan et al. [38]	Network for semantics and image, concept of decomposition of features	Image semantic based	SCN network

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. [66] 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. [14] 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. [61] 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. [154] 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. [35] 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. [16] 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. [24] 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. [171] 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. [80] 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. [13] 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. [136] 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. [15] 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. [160] 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. [147] 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. [18] 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. [17] 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. [83] 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. [44] introduced a model 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. [85] 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.

Park et al. [106] 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. [119] 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. [158] 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. [167] 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. [34] 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. [113] 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. [84] 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. [22] 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. [82] 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. [159] 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. [88] 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. [145] 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. [170] 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. [107] 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. [148] 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. [114] 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. [152] 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. Here attribute prediction layer was used for high level semantic concept layer.

Vinyals et al. [143] 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. [64] proposed a bidirectional retrieval model capable of retrieving description of images through the construction of sentences through a deep, multi-modal 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. [155] 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. [31] 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 and Li [63] 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 inter-modal connection between language and visual data and aligned the two modalities through a multimodal embedding.

Hendricks et al. [47] 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 leveraged large object recognition datasets and external text corpora and through transferring knowledge between semantically similar concepts.

Chen and Zitnick [10] 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. [26] 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. [29] 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. [38] 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. [62] 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. [67] 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. [68] 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. [92] 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 and Hinton [96] 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. [111] 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. [124] 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. [132] 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. [133] 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.

Tran et al. [138] 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. [139] 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. [151] 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. [157] 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. [161] 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. [163] 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. [32] 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. [37] 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. [39] 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. [52] 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. [60] 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. [71] 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. [73] 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. [79] studied an effective phenomenon for automatic composition of image descriptions from the visual features and using web-scale n-grams, unlike previous works where the task 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. [75] 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. [93] 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. [94] 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. [98] 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. [105] 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. [156] 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.

7 Conclusion

So in conclusion we can say that the deep learning is still much in its primitive form and there is way ahead before it can literally replicate or rather mimicry the cognitive and computational neuroscience. Till now, it is a numerical based mathematically favorable structures which can neither generalize nor it can develop conceptual and relational learning. However from the prospect of computational intelligence and certain problem solving,

artificial neural network and deep learning have provided a lot of scalability and the ability to generate better transfer functions which can identify entities it has learnt and created efficient hyper-planes for very high dimensional feature space. The main reason deep learning has evolved because of its ability to transfer the learnt knowledge through its neural components efficiently and effectively. Also deep learning has been very efficient in extraction of better features which are both discriminative and descriptive and thus produces the best set of very rich feature sets which can help in better classification and regression analysis. There was a time, when people were dependent on feature engineering for better artificial intelligence and decision making, but with the robustness of deep learning, that sole requirement has been transformed to mere refinement.

Image captioning deals with generation of sentences from image context and the performance of different architectures are yet to achieve the best in terms of different evaluation techniques. The main drawback is the ineffectiveness in defining the context combination and limitation in generating the interaction among different entities in the image. This is mainly because of ineffectiveness of the defined context and inability of the recurrent units to generalize and identify them. New works must progress towards generalization and context building, with more accurate sentence generation. The future works can range from large scale integration of representation for natural languages and definition of image transformation towards selective combination of regions. New work in industry are progressing towards neuromorphing of technology and computational progresses are heading towards more robust models which can identify and correlate different sources of data and provide inference.

