ENERGY-EFFICIENT CACHE LOCALIZATION IN DEVICE-TO-DEVICE NETWORK

Thesis submitted in partial fulfilment of the requirements for the award of degree of

Master of Engineering

in

Computer Science and Engineering

Submitted By

DIVYA PRERNA

(801732010)

Under the supervision of:

Dr. Rajkumar Tekchandani

Assistant Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY

PATIALA – 147004

June 2019

CERTIFICATE

I hereby declare that the work that is being presented in this thesis entitled, "Energy-efficient Cache Localization in Device-to-Device Network", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Rajkumar Tekchandani and refers other researcher's work which is duly listed in the reference section.

The matter presented in the thesis has not been for award of any other degree of this or any other University.

Divya Prerna

801732010

ME-CSE

This is to certify that the above statement framed by the student is correct and true to the best of my knowledge.

Dr. Rajkumar Tekchandani

Assistant Professor

Computer Science and Engineering Department

TIET, Patiala

ii

ACKNOWLEDGEMENT

I hereby declare that the work that is being presented in this thesis entitled, "Energy-efficient Cache Localization in Device-to-Device Network", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Rajkumar Tekchandani and refers other researcher's work which is duly listed in the reference section.

The matter presented in the thesis has not been for award of any other degree of this or any other University.

Divya Prerna

801732010

ME-CSE

This is to certify that the above statement framed by the student is correct and true to the best of my knowledge.

Dr. Rajkumar Tekchandani

Assistant Professor

Computer Science and Engineering Department

TIET, Patiala

ABSTRACT

Last few years has seen one of the most important challenges for smart devices that is resource allocation. This has been a major bottleneck as there have been advancements in the cellular technology, especially 5G. With the increase in the enormous amount of number of users, there has been an increase in the demand of services and content within fraction of seconds. This has increased the burden on the network with respect to QoS and QoE to the end users and service providers. Caching the most popular contents on the user's equipments (UE's) can solve the problems because contents will be nearer to the users and can be shared using Device-to-Device (D2D) communication without the involvement of the core network. Motivated by this fact, in this thesis we present an extensive survey on the present caching techniques for D2D communication in 5G. Further, we present a model to address the problem of cache location decision. First, we find form a network of users using Matlab. Then we create dataset for predicting the cache locations in a network. Further, we predict the location where the user caches the most accessed content using machine learning classification models. The classification models used are decision tree and random forest. We compare the results obtained from decision tree and random forest classification model. On comparison we observe that the random forest model considering the trust factor between the users has higher accuracy. Then we use iFogSim simulator, to simulate the sending and receiving of contents in the network in a Software Defined Network environment. The metrics used for evaluating the simulations are access delay and energy consumption of the user's equipment (UE's). On analyzing the obtained result from simulation, we observe that the access delay is maximum at the users end when the sharing is with the gateway and energy consumption of the UE's is also the maximum when the contents are accessed from the gateway.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

D2D	Device-to-Device
UE	User Equipment
BS	Base Station
SBS	Small Base Station
MBS	Macro Base Station
WSN	Wireless Sensor Networks
HetNet	Heterogeneous Network
P2P	Peer-to-Peer
MWSN	Mobile Wireless Networks
QoS	Quality of Service
MEC	Mobile Edge Computing
MANET	Mobile Ad Hoc Network
MAC	Medium Access Protocol
MIMO	Multiple Input Multiple Output
CSV	Comma Separated Values
SDN	Software Defined Network
IoT	Internet of Things
JDK	Java Development Kit
IDE	Integrated Development Environment

1. INTRODUCTION

There has been an enormous increase in the wireless data traffic and is still growing many folds. One of the main reasons for this data expansion is the increase in the number of smart connected devices and data specific applications. This tends to add to the amount of data that is generated. Keeping under consideration these issues, the telecom service providers are facing great difficulties in keeping up with demands of the mobile users and also providing QoS to the mobile users. Hence, the traditional cellular networks need to be enhanced with respect to the bandwidth provisioning in order to provide increased data rates to the mobile users. The upcoming generation of cellular network, 5G, is based on a infrastructure which is expected to fulfill the demands of the mobile users by preserving the QoS and reducing the latency. Fig. 1 [1] shows the increase in the number of social media users in the span of next 3 years.

