DOI: 10.19729/j.cnki.1673-5188.2018.03.006
http://kns.cnki.net/kcms/detail/34.1294.TN.20180815.1424.002.html, published online August 15, 2018

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang
(School of Computer Science and Engineering, Nanyang Technological University, 639798, Singapore)

Abstract

Nowadays, media cloud and machine learning have become two hot research domains. On the one hand, the increasing user demand on multimedia services has triggered the emergence of media cloud, which uses cloud computing to better host media services. On the other hand, machine learning techniques have been successfully applied in a variety of multimedia applications as well as a list of infrastructure and platform services. In this article, we present a tutorial survey on the way of using machine learning techniques to address the emerging challenges in the infrastructure and platform layer of media cloud. Specifically, we begin with a review on the basic concepts of various machine learning techniques. Then, we examine the system architecture of media cloud, focusing on the functionalities in the infrastructure and platform layer. For each of these function and its corresponding challenge, we further illustrate the adoptable machine learning based approaches. Finally, we present an outlook on the open issues in this intersectional domain. The objective of this article is to provide a quick reference to inspire the researchers from either machine learning or media cloud area.

Keywords

machine learning; media cloud

1 Introduction

Recently, the increasing user demand on rich media experience has triggered an exponential growth of media services worldwide. According to the Cisco Visual Networking Index (VNI) report [1], the Internet video traffic would increase 3-fold from 2016 to 2021, contributing up to 82% of all Internet traffic by 2021. This trend may bring tremendous opportunities for all the stakeholders in the media service chain. Application developers can attract more customers by providing novel media experiences, such as video-on-demand, multi-screen interactions, and face/expression recognition. Platform service providers can host more applications and get more revenue. Content service providers can generate more contents and have them viewed by billions of users. Network service operators can expect to deliver significantly more network traffic. Nevertheless, such a trend also calls for novel paradigms to properly fulfil all the requirements.

Media cloud [2]–[5], inheriting the advances from cloud computing, has emerged as a promising computing paradigm to provide novel multimedia services with satisfied Quality of Service (QoS) and reduced cost. Specifically, media cloud adds media-related functions to each cloud computing layers (i.e., Infrastructure-as-a-Service(IaaS),Platform-as-a-Service(PaaS), and Software-as-a-Service (SaaS)), following the cloud computing paradigm. In the infrastructure layer, media cloud schedules more virtual resources in a more dynamic style. In the platform layer, it integrates a list of media-specific functions, such as media adaptation, media streaming, and media traffic analysis, to meet various QoS requirements from different media services. In the software layer, media cloud is able to host novel media services with higher complexity than traditional web services with only text and images.

The uniqueness of media cloud posits a list of new challenges, especially in the infrastructure and platform layer. First, the process, storage, and transmission of multimedia contents need more resources, leading to more power consumption and higher failure ratio of physical and virtual resources. Second, most media services need to be delivered with low latency and high volume, thus requiring precise workload prediction and careful resource scheduling accordingly. Third, the media distribution and adaption are more resource-intensive and thus more complicated than traditional web services. Last but not least, different media functions must be orchestrated properly to better serve the media users with optimized cost.

Machine learning, which have been intensively applied in various multimedia applications, provides a nature solution to address these challenges in media cloud. In particular, machine learning represents the set of algorithms that can progressively improve the performance of a specific task without being explicitly programmed. As a result, the adoption of machine learning makes the development of new media services and the optimization of existing media systems much easier than ever before. For example, machine learning has been already widely used in image and video processing such as face recognition, image classification, and video surveillance. However, the machine learning research in the infrastructure and platform layer of media cloud has not been as hot as the upper layer media applications.

In this article, we present a survey of how machine learning addresses the challenges in media cloud, from the infrastructure and platform perspectives. In particular, we start with the tutorial study on different machine learning strategies, as well as the concept and the challenges of media cloud. Then, we substantiate the ways of applying these machine learning techniques into media cloud via a literature review. The map between machine learning techniques and the challenges in the infrastructure and platform layer of media cloud are illustrated respectively. As a result, this allows the researchers from either machine learning or media cloud domain to quickly grasp the state-of-the-art knowledge in the overlaps of these two domains.

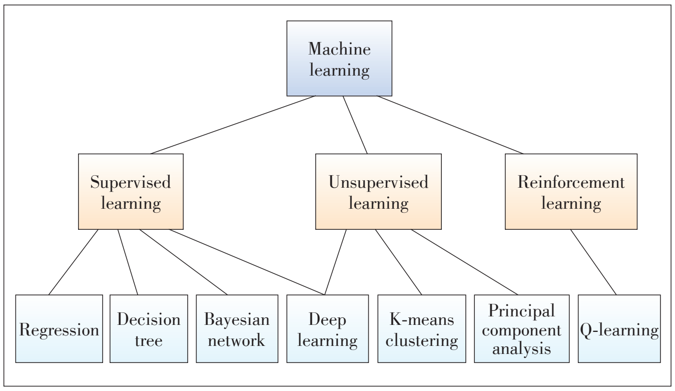
The rest of this paper is organized as follows. In Section 2, we introduce the basic ideas of machine learning as well as a layered media cloud framework and the functionalities in each layer. In Section 3, we review the machine learning efforts towards the challenges in the infrastructure layer of media cloud. In Section 4, we investigate the machine learning solutions to address the issues in the platform layer of media cloud. In Section 5, we highlight a list of open research issues in media cloud that could be addressed by machine learning techniques in the near future. Finally, in Section 6, we conclude this article.

2 Overview of Machine Learning and Media Cloud

In this section, we first introduce the basic machine learning algorithms to provide the necessary background knowledge, which will be referred to in the later sections. Then, we illustrate the media cloud framework and decompose it into a layered model. This model will serve as the blueprint to survey existing research efforts.

2.1 Machine Learning Algorithms

Existing machine learning algorithms can be generally categorized into three types [6], including supervised learning, unsupervised learning, and reinforcement learning. **Fig. 1** depicts such categorization, where each category further consists of one or more sub-categories. The brief concepts of these techniques are presented as follows.