References

1. Anderson P, He X, Buehler C, Teney D, Johnson M, Gould S, Zhang L (2018) Bottom-up and top-down attention for image captioning and visual question answering. In: CVPR, vol 3, p 6
2. Baldi P (2012) Autoencoders, unsupervised learning, and deep architectures. In: ICML Unsupervised and Transfer Learning, vol 27, p 1
3. Bayer J, Wierstra D, Togelius J, Schmidhuber J (2009) Evolving memory cell structures for sequence learning. In: International conference on artificial neural networks. Springer, Berlin, pp 755–764
4. Bebis G, Georgiopoulos M (1994) Feed-forward neural networks. IEEE Potentials 13(4):27–31
5. Bengio Y, Ducharme R, Vincent P, Jauvin C (2003) A neural probabilistic language model. J Mach Learn Res 3:1137–1155
6. Bengio Y, Lamblin P, Popovici P, Larochelle H (2007) Greedy layer-wise training of deep networks. In: Advances in neural information processing systems 19. MIT Press, Cambridge
7. Bengio Y, Boulanger-Lewandowski N, Pascanu R (2013) Advances in optimizing recurrent networks. In: 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, pp 8624–8628
8. Bordes A, Weston J Learning end-to-end goal-oriented dialog. arXiv:1605.07683
9. Bordes A, Usunier N, Chopra S, Weston J Large-scale simple question answering with memory networks. arXiv:1506:02075
10. Chen X, Zitnick CL (2015) Mind's eye: a recurrent visual representation for image caption generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition
11. Chen S, Cowan CF, Grant PM (1991) Orthogonal least squares learning algorithm for radial basis function networks. IEEE Trans Neural Netw 2(2):302–309
12. Chen W, Wilson JT, Tyree S, Weinberger KQ, Chen Y (2015) Compressing neural networks with the hashing trick. arXiv:1504.04788
13. Chen M, Ding G, Zhao S, Chen H, Liu Q, Han J (2017) Reference based LSTM for image captioning. In: AAAI, pp 3981–3987
14. Chen F, Ji R, Su J, Wu Y, Wu Y (2017) Structcap: structured semantic embedding for image captioning. In: Proceedings of the 2017 ACM on multimedia conference. ACM, pp 46–54
15. Chen H, Zhang H, Chen PY, Yi J, Hsieh CJ (2017) Show-and-fool: crafting adversarial examples for neural image captioning. arXiv:1712.02051

16. Chen H, Ding G, Lin Z, Zhao S, Han J (2018) Show, observe and tell: attribute-driven attention model for image captioning. In: IJCAI, pp 606–612
17. Chen F, Ji R, Sun X, Wu Y, Su J (2018) GroupCap: group-based image captioning with structured relevance and diversity constraints. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1345–1353
18. Chen T, Zhang Z, You Q, Fang C, Wang Z, Jin H, Luo J (2018) "Factual" or "Emotional": stylized image captioning with adaptive learning and attention. arXiv:1807.03871
19. Cho Y, Saul LK (2009) Kernel methods for deep learning. In: Advances in neural information processing systems, pp 342–350
20. Cho K, Van Merriënboer B, Bahdanau D, Bengio Y (2014) On the properties of neural machine translation: encoder-decoder approaches. arXiv:1409.1259
21. Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078
22. Cohn-Gordon R, Goodman N, Potts C (2018) Pragmatically informative image captioning with character-level reference. arXiv:1804.05417
23. Collobert R, Weston J (2008) A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning. ACM, pp 160–167
24. Cornia M, Baraldi L, Serra G, Cucchiara R (2018) Paying more attention to saliency: image captioning with saliency and context attention. ACM Trans Multimed Comput Commun Appl (TOMM) 14(2):48
25. Courville AC, Bergstra J, Bengio Y (2011) A spike and slab restricted Boltzmann machine. In: AISTATS, vol 1, p 5
26. Devlin J et al (2015) Language models for image captioning: the quirks and what works. arXiv:1505.01809
27. Dhillon PS, Foster DP, Ungar LH (2015) Eigenwords: spectral word embeddings. J Mach Learn Res 16(1):3035–3078
28. Doersch C (2016) Tutorial on variational autoencoders. arXiv:1606.05908
29. Donahue J et al (2015) Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition
30. Du B, Xiong W, Wu J, Zhang L, Zhang L, Tao D (2016) Stacked convolutional denoising auto-encoders for feature representation. IEEE trans Cybern 47(4):1017–1027
31. Fang H et al (2015) From captions to visual concepts and back. In: Proceedings of the IEEE conference on computer vision and pattern recognition
32. Farhadi A et al (2010) Every picture tells a story: Generating sentences from images. In: European conference on computer vision. Springer, Berlin
33. Fu J et al (2016) Deep Q-networks for accelerating the training of deep neural networks. arXiv:1606.01467
34. Fu K, Jin J, Cui R, Sha F, Zhang C (2017) Aligning where to see and what to tell: image captioning with region-based attention and scene-specific contexts. IEEE Trans Pattern Anal Mach Intell 39(12):2321–2334
35. Fu K, Li J, Jin J, Zhang C (2018) Image-text surgery: efficient concept learning in image captioning by generating pseudopairs. IEEE Trans Neural Netw Learn Syst 29.12(2018):5910–5921
36. Funahashi KI, Nakamura Y (1993) Approximation of dynamical systems by continuous time recurrent neural networks. Neural Netw 6(6):801–806
37. Gan Z et al (2016) Semantic compositional networks for visual captioning. arXiv:1611.08002
38. Gan C et al (2017) Stylenet: generating attractive visual captions with styles. In: CVPR
39. Girshick R et al (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition
40. Goldberg Y, Levy O (2014) word2vec explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv:1402.3722
41. Graupe D (1997) Large scale memory storage and retrieval (LAMSTAR) network. In: Principles of artificial neural networks, pp 191–222
42. Graves A, Wayne G, Danihelka I (2014) Neural turing machines. arXiv:1410.5401
43. Han S, Mao H, Dally WJ (2015) Deep compression: compressing deep neural network with pruning, trained quantization and Huffman coding. arXiv:1510.00149
44. Harzig P, Brehm S, Lienhart R, Kaiser C, Schallner R (2018) Multimodal image captioning for marketing analysis. arXiv:1802.01958
45. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv:1512.03385