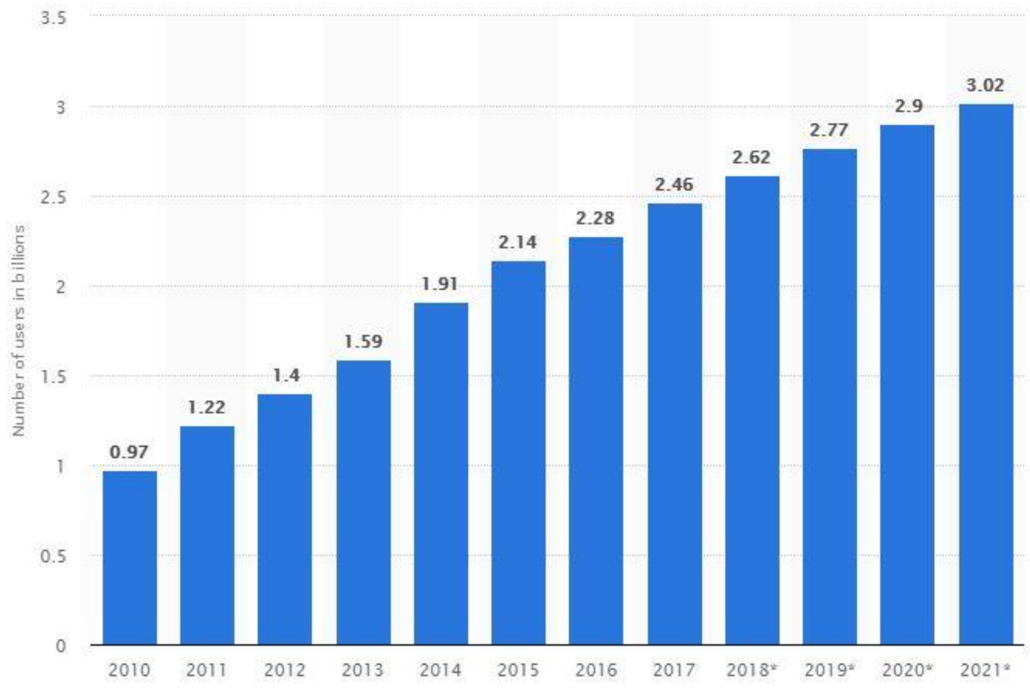

Figure 1: Trend of growth of users [1]

The main reason for this enormous increase in the amount data being generated is the increase in the number of connected devices. For example, a huge amount of data is generated from the social media platform, on-the-top processing units, and other smart devices which are increasing at a very fast pace. Popularity of content and data volume makes a huge impact on the contents that can be cached on the mobile devices. Recently, we have seen an increase in on-the-top (OTT) content. Service that makes available audio and video contents to the smart users streaming over the internet like, YouTube, Amazon Prime is called as OTT services. The use of these services has increased the interest of users towards it and also has led to an increase in the number of data intensive applications. For example, Netflix is one such service that provides subscriptions to the users to access the content on different platforms.

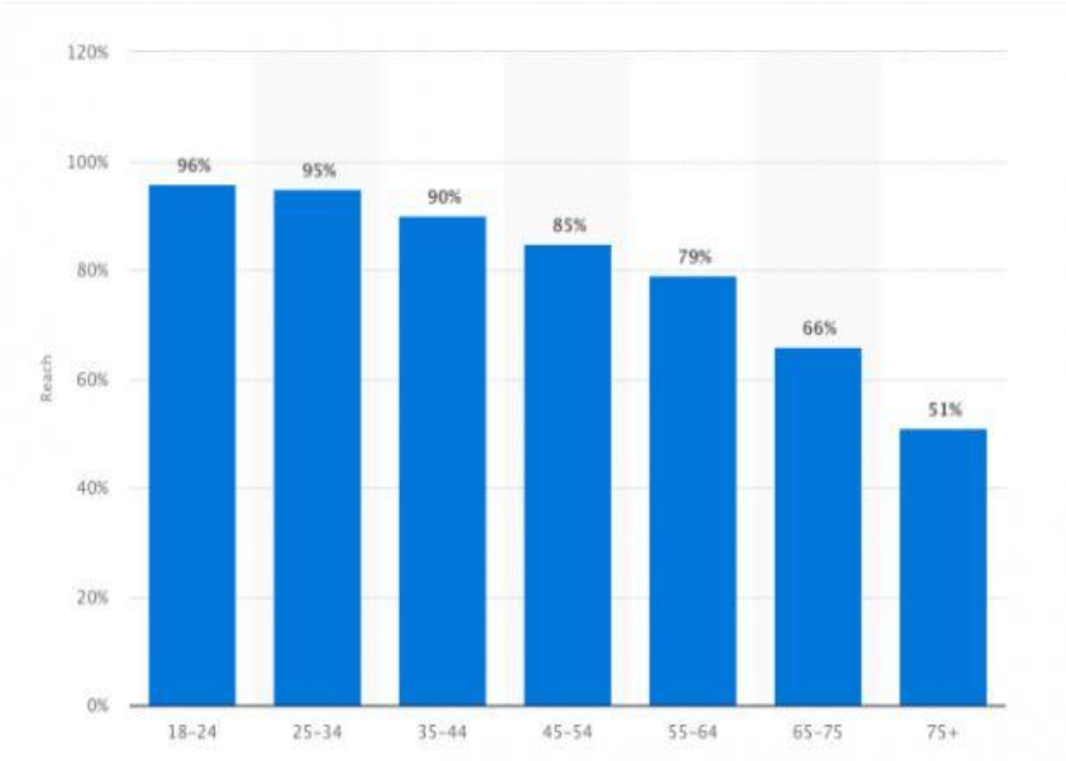

Figure 2: Age group of YouTube users in one month [2]

The challenges faced in 5G are high bandwidth, low latency, high data rates, Device-to-Device (D2D) communication and increase in the number of connected devices. D2D treats the mobile user's devices as data centres for sharing content. D2D communication enables the mobiles users in close proximity to connect directly connect with each other without going back to form a connection with backhaul. Backhaul in telecommunication network is defined as an intermediate link network between the core network and the small stations at the edge of the network. Thus, in order to reduce the traffic on the core network, caching at the mobile user's device for sharing content via D2D communication becomes unavoidable.

