A Novel DC-GCN with Attention Mechanism for Accurate Near-Duplicate Video Data Cleaning

[1]Jayalakshmi D, [2]Hemavathi R, [3]Murali L, [4]Baskar Duraisamy, [5]Banda SNV Ramana Murthy and [6]Sunita
[1,2]Department of Computer Science and Engineering, Saveetha School of Engineering,
Saveetha Institute of Medical and Technical Science, Saveetha University, Thandalam, Chennai, Tamil Nadu, India.
[3]Department of Electronics and Communication Engineering, P. A. College of Engineering and Technology,
Pollachi, Tamil Nadu, India.
[4]Department of Electronics and Communication Engineering, Karpagam Institute of Technology,
Coimbatore, Tamil Nadu, India.
[5]Department of Computer Science and Engineering -AIML, Aditya University, Surampalem, Andhra Pradesh, India.
[6]Department of Information Science and Engineering, Dayanand Sagar Academy of Technology and Management
Udaypur, Bangalore, Karnataka, India.
[1]jayalakshminandakumar2@gmail.com, [2]saihema01@gmail.com, [3]murlak37@gmail.com, [4]baskardr@gmail.com,
[5]ramanamurthy.banda@gmail.com, [6]sunitajeevangi@gmail.com

Correspondence should be addressed to Jayalakshmi D : jayalakshminandakumar2@gmail.com

Article Info
Journal of Machine and Computing (http://anapub.co.ke/journals/jmc/jmc.html)
Doi : https://doi.org/10.53759/7669/jmc202404093
Received 15 March 2024; Revised from 20 May 2024; Accepted 30 July 2024.
Available online 05 October 2024.

Abstract – There has been a steady emergence of nearly identical recordings in the last several decades, thanks to the exponential development of video data. The use of regular videos has been impacted by data quality difficulties produced by near-duplicate movies, which are becoming increasingly noticeable. While there has been progress in the field of near-duplicate video detection, there is still no automated merging method for video data characterised by high-dimensional features. As a result, it is challenging to automatically clean near-duplicate videos in advance video dataset data quality. Research on removing near-duplicate video data is still in its early stages. The precision of near-duplicate video data cleaning is severely compromised by the delicate issues of video data organization besides initial clustering centres in the current research, which arise when the previous distribution is unknown. In tackle these problems, we offer a new kind of Graph Convolutional Neural Network (GCN) that uses dense influences and a categorization attention mechanism. Deeply connected graph convolutional networks (DC-GCNs) learn about faraway nodes by making GCNs deeper. By using dense connections, the DC-GCN is able to multiplex the small-scale features of shallow layers and generate features at diverse scales. Finally, an attention mechanism is incorporated to aid in feature combination and importance determination. Sparrow Search Optimisation Algorithm (SSA) is used to pick the parameters of the given model in the most optimal way. In the end, experiments are carried out using a coal mining video dataset and a widely known dataset called CC_WEB_VIDEO. The simulation findings show that the suggested strategy performs better than certain previous studies.

Keywords – Video Data, Graph Convolutional Neural Network, Densely Connected Graph Convolutional Network, Sparrow Search Optimization Algorithm, Duplicate Video Data Cleaning.

I. INTRODUCTION

A tremendous amount of video data has been added to the Internet due to the expansion of information technology. Demand for efficient and effective video retrieval systems is on the rise [1]. With its combination of low memory cost and fast retrieval speed, hashing technology is highly promising for content-based retrieval in real-time [2]. Aiming at efficient video retrieval, this work investigates unsupervised video hashing. Unsupervised video hashing removes the need for human annotations in comparison to the supervised case [3]. Because most methods rely on estimating fundamental video similarities for content-based retrieval, it is difficult [4]. The objective of video hashing is to quickly retrieve videos by encoding them into a collection of binary codes and then utilising the Hamming distance to find them. It follows that the hash codes should represent the video's educational substance [5]. The initial challenge is to record as much video as you can. This is also where the focus of current approaches lies. Prior methods of video hashing attempted

to tackle this problem by painstakingly extracting detailed visual data and constructing a number of complex mathematical models [6].

Actually, as video data becomes increasingly large, numerous identical films (sometimes called near-duplicate videos, or NDVs) keep cropping up following editing, with a revised original being reissued and other processes performed on the videos [7]. According to [8], videos are considered near-duplicates if they are nearly identical or very similar in appearance but differ in minor details. It is common practice to generate videos from the original, which has two problems: first, it lowers the quality of video datasets' data [9] and second, it violates the copyright of the video's creator.