▲**Figure 1. A categorization of machine learning.**

2.1.1 Supervised Learning

Supervised learning aims to build a model to map an input to an output based on pre-labelled input-output pairs. Typically, the input objective is a high dimension vector, the output is a low dimension or even one-dimension decision, while the objective is to minimize the difference between the labels and the output from the model. Regression, decision tree, Bayesian network, and deep neural network/deep learning are supervised learning algorithms

Regression tries to find a single function with proper parameters to represent the relationship between the input and output. There are a list of different regression models with different function types to deal with different input. For example, linear regression uses a linear function to deal with continuous input, logistic regression uses a logistic function to deal with categorical input, and non-linear regression uses non-linear functions (e.g., polynomial, logarithmic).

A decision tree uses a tree-like graph to deduct the consequences from the input. In a decision tree, each internal node refers to a control variable on an attribute, each branch refers to the consequence from the control decision, each leaf node refers to one final output, and the paths from the root to each leaf node refer to the rules.

Bayesian network is a probabilistic graphical model that represents a set of variables and their conditional dependencies via a directed acyclic graph. It calculates an estimate for the class probability from the training set based on the Bayes' theorem, and uses the estimation to map the input and output.

Deep neural network/deep learning is generally based on artificial neural networks, which consist of a collection of multiple layers of connected units (i.e., neurons). The weights between each pair of neurons are tunable to optimize the objective function. It can be used as a supervised learning approach for classification tasks.

2.1.2 Unsupervised Learning

On the contrary to supervised learning, unsupervised learning algorithms, such as K-means clustering, principle component analysis (PCA), and deep learning, focuses on inferring a function to describe the hidden structure from unlabeled data.

K-means clustering aims to partition n observations into k clusters, where each observation belongs to one cluster. The criteria is to ensure the overall shortest distance between the observations and the centroid of their assigned clusters accordingly.

Principle component analysis uses an orthogonal trans- formation to convert given observations into a set of values of linearly uncorrelated variables in lower dimension. The generated variables are often called as principal components. They serve as a projection of the original higher dimension input from its most informative perspective.

Deep learning can be also used in an unsupervised manner. Due to its multi-layer structure of fully connected neurons, deep learning can well represent complex non-linear relationships. As a result, it is able to compact the input in higher dimension into informative output with much lower dimension. Deep auto-encoder is one example in this category.

2.1.3 Reinforcement Learning

Reinforcement learning trains the model by interacting with the environment using different actions and receiving the incurred rewards iteratively. Specifically, it relies on two operations, including exploration of uncharted territory and exploitation of current knowledge to maximize the received rewards. On the one hand, exploration operation enables the algorithm to keep trying different decisions so that it can evolve without explicitly giving labelled data. On the other hand, the exploitation allows the algorithm to be aware of the explored point and move closer to the optimal decision strategy. Q-learning is a reinforcement learning algorithm.

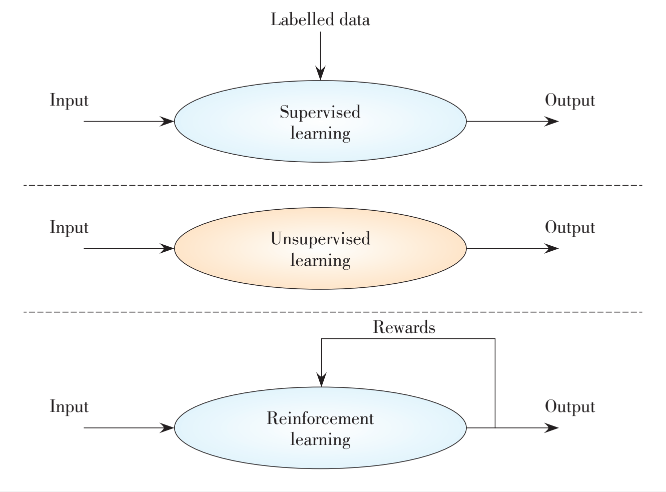
Q-learning is most representative reinforcement learning technique. Specifically, it uses Q-value to represent the quality of a state-action combination, and iteratively update this Q-value for the improvement. Q-learning can compare the expected utility of the available actions without requiring a model of the environment. Moreover, it has been proven that Q-learning is able to eventually get the optimal action-selection policy, for any finite Markov decision process.

Fig. 2 illustrates a comparison of supervised learning, unsupervised learning, and reinforcement learning. In particular, supervised learning relies on the pre-defined labelled input and output pairs as the target. On the other hand, unsupervised learning does not need labelled data, and it uses the internal features of the dataset instead of any labelled data as the objective. Whereas reinforcement learning does not have the labelled data in advance. It has to sense the results by performing different actions, and use the previous outputs as the objective.

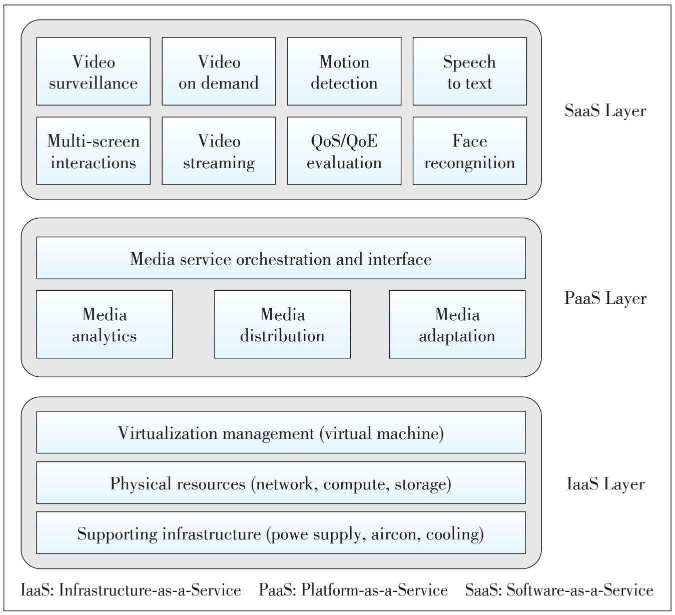
We will illustrate how the machine learning techniques from different categories can be applied in media cloud in Sections 3 and 4.