Caching the most popular contents and services at the edge of the user tends to bring content and end user nearer. As the mobile user's device have sufficiently large amount of storage spaces, so using the D2D communication link, the users can exchange the popular contents with other users in the network directly. But, there is a challenge in deploying D2D communication network that is discovery of devices. The discovery of devices can be performed in two ways that is by using either distributed or centralized technique. [3][4][5]. In contrast to D2D, the traditional communication in cellular networks involved the connectivity with the core network irrespective of the fact that the two users intending to share content are in proximity to perform sharing over D2D communication. Moreover, the 5G cellular network's key enabler is D2D communication which enables the UE's to fetch and share the cached content via direct wireless communication link, instead of communicating through the core network. Before the emergence of D2D wireless communication, there were different wireless technologies like WiFi and Bluetooth which allowed users to perform short range communications. These technologies may help in enabling D2D communication.

Figure 3: Wireless technologies in cellular networks [6]

These technologies are different from each other with respect to distance between the two communicating equipments and data rates provided. Bluetooth 4 provides 25 Mbps data rate and allows coverage of 100m, WiFi Direct supports a data rate of 250 Mbps with coverage of upto 200m and LTE has a data rate of 14 Mbps and range upto 500m [7].

	Bluetooth®	Wi-Fi	
Frequency Band	2.4GHz	2.4GHz	5.0GHz
Communication Distance	~10m	Several hundred m	
Transmission Speed	1Mbps~	11Mbps~	
Available Channels	79	13	19
Transmission Time Limit	No	No	
Modulation Method	FSK/PSK	DSSS/OFDM	

Figure 4: Features of wireless technologies [7]

Hence, popularity of content and the data's volume are considered and modelled to recognize which content should be cached such that the storage spaces of the mobile user's device are utilized efficiently. Along with improving the spectral reuse, D2D communication technology also provides reduced latency, optimal energy consumption, and improves overall throughput of the system. Thus, D2D communication technology proves to be a key enabler in the upcoming 5G of the cellular networks ensuring reduced latency among the mobile users.

1.1 Scope

The existing cellular networks make use of the core network for accessing and sharing contents between mobile devices. So, the increased demand of high data rates leads to an overall increase in latency and traffic on the network. To reduce the delay and pressure on the network, the concept of caching at the mobile user's device is implemented in 5G cellular network. It would bring the mobile user nearer to the content which will enable the user to share contents through D2D communication which reduce the burden on the network and minimize delay. There are different parameters that are considered for caching contents in 5G. They are as:

1. Popularity- Popularity of content is evaluated and similar popularity contents are cached together in clusters.
2. Cooperative - All the UE's in the network collectively perform caching.
3. Hierarchical - There are different levels in the cache. Communication is established between levels whenever a file is to be fetched or cached.
4. Social- The caching is done based on the friendliness and behaviour of the user. While caching, the behaviour of the user is observed.
5. Competitive - The contents are cached based on a price which is put by the operator.

Caching at mobile devices in 5G distributes the load of the core network and minimizes the pressure of the network at the peak hours. But there are various constraints while content caching as:

1. Mobility speed - As the users keep moving, the mobility speed of the user plays an important role in determining the contact time of the user and cached content. Thus it makes a hug impact on the caching mechanism with respect to the cache hit ratio.
2. Social network - The general behaviour of users is to connect with friends or users whom they trust. So, social awareness plays a significant role in caching contents.
3. Storage - The caching space of the user's device plays a significant role while caching in 5G. Although, there are large storage spaces available at low prices but are finite. Therefore, the caching technique used should utilize the space efficiently.
4. Energy consumption - As 5G increases the content sharing and the number of connection, this tends to drains the battery of the device. Thus, an energy efficient caching technique is required to limit the power consumption of the device.

5G is said to increase the number of connected devices using D2D communication link. D2D enables two users to connect directly without the core network. Caching at the UE's would encourage the users to perform sharing of contents via D2D communication. But, there are challenges such as, load balancing, content offloading decisions, privacy and security of the content. As D2D enables direct communication, the user's content and all files are prone to threat. Also, there are possibilities of too many files cached at one users and very few at another. Resource allocation and consumption is also a major challenge in 5G. The best solution mentioned in the existing literature, is to allocate resources according to the channel, traffic and interference in the network.

1.2 Objective

1. Predict the cache locations in a network of users connected through wireless networks using classification models.
2. A comparative analysis of the results obtained from different classification models on the basis of accuracy.
3. Simulate the network for sharing of contents such that it yields minimum latency and minimum energy consumption.

1.3 Outline of the Thesis

Chapter-2: Illustrates the survey done on caching techniques in 5G. We compared the existing surveys and further this section talks about the constraints and elaborates them.

Chapter-3: This section talks about the review questions the reason behind analyzing the question, the gap identified and the problem statement addressed.

Chapter-4: In this section we talk about the approach used for solving the problem, the algorithm used and execution details.

Chapter-5: We show the obtained results and output in this section.

Chapter-6: This section includes the conclusion and the future scope of the proposed approach.

2. LITERATURE REVIEW

2.1 Background

Caching techniques for D2D in 5G is categorized into three types. Synchronous caching, asynchronous caching and hybrid technique for content caching as illustrated in Fig. 3. These techniques are also addressed as cache consistency techniques. Synchronous caching involves the BS that maintains all the information of the connected mobile user devices. Also, it broadcasts messages in the network for invalidating the information of each user. No mobile user can request to access the cache before the next broadcast message is received. In asynchronous caching, it does not have any information about the mobile devices connected but broadcasts messages for each entry of content. The hybrid technique is a combination of synchronous and asynchronous techniques. It stores information of mobile devices in its cell but no mobile device is allowed to request for a cache access until an acknowledgment is received.