When considering video data quality, it is important to focus on the entire dataset quality and ensure that information systems meet standards for data consistency, correctness, completeness, and minimization [10]. Data reduction and consistency in video datasets will suffer when near-duplicate films proliferate. These movies are almost identical; they have extensive coverage and come in a variety of forms; and they may be considered dirty data [11]. Concretely, it is feasible to produce almost identical movies at any point in the video acquisition, integration, or processing processes. In the video gathering stage, for example, footage can be filmed from various perspectives scene. Then, in the video integration stage, footage from various data sources can be combined to create near-duplicate videos. Lastly, in the video processing stage, operations such as video copying and editing can result in a large sum of videos [12]. Research on detecting near-duplicate movies can lead us to previously unseen near-duplicate recordings in video collections. Feature extraction, index are the primary steps in the implementation process, and there are now many different types of approaches suggested in the literature [13]. For near-duplicate video identification in both of these approaches, feature extraction is an essential step.

There are two main tactics to near-duplicate video detection that are based on video feature representation: high-level feature-based practice and hand-crafted feature-based practice [14]. But near-duplicate video recognition methods can only find films that are almost identical in a video dataset [15] that doesn't have a way to automatically merge and sort high-dimensional characteristics that describe video data. Thus, it is extremely difficult for them to automate the process of removing unnecessary near-duplicate videos in order to lessen instances of video copyright infringement and associated problems that arise from manual video editing, copying, and other related tasks [16].

A novel compositional approach is developed in this study to construct an individual graph for every sentence. Present a novel DC-GCN network architecture and expand dense connection to GCN network. The network uses an attention apparatus to routinely rank the word nodes according to their significance. Even with the coal mining video collection's complicated context scenarios, the approach given in this research was able to succeed on the very challenging CC_WEB_VIDEO dataset. The following is the outline of the article's subsequent sections. A concise summary of relevant literature is provided in Section 2. In Section 3, a method for automatically cleaning videos of near-duplicates is shown. Section 4 presents the experimental data that illustrate the method's efficacy. Lastly, Section 5 delivers a summary of the paper.

II. RELATED WORKS

Jo et al., [17] shown that the present challenges of video-level methods can be better understood by appropriately removing extraneous frames. As an additional measure, we suggest a VVS network, which stands for Video-to-Video Suppression. VVS is a full-stack framework with two primary parts: a simple distractor removal stage for picking out which frames to crop out and a suppression step for figuring out how much to crop out of the rest. An uncut video with varied material and irrelevant details is what this structure is trying to portray. Our solution not only has state-of-the-art video-level competences but also a rapid inference time and retrieval approaches, as demonstrated by comprehensive trials, proving its usefulness.

For the purpose of pragmatically retrieving relevant anomalous films using cross-modalities, such as linguistic descriptions and synchronous audios, Wu et al., [18] have proposed a new task named Video Anomaly Retrieval (VAR). Unlike traditional video retrieval methods, which presume that movies are short and well-trimmed in terms of time, VAR may retrieve longer, untrimmed videos that may only partially relate to the query. Our Anomaly-Led Alignment Network (ALAN) model for VAR and two large-scale VAR benchmarks help us reach this goal. Our proposal in ALAN is to use anomaly-led sampling to zero down on important parts of lengthy, uncut films. Next, we provide a pretext task that is both efficient and effective in order to improve the semantic linkages between the fine-grained representations of video and text. Finally, to further match contents, we use two complimentary alignments. The experimental findings on two benchmarks show the benefits of our customised approach and the difficulties of the VAR job.

As a video retrieval, Mounika et al. [19] presented an LBP-TOP, a form of dynamic texture. LBP-TOP may describe both look and motion simultaneously. Light, rotation, and local translation have no effect on the LBP-TOP characteristics. These substantial advantages lend credence to the idea that the suggested approach might benefit from LBP-TOP. The query video clip, which includes 10 randomly chosen sample frames, is utilised by the suggested approach. processing, and a matching and retrieval stage make up the three phases of the suggested CBVR. During offline processing, we first utilise the Pearson Correlation Coefficient (PCC) and Colour Moments (CM) to extract keyframes from the database movies. Then, we use the LBP-TOP feature of these keyframes to represent the complete database video. We extract LBP-TOP features from the query video during online processing. These features are then passed on to the matching and retrieval step, where we find the films with the shortest distance by calculating the

Euclidean distance between the LBP-TOP features of the database keyframes and the frames in the query video. In demonstrate the efficacy of the suggested technique, it has been evaluated using 108 movies from a publicly available standard traffic dataset and contrasted quantitatively and qualitatively with other cutting-edge methodologies. Precision, recall, accuracy, specificity, besides the E-measure were the assessment metrics used in the quantitative performance evaluation. Incorporating dynamic textures is key to the effectiveness of the suggested technique, which outperformed previous state-of-the-art approaches in both qualitative besides quantitative performance evaluations. Use cases for the proposed technique include traffic monitoring and other real-time applications. Through feature matching among query scenes and database videos, the suggested CBVR system may be utilised for traffic monitoring. The algorithm will present the outcome as low traffic time if the query matches a video in the database that has low traffic. Similarly, the system will be able to identify periods of medium or heavy traffic.

