AUTOMATIC IMAGE CAPTIONING USING DEEP LEARNING

Report submitted in partial fulfilment of the requirement for degree of

Bachelor of Technology

In

Computer Science & Engineering

By

Parul Diwakar

To

Shaily Malik, Assistant Professor, Dept of CSE

Maharaja Surajmal Institute of Technology

Affiliated to Guru Gobind Singh Indraprastha University

Janakpuri, New Delhi-58

Batch 2018-22

Can you caption this image?

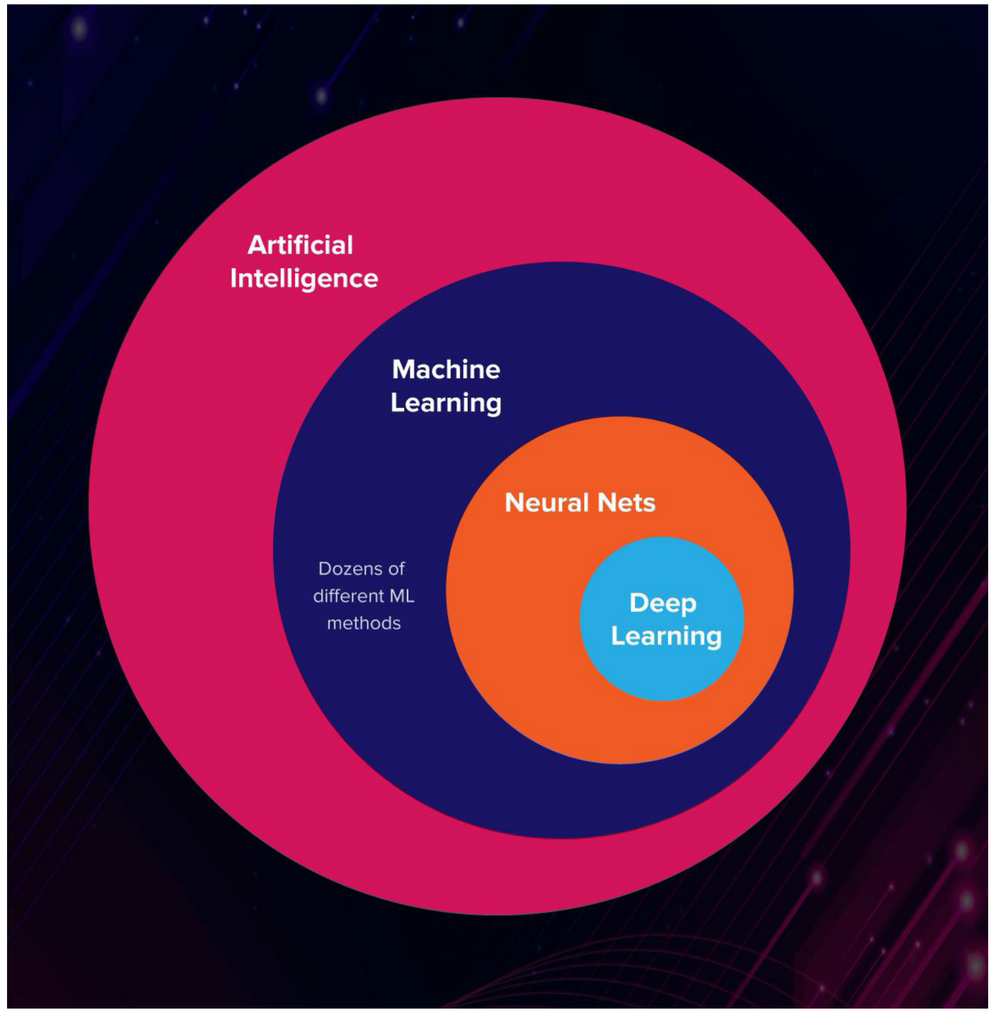
Some of us might say "**A girl in white dress**" or some may say "**A girl is smiling**" and some others may caption it "**A little girl in a white dress is smiling**". All these captions are completely valid for this image.

It is so easy for us, as humans, to just have a glance at a picture and describe it in an appropriate language. Even a child could do this with utmost ease.

But can a computer do this?

The answer is **yes**, with the advent of advanced deep learning models it is possible.

What is Deep Learning ?



In layman's words **Deep learning** is an AI function that mimics the workings of the human brain in processing data for use in detecting objects, recognizing speech, translating languages, and making decisions. **Deep learning** AI is able to **learn** without human supervision, drawing from data that is both unstructured and unlabeled.