2.2 Media Cloud Framework

Media cloud aims to leverage cloud computing paradigm together with a list of media-related functions to enhance the media experience. From a cloud-centric view [2], [5], it still can be defined as a cloud-centric layered model as shown in **Fig. 3**. Each layer consists of traditional cloud services (e.g., virtualization and resource management) and corresponding media services (e.g., media adaptation and media analytic). This conceptual hierarchy provides a clear clue for us to characterize the technical challenges and existing works in different layer. Note that, this paper mainly focuses on the machine learning works for infrastructure and platform layer, whereas the efforts



▲Figure 2. Comparison of three different machine learning categories.


Video surveillance
Video on demand
Motion detection
Speech to text
Multi-screen interactions
Video streaming
QoS/QoE evaluation
Face recongnition
SaaS Layer
Media service orchestration and interface
Media analytics
Media distribution
Media adaptation
PaaS Layer
Virtualization management (virtual machine)
Physical resources (network, compute, storage)
Supporting infrastructure (powe supply, aircon, cooling)
IaaS Layer
IaaS: Infrastructure-as-a-Service
PaaS: Platform-as-a-Service
SaaS: Software-as-a-Service

▲Figure 3. A layered architecture of media cloud, consisting of three layers, including IaaS, PaaS, and SaaS from the bottom up.

towards multimedia software applications have been intensively reviewed by many other literatures [7]–[11].

The infrastructure layer aggregates all the physical ICT resources together via virtualization technology, with the objective to allocate them in a fine-granular, on-demand, and fault-free manner. According to the different functionalities, we can further classify this layer into the following three sub-layers.

- Supporting Infrastructure: This layer refers to the power supply, air-conditioner, and other cooling systems, which support the smooth operations of datacenters [12] as well as the cloud services on top of them. It focuses on those bottom level schemes, for instance the datacenter layout schedule, power consumption optimization, and cooling system management [13].
- Physical Resources: This layer consists of servers (including CPUs, hard disks, memory, network interface, etc.) and switches/routers, which provide the networking, computation, and storage capacity. These resources need to be monitored in real time and well maintained once there is any fault [14]. As a result, the hosted cloud services will not be affected.
- Virtualization Management: This layer virtualizes the underlying physical resources into a virtual resource pool in terms of virtual machines [15], [16]. These resources are then exposed to the cloud platform services to meet the specific Service Level Agreement (SLA) with the lowest possible cost. To achieve this target, the resource provision needs be allocated elastically and dynamically via virtual machine configuration and migration.

Following this hierarchy, machine learning algorithms can be developed and applied to each of the sub-layer to address the corresponding challenges. This will be the main focus of Section 3.

The platform layer encapsulates various fundamental media services into a layer of middle-ware, by utilizing the virtual resources provided by the infrastructure layer. This middle-ware is then exposed to the software layer via a set of APIs. According to the functionalities, we cast these media services into the following four types.

- Media Analytics: This service refers to the data mining schemes that focus on the nature of media contents as well as the user request patterns. Typical examples include media content popularity prediction [17] and content recommendation [18].
- Media Distribution: This service is in charge of acquiring media contents from the origin servers, and delivering them to end users throughout the media cloud. The objective is to improve the delivery efficiency. Content caching [19] and pre-fetching [20] are two representative examples under this category.
- Media Adaptation: This service modifies the original media contents into the target ones with different domains (e.g., format, rate, resolution, and annotation). Typical examples of such services are content encoding/transcoding [21], content quality estimation/assessment [22], and media mashup [23].

Similarly, intelligent mechanisms powered by machine learning algorithms can be adopted by these services to improve the quality. The in-depth survey of adopting machine learning in the cloud-based media platform services will be covered in Section 4.

3 Machine Learning in Infrastructure Layer

In this section, we present three typical scenarios that can be benefited from machine learning techniques in the infrastructure layer of media cloud (Fig. 3). They are datacenter power consumption prediction and control, cloud resources failure prediction and operation, and virtual machine configuration and operation.

3.1 Power Consumption Prediction and Control

Datacenters nowadays have become a large energy consumption center, resulting in the fact that even modest improvements are able to yield significant cost cut and avert millions of carbon emissions globally. In particular, power consumption from datacenter comprises around 1.4% of global energy usage and 2% of global carbon emissions [24]. Among all datacenters in the world, the majority of them has a power usage effectiveness (PUE) at 1.6–2.0 while the ideal efficiency should be around 1.1–1.2 [24]. As a result, there are sufficient improvement spaces, and any small one improvement can bring great impact.

Regression is one of the classic ways to predict the power consumption. Choi et al. [25] proposed a three-dimension regression model to predict the power usage using work intensity and CPU utilization. This model achieved 9% error margin on average comparing with real observed usage. Similarly, Lewis et al. [26] developed a linear regression model to correlate processor power, bus activity, and system ambient temperatures with real-time server power consumption by considering. Their model gave an error of 4% as verified using a set of benchmarks. In addition, some further studies [27] indicated that the linear regression model based on CPU usage and workload is only able to provide reasonable prediction accuracy for CPU-intensive jobs, while a Gaussian mixture regression model can perform consistently well with different workload (e.g., IO or memory intensive jobs).

Neural network/deep learning is another powerful tool to predict the data-center power consumption with the ability to take into much more input parameters. Gao [28] from Google, presented a simply three layers neural network model by considering 19 different factors as the input, including the IT load, weather conditions, number of chillers and cooling towers running, equipment set points and so on. This generated a promising prediction performance already, with a mean absolute error of 0.4% and standard deviation of 0.005. Li et al. [29] further deepened the neural network with more layers. Specifically, it used a linear recursive auto-encoder to process the input, and added an additional layer before the final output to correct the prediction results of auto-encoder. This model was fed by 11-dimension input, including CPU/memory/disk usage, network traffic, and file system workload. The results pushed the performance a bit further by reducing around 40% prediction error comparing with a widely-used regression based time-series prediction model.

3.2 Cloud Failure Prediction and Operation

Given the scale and complexity of cloud infrastructure, the failure prediction and operation desires significant high levels of automation. In particular, such failure consists of physical hardware failure such as disk/CPU/memory/network error and virtual jobs failure due to software or configuration issues. It is challengeable but important to properly identify these failures and take actions accordingly on time if not in advance, so that the high standard Service Level Agreement can be well maintained.

