LINGUISTICA ANTVERPIENSIA, 2021 Issue-1
www.hivt.be
ISSN: 0304-2294

Comparison of LSTM and GRU in Video Activity Recognition Using Transfer Learning and Time Distributed Layer in Keras

Dr. S Praveen Kumar[1], P Mansa Devi[2],G Karthika[3], A Yeshwanth[4], M Krishna sampath[5]

[1,2,3,4] Dept. of CSE, GIT, GITAM (Deemed to be University),
[1]spkmtech@gmail.com,[2]mpappu@gitam.edu,[3]kgidijal@gitam.edu,yamanapu@gitam.edu, [5]krishnasampath23@gmail.com

Issue Details
Issue Title: Issue 1
Received: 15 January, 2021
Accepted: 08 February, 2021
Published: 31 March, 2021
Pages: 3637 – 3645

Abstract

With the advancements in Deep Learning, many feats are achievable today that can be considered technologically appealing and also very useful in real life. Our paper aims to recognize human activity in videos using Deep Learning and compare Gated Recurrent Unit (GRU) efficiency with Long Short Term Memory (LSTM) and various transfer learning models to determine an optimal and simplistic action recognition mode. Taking video as input, we convert it into sequences of frames, process each frame through a Convolutional Neural Network (CNN), and connect the entire sequence, using a time distributed layer, to LSTM or a GRU. To achieve this, we train our model on UCF101 and HMDB51, two large video activity datasets. UCF101 has 101 action classes that are widespread across various activities, and HMDB51 has 51 action classes that are more focused on humans' movement.

Activity recognition is beneficial as it can help us refine many parts of society, especially post-pandemic. This project is our attempt at activity recognition and analysis on a couple of methods that can hopefully be helpful.

Keywords

Deep Learning, Activity Recognition, LSTM, GRU, CNN

1. Introduction

Activity recognition in videos is identifying the action that is being performed in a video clip. We have seen many advancements in Deep Learning and Computer Vision in recent times, and many novel methods are available to process image data. However, when it comes to video data, there is still a lot of room for betterment. Video activity recognition has never been straightforward, and that is because videos cannot be directly processed like images. First, the video data should be converted to a sequence of frames, and only by processing the individual frames and their arrangement can we work on video data.

In this paper, we approach the conversion of video input to a sequence of frames using the video generators module provided by keras. The number of frames in a video can be very high. There is a significant possibility that consecutive frames are similar to one another and do not represent the movement that occurs in the video. For this reason, we skip consecutive frames and only select the frames that are apart from one another as it increases the possibility of capturing the frame sequence that shows more movement. Next, we should process the series of frames through a

classifier model. In deep learning, convolutional neural networks (CNN) generally provide excellent results when classifying image data. A traditional convolutional neural network cannot take a series of images, and it can only process a single frame at a time. A CNN can only accept a two-dimensional entity like an image as its input. However, in this case, the input has an additional dimension: the number of frames in the sequence. So we use the time distributed layer, which considers a temporal dimension. The individual images are processed by the CNN according to the temporal dimension, and all the processed images connect to a recurrent neural network (RNN) in the respective sequence. We use either an LSTM layer and separately a GRU layer here and compare the results obtained. We operate CNN using transfer learning to repurpose pre-trained models to suit our use case. Transfer learning improves the accuracy of the model quite drastically and proves to be essential.

The proposed approach is simple to understand and implement. While still requiring some suitable hardware, considering modern computing standards, one does not need a laboratory with cutting-edge graphics cards to execute this method. Many solutions to video activity recognition take flow and RGB images of the datasets as inputs which require lots of pre-processing and larger disk spaces. In contrast, the proposed model takes videos as input directly and converts them into sequences of frames intrinsically while not compromising the model's final accuracy.