Figure 5: Types of caching

2.1.1 Advantages

In synchronous caching, the downlink is utilized efficiently because the traffic volume is less and effectively manages the sleep-wake-up trend of the mobile devices. But, this technique increases the average latency due to the reason the mobile device is in waiting phase until the next broadcast message is received. On the other hand, asynchronous caching increases the amount of traffic in the downlink but has low average latency. The reason being that the BS needs to send acknowledgement very frequently which sometimes is unnecessary. The hybrid technique takes the advantage from both, synchronous and asynchronous technique. It minimizes latency and performs bandwidth saving.

2.2 Existing Systems

There are many survey articles in the literature which cover caching of the contents and sharing through D2D communication. Wang et al. [8] analyzed the utilization of caching, computation and communication resources in mobile edge networks. The objective is to summarize computation, communication and caching resource utilization of MEC. The authors explored MEC, its key enablers and applications. Mao et al. [9] analyzed the edge computing with respect to communication with respect to latency and energy parameters of the network. The objective is to explore MEC with respect to communication. The authors only considered latency and energy parameters. But, effectively analyzes the resource management. Zhang et al. [10] carried out a comparison between different architectures of computational offloading in mobile cloud computing (MCC). The objective is to consider the offloading probability of MCC. Anugraha et al. [11] analyzed different trust management approaches in Mobile Ad Hoc Networks (MANETs). It aims at enhancing the performance by dealing with the selfish and maliciously affected devices. The authors do not consider a secured routing scheme in the network.

Next, we categorize the literature on the basis of four parameters that form the pillars of content caching for D2D communication in 5G. The parameter on which the classification is done are communication and connectivity of the mobile users, the type of storage used for caching, quality of experience gained by the end user and the type of trust between two mobile users.

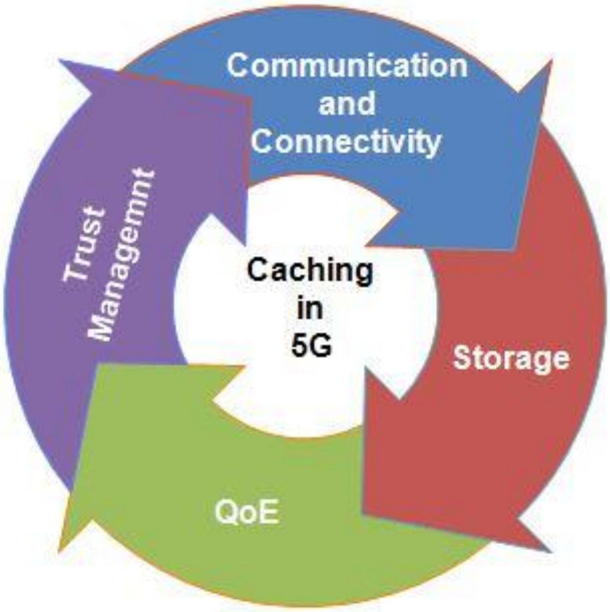

Figure 6: Parameters for caching in 5G

Communication and Connectivity - Communication and Connectivity for D2D content caching in 5G is distinguished in three parts as connectivity, communication and deployment model. The mobile users in the network make groups to perform communication. The groups formed are called clusters and the communication can be peer-to-peer (P2P), in this the sharing of content is between two mobile users in a cluster. Cluster can have different type constitution depending on the types of users in the cluster. If there are different types of mobile devices using different technologies are called as heterogeneous network (HeNet). Other cluster types include micro cell, micro cell and homogeneous network. Micro cell is defined as a cellular data centre that has small coverage area, macro cell which has larger area coverage than micro cell. A group similar mobile device is named as homogeneous networks. The mobile users in a network can be deployed uniformly, randomly or stochastically. The wireless communication can be further performed in two ways as cooperatively and collaboratively.

Storage- Different types of data structures are used for caching the contents in the UE's. The various types of data structure used in the literature are hash tables, linear model, undirected and directed graphs. One of the proposed approach used depth first search traversal for fetching data in undirected graph.

QoS- The QoS for the end user can be enhanced by reducing the power consumption of the UE's, the overall cost for the user, and the delay while retrieving the content.

Trust Management- Trust management plays a significant role in preserving the privacy and security of the mobile user. Broadly three types of trust management approaches is presented in the existing work. Direct trust between two mobile users, indirect trust based on recommendation and hybrid method which a combination of direct and indirect approach. This approach helps in establishing trust considering direct trust value and also recommendation.

2.2.1 Communication and connectivity

The growth of data specific applications and connected smart devices has led to the birth of the concept of direct link between two end users communicating with one another for sharing content. The mobile users that take part in the wireless sharing of content are deployed within clusters as micro cell, macro cell, HetNet and homogeneous network. With respect to the type of network and coverage area of the network, a connection can be established via various types of wireless connectivity, for example, D2D communication.