Decision trees have become a popular method for failure prediction and detection. Pelleg et al. [30] collected system metrics including execution count, CPU usage, waiting time, blocked time, and IO count, on top of Xen virtual machine, and fed them into a decision tree classifier. By using this classifier, they were able to detect the potential system problems with 0.94 receiver operating characteristic (ROC) curve as the accuracy. Fu [31] proposed a framework to combine the decision tree model together with the principle component analysis algorithm. Specifically, the principle component analysis is first used to reduce feature dimensions from the set of collected cloud infrastructure parameters, then only the principle components are input into a decision tree classifier to identify anomalies in the cloud. A following-up work [32] further integrated a Bayesian model with the decision tree. It first reduced 50 plus system metrics including CPU statistics, memory swap statistics, IO requests, and network traffic into principle components. Then this solution fed these generated components into both the Bayesian predictor and decision tree, and did an ensemble between the output. This generated a promising result with 0.99 ROC curve.

At the same time, there is an increasing popularity of applying neural network or deep learning into the cloud failure prediction task recently. Prevost et al. [33] presented a neural network model to predict the cloud datacenter work- load. Specifically, the model takes historical data points as input to predict the future trend, with the objective to minimize the Rooted Mean Squared Error (RMSE) of sample data. Chen et al. [34] developed a recurrent neural network (RNN) based model to learn the temporal characteristics of resource usage metrics including CPU and memory usage, which are in turn used to calculate the failure possibility of a running job in the cloud. They then verified the model by using real Google cluster workload traces. The results indicated a reasonable accuracy with a false positive rate at around 6%, and the following-up operations based on the prediction were able to save 6% to 10% cost saving by early killing and restarting jobs with high failure possibility. Zhu et al. [35] also explored the performance of back propagation based neural network combined with a boosting approach, in driver failure prediction for large scale storage system. Moreover, it compared the results with a traditional Supported Vector Machine (SVM) model on a real world database. The evaluation showed the proposed neural network model achieves over 95% detection accuracy which is much better than 68% achieved by SVM.

3.3 Virtual Resource Configuration and Consolidation

Cloud infrastructure virtualizes the physical resources into a virtual machine pool and operates them in a fine-grained model, thus providing significant flexibilities to host different services. In particular, virtual machines can be dynamically turned on/off, migrated from one physical machine to another. As a result, there is a chance to significantly increase the cost efficiency by properly orchestrating the virtual machines to consolidate the workload in an on-demand manner.

Bayesian networks have become a popular tool to consolidate virtual resources for cloud environment. Sohrabi et al. [36] proposed a virtual machine migration heuristic based on Bayesian networks. In particular, this solution evaluates the probability of a physical server host to be overloaded, then migrates the virtual machines away from those servers. As a result, not only energy consumption can be saved by consolidating virtual machines, but also the performance is improved by balancing the workload into multiple hosts. Li et al. [37] discussed a very similar Bayesian based approach to estimate the resource utilization in physical machines and then used it to predict the migration probability of virtual machines. Shyam et al. [38] presented a Bayesian model to determine both short and long term virtual resource requirements for CPU or memory intensive applications running in cloud environment. They built the Bayesian model based on a list of parameters, including day of week, time-interval of application access, workload, benchmarks, and availability of virtual machines. All of these works were able to generate better performance in terms of either lower energy consumption of cloud infrastructure or higher accuracy in predicting virtual resource utilization, by comparing with a few other non-machine-learning methods.

Reinforcement learning has also been applied into this task. Masoumzadeh et al. [39] presented a Q-learning based model, which takes multiple virtual machine metrics (including CPU performance, disk storage, memory usage and network bandwidth) as the input, the migration action as the output, and the energy consumption combined with SLA score as the reward function. The trained model outperforms virtual machine selection policies using fixed criteria for decision making. Jin et al.

[40], [41] built the virtual machine migration model specifically for the cloud media scenario by using the same technique. In particular, this work used the user interactive behaviors in multi-screen applications as the input, the backend virtual machine migration decision as the output, and the total monetary cost of operating cloud resources as the rewards. The result revealed a significant cost saving compared with some heuristics. The model also showed a very closed performance to an offline optimal solution. Liu et al. [42] introduced deep reinforcement learning into the virtual machine allocation and consolidation problem. Specifically, deep reinforcement learning integrates deep neural network with reinforcement learning, enabling the algorithm to deal with larger state space while keeping the fast coverage speed. Thus, this work is able to take the real-time metrics for each job and virtual machine pair as the input, the job and virtual machine matching decision as the output, and the combined job latency and energy consumption as the rewards. Similarly, this approach also achieves cost saving while at the same time the latency improvement.

3.4 Summary

We demonstrate a list of works that use different machine learning techniques to tackle three major infrastructure challenges in this section. **Table 1** matches the specific machine learning approaches with the topic domains for each work, so that the interested readers can quickly obtain the ways how machine learning can be applied in the infrastructure layer in media cloud.

4 Machine Learning in Platform Layer

In this section, we discuss the machine learning applications in three major media platform services (Fig. 3). Specifically, this section covers content popularity prediction and recommendation in media analysis domain, content caching and pre-fetching in media distribution domain, and content transcoding in media adaptation domain.

4.1 Content Popularity Prediction and Recommendation

The tremendous growth of multimedia content generation has changed not only the user content consumption behaviors, but also the way of operating the media services. Millions of hours of video are generated and uploaded to YouTube every day [43]. As opposed to the traditional TV programs where all the audiences watched the same content at the same time, multimedia content users have much more options to spend their video watching time. As a result, given such a large amount of available user generated content, their popularity are much more difficult to be predicted. Moreover, the personalized video recommendation becomes increasingly important for better user experience.

Regression is the simplest yet feasible machine learning tool for dealing with the content popularity prediction task. Szabo et al. [44] found the long-term content popularity on YouTube had a strong correlation with their early popularity. Such correlation can be represented by a linear regression model to predict the long-term content popularity. Borgho et al. [45] confirmed the efficiency of using the linear model to predict the popularity, and further derived a multi-linear regression model by taking more factors such as video quality, number of keywords, uploader view count, uploader followers, and uploader video count. Chu et al. [46] adopted a similar approach by using a bilinear regression framework to achieve a personalized content recommendation system. They used this regression model to associate the attributes in user profiles with the potential content that might be interested to the user. Unsupervised learning tools provide another angle to examine the content popularity task. Szabo et al. [44] used k-means algorithm to separate video contents into two clusters, where the content popularity in one group grew faster than the average, and the other grew slower. Borgho et al. [45] applied PCA to characterize the relationships between different content/user profiles and the content popularity. In this way, they were able to identify the groups of variables which were responsible for the variation of popularity prediction. Ahmed et al. [47] introduced another clustering algorithm known as affinity propagation to the content popularity prediction task. This method does not require a predefined number of clusters, which differs from the k-means algorithm. By properly cluster the similarity score for the content popularity, this approach is able to outperform the traditional k-means and the linear regression models.