In their generative diffusion-based system MomentDiff, Li et al. [20] have replicated the steps involved in a natural human retrieval process, from exploratory browsing to incremental localization. In particular, we utilise text-video similarity as a guide to learn how to denoise the random noise back to the original span after diffusing it to random noise from the genuine span.Finding segments from a random initialization is now within the model's capabilities, thanks to its capacity to learn a random places to real moments. Upon training, MomentDiff has the ability to produce an accurate temporal boundary by iteratively refining initial predictions derived from random temporal segments. When contrasted with discriminative efforts (such as those based on learnable proposals or queries), MomentDiff with randomised initialised spans might be able to withstand datasets' temporal location biases. Two "anti-biases" datasets, Mom, are proposed to assess the impact of temporal location biases. These datasets involve changes in the distribution of locations. Experiments on three public benchmarks show that our efficient system routinely beats state-of-the-art approaches, and on the suggested anti-bias datasets, it shows higher generalisation and resilience. Public access will be granted to the code, model, and datasets used for anti-bias evaluation.

An example of object tracking was presented by Han et al., [21]. They used the better responsiveness of event cameras across a wide intensity range to suggest an event-assisted object tracking procedure that can reliably follow objects even when the intensity levels vary greatly. Our proposal is to first create a U-Net-based image improvement procedure that uses nearby frames in the time domain to balance the RGB intensity, and then use this to build a dual-input tracking perfect that can track moving objects in video and event sequences. This will allow us to better understand and analyse dense event signals for feature matching. Comprehensive validation of the suggested technique is achieved through both simulation and real-world trials.

III. PROPOSED APPROACH

In this section, the brief explanation of each block for proposed methodology is explained and it is shown in **Fig 1**.

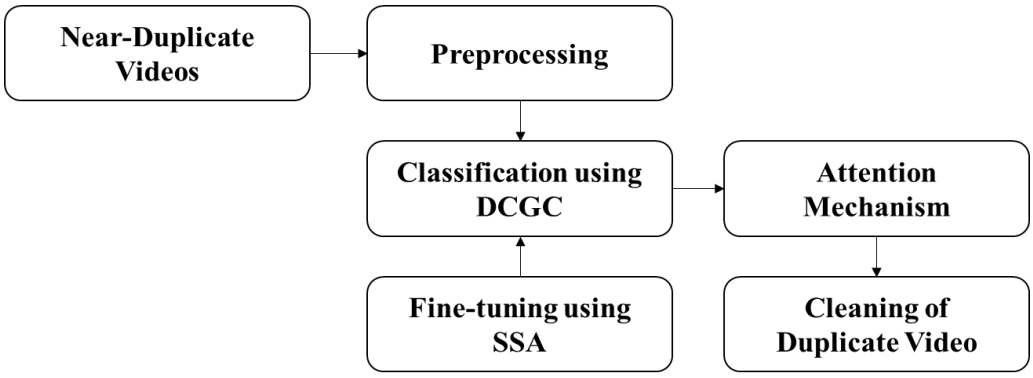

Fig 1. Workflow of the Proposed Model.

Video Data Preprocessing

While structured data is typically the focus of information searches, video is an example of unstructured data that needs pre-processing in order to be directly searched. The term "frame" can be used to describe the building blocks of a video. This research takes into account the fact that the storage space and processing capacity needs of the network model would increase if the characteristics of the video are retrieved from each frame using the network model. So, it's crucial to extract a number of critical frames that may stand in for the full video content if you want to get useful data characteristics out of video data. When it comes time to extract video features and decipher the video's semantic content, this is crucial.

When choosing video key frames, this study uses the random extraction approach. Put simply, the key frames of the movie are chosen from a predetermined number of frames according to the fixed interval. Concurrently, data augmentation procedures like resizing, flipping, regularisation, etc. are performed during the extraction of video key frames. This contributes to the network model's stability while also increasing its accuracy.