Ji et al. [12] reviewed in a compact form the throughput of the system with respect to asynchronous reuse of content. In the simulation of the problem, the authors considered a randomly distributed cluster of users forming a network. Random distribution is chosen due to its low cost. The type of communication used by the users is D2D communication. After the simulation of the network, the obtained results illustrate the system helps in efficiently utilizing the storage resources. Li et al. [13] created a strategy that focuses on cache replacement and named the strategy as Energy-efficient Coordinated cache Replacement Problem (ECORP). The cache replacement problem is formulated as a peer-to-peer (P2P) model. All the users in this P2P network are deployed randomly for exchanging contents via wireless cooperative communication link. Exchanging contents with mobile devices that are not present in the coverage area is performed by multi-hoping devices till the nearest BS is encountered. The obtained results depict that the system increases the number of cache hit. Wang et al. [14] focussed on the problem of resource allocation for a randomly deployed HetNet. The communication medium chosen is wireless D2D communication. The obtained results show a very small deviation from centralized techniques. Wang et al. [15] presented a framework keeping in consideration MEC and data centre (DC). The authors simulated the framework for a macro cell deployed randomly and communicating cooperatively. The proposed framework uses Genetic algorithm (GA) and Simulated Annealing (SA) for predicting caching and offloading decision. The framework obtains a steady state and when compared with traditional frameworks, it performs better. Liu et al. [16] presented a blockchain architecture for a MEC-enabled wireless network. The authors simulate the architecture in a randomly deployed HetNet with D2D communication. The obtained results show that there is a very narrow gap between the proposed architecture and centralized. Zhao et al. [17] addressed MEC based server that has the capability of cloud for caching the results of the computation tasks. The authors address the resource allocation problem under the constraints of storage and radio computation constraints. The authors proposed to deploy the network randomly and communicate through collaborative wireless link. A fluctuation is observed in the transmission rates under when simulated considering real environment parameters. The results show the system is effective when compared with other baseline systems. The three baseline systems considered are as proposed system without caching, local computation and cloud computing. Park et al. [18] explored cooperatively caching of contents using CCN technology. The authors formulated traffic offloading using X2 link for supporting video streaming. While simulating results the network is uniformly deployed with D2D communication. The proposed system was compared with five similar existing techniques. The existing techniques are broadly categorized as non-cooperative and cooperative caching techniques. Least recently used (LRU), least frequently used (LFU), and MPC are non-cooperative techniques. Extensive cooperation caching (EC-Caching) and caching as clusters are cooperative caching techniques. All the existing techniques address traffic offloading decisions. The results show that the presented technique minimizes the traffic upto 13%. Wang et al. [19] presented an incentive framework for caching. The authors consider mobility speed in order to increase the utility and minimize the cost of the base station. The network chosen is deployed randomly in a wireless HetNet. The presented framework is compared with greedy, random and fair caching. The obtained results prove the superiority of over other techniques under similar environment. Zhan et al. [20] performed network coding to implement caching in a HetNet. The cluster is uniformly deployed and communicates using wireless links. The proposed solution is compared with two old approaches as Femtocaching and uncoded caching approach considering mobility. On comparing with the old approaches, the simulation results depicted that the proposed solution downloads the maximum number of packets from small base stations. Naderializadeh et al. [21] constructed an algorithm addressing the problem of communication scheduling. The problem is based on an information-theoretic optimality rule (ITLinQ). The simulation environment is a HetNet which deployed uniformly. The obtained numerical results show that the proposed solution improves the throughput gains. Xu et al. [22] proposed a theoretic model. The problem is formulated using a game. The model consists of edge nodes that saves energy for green cities and makes sure that the content delivery takes place through a secure channel. The simulation environment is a cooperative macro cell randomly deployed. The obtained results depict significant reduction of average time of execution of the proposed game theoretic approach. Sivashankari et al. [23] proposed a model for detecting compromise attack in the early stage to preserve the privacy and authenticity of the locations at which the source and sink nodes are present. The problem is simulated in a sensor network comprised of sensor nodes. The nodes communicate with one another via wireless communication links. The sensor nodes are deployed uniformly in the sensor network. The analytical results illustrate that the proposed model works efficiently in the presence of a global attacker. Zhang et al. [24] presented a technique that detects the beacon nodes by evaluating the trust by local detection. The simulation network is a sensor network. It is deployed randomly by sensor nodes. The obtained results reveal that the proposed technique performs accurately for nodes that are not known to the network.

2.2.2 Storage

With the increase in the number of smart devices and exhaustive services that are being provided through the cellular networks tends to increase the storage demands of the mobile users many folds. Although many storage devices are easily available that provide larger storage spaces at very low price but, the space still remains finite. With increasing demand of content, the storage scarcity will be a challenge. Thus, to efficiently utilizing the storage resources play a very important role in 5G. Different types data structures have been used for efficiently storing contents in mobile devices of the users.

Ji et al. [12] presented some outcomes of throughput of wireless networks along with asynchronous reuse of content. The contents are stored linearly in the UE and depend on the throughput of the system communicating via D2D communication. Hence, the obtained results show that even for small outage the throughput scales very rapidly. Luo et al. [25] discussed an existing approach B-REMiT which aims at minimizing energy and is said to increase the lifetime of the system. The approach used in development of B-REMiT is a distributed algorithm formed by creating multi-cast group shared trees in Ad Hoc networks. The obtained results depict that the total energy consumed is minimized and lifetime of system is extended by B-REMiT. Deb et al. [26] presented a technique for improving the lookups in P2P networks. The technique uses distributed hash tables (DHT) for storing contents. The improved lookups are achieved by caching the neighbours with respect to their access frequencies. The results prove that the proposed technique is scalable and efficient in improving the look-up time of the P2P model. Naderializadeh et al. [27] analyzed an existing scheduling technique based on information-theoretic link (ITLinQ). The technique aims at scheduling users in a D2D network. The authors proposed to improve the performance of ITLinQ. This is done by efficiently utilizing the storage resources at the user ends. Further, the memory utilization of UE is achieved by an association mechanism with the end users. This association approach is formulated as a greedy closest-source scheme using undirected graph. The obtained result is compared with a conventional approach which focuses on avoiding interference amoung large number of users. The results show higher gains in throughput.