There are also a few works making use of deep learning for content recommendation. Ma et al. [48] developed an auto-encoder model backed by unsupervised deep learning technique, to cluster the similarity among different videos. They could recommend different videos to different users according to their categories. Covington et al. [49] designed a YouTube recommendation system based on a fully-connected deep neural network. It first embeds the video profile, video watch history, search tokens, previous impressions, and user profile into high-

▼Table 1. Mapping between machine learning methods and cloud infrastructure services for each literature work

	Regression	Decision tree	Bayesian network	PCA	Q-learning	Deep learning
Power predict and control	[25], [26], [27]					[28], [29]
Failure predict and operate		[30], [31], [32]		[31], [32]		[33], [34], [35]
VM configure and consolidate			[36], [37], [38]		[39], [40], [41], [42]	

PCA: principle component analysis

dimension vectors, and uses the concatenation of them as the input to the neural network. And the output can be directly used as ranked recommendations for each individual user.

4.2 Media Content Caching and Pre-fetching

It is a common practice nowadays to cache multimedia content data in the intermediate nodes between users and the host servers, to improve the user experience as well as the media service operational cost. In particular, because the sizes of multimedia contents are much larger than the traditional text or images, it takes more time to transmit them from the host to the end-users. To relieve the pain, content delivery networks has been proposed to cache contents in some middle places. However, it is not efficient to cache all or just blindly choose a few at all the time. Therefore, the key factor of this task is how to choose the right content to be cached at the right time. Bayesian network is a promising tool for content personalization prefetching task by identifying the right content to be cached in the content delivery network [50]. Venketesh et al. [51] introduced the naive Bayesian classifier to calculate the probability of viewing a potential content based on the browsing pattern exhibited by the end users in sessions. This approach helps to increase the cache hit rate and minimize access latency, especially when user has long browsing sessions. Ali et al. [52] used naive Bayesian classifier in the same task but in a different way. Specifically, they incorporated the Bayesian classifier with the classical caching strategy (e.g., Least-Recently-Used and Greedy-Based), by learning the dependency probability between the content access log and the content re-visit event. As a result, when doing cache eviction, the content with higher probability of re-visit will be kept. Clustering is another promising way for the content personalization prefetching task [50]. Yan et al. [53] uses K-means to cluster users based on their geo-location and temporal access patterns. In this way, the contents for different mobile applications can be prefetched into the mobile device, thus reducing the app launching delay and improving the user experience. Hu et al. [54] applied the affinity propagation clustering algorithm to group users in different communities, based on their social relationships, geo-locations, and video watching interests. As a result, the content caching decision can be made specifically for different communities, thus improving the caching efficiency.

It is also possible to use reinforcement learning to optimize this content prefetching process. Hu et al. [55] formulated the content prefetching problem as a Markov Decision Process (MDP), with the objective to strike a balance between the increased content caching cost from incorrect prediction and the reduced content download delay from correct prediction. A Q-learning based approach was then proposed to address this problem.

4.3 Media Content Adaptation

There is an increasing trend to consume online video via mobile phones rather than via fixed terminals like TV and PCs. This means the video contents must adapt to the terminals by providing different resolution, bitrates, and quality versions for different screen sizes under different network environments. Such video adaptation tasks can be computation intensive, but at the same time, they also pose an opportunity to improve the user experience with different devices.

Deep learning is the most widely used machine learning tool for such tasks. Covell et al. [56] explored a neural network based framework to predict the parameters of a model that relates the bitrate to various video properties. Specifically, in video transcoding, the perceptual video quality for a given bandwidth constraint can be achieved by controlling the quantization levels. In this context, they used deep neural network to correlate this quantization level with the bitrate, and achieved a much higher accuracy than the traditional alternative. Dash et al. [57] proposed to use deep neural network to assess the quality of images after encoding/decoding or transcoding, and the model is able to achieve as high as 98% image-level accuracy for the assessment. Zhang et al. [58] further extended the quality of experience assessment from images into video by using an even deeper neural networks with more hidden layers and unique structures.

4.4 Summary

In this section, we demonstrate the way of using machine learning techniques for three important media platform services. **Table 2** maps each work according to the adopted machine learning technique as well as the detailed platform services that it focused on. As a result, the interested readers can quickly obtain the ways how machine learning can be used in the platform layer services in media cloud.

5 Open Research Issues

The research on applying machine learning to media cloud is at the infancy stage, while there are still many open challenges. In this section, we present a brief outlook on these open issues, aiming to provide insights for researchers from either machine learning or media cloud area.

5.1 Media Traffic Classification and Flow Control

Real-time media traffic classification and flow information are important for network management and optimizing the service operational cost. The traditional way of classifying Internet traffic is based on the network protocols (e.g., TCP or UDP). However, such static methods are not enough for media contents as they roughly use the same protocol for network transmission.

Machine learning is able to either learn from the historical media traffic data with fine-grained categories as per different metric set in a supervised manner, or cluster the real-time media traffic based on their internal features into different groups

▼Table 2. Mapping between machine learning methods and cloud platform services for each literature work

	Regression	Bayesian network	K-means	PCA	Affinity propagation	Q-learning	Deep learning
Content recommendation	[44], [45], [46]		[44]	[45]	[47]		[48], [49]
Content prefetching		[50], [51], [52]	[53]		[54]	[55]	
Media data adaptation							[56], [57], [58]

PCA: principle component analysis

in an unsupervised manner. For example, for the former one, it is possible to make use of distributed SVM [59], and deep learning [60]. While for the later one, K-means [61] and deep auto-encoder [62] can be the right tools.