46. Heermann PD, Khazenie N (1992) Classification of multispectral remote sensing data using a back-propagation neural network. IEEE Trans Geosci Remote Sens 30(1):81–88
47. Hendricks LA et al (2016) Deep compositional captioning: describing novel object categories without paired training data. In: Proceedings of the IEEE conference on computer vision and pattern recognition
48. Herculano-Houzel S (2009) The human brain in numbers: a linearly scaled-up primate brain. Front Hum Neurosci 3:31
49. Hinton GE (1986) Learning distributed representations of concepts. In: Proceedings of the eighth annual conference of the cognitive science society, vol 1, p 12
50. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18:1527–1554
51. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
52. Hodosh M, Young P, Hockenmaier J (2013) Framing image description as a ranking task: data, models and evaluation metrics. J Artif Intell Res 47:853–899
53. Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci 79(8):2554–2558
54. Huang EH, Socher R, Manning CD, Ng AY (2012) Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th annual meeting of the association for computational linguistics: long papers, vol 1. Association for Computational Linguistics, pp 873–882
55. Hubel DH, Wiesel TN (1959) Receptive fields of single neurones in the cat's striate cortex. J Physiol 148(3):574–591
56. Hutchinson B, Deng L, Yu D (2013) Tensor deep stacking networks. IEEE Trans Pattern Anal Mach Intell 35(8):1944–1957
57. Irsoy O, Cardie C (2014) Deep recursive neural networks for compositionality in language. In: Advances in neural information processing systems, pp 2096–2104
58. Iyyer M, Manjunatha V, Boyd-Graber J, Daumé H III (2015) Deep unordered composition rivals syntactic methods for text classification. In: Proceedings of the association for computational linguistics
59. Izhikevich EM (2004) Which model to use for cortical spiking neurons? IEEE Trans Neural Netw 15(5):1063–1070
60. Jia X et al (2015) Guiding the long-short term memory model for image caption generation. In: Proceedings of the IEEE international conference on computer vision
61. Jiang W, Ma L, Chen X, Zhang H, Liu W (2018) Learning to guide decoding for image captioning. arXiv:1804.00887
62. Jin J et al (2015) Aligning where to see and what to tell: image caption with region-based attention and scene factorization. arXiv:1506.06272
63. Karpathy A, Li F-F (2015) Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE conference on computer vision and pattern recognition
64. Karpathy A, Joulin A, Li FFF (2014) Deep fragment embeddings for bidirectional image sentence mapping. In: Advances in neural information processing systems
65. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L (2014) Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1725–1732
66. Kilickaya M, Akkus BK, Cakici R, Erdem A, Erdem E, Ikizler-Cinbis N (2017) Data-driven image captioning via salient region discovery. IET Comput Vis 11(6):398–406
67. Kiros R, Salakhutdinov R, Zemel RS (2014) Unifying visual-semantic embeddings with multimodal neural language models. arXiv:1411.2539
68. Kiros R, Zemel R, Salakhutdinov R (2014) A multiplicative model for learning distributed text-based attribute representations. In: Advances in neural information processing systems
69. Kohonen T (1995) Learning vector quantization. In: Self-organizing maps. Springer, Berlin, pp 175–189
70. Kohonen T, Somervuo P (1998) Self-organizing maps of symbol strings. Neurocomputing 21(1):19–30
71. Krishna R et al (2017) Visual genome: connecting language and vision using crowdsourced dense image annotations. Int J Comput Vis 123(1):32–73
72. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
73. Kulkarni G et al (2013) Babytalk: understanding and generating simple image descriptions. IEEE Trans Pattern Anal Mach Intell 35(12):2891–2903
74. Kumar A, Irsoy O, Su J, Bradbury J, English R, Pierce B, ..., Socher R (2015) Ask me anything: dynamic memory networks for natural language processing. arXiv:1506.07285
75. Kuznetsova P et al (2014) TREETALK: composition and compression of trees for image descriptions. TACL 2(10):351–362