2.2.3 QoS

The key enabler of 5G cellular network being D2D technology, it has the advantages of minimizing latency for increased traffic volumes and high data rates. Irrespective of the promises and advantages of 5G, one of the major challenges is to cater he needs of the end user by meeting their requirements. One such requirement is to maintain the QoS of the services delivered to the mobile users by the telecom operators. The QoS can be maintained by performing content delivery with minimized latency in traffic hours, reducing the battery drainage of the UE's, optimally using the resources and minimizing the cost of the network to be incurred by the users.

Datsika et al. [28] presented a D2D network using medium access protocol (MAC). The constraints applied to the network are social-awareness which contains a social data. The authors develop an energy efficient model for D2D MAC design. The design is called as Social-aware Cooperative D2D MAC Protocol (SCD2D-MAC). The objective is to focus on the real issues of social-aware network. The amount of energy consumed by the system is the performance metric. The performance of the proposed scheme varies with the number of friendly transmissions. This is because less number of rounds have to explored cooperatively by the unknown users in the network. The proposed scheme is compared with two protocols. (1) ACNC-MAC which has the objective to reduce channel access issues. The network chosen is a D2D hierarchical network. It uses the approach of relaying cooperatively. (2) NCCARQ-MAC addresses the problem of transmission coordination in a wireless network. The approach used is ARQ bases coded cooperative caching. The obtained results illustrate that SCD2D-MAC reduces the draining of battery of UE’s by 44% and 58% when compared to ACNC-MAC and NCCARQ-MAC approach, respectively.

Jia et al. [29] presented a coded caching approach cooperative in nature. The author’s objective is to improve the delivery of content and provide improved QoS to the end users. To formulate the approach, a framework is proposed with the aim of optimizing the consumption of energy consumption while caching contents. A greedy algorithm is developed with the objective of optimizing the placement of cache to ensure energy efficiency. Energy is the performance metric, which depicts that the total consumption of energy varies and the trend is observed which reveals that initially the consumption increases but tends to minimize as the number of SBS are increased. Moreover, the results show that the total consumption of energy is increased with the increase in the requests. The proposed mechanism succeeds in saving the overall consumed energy, thereby, enhancing the performance of the system. Chao et al. [30] developed an interface for accessing contents that are cached. The approach used to access the content is through dynamic backing-up methodology of routing protocols. A cache sharing interface is used for configuring a mobile node. This approach minimizes the power consumption and usage of bandwidth by the network. Moreover, the proposed approach performs better than other routing protocols.

Malak et al. [31] addressed distributed caching in a D2D network. The objective of the authors is to increase the probability of delivery content. The problem is formulated by Poisson Point Process which models the devices used for sending and receiving the contents. Stochastic geometry is used for obtaining results from the system. The results show a variation with respect to different noise levels. A relation is obtained between the demand for the content and caching distribution for the contents. Huang et al. [32] developed caching scheme that is multicast aware and cooperative in nature. The scheme is named as Cooperative Multicast-Aware Caching (CMAC). Moreover, the objective of the authors is to increase the hit ratio and number of successful delivery of content and also to minimize the delay. QoS and access latency are the performance metrics. The results reveal that CMAC gives better performance when compared to MAC. CMAC minimizes the average access delay by 13% when simulating under similar conditions of environment.

Zhang et al. [33] proposed to implement hierarchical caching. The constraints considered in the implementation of the system are mobility speed of smart vehicles. The objective of the authors is to improve the utilization of resources in the network. The formulation of the problem is done by the concept of edge caching of contents in vehicles by the assistance of cache-enabled cloud. The role of cloud caching is to jointly schedule the allocation of caching and computing resources at the edge of the wireless network. Latency is the performance metric of the network. The numerical results of the proposed system are compared by the trace-based simulator's results. The comparative analysis yields that the average difference between the two obtained results is of 1.105 sec and 2.073 sec when different number of content types as 10 and 20 respectively are chosen. Thus, the proposed system minimizes the download latency and effectively improves resource consumption.

Mehamel et al. [34] developed a fuzzy mechanism for caching contents. The objective of the authors is to develop an energy-efficient mechanism considering mobile users and a resource constrained environment. The proposed mechanism uses Field-Programmable Gate Array (FPGA) which focuses on minimizing the power required by the mechanism. The FPGA is a hardware implementation that is believed to consume 45W energy. The results illustrate that the proposed hardware implementation more energy-efficient than most of the software implementation addressing similar problems. Cui et al. [35] addressed the problem of caching and offloading. The authors also included upload and download of uncached results of computation tasks of all the tasks executed at the BS. The objective of the authors is to reduce the overall energy consumption and perform efficient resource allocation of storage resources. The simulation environment takes latency and caching as constraints. The proposed solution is compared with existing solutions to the problem. The obtained results show that the proposed solution comes out to be best amoung the existing solutions with respect to attaining optimal and sub-optimal states for resource allocation of communication and storage resources. The advantage of this approach is that it ignores redundant transmissions. Liang et al. [36] presented a caching mechanism named as MEC based caching with multiple sources (MECC-MS). The mechanism is solved using source selection problem. The problem is decoupled from provisioned bandwidth through dual method of decomposition. Further, an ADMM based approach is developed by the authors to address the decoupled problem by establish coordination between links and BSs. The proposed mechanism is compared with four approaches. (1) MECC with single source (MECC-SS), (2) MECC at SBS (MECC-SBS), (3) MECC at MBS (MECC-MBS) and (4) without MECC. The obtained results yields that MECC-MS improve the energy efficiency.