5.2 Media Service Chain Orchestration

Recently, network function virtualization (NFV) emerges to transform the way of operating communication networks. Specifically, NFV implements network functions in software, orchestrating various services dynamically instead of follow the pre-defined workflows from hardware. As a result, it provides an opportunity to dramatically increase the infrastructure flexibility, simplify the resource management process, and reduce both hardware and operational cost.

The emergence of NFV-enabled media cloud framework [63] offers the opportunity to further improve the performance of media services running on top of the media cloud, and machine learning can be one of the best candidates to achieve this target. In particular, media services are not standalone. Most media services require a list of functions to be orchestrated in a chain. For example, the consumption of online video streaming via a mobile phone involves content caching, prefetching, adaptation and personalization. Machine learning can be used to learn the most efficient pattern on how to orchestrate these services in the large scale.

5.3 Media Security

Nowadays, it is much easier to access, download, and upload multimedia contents via the Internet, making the Digital Rights Management (DRM) much more difficult and complicated than before. Previously, audio and video DRM was usually achieved by physical subscription and rental, but this method does not work well today, because any subscriber can simply upload the copyrighted contents as user-generated contents to popular video distribution platforms like YouTube. It is hard to restrict such behavior giving the huge amount of uploaded contents every day.

Machine learning can be applied in this field too. In particular, it can be used to classify or identify if the uploaded audio or video has a copyright issue, by learning from a set of labelled contents from their commercial owners. It is also possible to use machine learning to improve the performance of DRM techniques such as digital watermarking by learning from the failed cases. In fact, such operation has been introduced to the music and audio DRM system [64], while it is on the way to extend to video contents.

6 Conclusions

In this article, we presented a tutorial survey on applying machine learning techniques to address challenges in the infrastructure and platform layers of media cloud. In particular, we first reviewed the basic concept of different machine learning techniques. Then, we examined the system architecture of media cloud framework, focusing on the functionalities in the infrastructure and platform layers. For each functionality and its corresponding challenge, we further illustrated the adopted machine learning techniques. Finally, we present an outlook on a few open issues in this domain, aiming to inspire the researchers from either machine learning or media cloud area.

References

[1] Cisco. (2017). Cisco visual networking index: Forecast and methodology, 2016-2021 [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/ visual-networking-index-vni/complete-white-paper-c11-481360.pdf

[2] W. Zhu, C. Luo, J. Wang, and S. Li, "Multimedia cloud computing," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 59–69, May 2011. doi: 10.1109/MSP.2011.940269.

[3] M. Tan and X. Su, "Media cloud: when media revolution meets rise of cloud computing," in *IEEE 6th International Symposium on Service Oriented System Engineering (SOSE)*, Irvine, USA, 2011, pp. 251–261. doi: 10.1109/SOSE.2011.6139114.

[4] Y. Xu and S. Mao, "A survey of mobile cloud computing for rich media applications," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 46–53, Jun. 2013. doi: 10.1109/MWC.2013.6549282.

[5] Y. Wen, X. Zhu, J. Rodrigues, and C. Chen, "Cloud mobile media: Reflections and outlook," *IEEE Transactions on Multimedia*, vol. 16, no. 4, pp. 885–902, Jun. 2014. doi: 10.1109/TMM.2014.2315596.

[6] S. Marsland, *Machine Learning: An Algorithmic Perspective*. Boca Raton, USA: CRC Press, 2015.

[7] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recogni- tion: A literature survey," *ACM computing surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[8] R. Poppe, "A survey on vision-based human action recognition," *Image and vision computing*, vol. 28, no. 6, pp. 976–990, 2010.

[9] M. Wang and X.-S. Hua, "Active learning in multimedia annotation and retrieval: a survey," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 2, article no. 10, Feb. 2011. doi: 10.1145/1899412.1899414.

[10] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 41, no. 6, pp. 797–819, Nov. 2011. doi: 10.1109/TSMCC.2011.2109710.

[11] L. Deng and X. Li, "Machine learning paradigms for speech recognition: an overview," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1060–1089, May 2013. doi: 10.1109/TASL.2013.2244083.

[12] L. A. Barroso, J. Clidaras, and U. Ho¨lzle, "The datacenter as a computer: an introduction to the design of warehouse-scale machines," *Synthesis Lectures on Computer Architecture*, vol. 8, no. 3, pp. 1–154, 2013.

[13] W. Zhang, Y. Wen, Y. Wong, K. Toh, and C.-H. Chen, "Towards joint optimization over ict and cooling systems in data centre: a survey," *IEEE Communica-*

tions Surveys and Tutorials, vol. 18, no. 3, pp. 1596–1616, 2016. doi: 10.1109/COMST.2016.2545109.

[14] M. Isard, "Autopilot: automatic data center management," *ACM SIGOPS Operating Systems Review*, vol. 41, no. 2, pp. 60–67, Apr. 2007. doi: 10.1145/1243418.1243426.

[15] P. Barham, B. Dragovic, K. Fraser, et al., "Xen and the art of virtualization," in *ACM Symposium on Operating Systems Principles*, New York, USA, 2003.

[16] S. Soltesz, H. Pötzl, M. E. Fiuczynski, A. Bavier, and L. Peterson, "Container-based operating system virtualization: a scalable, high- performance alternative to hypervisors," *ACM SIGOPS Operating Systems Review*, vol. 41, no. 3, pp. 275–287, 2007. doi: 10.1145/1272996.1273025.

[17] A. Tatar, M. D. de Amorim, S. Fdida, and P. Antoniadis, "A survey on predicting the popularity of web content," *Journal of Internet Services and Applications*, vol. 5, no. 1, pp. 1–20, 2014. doi: 10.1186/s13174-014-0008-y.

[18] F. Xia, N. Y. Asabere, A. M. Ahmed, J. Li, and X. Kong, "Mobile multimedia recommendation in smart communities: a survey," *IEEE Access*, vol. 1, pp. 606–624, Sept. 2013. doi: 10.1109/ACCESS.2013.2281156.

[19] S. Podlipnig and L. Böszörmenyi, "A survey of web cache replacement strategies," *ACM Computing Surveys*, vol. 35, no. 4, pp. 374–398, 2003.