2. Literature Review

In 'UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild'[1], Khurram Soomro orchestrated a state-of-the-art action recognition dataset called UCF101. This dataset contains 101 action classes with more than 13,000 videos that, combined, exceed 27 hours. The 101 action classes belong to various categories like human-human interactions, sports, playing musical instruments, body motion activities, and human-object interaction. Nearly 7 GB in size, training a model successfully on the UCF101 dataset can give us solid activity recognition results. Another state-of-the-art dataset is the HMDB51 dataset introduced by H. Kuehne in the paper 'HMDB: A Large Video Database for Human Motion Recognition'[2]. HMDB51 is a dataset on human motion only and spans 51 classes, and has over 7000 videos with a size of nearly 2GB. The contrast between the two datasets is that UCF101 is a significantly larger dataset and its classes are broader in range and category in comparison.

In 'Two-Stream RNN/CNN for Action Recognition in 3D Videos'[3], Rui Zhao pointed out how GRU can be faster, more efficient, and provide better accuracy compared to LSTM. LSTM and GRU are further tested and compared in this paper. 'Late Temporal Modeling in 3D CNN Architectures with BERT for Action Recognition'[4] introduced a novel architecture called BERT using temporal modeling of 3D CNNs. This model holds the flagship results for the highest accuracy on UCF101 and HMDB51 datasets with 98.69% and 85.10% accuracy respectively. However, [4] used additional training data and pre-processed image datasets to achieve those results instead of using videos directly as in our approach. In 'Action Recognition in Video Sequences using Deep Bi-directional LSTM with CNN

Features,' [14] Amin Ullah introduced a bi-directional LSTM architecture which garnered a high accuracy of 91.21% and 87.64% on the two datasets. In this paper, we compare our results with these models to realize our models' efficiency.

In the paper 'Receptive Fields of Single Neurones in the Cat's Striate Cortex,'[10] David Hubel and Torsten Wiesel introduced simple and complex cells for visual pattern recognition. Simple cells can recognize boundaries like edges of a particular orientation of the given input. The complex cells do this at a more holistic scale all over the provided input. This work laid the foundation for artificial neural networks. In 'Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,' [11] Kunihiko Fukushima designed an artificial neural network architecture where complex patterns are captured using lower-level complex cells and simple cells that detect small patterns. While this was a conceptual breakthrough, in the paper 'Gradient-Based Learning Applied to Document Recognition,'[12] Yann LeCun from NYU's Courant Institute made the first modern use of artificial neural networks. Yann LeCun and C. Cortes introduced the 'MNIST handwritten digit database,'[13] a dataset having 70,000 images of greyscale codes of handwritten digits (0 to 9). Based on Fukushima's artificial neural network accumulating from simple cells and smaller complex cells for processing complex patterns, LeCun devised a convolutional neural network and trained it on the MNIST dataset, which was the gold standard for image and computer vision applications for a long time.

In the more recent 'MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,'[5] Andrew G. Howard proposed a depthwise separable architecture for vision applications called MobileNets. The MobileNet model allowed in building light weight deep neural networks and provided good accuracy for image classification problems. In the paper, 'Deep Residual Learning for Image Recognition,'[6] Kaiming He proposed a residual learning framework architecture called ResNet to train substantially deep neural networks effectively. ResNets have up to 152 layers and perform very well on large image classification datasets like ImageNet or COCO. Kaiming He's work in 'Identity Mappings in Deep Residual Networks'[7] helped build improved ResNet versions that enhance performance.

In the paper 'Long Short-term Memory,'[8] back in 1997, Sepp Hochreiter presented a way to store information for extended sequences of time and called it long short-term memory (LSTM). LSTM has multiplicative gates that learn to keep data for a long time by highlighting only particular essential parts. LSTM structure is not affected by the reducing gradient descent problem present in vanilla recurrent neural networks and proved to be a go-to technique for deep learning problems, including sequences like text classification, speech recognition, and our use case, video activity recognition. In the more recent paper, 'On the Properties of Neural Machine Translation: Encoder-Decoder Approaches,'[9] Kyunghyun Cho introduced a gated recursive convolutional network with a focus on producing quick and efficient classification for short sequences. This method is known to us as Gated Recurrent Units(GRU), which contains a simpler two gate architecture and provides results faster than LSTM for short bursts or sequences.