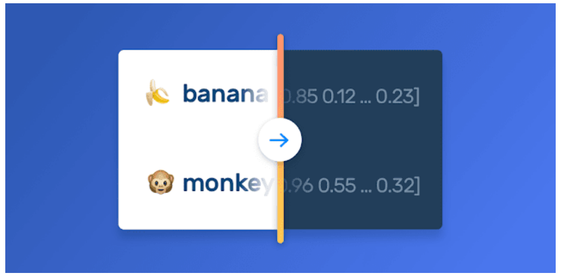
Platform and tools

Platform : Google Colaboratory and Drive

ResNet50 CNN model pretrained with ImageNet Dataset

Keras with Tensorflow
(Framework & Libraries)

Flickr8k Dataset
(Testing & Training Dataset)



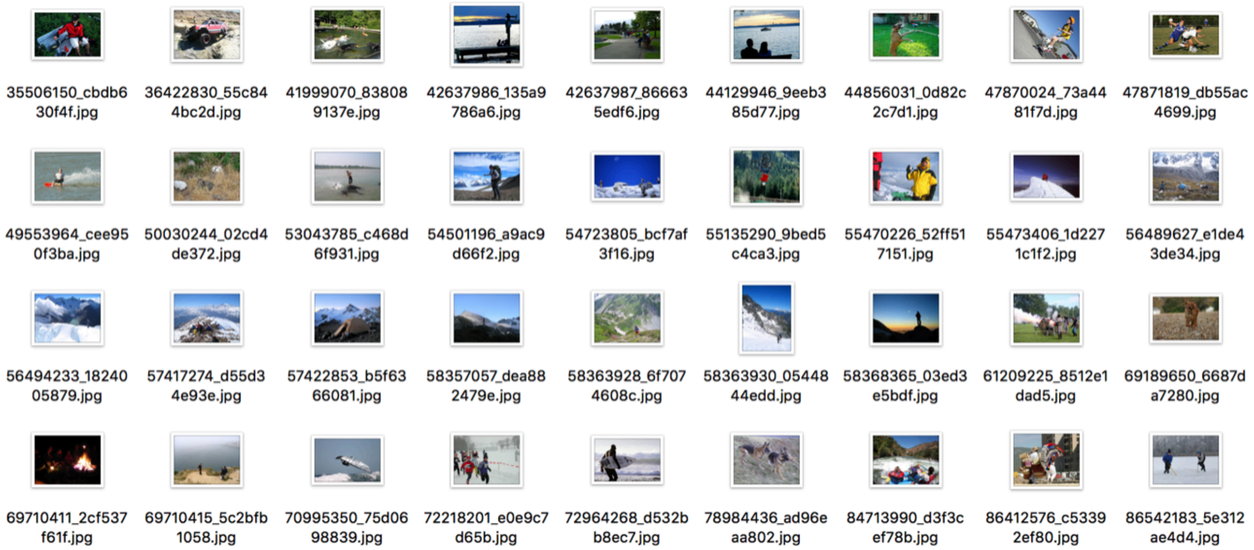
Pretrained Glove Model
(Word Embeddings)