[20] J. Famaey, F. Iterbeke, T. Wauters, and F. De Turck, "Towards a predictive cache replacement strategy for multimedia content," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 219–227, 2013. doi: 10.1016/j.jnca.2012.08.014.

[21] I. Ahmad, X. Wei, Y. Sun, and Y.-Q. Zhang, "Video transcoding: an overview of various techniques and research issues," *IEEE Transactions on multimedia*, vol. 7, no. 5, pp. 793–804, Oct. 2005. doi: 10.1109/TMM.2005.854472.

[22] Y. Chen, K. Wu, and Q. Zhang, "From QoS to QoE: a tutorial on video quality assessment," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 1126–1165, 2015. doi: 10.1109/COMST.2014.2363139.

[23] M. K. Saini, R. Gadde, S. Yan, and W. T. Ooi, "Movimash: online mobile video mashup," in *Proc. 20th ACM International Conference on Multimedia*, Nara, Japan, 2012, pp. 139–148.

[24] M. Avgerinou, P. Bertoldi, and L. Castellazzi, "Trends in data centre energy consumption under the european code of conduct for data centre energy efficiency," *Energies*, vol. 10, no. 10, p. 1470, 2017.

[25] J. Choi, S. Govindan, B. Urgaonkar, and A. Sivasubramaniam, "Pro- filing, prediction, and capping of power consumption in consolidated environments," in *IEEE International Symposium on Modeling, Analysis and Simulation of Computers and Telecommunication Systems*, Baltimore, USA, 2008, pp. 1–10. doi: 10.1109/MASCOT.2008.4770558.

[26] A. W. Lewis, S. Ghosh, and N.-F. Tzeng, "Run-time energy consumption estimation based on workload in server systems," *HotPower*, vol. 8, pp. 17–21, 2008.

[27] G. Dhiman, K. Mihic, and T. Rosing, "A system for online power prediction in virtualized environments using gaussian mixture models," in *ACM/IEEE 47th Design Automation Conference (DAC)*, Anaheim, USA, 2010, pp. 807–812. doi: 10.1145/1837274.1837478.

[28] J. Gao, "Machine learning applications for data center optimization," Google White Paper, 2014.

[29] Y. Li, H. Hu, Y. Wen, and J. Zhang, "Learning-based power prediction for data centre operations via deep neural networks," in *Proc. 5th International Workshop on Energy Efficient Data Centres*, Waterloo, Canada, 2016, pp. 1–10. doi: 10.1145/2940679.2940685.

[30] D. Pelleg, M. Ben-Yehuda, R. Harper, L. Spainhower, and T. Adeshiyan, "Vigilant: out-of-band detection of failures in virtual machines," *ACM SIGOPS Operating Systems Review*, vol. 42, no. 1, pp. 26–31, Jan. 2008.

[31] S. Fu, "Performance metric selection for autonomic anomaly detection on cloud computing systems," in *IEEE Global Telecommunications Conference (GLOBECOM 2011)*, Kathmandu, Nepal, 2011, pp. 1–5. doi: 10.1109/GLOCOM.2011.6134532.

[32] Q. Guan, Z. Zhang, and S. Fu, "Ensemble of bayesian predictors and decision trees for proactive failure management in cloud computing systems," *Journal of Communications*, vol. 7, no. 1, pp. 52–61, 2012. doi: 10.4304/jcm.7.1.52-61.

[33] J. J. Prevost, K. Nagothu, B. Kelley, and M. Jamshidi, "Prediction of cloud data center networks loads using stochastic and neural models," in *IEEE International Conference on System of Systems Engineering (SoSE)*, Albuquerque, USA, 2011, pp. 276–281. doi: 10.1109/SYSOSE.2011.5966610.

[34] X. Chen, C.-D. Lu, and K. Pattabiraman, "Failure prediction of jobs in compute clouds: A google cluster case study," in *IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, Naples, Italy, 2014, pp. 341–346. doi: 10.1109/ISSREW.2014.105.

[35] B. Zhu, G. Wang, X. Liu, et al., "Proactive drive failure prediction for large scale storage systems," in *IEEE 29th Symposium on Mass Storage Systems and Technologies (MSST)*, Long Beach, USA, 2013, pp. 1–5. doi: 10.1109/MSST.2013.6558427.

[36] S. Sohrabi, A. Tang, I. Moser, and A. Aleti, "Adaptive virtual machine migration mechanism for energy efficiency," in *Proc. 5th International Workshop on Green and Sustainable Software*, Austin, USA, 2016, pp. 8–14. doi: 10.1145/2896967.2896969.

[37] Z. Li, C. Yan, X. Yu, and N. Yu, "Bayesian network-based virtual machines consolidation method," *Future Generation Computer Systems*, vol. 69, pp. 75–87, 2017. doi: 10.1016/j.jnca.2016.03.002

[38] G. K. Shyam and S. S. Manvi, "Virtual resource prediction in cloud environment: a bayesian approach," *Journal of Network and Computer Applications*, vol. 65, pp. 144–154, Apr. 2016.

[39] S. S. Masoumzadeh and H. Hlavacs, "Integrating VM selection criteria in distributed dynamic VM consolidation using fuzzy q-learning," in *IEEE International Conference on Network and Service Management (CNSM)*, Zurich, Switzerland, 2013, pp. 332–338. doi: 10.1109/CNSM.2013.6727854.

[40] Y. Jin, Y. Wen, and H. Hu, "Minimizing monetary cost via cloud clone migration in multi-screen cloud social tv system," in *IEEE Global Communications Conference (GLOBECOM)*, Atlanta, USA, 2013, pp. 1747–1752. doi: 10.1109/GLOCOM.2013.6831326.

[41] Y. Jin, Y. Wen, H. Hu, and M. Montpetit, "Reducing operational costs in cloud social TV: an opportunity for cloud cloning," *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1739–1751, Oct. 2014. doi: 10.1109/TMM.2014.2329370.

[42] N. Liu, Z. Li, J. Xu, et al. (2017, Mar. 13). A hierarchical framework of cloud resource allocation and power management using deep reinforcement learning [Online]. Available: https://arxiv.org/abs/1703.04221

[43] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proc. 7th ACM SIGCOMM Conference on Internet Measurement*, San Diego, USA, 2007, pp. 1–14. doi: 10.1145/1298306.1298309.