Most methods for video action recognition use LSTMs, but they are slower when compared to GRU. There is a great scope for using GRUs, and in this paper, we exploit the faster nature of GRUs and compare how they perform on large data with LSTM.

3. Methodology

3.1 Architecture

We aim to provide a simple model to highlight the differences in performance between LSTM and GRU and provide efficient activity recognition. The first part of the model is converting video input to a sequence of frames. We do this with the help of the video generators module in keras. We can use the datasets directly without the need to use datasets having pre-processed frames and flow images. Such datasets provide a base for good results, and many state-of-the-art models like [4] use them, but those datasets take up a much larger space, and training them would require more hardware power. Instead, by using the videos from the datasets directly, we can focus more on how LSTM and GRU perform on them. Once we generate the sequence, we process each image from the series through a CNN. We do this by using the time-distributed layer in keras. The processed outputs then connect to LSTM/GRU. While this is a simple architecture, we can optimize the performance and get high accuracies on our datasets.

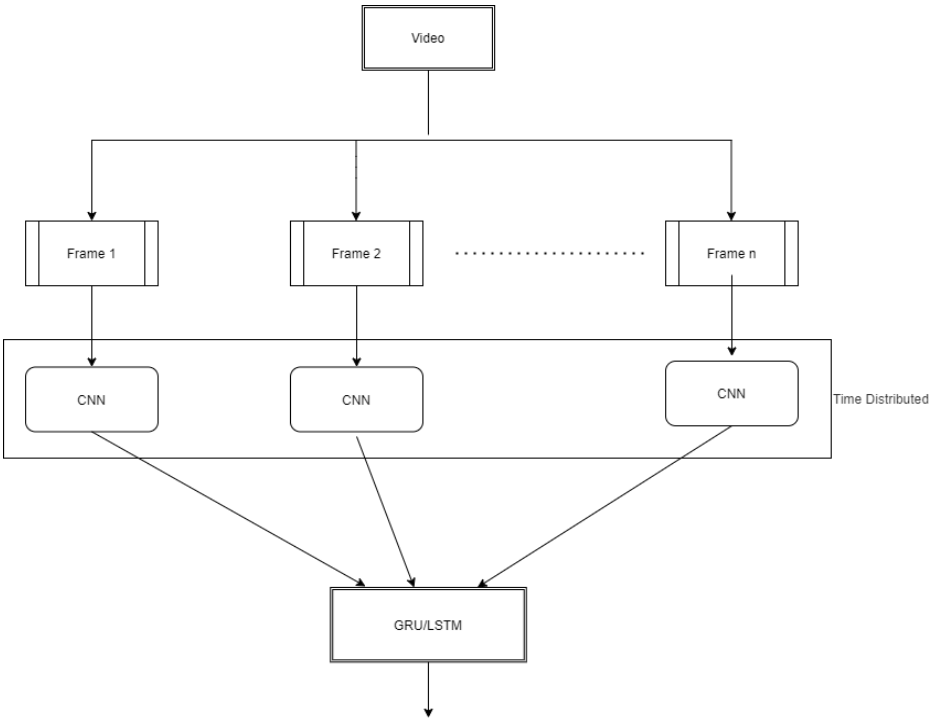

Figure 1: Basic architecture of the model

3.2 Transfer Learning for CNN

When creating the CNN, we have two options: make one from scratch or use an existing model. The latter is called transfer learning. We used both methods to find out which is the better-suited technique for this case. We used a small part of the

datasets, to be precise, four classes out of the 101 classes from the UCF101 dataset (classes archery, cricket shot, trampoline jumping, and walking a dog) and three classes out of the 51 classes from the HMDB51 dataset (classes run, jump and walk). The CNN model created from scratch gave an 80% accuracy for the four classes from UCF101 and 63% for the three classes from HMDB51. The model struggled to differentiate between the activities of walking and running. To improve this, we decided to use transfer learning instead of creating a CNN from scratch. [5] introduced a famous model called Mobilenet, which is available openly in TensorFlow Keras. Using transfer learning by changing the CNN to Mobilenet's CNN model boosted the accuracy to 96% for the four classes from UCF101 and 80% for the three classes from HMDB51. It also provided significantly better results in distinguishing between walking and running. [6] introduced ResNet, another revolutionary model for image classification. Various versions of ResNet are available in Tensorflow Keras. We used a version called ResNet152V2, a more sophisticated version of ResNet with 152 layers introduced by the same team in [7]. Using this model provided similar results for the smaller datasets but outperformed Mobilenet when trained on the entire datasets. This is further discussed in detail in the results section.