About the Dataset

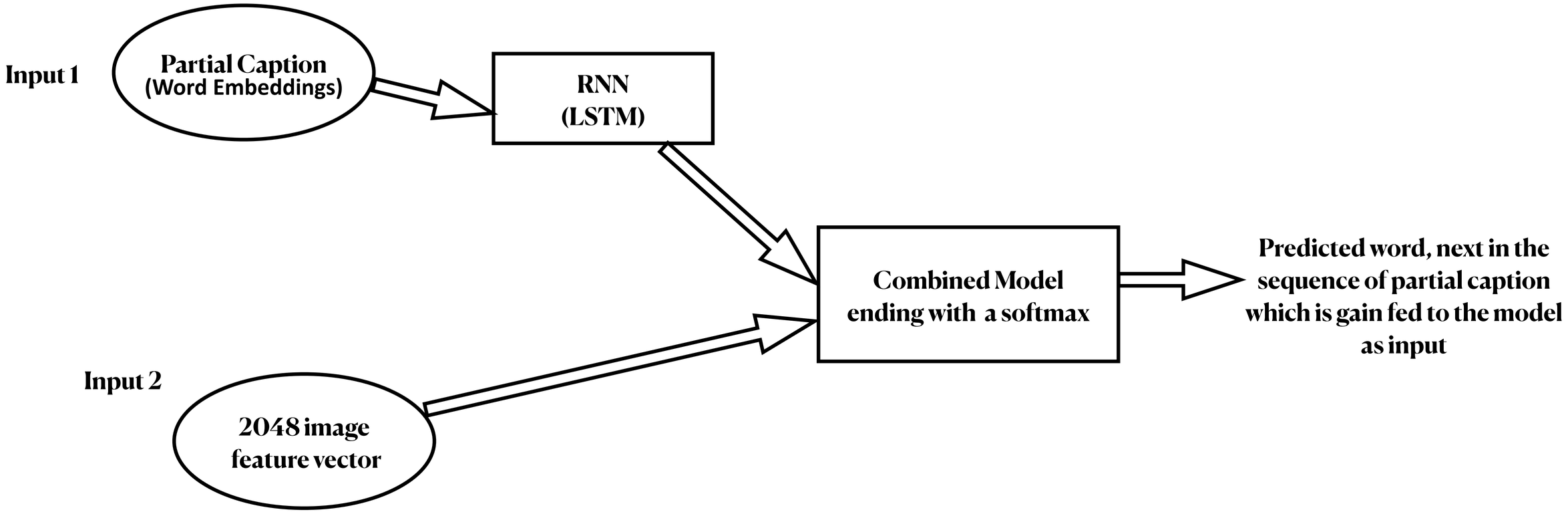
The **Flickr8k** Dataset consists of 8000 images and each image has 5 corresponding captions.The Dataset is divided into **Training Dataset (6000 images)** and **Testing Dataset (2000 images).** This Dataset was chosen because:

- Data is properly labelled. For each image 5 captions are provided.
- The dataset is available for free.
- It is small in size. So, the model can be trained easily on low-end laptops/desktops.

```
['A little tan dog with large ears running through the grass .',
 'A playful dog is running through the grass .',
 'A small dogs ears stick up as it runs in the grass .',
 'The small dog is running across the lawn .',
 'this is a small beige dog running through a grassy field']
```