Densely Connected Graph Convolution

Graph Convolution

There are three layers that make up a GCN neural network: input, hidden, besides output. It learns node characteristics from the adjacency matrix and uses word co-occurrence vectors as nodes, relationships as edges. That is, with regard to the feature matrix of nodes $X = [x_1, x_2, x_3, \ldots, x_n]$ in addition to the adjacency matrix A. The diagonal elements of A are initialised to 1 by self-loops. Moreover, we include a degree matrix as well. $D \in R^{\mathrm{n*n}}$, where each element D_{ii} in D is:

$$D_{ii} = \sum_j A_{ij} \tag{1}$$

Hence, the procedure for learning node features in a one-layer GCN is:

$$L^{(1)} = relu\left(\tilde{A}XW_0\right) \tag{2}$$

where $\tilde{A} = D^{-1/2}AD^{-1/2}$, $W_0$is a trainable limit.

Dense Connection

As seen in Section 3.3.1, a single-layer GCN is limited to capturing the features of its immediate neighbours. In contrast, a GCN can acquire the crucial aspects of a multi-node graph by capturing the features of faraway nodes via numerous neighbouring nodes. This is the formula that represents multi-layer GCN stacking:

$$L^{(m)} = relu\left(\tilde{A}L^{(m-1)}W_{m-1}\right) \tag{3}$$

where m is the sum of layers of GCN.

The superficial features may become unimportant as the network develops deeper. With dense connections, it is feasible to directly combine deep and shallow properties. Drawing design inspiration from DenseNet, our densely linked GCN network (DC-GCN) employs multiplexing in shallow GCN layers to enhance feature capture at long distances between nodes.

In light of this, DC-GCN's m-th layer propagation becomes

$$L^{(m)} = relu\left(\tilde{A}\left[L^{(1)} \oplus L^{(2)} \oplus \ , \ldots . , L^{(m-1)} \oplus\right] W_{m-1}\right) \tag{4}$$

where $\oplus$ denotes joining together of different parts. Also, previous GCN models could only save the scale attributes of the last layer because they used only that layer's output. We depart from the conventional approach by retaining the outputs of every layer and relying on the intermediate features at different scales for the last product:

$$h_i = L_i^{(1)} \oplus L_i^{(2)} \oplus \ldots \oplus L_i^{(m)} \tag{5}$$

$$H = [h_1, h_2, \ldots h_n] \tag{6}$$

where hi is the $i - th$ word node. Therefore, complete this layer, we become matrix $H \in R^{n*(k*m)}$, where k dimension of GCN.

Attention Mechanism

The retrieved physiognomies must be mutual classified. For the feature matrix $H = [h_1, h_2, h_3, \ldots, h_n]$, Not all features are equally useful for the job. So, we make the attention module to draw attention to them because of how important they are. An attention score ai is computed for each feature hi; this score signposts the feature's meaning in relation to the classification task.

$$a_i = \frac{exp(e_i^T u_s)}{\sum_i exp(e_i^T u_s)} \tag{7}$$

$$e_i = relu(W_s h_i + b) \tag{8}$$

where W_{s} trainable limits, besides b is the bias term.

Finally, we increase the score on attention $\alpha = [a_1, a_2, a_3, \ldots, a_n] \in R^n$ and the feature matrix H. Sum and become the over illustration V:

$$V = \sum_{i \in n} a_i h_i \tag{9}$$

Finished this layer, we become the last representation$V \in R^{km}$.

Classification

There is just one fully connected layer in the categorization module. Based on the final representation V, this layer's objective is to determine the category's probability distribution. The equation that follows represents this layer.:

$$P = softmax(WV + b) \tag{10}$$

where P is the group's likelihood delivery. W is the limit, besides b is the bias besides $softmax$ is function.

Implementation

The graph $G = (V, E)$ is produced for each video discretely, where the $V = [w_1, w_2, w_3, \ldots, w_n]$ characterizes a video's entire list. The p is 2, besides the edge set E among window p. Then over layer $X = [x_1, x_2, x_3, \ldots; x_n] \in R^{n*d}$ is node vectors. Using the concat connection approach, the DC-GCN block densely connects five GCN layers for feature extraction. By assigning each feature an attention score ai, we can choose those that are most suited for the task at hand in video hash categorization. hello, AI shows how important the feature is for the classification job. After that, focus twice as much score $\alpha = [a_1, a_2, a_3, .., a_n] \in R^n$ with feature matrix H. Finally, add everything together to get V, the video representation. The classifier, which uses a fully connected layer and function, receives V as input and returns the judged label.

Fine-tuning using SSA

By modelling its operations after those of the sparrow, the sparrow search algorithm (SSA) is able to get the best possible answer. A sample of sparrows to serve as guards should be randomly selected after the discoverer-joiner sparrow population model has been established. Foraging directions and places should be provided by the discoverer to the sparrow population. In order to get food, the joiners will follow the finder, keep an eye on them, and even steal from them. Immediately upon becoming aware of the threat, the sparrow population will begin to exhibit anti-predation behaviours. At last, the optimal site for the whole population is determined by repeatedly iterating the positions of discoverers and joiners.