3.3 GRU vs. LSTM

GRU and LSTM are two widely used versions of Recurrent Neural Networks in Deep Learning. LSTM was first proposed by Hochreiter and Schmidhuber in [8]. LSTM has three gates: input gate, output gate, and forget gate. The forget gate is the heart of the concept of LSTM as it decides what parts of the sequence can be forgotten so that only useful information is retained. GRU, a more recent concept, was introduced in [9]. GRU has a simpler architecture with only two gates: reset and update gates. Virtually, GRU is a simplified LSTM that combines the input and forget gates. In GRU, either the sequence is updated with new information or is reset. The general consensus is that due to its lesser parameters, GRU is quicker, but when dealing with large data, LSTM is more efficient. For highly intense tasks like training a model for activity recognition, LSTM is heavily preferred. However, in our findings, it is not as simple as that.

4. Results

The first comparison is between Mobilenet and ResNet. Table 1 shows the results that both models fetched (with LSTM) when trained on the entire UCF101 dataset. There is a massive difference between them in terms of accuracy and loss. The Mobilenet model gave an accuracy of 72% with a training loss of 0.76. While the ResNet scored an accuracy and loss of 96.06% and 0.0996, respectively. The Mobilenet model's best epoch is 192, while ResNet's is 110. The ResNet model trained significantly quicker than MobileNet. With these results, it is safe to assert the ResNet model is the better one for processing the frames, and we use this in the comparison between LSTM and GRU.

Table 1: Comparison of Resnet and Mobilenet with LSTM on UCF101 dataset

Model	Best Epoch	Accuracy (in %)	Loss
Mobilenet	192	72.00	0.7616
Resnet	110	96.06	0.0996

Table 2 and Figure 2 disclose the results for training ResNet+LSTM/GRU on the entire datasets of UCF101 and HMDB51. For the UCF101 dataset, the accuracy and loss are pretty close for both LSTM and GRU. GRU has accuracy and loss of 97.03 and 0.0883, respectively, while LSTM fetched 96.06 and a loss of 0.0996. While they are almost identical, it is essential to note that GRU's best epoch is 102, and LSTM is 110. GRU is quicker while providing slightly better results when training on nearly 7GB worth of data.