76. Landauer TK, Foltz PW, Laham D (1998) An introduction to latent semantic analysis. Discourse Process 25(2–3):259–284
77. Lee H, Grosse R, Ranganath R, Ng AY (2009) Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th annual international conference on machine learning. ACM, pp 609–616
78. Levy O, Goldberg Y (2014) Dependency-based word embeddings. In: ACL (2), pp 302–308
79. Li S et al (2011) Composing simple image descriptions using web-scale n-grams. In: Proceedings of the fifteenth conference on computational natural language learning. Association for Computational Linguistics
80. Li X, Wang X, Xu C, Lan W, Wei Q, Yang G, Xu J (2018) COCO-CN for cross-lingual image tagging, captioning and retrieval. arXiv:1805.08661
81. Lin Y, Tong Z, Zhu S, Yu K (2010) Deep coding network. In: Advances in neural information processing systems, pp 1405–1413
82. Liu C, Mao J, Sha F, Yuille AL (2017) Attention correctness in neural image captioning. In: AAAI, pp 4176–4182
83. Liu C, Sun F, Wang C, Wang F, Yuille A (2017) MAT: a multimodal attentive translator for image captioning. arXiv:1702.05658
84. Liu S, Zhu Z, Ye N, Guadarrama S, Murphy K (2017) Improved image captioning via policy gradient optimization of spider. In: Proceedings IEEE international conference on computer vision, vol 3, p 3
85. Liu X, Li H, Shao J, Chen D, Wang X (2018) Show, tell and discriminate: image captioning by self-retrieval with partially labeled data. arXiv:1803.08314
86. Lo SCB, Chan HP, Lin JS, Li H, Freedman MT, Mun SK (1995) Artificial convolution neural network for medical image pattern recognition. Neural Netw 8(7):1201–1214
87. Lotter W, Kreiman G, Cox D (2016) Deep predictive coding networks for video prediction and unsupervised learning. arXiv:1605.08104
88. Lu J, Xiong C, Parikh D, Socher R (2017) Knowing when to look: adaptive attention via a visual sentinel for image captioning. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), vol 6, p 2
89. Lu D, Whitehead S, Huang L, Ji H, Chang SF (2018) Entity-aware image caption generation. arXiv:1804.07889
90. Lu J, Yang J, Batra D, Parikh D (2018) Neural baby talk. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7219–7228
91. Luong T, Socher R, Manning CD (2013) Better word representations with recursive neural networks for morphology. In: CoNLL, pp 104–113
92. Mao J et al (2014) Deep captioning with multimodal recurrent neural networks (m-rnn). arXiv:1412.6632
93. Mao J et al (2015) Learning like a child: fast novel visual concept learning from sentence descriptions of images. In: Proceedings of the IEEE international conference on computer vision
94. Mathews AP, Xie L, He X (2016) SentiCap: generating image descriptions with sentiments. In: AAAI
95. Melnyk I, Sercu T, Dognin PL, Ross J, Mroueh Y (2018) Improved image captioning with adversarial semantic alignment. arXiv:1805.00063
96. Memisevic R, Hinton G (2007) Unsupervised learning of image transformations. In: IEEE conference on computer vision and pattern recognition, 2007. CVPR'07. IEEE
97. Mikolov T, Karafiát M, Burget L, Cernocký J, Khudanpur S (2010) Recurrent neural network based language model. In: Interspeech, vol 2, p 3
98. Mitchell M et al (2012) 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
99. Mnih A, Hinton GE (2009) A scalable hierarchical distributed language model. In: Advances in neural information processing systems, pp 1081–1088
100. Mnih A, Kavukcuoglu K (2013) Learning word embeddings efficiently with noise-contrastive estimation. In: Advances in neural information processing systems, pp 2265–2273
101. Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, Riedmiller M (2013) Playing atari with deep reinforcement learning. arXiv:1312.5602
102. Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, ..., Petersen S (2015) Human-level control through deep reinforcement learning. Nature 518(7540):529–533
103. Nakkiran P, Alvarez R, Prabhavalkar R, Parada C (2015) Compressing deep neural networks using a rank-constrained topology
104. Nie L, Wang M, Zhang L, Yan S, Zhang B, Chua TS (2015) Disease inference from health-related questions via sparse deep learning. IEEE Trans Knowl Data Eng 27(8):2107–2119