In the space of N×D, where N is the whole sum of sparrows and D is dimension, the i-th sparrow's position in space i is located.

s $X_i = (x_{i1}, x_{i2}, \cdots; x_{id}), i \in [1; N], d \in [1; D], x_{id}$ stands for the i-th sparrow's location in the d-dimensional space. Formula for updating the discoverer's location:

$$x_{id}^{t+1} = \begin{cases} x_{id}^t . exp\left(\frac{-i}{\alpha . T}\right), R_2 < ST \\ x_{id}^t + Q . L \qquad R_2 \geq ST \end{cases} \tag{11}$$

Among them, t characterizes the current sum of repetitions; T is the extreme amount of iterations; α is random sum among $[0; 1]$; Q is a random sum with normal distribution; L is a matrix are totally 1, and the size is $1 \times d$; $R_2 \in [0; 1]$ Characterizes the warning value; $ST \in [0.5; 1]$ characterizes the safety charge.

When R2 < ST, it resources search; When R2 ≥ ST, stands for the i-th sparrow's location in space. Formula for updating the discoverer's site Joiner location updates formula:

$$x_{id}^{t+1} = \begin{cases} Q . exp\left(\frac{x_{worst\ d}^t - x_{id}^t}{i^2}\right) & i > \frac{N}{2} \\ x_{best\ d}^{t+1} + \frac{1}{D}\sum_{d=1}^{D}(rand(-1,1). |x_{id}^t - x_{best\ d}^{t+1}|), & i \leq \frac{N}{2} \end{cases} \tag{12}$$

Among them, $x_{worst\ d}^t$ characterises worst site at t-th repetition; $x_{best\ d}^{t+1}$ characterizes the global best site at $the\ t + 1th$ iteration.

When i > $\frac{N}{2}$, What this signifies is that the i-th joiner is in need of food and will have to fly to other locations to get it. When i ≤ $\frac{N}{2}$, If the i-th joiner is aimlessly wandering, it indicates that it is near the global ideal position.

The apprise formula of the vigilant site:

$$x_{id}^{t+1} = \begin{cases} x_{worst\ d}^t + \beta(x_{id}^t - x_{worst\ d}^t), & f_i \neq f_g \\ x_{id}^t + K\left(\frac{x_{id}^t - x_{worst\ d}^t}{|f_i - f_w| + e}\right), & f_i = f_g \end{cases} \tag{13}$$

Among them, β describes the pace parameter, which is a usually distributed random sum with a mean of 0 besides a variance of 1; K stands for the sparrow's movement direction, and its value is a sum of random numbers between -1 and 1; e is a tiny constant.; f_{i} characterizes the fitness of the i-th sparrow; f_{g} embodies the populace; f_{w} signifies populace.

When $f_i \neq f_g$, it incomes that the i-th sparrow populace and is straightforwardly attacked by marauders; when $f_i = f_g$, it earnings that the i-th sparrow of the populace, and since it is alert of the threat, it requirements to be close to additional sparrows to lessen the catch risk.

Position Update Based on Learning Coefficient Besides Mutation Operator

Using learning mutation operators, this study aims to increase the search aptitude of the SSA, which is an issue with the classic SSA since it is easy to slip local extremum. A significant aptitude for foraging is possessed by the discoverer. If the discoverer goes too far in one direction, the algorithm as a whole will end up at a local optimal key. In this study, learning coefficients are introduced into the formula for the discoverer's site update with the aim of improving the discoverer's international search capabilities. When joiner is i > $\frac{N}{2}$, it extremes.

The purpose of this study is to enhance the capacity of certain joiners to escape from local extremes by incorporating a mutation operator into their position updating formula.

The method for recalculating the discoverer's position following an upgrade

$$x_{id}^{t+1} = \begin{cases} v(t)x_{id}^{t}.exp\left(\frac{-i}{a.T}\right), & R_2 < ST \\ v(t)x_{id}^{t} + Q.L, & R_2 \geq ST \end{cases} \tag{14}$$

Among them, $v(t)$ is the learning coefficient of the discoverer.

The appearance of $v(t)$ is:

$$v(t) = v_{min} + (v_{max} - v_{min}) \times sin\left(\frac{t}{T}\pi\right) \tag{15}$$

Among them, v_{max} and v_{min} are the learning coefficients correspondingly.