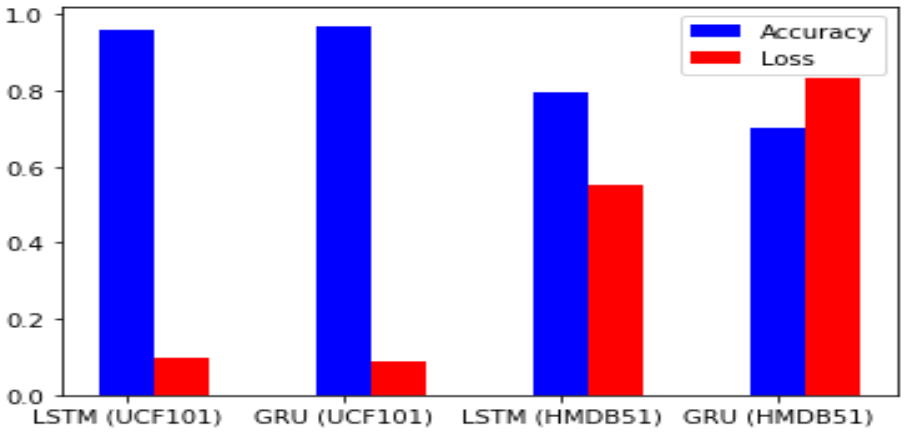

Figure 2: Graph plotting accuracy and loss for LSTM/GRU+ResNet model

Table 2: Comparison of LSTM and GRU with Resnet on UCF101 and HMDB51 datasets

Model	Best Epoch	Accuracy (in %)	Loss	Validation Accuracy (in%)	Validation Loss
LSTM (UCF101)	110	96.06	0.0996	93.93	0.3750
GRU (UCF101)	102	97.03	0.0883	95.09	0.3364
LSTM (HMDB51)	87	79.65	0.5540	58.24	2.36
GRU (HMDB51)	73	70.03	0.8350	54.02	2.25

However, it is a different scenario for HMDB51. LSTM gave an accuracy of nearly 80% with a 0.554 training loss. In contrast, GRU scored an accuracy of only 70% and a loss of 0.835. The gap is narrow if you consider validation accuracy and loss, but

LSTM still performs better. GRU proved to be quicker, with its best epoch being 73, while LSTM's best epoch is 87. Though quicker, GRU's results are not as good considering the superior performance of LSTM.

Figure 3 shows the accuracy of each of the five kinds of activities in the USF101 dataset: body motion activities, human-human interactions, sports, playing musical instruments, and human-object interaction. All the different breadth of activities achieved high accuracy. The lowest accuracy achieved is 94.91% for sports, and the highest achieved is 99.37% for playing musical instruments.

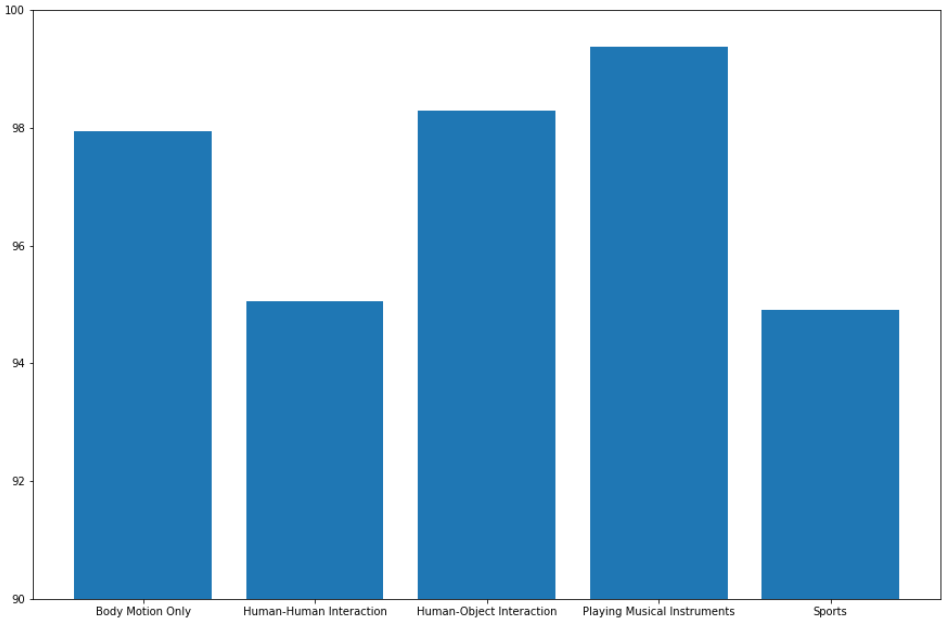

Figure 3: Graph comparing the accuracy achieved on the different groups of activities for LSTM+ResNet model on UCF101 dataset

In addition to that, the accuracies achieved are not far from [4] 's flagship action recognition model, which has an accuracy of 98.69 for the UCF101 dataset and 85.10 for the HMDB51 dataset. Our model fares better than [14] 's accuracy of 91.21% on the UCF101 dataset, but it is interesting to note that [14] 's bi-directional LSTM structure fared a much better accuracy of 87.64% on the HMDB51. LSTM performed better than GRU on the HMDB51 dataset, and GRU was able to perform better on the more extensive and broader UCF101 dataset.