Hao et al. [37] addressed caching and offloading of completed computation tasks on the edge cloud. The simulation environment is kept resource constrained. The authors developed a task caching and offloading (TCO) algorithm. The algorithm is formulated as mixed integer problem by iterative alternating programming. The result of TCO is compared with three approaches. (1) task popular caching and offloading (TPO) which caches the tasks at the edge cloud, (2) task random caching and offloading (TRO) which caches tasks randomly on the edge cloud and (3) task femtocaching and offloading (TFO) which caches only those tasks which shows maximum reduction in energy consumed at the edge cloud. The caching of the tasks are done upto the maximum capacity of the edge cloud's storage. The results obtained after the comparative analysis show that TCO reduces the maximum amount of energy than the other approaches.

Yu et al. [38] addressed the problem of caching and offloading. The authors developed an optimal fine grained offloading and caching technique (OOCS) in order to reduce the overall delay for the users. The proposed technique is compared with six other techniques. (1) The No objection scheme (NOS), (2) optimal offloading without caching (OOS), (3) total offloading without caching (TOS), (4) total offloading and caching scheme (TOCS), (5) D2D offloading and caching (DOCS) and (6) D2D offloading without caching (DOS). The proposed technique is simulated in two types of environment as (1) high-end and (2) low-end deployment. High-end deployment refers to that state of the network which consists of standard servers as access points and low-end deployment consists of routers acting as servers. The obtained results yield that under the high-end setup, OOCS yields the best result with respect to latency by showing reduction upto 33.28%, 28.04%, 32.31%, 15.42%, 11.89%, 29.19% in comparison to NOS, OOS, TOS, TOCS, DOCS and DOS, respectively. In low-end scenario OOCS again has the best performance in reducing latency upto 42.83%, 36.49%, 41.51%, 12.25%, 24.50% and 39.33% in comparison to NOS, OOS, TOS, TOCS, DOCS and DOS, respectively. In addition to this, the multi-user OOCS reduces latency by 11.71% as compared with single user OOCS.

Yeung et al. [39] focused on maintain cache consistency for UE's considering the minimum communication between the users. The authors proposed to solve the problem by a fully distributed protocol. The performance metric of the protocol is the power consumption of UE's. The authors also consider selfish clients in the network. The obtained results show that the proposed protocol increases the amount of queries completed successfully, increases the lifetime of the users in the network and reduces latency. The system is said to be in pure Nash Equilibrium. Nash equilibrium is defined as the stable state which the system attains in the presence of clients having selfish behaviour. Liu et al. [40] developed a caching framework. It is addresses as a fine grained proxy adaptive network which enables low cost scalable videos. The authors also developed a cache management algorithm which caches contents in an efficient and effective way. The proposed algorithm provides optimized streaming data rates to all users. The results of the proposed framework are in comparison with a replicated video streaming technique without proxy caching. The results obtained illustrate that the proposed framework minimizes the consumption of bandwidth and is flexible in nature when simulated in the presence of heterogeneous users. The reduction is bandwidth is by 40%-60%.

Naor et al. [41] addressed the last mile problem of video delivery. The last mile problem is defined as improving the quality of services delivered by telecom operators to the mobile users. The solution to the problem is done with the help of content aware caching dependent on the content popularity, by multicast enabled delivery of content. The obtained results are compared with content distribution in a user specific network. The proposed solution performs better with respect to delay, and increases the QoS for its users. Li et al. [42] presented a caching mechanism which uses more number of storage nodes for conserving the cost of repair and thereby, reducing the cost of video downloading. The mechanism is named as double replication maximum distance separable code (DR-MDS). The proposed mechanism performs better than the traditional MDS code approach on comparison. The obtained results illustrate that cost of communication tends to decrease when the number of users are increased. Further, it also reduces the total cost of communication by preserving the repair cost and cost of downloading videos. Guo et al. [43] presented a caching technique for D2D enabled wireless network. The caching is performed in two stages that are employment of MIMO and hierarchical cooperation. The authors addressed the problem of cooperative delivery of cached contents by developing a uncoded random cache placement technique. The proposed technique is compared with multi-hoping caching technique. The obtained numerical results show that the throughput increases linearly with the increase in the number of user nodes. The proposed technique said to outperform the multi-hoping caching technique when simulated in similar environments. Han et al. [44] addressed the issue of frequency reuse and caching. Moreover, the authors aim at removing interference in the channel when the content has not been cached at the nearest BS. The problem is formulated by developing an optimization algorithm which addresses cache storage allocation and frequency reuse. The numbers of successful transmissions is the performance metric. The simulation is done considering limited storage and backhaul constraints. The obtained results illustrate that the proposed solution is better than other techniques with respect to performance gain. Wu et al. [45] designed a protocol for improving the quality of the videos and also minimize the consumption of bandwidth. The protocol is named as bandwidth-efficient multipath streaming (BEMA). The protocol is formulated through a mathematical model to transmit videos concurrently. The approach used in the proposed protocol is Raptor coding and data distribution scheme. The performance metrics are PSNR, end-to-end delay and the number of successful transmissions of videos. BEMA offers resistance for change in traffic rate and bandwidth for delivering high quality videos. The obtained results depict 20% gains in performance when the proposed protocol is compared to multipath transport protocols.