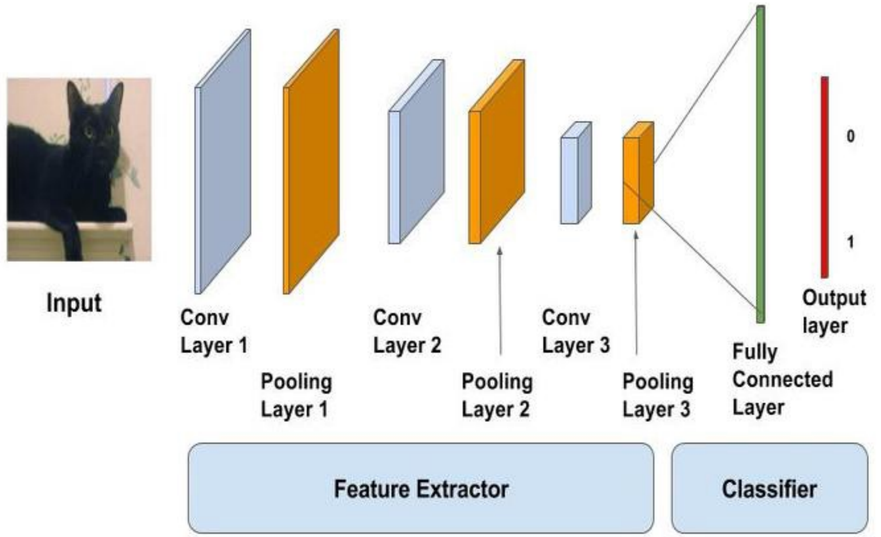
Brief Approach

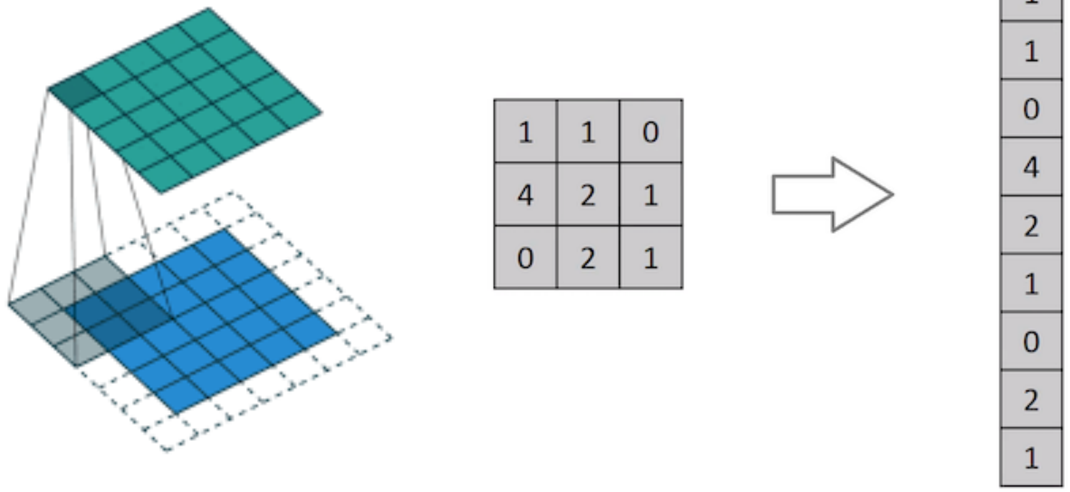


CNN

Convolutional Neural Network (for image processing)

A **Convolutional Neural Network (ConvNet/CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.





Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to **decrease the computational power required to process the data** through dimensionality reduction. Furthermore, it is useful for **extracting dominant features** which are rotational and positional invariant, thus maintaining the process of effectively training of the model.

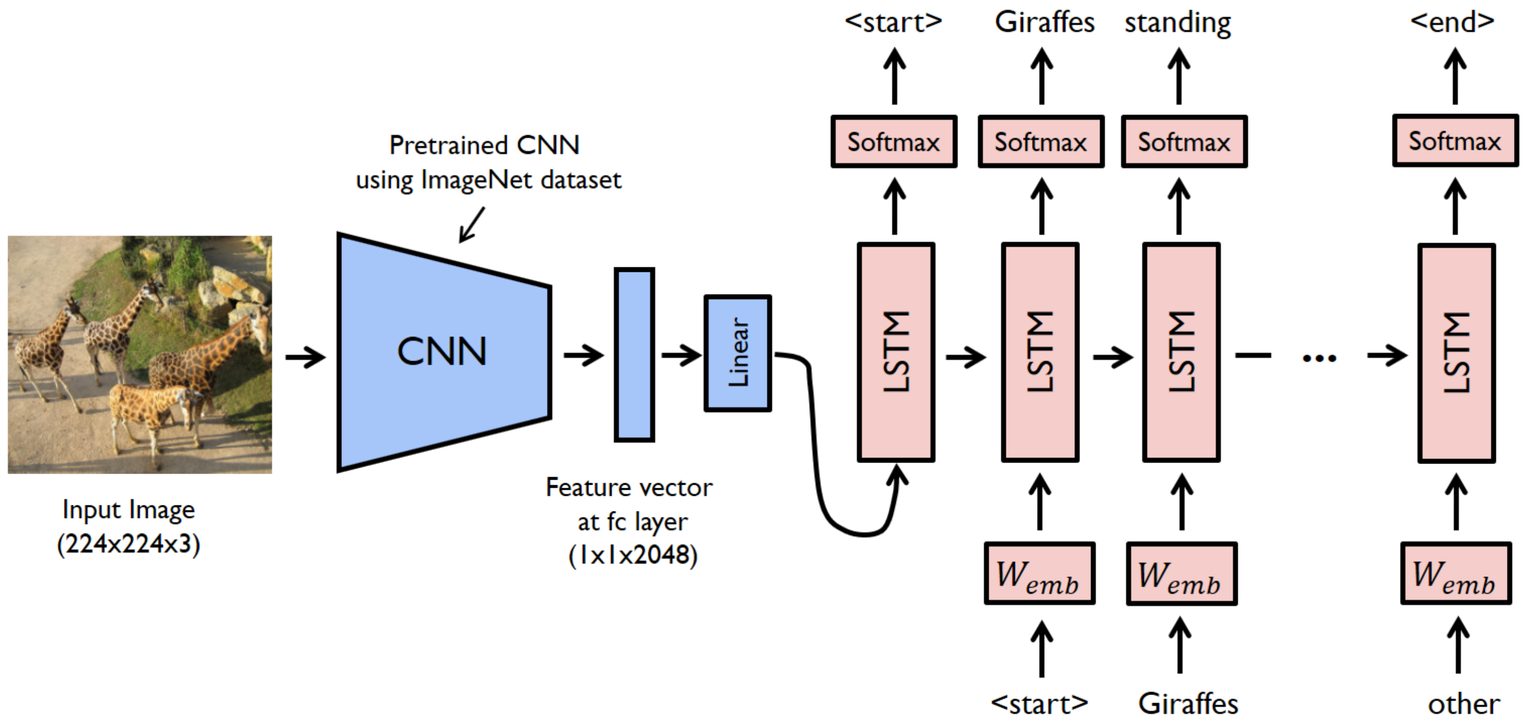
RNN

Recurrent Neural Network (for caption processing)