5. Conclusion

When comparing LSTM and GRU in activity recognition, the results are nothing short of surprising. While the preconceived consensus is that GRU is preferable for smaller datasets and LSTM for the larger ones, this paper's findings contrast that assumption. GRU performed slightly better and faster than LSTM when training on UCF101, a large dataset that is nearly 7GB in size. However, LSTM is significantly superior when dealing with HMDB51, a 2GB dataset. Another essential factor we should consider is the differences between the two datasets. UCF101 has a wide range of classes focusing on various activities, while HMDB51 focuses more on human

movement. It is not unusual that GRU didn't perform better on the HMDB51 dataset, as it has a simpler architecture, and LSTM has the internal structure to deal with the complexity of the similarity of the actions.

6. Future Scope

GRU's performance on the UCF101 dataset proves that it can handle large amounts of data. However, that depends on the data itself. HMDB51, though a smaller dataset, has classes like run and walk, which, when stripped down to frames, look similar. There is massive scope for improvement for GRU in distinguishing such complex data. A customized and capable GRU model can potentially provide equivalent or better results than LSTMs while being faster.

References

[1] Khurram Soomro, Amir Roshan Zamir and Mubarak Shah, UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild, CRCV-TR-12-01, November, 2012.

[2] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A Large Video Database for Human Motion Recognition. ICCV, 2011

[3] Zhao, Rui & Ali, Haider & van der Smagt, Patrick. (2017). Two-stream RNN/CNN for action recognition in 3D videos. 4260-4267. 10.1109/IROS.2017.8206288.

[4] Kalfaoglu, Esat & Kalkan, Sinan & Alatan, A.. (2020). Late Temporal Modeling in 3D CNN Architectures with BERT for Action Recognition. 10.1007/978-3-030-68238-5_48.

[5] Howard, Andrew & Zhu, Menglong & Chen, Bo & Kalenichenko, Dmitry & Wang, Weijun & Weyand, Tobias & Andreetto, Marco & Adam, Hartwig. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.

[6] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Deep Residual Learning for Image Recognition. 770-778. 10.1109/CVPR.2016.90.

[7] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Identity Mappings in Deep Residual Networks. 9908. 630-645. 10.1007/978-3-319-46493-0_38.

[8] Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation. 9. 1735-80. 10.1162/neco.1997.9.8.1735.

[9] Cho, Kyunghyun & van Merriënboer, Bart & Bahdanau, Dzmitry & Bengio, Y.. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. 10.3115/v1/W14-4012.

[10] Hubel, D.H. & Wiesel, T.N.. (1959). Receptive Fields of Single Neurons in the Cat's Striate Cortex. JP. 148. 574-591.

[11] Fukushima, Kunihiko. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological cybernetics. 36. 193-202. 10.1007/BF00344251.

[12] Lecun, Yann & Bottou, Leon & Bengio, Y. & Haffner, Patrick. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE. 86. 2278 - 2324. 10.1109/5.726791.

[13] LeCun, Y. & Cortes, C. (2010), 'MNIST handwritten digit database', .

[14] Ullah, Amin & Ahmad, Jamil & Muhammad, Khan & Sajjad, Muhammad & Baik, Sung. (2017). Action Recognition in Video Sequences using Deep Bi-directional LSTM with CNN Features. IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2778011.

[15] Praveen Kumar S , Kesava Jayendra Varma V, Subramanya V, Venkata Sai Harish A. A Multiple Face Recognition System With Dlib's Resnet Network Using Deep Metric Learning . JCR. 2020; 7(6): 856-859. doi:10.31838/jcr.07.06.147

[16] P.Raja Mani, Dr.S.Praveen Kumar, P.Surya Chandra,G.Santoshi Kumari. A Robust Copy Move Tamper Detection and Localization Method for Image Forensics . JCR. 2020; 7(17): 1522-1530. doi:10.31838/jcr.07.17.192

[17] Dr. S.Praveen Kumar , Dr. K.Naveen Kumar, Dr. Y.Srinivas, Dr. G.V.S Rajkumar 2020. Defect Detection On Manufacturing Product Images by Applying Weekly Detector Algorithm using Deep Learning Technology. International Journal of Advanced Science and Technology. 29, 7 (May 2020), 186 - 194.