[44] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Communications of the ACM*, vol. 53, no. 8, pp. 80–88, 2010.

[45] Y. Borghol, S. Ardon, N. Carlsson, D. Eager, and A. Mahanti, "The untold story of the clones: content-agnostic factors that impact youtube video popularity," in *Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Beijing, China, 2012, pp. 1186–1194. doi: 10.1145/2339530.2339717.

[46] W. Chu and S.-T. Park, "Personalized recommendation on dynamic content using predictive bilinear models," in *Proc. 18th International Conference on World Wide Web*, Madrid, Spain, 2009, pp. 691–700. doi: 10.1145/1526709.1526802.

[47] M. Ahmed, S. Spagna, F. Huici, and S. Niccolini, "A peek into the future: Predicting the evolution of popularity in user generated content," in *Proc. Sixth ACM International Conference on Web search and Data Mining*, Rome, Italy, 2013, pp. 607–616. doi: 10.1145/2433396.2433473.

[48] X. Ma, H. Wang, H. Li, J. Liu, and H. Jiang, "Exploring sharing patterns for video recommendation on youtube-like social media," *Multimedia Systems*, vol. 20, no. 6, pp. 675–691, 2014. doi: 10.1007/s00530-013-0309-1.

[49] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proc. 10th ACM Conference on Recommender Systems*, Boston, USA, 2016, pp. 191–198. doi: 10.1145/2959100.2959190.

[50] G. Pallis and A. Vakali, "Insight and perspectives for content delivery networks," *Communications of the ACM*, vol. 49, no. 1, pp. 101–106, 2006.

[51] P. Venketesh, R. Venkatesan, and L. Arunprakash, "Semantic web prefetching scheme using naïve bayes classifier," *International Journal of Computer Science and Applications*, vol. 7, no. 1, pp. 66–78, 2010.

[52] W. Ali, S. M. Shamsuddin, and A. S. Ismail, "Intelligent na¨ıve bayes- based approaches for web proxy caching," *Knowledge-Based Systems*, vol. 31, pp. 162–175, 2012.

[53] T. Yan, D. Chu, D. Ganesan, A. Kansal, and J. Liu, "Fast app launching for mobile devices using predictive user context," in *Proc. 10th International Conference on Mobile Systems, Applications, and Services*, Low Wood Bay, UK, 2012, pp. 113–126. doi: 10.1145/2307636.2307648.

[54] H. Hu, Y. Wen, T.-S. Chua, et al., "Joint content replication and request routing for social video distribution over cloud CDN: a community clustering method," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 7, pp. 1320–1333, Jul. 2016. doi: 10.1109/TCSVT.2015.2455712.

[55] W. Hu, Y. Jin, Y. Wen, Z. Wang, and L. Sun, "Towards wi-fi AP-assisted content prefetching for on-demand TV series: a learning-based approach," *IEEE Transactions on Circuits and Systems for Video Tech- nology*, vol. 28, no. 7, pp.

1665–1676, Jul. 2017. doi: 10.1109/TCSVT.2017.2684302.

[56] M. Covell, M. Arjovsky, Y.-C. Lin, and A. Kokaram, "Optimizing transcoder quality targets using a neural network with an embedded bitrate model," *Electronic Imaging*, vol. 2016, no. 2, pp. 1–7, 2016. doi: 10.2352/ISSN.2470-1173.2016.2.VIPC-237.

[57] P. P. Dash, A. Mishra, and A. Wong. (2016, Sept. 22). Deep quality: a deep no-reference quality assessment system [Online]. Available: https://arxiv.org/abs/1609.07170v1

[58] H. Zhang, H. Hu, G. Gao, Y. Wen, and K. Guan. (2018, Apr. 10). Deepqoe: a unified framework for learning to predict QoE [Online]. Available: https://arxiv.org/abs/1804.03481

[59] V. D'Alessandro, B. Park, L. Romano, *et al.*, "Scalable network traffic classification using distributed support vector machines," in *IEEE 8th International Conference on Cloud Computing (CLOUD)*, New York, USA, 2015, pp. 1008–1012. doi: 10.1109/CLOUD.2015.138.

[60] L. Vu, C. T. Bui, and Q. U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proc. Eighth International Symposium on Information and Communication Technology*, Nha Trang, Vietnam, 2017, pp. 333–339.

[61] J. Ran, X. Kong, G. Lin, D. Yuan, and H. Hu, "A self-adaptive network traffic classification system with unknown flow detection," in *3rd IEEE International Conference on Computer and Communications (ICCC)*, Chengdu, China, 2017, pp. 1215–1220.

[62] J. Hochst, L. Baumgartner, M. Hollick, and B. Freisleben, "Unsupervised traffic flow classification using a neural autoencoder," in *IEEE 42nd Conference on Local Computer Networks (LCN)*, Singapore, Singapore, 2017, pp. 523–526. doi: 10.1109/LCN.2017.57.

[63] Y. Jin and Y. Wen, "When cloud media meets network function virtualization: challenges and applications," *IEEE MultiMedia*, vol. 24, no. 3, pp. 72–82, 2017. doi: 10.1109/MMUL.2017.3051519.

[64] H. Jagadish, J. Gehrke, A. Labrinidis, et al., "Big data and its technical challenges," *Communications of the ACM*, vol. 57, no. 7, pp. 86–94, 2014. doi: 10.1145/2611567.

Manuscript received: 2018-04-05

Biographies

JIN Yichao (yjin3@ntu.edu.sg) received the B.S and M.S degree from Nanjing University of Posts and Telecommunications (NUPT), China, in 2008 and 2011 respectively, and Ph.D degree from School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore, in 2016. His research interests are cloud computing and multimedia network.

WEN Yonggang (ygwen@ntu.edu.sg) is an associate professor with School of Computer Science and Engineering at Nanyang Technological University, Singapore. He received his PhD degree in Electrical Engineering and Computer Science (minor in Western Literature) from Massachusetts Institute of Technology (MIT), Cambridge, USA. Previously he has worked in Cisco to lead product development in content delivery network, which had a revenue impact of 3 Billion US dollars globally. Dr. Wen has published over 150 papers in top journals and prestigious conferences. His research interests include cloud computing, green data center, big data analytics, multimedia network and mobile computing.