105. Ordonez V, Kulkarni G, Berg TL (2011) Im2text: describing images using 1 million captioned photographs. In: Advances in neural information processing systems
106. Park CC, Kim B, Kim G (2017) Attend to you: personalized image captioning with context sequence memory networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 895–903
107. Park CC, Kim B, Kim G (2018) Towards personalized image captioning via multimodal memory networks. IEEE Trans Pattern Anal Mach Intell 41.4(2018):999–1012
108. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: EMNLP, vol 14, pp 1532–43
109. Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 701–710
110. Pfister T, Simonyan K, Charles J, Zisserman A (2014) Deep convolutional neural networks for efficient pose estimation in gesture videos. In: Asian conference on computer vision. Springer International Publishing, pp 538–552
111. Pu Y et al (2016) Variational autoencoder for deep learning of images, labels and captions. In: Advances in neural information processing systems
112. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99
113. Ren Z, Wang X, Zhang N, Lv X, Li LJ (2017) Deep reinforcement learning-based image captioning with embedding reward. arXiv:1704.03899
114. Rennie SJ, Marcheret E, Mroueh Y, Ross J, Goel V (2017) Self-critical sequence training for image captioning. In: CVPR, vol 1, p 3
115. Salakhutdinov R, Hinton G (2009) Semantic hashing. Int J Approx Reason 50(7):969–978
116. Salakhutdinov R, Hinton GE (2009) Deep Boltzmann Machines. In: AISTATS, vol 1, p 3
117. Salakhutdinov R, Tenenbaum JB, Torralba A (2013) Learning with hierarchical-deep models. IEEE Trans Pattern Anal Mach Intell 35(8):1958–1971
118. Schmidhuber J (1992) Learning complex, extended sequences using the principle of history compression. Neural Comput 4(2):234–242
119. Sharma P, Ding N, Goodman S, Soricut R (2018) 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 (volume 1: long papers), vol 1, pp 2556–2565
120. Shaw AM, Doyle FJ, Schwaber JS (1997) A dynamic neural network approach to nonlinear process modeling. Comput Chem Eng 21(4):371–385
121. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems, pp 568–576
122. Simonyan K, Vedaldi A, Zisserman A (2013) Deep fisher networks for large-scale image classification. In: Advances in neural information processing systems, pp 163–171
123. Socher R, Lin CC, Manning C, Ng AY (2011) Parsing natural scenes and natural language with recursive neural networks. In: Proceedings of the 28th international conference on machine learning (ICML-11), pp 129–136
124. Socher R et al (2014) Grounded compositional semantics for finding and describing images with sentences, vol 2, pp 207–218
125. Srivastava N, Salakhutdinov RR (2012) Multimodal learning with deep Boltzmann machines. In: Advances in neural information processing systems, pp 2222–2230
126. Srivastava N, Hinton GE, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958
127. Strobl EV, Visweswaran S (2013) Deep multiple kernel learning. In: 2013 12th international conference on machine learning and applications (ICMLA), vol 1. IEEE, pp 414–417
128. Sukhbaatar S, Szlam A, Weston J, Fergus R End-To-End memory networks. NIPS 2015 (and arXiv:1503.08895)
129. Sur C (2018) DeepSeq: learning browsing log data based personalized security vulnerabilities and counter intelligent measures. J Ambient Intell Humaniz Comput (2018):1–30
130. Sur C (2018) Ensemble one-vs-all learning technique with emphatic & rehearsal training for phishing email classification using psychology. J Exp Theor Artif Intell 30.6(2018):733–762
131. Sutskever I, Hinton GE, Taylor GW (2009) The recurrent temporal restricted boltzmann machine. In: Advances in neural information processing systems, pp 1601–1608
132. Sutskever I, Martens J, Hinton G (2011) Generating text with recurrent neural networks. In: Proceedings of the 28th international conference on machine learning (ICML-11)

133. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: Advances in neural information processing systems
134. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, ..., Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
135. Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) Line: large-scale information network embedding. In: Proceedings of the 24th international conference on World Wide Web. ACM, pp 1067–1077
136. Tavakoliy HR, Shetty R, Borji A, Laaksonen J (2017) Paying attention to descriptions generated by image captioning models. In: 2017 IEEE international conference on computer vision (ICCV). IEEE, pp 2506–2515
137. Torralba A, Tenenbaum JB, Salakhutdinov RR (2011) Learning to learn with compound hd models. In: Advances in neural information processing systems, pp 2061–2069
138. Tran D et al (2015) Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision
139. Tran K et al (2016) Rich image captioning in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops
140. Turian J, Ratinov L, Bengio Y (2010) Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, pp 384–394
141. Tymoshenko K, Bonadiman D, Moschitti A (2016) Convolutional neural networks vs. convolution kernels: feature engineering for answer sentence reranking. In: Proceedings of NAACL-HLT, pp 1268–1278
142. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11:3371–3408
143. Vinyals O et al (2015) Show and tell: a neural image caption generator. In: Proceedings of the IEEE conference on computer vision and pattern recognition
144. Vinyals O, Fortunato M, Jaitly N (2015) Pointer networks. In: Advances in neural information processing systems, pp 2692–2700
145. Vinyals O, Toshev A, Bengio S, Erhan D (2017) Show and tell: lessons learned from the 2015 mscoco image captioning challenge. IEEE Trans Pattern Anal Mach Intell 39(4):652–663
146. Wang Z, de Freitas N, Lanctot M (2015) Dueling network architectures for deep reinforcement learning. arXiv:1511.06581
147. Wang Y, Lin Z, Shen X, Cohen S, Cottrell GW (2017) Skeleton key: image captioning by skeleton-attribute decomposition. arXiv:1704.06972
148. Wang C, Yang H, Meinel C (2018) Image captioning with deep bidirectional LSTMs and multi-task learning. ACM Trans Multimed Comput Commun Appl (TOMM) 14(2s):40
149. Weston J, Chopra S, Bordes A Memory networks. ICLR 2015 (and arXiv:1410.3916)
150. Weston J, Bordes A, Chopra S, Rush AM, van Merriënboer B, Joulin A, Mikolov T (2015) Towards ai-complete question answering: a set of prerequisite toy tasks. arXiv:1502.05698
151. Wu Q et al (2016) 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
152. Wu Q, Shen C, Wang P, Dick A, van den Hengel A (2017) Image captioning and visual question answering based on attributes and external knowledge. IEEE Trans Pattern Anal Mach Intell 40.6(2017):1367–1381
153. Wu J, Hu Z, Mooney RJ (2018) Joint image captioning and question answering. arXiv:1805.08389
154. Wu C, Wei Y, Chu X, Su F, Wang L (2018) Modeling visual and word-conditional semantic attention for image captioning. Signal Process Image Commun 67(2018):100–107
155. Xu K et al (2015) Show, attend and tell: neural image caption generation with visual attention. In: International conference on machine learning
156. Yang Y et al (2011) Corpus-guided sentence generation of natural images. In: Proceedings of the conference on empirical methods in natural language processing. Association for Computational Linguistics
157. Yang Z et al (2016) Review networks for caption generation. In: Advances in neural information processing systems
158. Yao T, Pan Y, Li Y, Mei T (2017) Incorporating copying mechanism in image captioning for learning novel objects. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 5263–5271
159. Yao T, Pan Y, Li Y, Qiu Z, Mei T (2017) Boosting image captioning with attributes. In: IEEE international conference on computer vision, ICCV, pp 22–29