Recurrent neural networks (RNN) are a class of neural networks that is powerful for modeling sequence data such as time series or natural language.
After producing the output, it is copied and sent back into the recurrent network. For making a decision, it considers the current input and the output that it has learned from the previous input.

LSTM (Long Short Term Memory)

Long Short-Term Memory (LSTM) networks are a modified version of recurrent neural networks, which makes it easier to remember past data in memory. Hence the partial caption is fed to fetch the next probable word.



Prediction Process

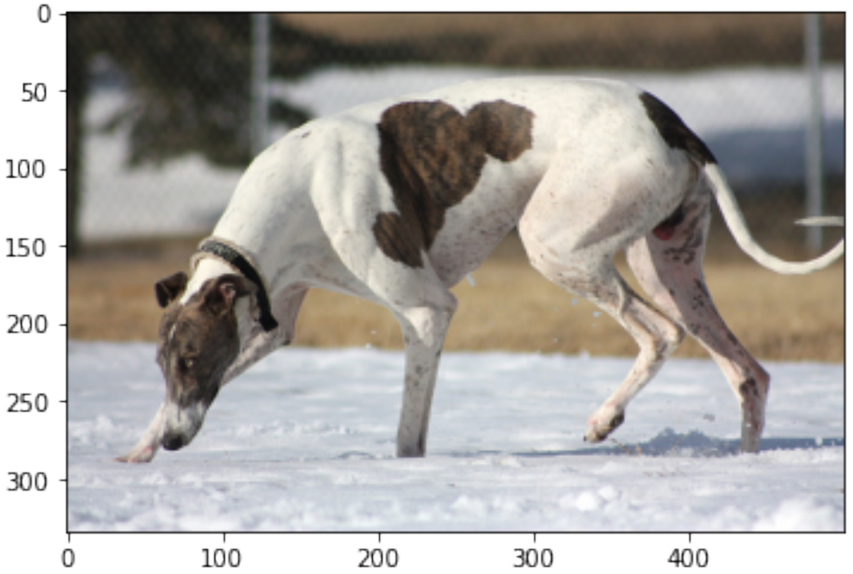
The vocabulary in the example = {and, black, dog, in, runs, snow, the, white}
The caption is generated iteratively, one word at a time as follows:

Iteration 1:
Input: Image vector + “start” (as partial caption)
Probable word: “black”

Iteration 2:
Input: Image vector + “start black”
Probable word: “and”

Iteration 3:
Input: Image vector + “start black and”
Probable word: “white”

Iteration 4:
Input: Image vector + “start black and white”
Probable word: “dog”



Test image
Caption : black and white dog runs in the snow .

Iteration 5:
Input: Image vector + “start black and white dog”
Probable word: “runs”

Iteration 6:
Input: Image vector + “start black and white dog runs”
Probable word: “in”

Iteration 7:
Input: Image vector + “start black and white dog runs in”
Probable word: “the”

Iteration 8:
Input: Image vector + “start black and white dog runs in the”
Probable word: “snow”

Iteration 9:
Input: Image vector + “start black and white dog runs in the snow”
Probable word: “end”

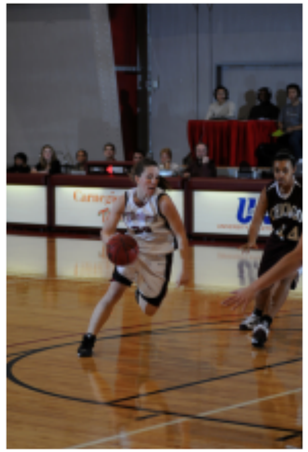
Token ‘start’ which is used as the initial partial caption for any image. We stop when either we encounter an ‘end’ token which means the model thinks that this is the end of the caption or maximum threshold of the number of words generated by the model is reached.

Output

two women dressed in dresses .

the dog jumps over bar .

black and white dog is running through the grass .



man in blue shirt is jumping into the air .

snowboarder is skiing down snowy hill .

two children are skiing down snowy hill .

Thank You