IV. RESULTS AND DISCUSSION

The presentation of the recommended method is evaluated in this study by extensive tests on two datasets: CC_WEB_VIDEO, a regularly used dataset, and a coal mining video dataset [22]. We ran all of our tests on a single system that had eight 2.10 GHz Intel Xeon CPUs, a graphics card from NVIDIA called a GP102, and software that was based on Python 3.6.5 and PyTorch 0.4.0. The outcomes and methodology of the experiment are then described in detail [23].

Dataset besides Evaluation Criteria

The proposed technique is tested in this research using the CC_WEB_VIDEO besides coal mining video datasets for comparison. With a grand entire of 13,129 video data points, the CC_WEB_VIDEO dataset encompasses 24 scenes. The efficiency of the proposed method is verified in this study by randomly selecting 63 movies from situations such as "The Lion Sleeps Tonight," "Evolution of Dance," "Folding Shirt," "Cat Massage," and "ok go-here it goes again." In order to evaluate the efficacy of the tactic detailed in this article, 125 videos containing 10 scenes were selected from the coal mining video dataset. The validation analysis of different classifiers with the projected model is presented in **Table 1** using various metrics.

Table 1. Comparative Analysis of Predictable Classical with Existing Procedures

Classifiers	Accuracy	Precision	Recall	F1-Score	AUC
MLP	90.12	91.09	90.43	91.13	92.02
AE	91.68	92.23	91.57	91.71	93.14
RNN	92.51	92.95	92.40	92.17	94.75
CNN	94.32	93.73	93.21	94.27	95.74
LSTM	95.46	95.21	95.21	95.31	96.43
GCN	95.82	96.28	96.12	97.76	97.06
DCGC-SSA	**97.17**	**98.28**	**97.76**	**98.93**	**98.43**

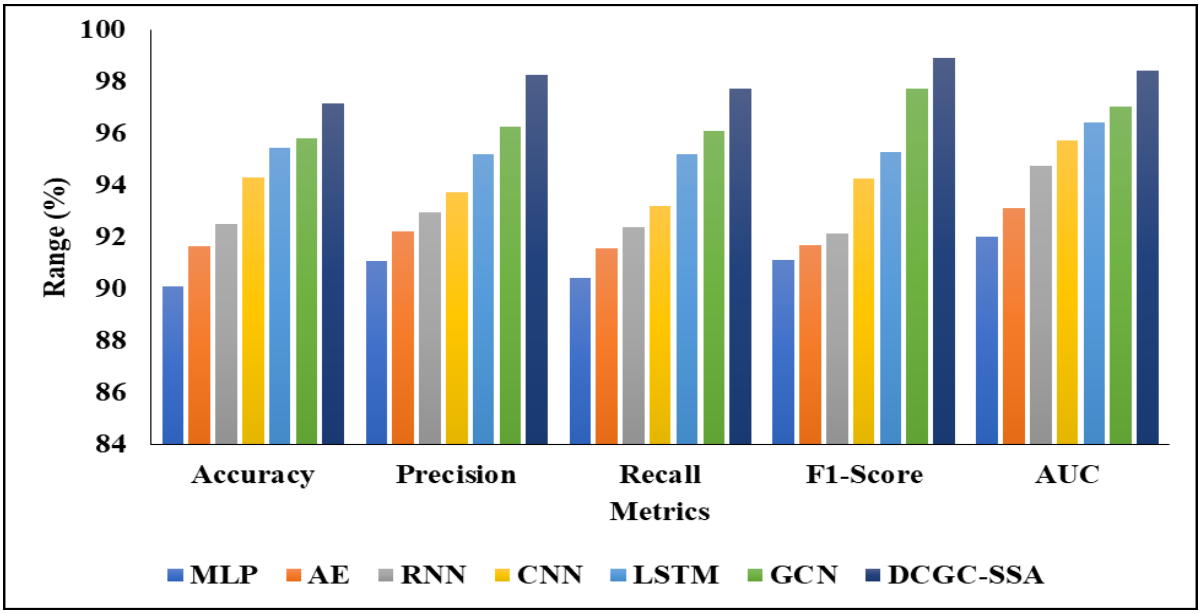

Fig 2. Graphical Illustration of Projected Prototypical with Existing Actions.