160. Ye S, Liu N, Han J (2018) Attentive linear transformation for image captioning. IEEE Trans Image Process 27.11(2018):5514–5524
161. You Q et al (2016) Image captioning with semantic attention. In: Proceedings of the IEEE conference on computer vision and pattern recognition
162. You Q, Jin H, Luo J (2018) Image captioning at will: a versatile scheme for effectively injecting sentiments into image descriptions. arXiv:1801.10121
163. Young P et al (2014) From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. Transactions of the Association for Computational Linguistics 2:67–78
164. Yu D, Deng L, Seide F (2013) The deep tensor neural network with applications to large vocabulary speech recognition. IEEE Trans Audio Speech Lang Process 21(2):388–396
165. Zhang X, LeCun Y (2015) Text understanding from scratch. arXiv:1502.01710
166. Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. In: Advances in neural information processing systems, pp 649–657
167. Zhang L, Sung F, Liu F, Xiang T, Gong S, Yang Y, Hospedales TM (2017) Actor-critic sequence training for image captioning. arXiv:1706.09601
168. Zhang Y-D et al (2017) Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. Multimed Tools Appl 78.3(2019):3613–3632
169. Zhang Y-D, Muhammad K, Tang C (2018) Twelve-layer deep convolutional neural network with stochastic pooling for tea category classification on GPU platform. Multimed Tools Appl 77(17):22821–22839
170. Zhang M, Yang Y, Zhang H, Ji Y, Shen HT, Chua TS (2018) More is better: precise and detailed image captioning using online positive recall and missing concepts mining. IEEE Trans Image Process 28.1(2018):32–44
171. Zhao W, Wang B, Ye J, Yang M, Zhao Z, Luo R, Qiao Y (2018) A multi-task learning approach for image captioning. In: IJCAI, pp 1205–1211
172. Zhuang J, Tsang IW, Hoi SC (2011) Two-layer multiple kernel learning. In: AISTATS, pp 909–917

Chiranjib Sur is a researcher in deep learning, machine learning, computational and artificial intelligence, soft computing, robotics and its allied branches for a various kinds of application for both engineering problems and social science problems related to intelligent data processing and optimization. He graduated with B.Tech degree in Electronics and Communication Engineering from National Institute of Technology, India, M.Tech degree in Computer Science & Engineering from Indian Institute of Information Technology, India and currently he is pursuing his PhD degree in Computer Science at Computer & Information Science & Engineering Depertment, University of Florida, USA. He is a reviewer of Reviewer of IEEE Transactions on Industrial Informatics, IEEE Transactions on Cybernetics, IEEE Access, International Journal of Engineering Science and Technology, International Journal of Control, International Journal of Automation and Computing, Applied Soft Computing, Journal of Engineering Optimization, IEEE Journal of Biomedical and Health Informatics, Computer Communications, Applied Computing and Informatics, Journal of Traffic and Transportation Engineering, International Journal of Electrical Power and Energy Systems (IJEPES), Journal of Experimental & Theoretical Artificial Intelligence, Journal of Industrial and Production Engineering, The Journal of Engineering, IEEE Transactions on Evolutionary Computation, IET Generation, Transmission & Distribution and other conferences.