[18] Praveen Kumar S, Srinivas Y, Bhargav K, "An n-gram analysis approach for sharing authentication of data using model based techniques", Test Engineering and Management, 2020.

[19] Praveen Kumar S, Sahithi Choudary A, "An Innovative ModelBased Approach for CreditCard Fraud Detection Using Predictive Analysis and Logical Regression.", International Journal of Innovative Technology and Exploring Engineering (IJITEE), Scopus, 2019, 8 , 1683-1688

[20] Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "Latent Semantic Indexing based Approach for Identification of Frequent Itemset.",Jour of Adv Research in Dynamical & Control Systems, Scopus, 2018, Vol. 10, 686-690.

[21] Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "A mechanism for identifying the guilt agent in a network using vector quantization and skew Gaussian distribution." International Journal of Engineering & Technology, Scopus, 2018, 7, 149-151

[22] Praveen Kumar S, Srinivas Y, Vamsi Krishna M, "A Data Leakage Identification System Based on Truncated Skew Symmetric Gaussian Mixture Model.", International Journal of Recent Technology and Engineering (IJRTE), Scopus, 2018, 7, 111-113

[23] S. Praveen Kumar, Gandham Bharathi, Chintala Neeraj, Killamsetty Akilesh, Surendranath, "An Efficient Key Transmission Model in Attribute Based Encryption Scheme in Cloud Computing", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.6, Issue 1, pp.1134-1137, March 2018, Available at :http://www.ijcrt.org/papers/IJCRT1872344.pdf

[24] Kumar, S. (2018). IoT-based Temperature and Humidity Monitoring System using Raspberry pi. International Journal for Research in Applied Science and Engineering Technology. 6. 288-291. 10.22214/ijraset.2018.4052.

[25] S Kumar, Y Srinivas, D Suba Rao, Ashish Kumar A Novel Model for Data Leakage Detection and Prevention in Distributed Environment International Journal of Engineering and Technical Research Posted: 2016

[26] S. Praveen Kumar, Y. Srinivas, M. Ranjan Senapat and A. Kumar, "An enhanced model for preventing guilt agents and providing data security in distributed environment," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), Paralakhemundi, India, 2016, pp. 337-339, doi: 10.1109/SCOPES.2016.7955847.

[27] A Complete Introspection on Big Data and Apache Spark – S.Praveen Kumar, Dr. Y .Srinivas, Dr. D.Subbarao ,Ashish Kumar—http://www.ijsdr.org/papers/IJSDR 1604006.pdf

[28] D. R. Giri, S. P. Kumar, L. Prasannakumar and R. N. V. V. Murthy, "Object oriented approach to SQL injection preventer," 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), Coimbatore, India, 2012, pp. 1-7, doi: 10.1109/ICCCNT.2012.6395979.

[29] L. P. Kumar, S. P. Kumar, D. R. Giri and V. Jayavani, "Object oriented approach to prefix based fast mining of closed sequential patterns," 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), Coimbatore, India, 2012, pp. 1-6, doi: 10.1109/ICCCNT.2012.6395980.

[30] Kumar S P , K. Anusha, R.Venkata Ramana, "A Novel Approach to Enhance Robustness in Steganography Using Multiple Watermark Embedding Algorithm". International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307 (Online), Volume-1, Issue-1, March 2011

[31] S. P. Kumar, K. N. Kumar, S. Sreenadh, B. Aravind, K. H. Kumar, Novel Advent for Add-On Security by Magic Square Intrication, Global Journal of Computer Science and Technology,Vol. 11, No. 21, December 2011.

[32] Kumar, N. Suresh & Ramakotireddy, D & Praveen, S & Anitha, J & Brahmaji, G. (2010). Method To Minimize The Clock Skew In Multiple Pipe Line By Uniform Clock Distribution Using Parallel Port. 1. 12-16.