In **Table 1** and **Fig 2** represent the comparative Study of Projected with Existing procedures. In the study of MLP classifier technique attained the accuracy as 90.12 and also precision of 91.09 after the recall as 90.43 besides f1-score as 91.13 and then AUC range of 92.02 correspondingly. Then the AE classifier technique attained the accuracy as 91.68 besides also precision of 92.23 after the recall as 91.57 besides f1-score as 91.71 and then AUC range of 93.14 congruently. Then the RNN classifier technique attained the accuracy 92.51 besides also precision of 92.95 after the recall as 92.40 after the recall as 92.17 and then AUC range of 94.75 correspondingly. Then the CNN classifier technique attained the accuracy of 94.32 and also precision of 93.73 after the recall as 93.21 and f1-score as 94.27 and then AUC range of 95.74 respectively. Then the LSTM classifier technique attained the accuracy as 95.46 and also precision of 95.21 after the recall as 95.21 after the recall as 95.31 and then AUC range of 96.43 correspondingly. Then the GCN classifier technique attained the accuracy as 95.82 and also precision of 96.28 after the recall as 96.12 besides f1-score as 97.76 and then AUC range of 97.06 congruently. Then the DCGC-SSA classifier technique attained the accuracy as 97.17 besides also precision of 98.28 after the recall of 97.76 and f1-score as 98.93 besides then AUC range of 98.43 similarly.

V. CONCLUSION

To enhance the quality of video datasets, this study proposes an automated approach to cleaning near-duplicate video data using a reliable feature hash ring. This study proposes an attention-enhanced densely linked GCN network for video hash retrieval categorization. It all begins with making a special graph specifically for each video. After that, incorporate these separate graphs into a densely connected GCN network. Because of the extensive connectivity in the GCN network, the model can easily adapt to different scales when extracting video information. In order to improve the model's performance, the attention module takes in the extracted characteristics and prioritises each facet when combining them. On top of that, SSA model is used to optimise the model's limits. Experimental results on video datasets show that the suggested strategy is effective in automatically cleaning up videos that are almost identical to one another. This paper's suggested approach, however, does not include training a single deep neural network model for both feature extraction and grouping. The cleaning calculation on the consistent ring is also somewhat massive. We will investigate future possibilities for building an end-to-end video data cleaning approach.

Data Availability

No data was used to support this study.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests

References

[1]. P. Pei, X. Zhao, J. Li, Y. Cao, and X. Lai, "Vision Transformer-Based Video Hashing Retrieval for Tracing the Source of Fake Videos," Security and Communication Networks, vol. 2023, pp. 1–16, Jun. 2023, doi: 10.1155/2023/5349392.

[2]. J.-M. Guo, A. W. H. Prayuda, H. Prasetyo, and S. Seshathiri, "Deep Learning-Based Image Retrieval With Unsupervised Double Bit Hashing," IEEE Transactions on Circuits and Systems for Video Technology, vol. 33, no. 11, pp. 7050–7065, Nov. 2023, doi: 10.1109/tcsvt.2023.3268091.

[3]. D. Hemanand, N. P. G. Bhavani, S. Ayub, M. W. Ahmad, S. Narayanan, and A. H, "Multilayer vectorization to develop a deeper image feature learning model," Automatika, vol. 64, no. 2, pp. 355–364, Dec. 2022, doi: 10.1080/00051144.2022.2157946.

[4]. Y. Yang, H. Wang, J. Wang, K. Dong, and S. Ding, "Semantic-Preserving Surgical Video Retrieval With Phase and Behavior Coordinated Hashing," IEEE Transactions on Medical Imaging, vol. 43, no. 2, pp. 807–819, Feb. 2024, doi: 10.1109/tmi.2023.3321382.

[5]. V. Srinivasan, V. H. Raj, A. Thirumalraj, and K. Nagarajan, "Detection of Data imbalance in MANET network based on ADSY-AEAMBi-LSTM with DBO Feature selection," Journal of Autonomous Intelligence, vol. 7, no. 4, Jan. 2024, doi: 10.32629/jai.v7i4.1094.

[6]. L. Yuan et al., "Learnable Central Similarity Quantization for Efficient Image and Video Retrieval," IEEE Transactions on Neural Networks and Learning Systems, pp. 1–14, 2023, doi: 10.1109/tnnls.2023.3321148.

[7]. D. Lakshmi Narayana Reddy, R. Mahaveerakannan, S. Kumar, J. Chenni Kumaran, and M. Bhanurangarao, "A Structure for Forecasting Stomach Cancer Using Deep Learning and Advanced Tongue Characteristics," Smart Trends in Computing and Communications, pp. 1–14, 2024, doi: 10.1007/978-981-97-1313-4_1.

[8]. P. Jing, H. Sun, L. Nie, Y. Li, and Y. Su, "Deep Multi-modal Hashing with Semantic Enhancement for Multi-label Micro-video Retrieval," IEEE Transactions on Knowledge and Data Engineering, pp. 1–12, 2024, doi: 10.1109/tkde.2023.3337077.

[9]. X. Gao, Z. Chen, B. Zhang, and J. Wei, "Deep Learning to Hash with Application to Cross-View Nearest Neighbor Search," IEEE Transactions on Circuits and Systems for Video Technology, pp. 1–1, 2024, doi: 10.1109/tcsvt.2023.3273400.

[10]. T. Yu, P. Mascagni, J. Verde, J. Marescaux, D. Mutter, and N. Padoy, "Live laparoscopic video retrieval with compressed uncertainty," Medical Image Analysis, vol. 88, p. 102866, Aug. 2023, doi: 10.1016/j.media.2023.102866.

[11]. Z. Xi, X. Wang, and P. Cheng, "Unsupervised Hashing Retrieval via Efficient Correlation Distillation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 33, no. 7, pp. 3529–3541, Jul. 2023, doi: 10.1109/tcsvt.2023.3234037.

[12]. Y. Huo et al., "Deep Semantic-Aware Proxy Hashing for Multi-Label Cross-Modal Retrieval," IEEE Transactions on Circuits and Systems for Video Technology, vol. 34, no. 1, pp. 576–589, Jan. 2024, doi: 10.1109/tcsvt.2023.3285266.

[13]. C. Anitha, S. Srinivasulu Raju, R. Mahaveerakannan, A. Rajasekaran, and N. Pathak, "White Blood Cells Classification Using MBOA-Based MobileNet and Coupling Pre-trained Models with IFPOA," Innovative Computing and Communications, pp. 573–588, 2024, doi: 10.1007/978-981-97-3588-4_46.

[14]. S. P. Jadhav, A. Srinivas, P. Dipak Raghunath, M. Ramkumar Prabhu, J. Suryawanshi, and A. H, "Deep learning approaches for multi-modal sensor data analysis and abnormality detection," Measurement: Sensors, vol. 33, p. 101157, Jun. 2024, doi: 10.1016/j.measen.2024.101157.

[15]. K. Nithya and V. Rajamani, "Triplet Label Based Image Retrieval Using Deep Learning in Large Database," Computer Systems Science and Engineering, vol. 44, no. 3, pp. 2655–2666, 2023, doi: 10.32604/csse.2023.027275.

[16]. L. Zhu, C. Zheng, W. Guan, J. Li, Y. Yang, and H. T. Shen, "Multi-Modal Hashing for Efficient Multimedia Retrieval: A Survey," IEEE Transactions on Knowledge and Data Engineering, vol. 36, no. 1, pp. 239–260, Jan. 2024, doi: 10.1109/tkde.2023.3282921.

[17]. K. Wu and L. Xu, "Deep Hybrid Neural Network With Attention Mechanism for Video Hash Retrieval Method," IEEE Access, vol. 11, pp. 47956–47966, 2023, doi: 10.1109/access.2023.3276321.

[18]. W. Jo, G. Lim, G. Lee, H. Kim, B. Ko, and Y. Choi, "VVS: Video-to-Video Retrieval with Irrelevant Frame Suppression," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 3, pp. 2679–2687, Mar. 2024, doi: 10.1609/aaai.v38i3.28046.

[19]. P. Wu, J. Liu, X. He, Y. Peng, P. Wang, and Y. Zhang, "Toward Video Anomaly Retrieval From Video Anomaly Detection: New Benchmarks and Model," IEEE Transactions on Image Processing, vol. 33, pp. 2213–2225, 2024, doi: 10.1109/tip.2024.3374070.

[20]. S. Silvia Priscila, S. K. Piramu Preethika, S. Radhakrishnan, R. Bagavathi Lakshmi, M. Sakthivanitha, and R. Mahaveerakannan, "Chaotic Map Cryptographic Hash-Blockchain Technology with Supply Chain Management," Innovative Computing and Communications, pp. 599–612, 2024, doi: 10.1007/978-981-97-3588-4_48.

[21]. Y. Han, X. Yu, H. Luan, and J. Suo, "Event-Assisted Object Tracking on High-Speed Drones in Harsh Illumination Environment," Drones, vol. 8, no. 1, p. 22, Jan. 2024, doi: 10.3390/drones8010022.

[22]. Y. Qin, O. Ye, and Y. Fu, "An Automatic Near-Duplicate Video Data Cleaning Method Based on a Consistent Feature Hash Ring," Electronics, vol. 13, no. 8, p. 1522, Apr. 2024, doi: 10.3390/electronics13081522.

[23]. K. Aravinda, B. Santosh Kumar, B. P. Kavin, and A. Thirumalraj, "Traffic Sign Detection for Real-World Application Using Hybrid Deep Belief Network Classification," Advanced Geospatial Practices in Natural Environment Resource Management, pp. 214–233, Mar. 2024, doi: 10.4018/979-8-3693-1396-1.ch011.