Sinnwell et al. [46] addressed the problem of combinatorial optimization with respect to distributed caching in a network of workstations (NOW). The authors also developed a cost model which derives efficient online heuristics. The obtained results show that the proposed solution to the problem works efficiently and a very small overhead is incurred. Pinho et al. [47] presented an on-demand video system. The proposed system is named as Global Video Environment (GloVE) which is formed on the underlying technique of cooperative caching (CVC) of videos. CVC is said to be very similar to chaining and patching. The active users cooperate with each other to form a shared cache. The performance metrics are occupation rate and delay. The results obtained from experiments show that there is an reduction in bandwidth upto 90% and delay of less than 20 seconds. The proposed technique also requires only one channel for transmitting high definition videos.

Araldo et al. [48] explored the size of caches, object placement and path selection problems. The authors propose to optimize the problems such that it reduces the overall cost or increases the hit ratio. The solution to the problem is formulated as a greedy algorithm. It is believed to function in polynomial time. Matlab is used for simulating the results. The obtained results show that the proposed algorithm succeeds in saving 30% cost but compromises on the hit ratio by 60%.

2.2.4 Trust Management

The D2D communication technology allows the user to connect with each other directly. The users in close proximity are enabled to share contents via D2D link without going to the core network. This makes the user’s files and identity vulnerable. So, it is very necessary to make this connection secure and such that the user’s privacy is ensured. Thus, it is important to embed a reliable and efficient trust evaluation mechanism for sharing content in a network via D2D communication.

Sutaone et al. [49] proposed developed a mechanism for detecting faults and a recovery mechanism for MWSN. The mechanism is named as Trust-based Cluster head Validation and Outlier Detection Technique (TCVOD). The cluster heads that are malicious are detected by evaluating trust using progressive approach. The cluster head is then replaced using outlier detection mechanism. The proposed mechanism is compared with trust-based Cluster head Selection Algorithm (TCSA). The performance metrics considered are packet delivery ratio, packet drop and average energy consumption. The simulation results show that TCVOD performs better than TCSA and the numerical results shown that while detecting cluster heads there is 33% greater packet delivery ratio, 66% lesser packet drop and 25% less consumption of power. TCVOD yields 30% greater packet delivery ratio, 60% lesser packet drop and 27% consumed energy while identifying outliers. Alsaedi et al. [50] presented a mechanism to remove Sybil attacks in an energy efficient way. The authors developed a trust system considering a hierarchical wireless sensor networks. The trust is evaluated by detecting true positives and false positives. The obtained results depict a success rate of 87% in detecting sybil attacks. Ding et al. [51] explored a trust evaluation method for establishing trust between two unknown users in the network. The network considered is a dynamic Ad Hoc network. This approach makes use of γ - Quasi-Clique technique for trust graphs. The results obtained show that it is a reliable approach for establishing relationships for discrete entities. Abuzaid et al. [52] proposed a system which calculates reputation of the nodes to calculate the trust values and respectively contributes in isolating the nodes that performs badly. By isolating the malicious node from the network, the chance of it initiating an attack is reduced. The nodes are assigned different roles as janitor, jury which helps in dealing with different situations in the network. Moreover, a performance certificate circulates in the network with the nodes which contains time stamps. The time stamps helps in regulating most recent updates so that they are effectively retained. Saini et al. [53] presented a mechanism for observing and analyzing the behaviour of nodes for evaluation of trust values. The obtained trust values are dynamic in nature. A table is maintained which has the possible trust values and the degree of trust corresponding to the values. The results show that the proposed approach works by optimally choosing a secure path for forwarding procedures. Gowridurga et al. [54] proposed an SMS alert system based on trust to the authentic users. The proposed approach is named as Zigbee alert system. For sending and receiving SMS this approach combines together WSN and GSM technology. The WSN is responsible for communication between the PC and the sensor nodes in the network. The proposed approach is believed to have its application in military services and applications.

3. Problem Identification

3.1 Review Questions

The aim of this review was to analyze existing literature focusing on D2D caching techniques in 5G and identify a gap in the existing literature. To carry out a systematic review, certain research questions with their objectives are listed in Table 1.

Research Questions	**Objectives**
What is caching?	Storing frequently accessed content in a temporary storage for subsequent reuse.
Why caching mobile service?	To simplify the access pattern and use of frequently used services.
What is Device-to-Device Communication?	Creating a direct link between two users in close proximity.
What is advantage of D2D communication?	Reduces access delay and reduces the traffic in the network.
What are the features of 4G?	Autonomous network, independent software, diverse users, higher coverage.
What is LTE/LTE-A?	Long Term Evolution Its is a communication standard of mobile network. It provides speed, coverage over the internet.
What is Mobile Edge Computing?	Bringing the storage and computation of contents at the edge of the cellular network.
What are the features of 5G?	It promises to reduce latency, provide high data rates and increase the number of connected devices.
Why caching in 5G using D2D communication?	Caching at the UE's would bring the content nearer to the users and sharing using D2D will reduce latency. Thus, ensuring the advantages of D2D in 5G.

Table 1: Review questions

3.2 Gap identified

While content caching for D2D communication in 5G, one of the most important question that comes forward is ”where to cache the contents”. There are many existing work that focus on the D2D content caching technique in 5G. Moreover, there are various constraints which have been addressed in the existing literature as content offloading decisions, cost incurred, mobility speed of the user, storage capacity of the device, trust, energy/power consumption, latency, bandwidth provisioning and resource consumption. These constraints are explored in most of the research addressing content caching techniques. In this thesis, we aim at predicting the location where contents can be cached in 5G. In 5G, the contents can be cached at the UE’s, at the BS or at the gateway which is also called as the core network. We use machine learning classification models with multi-class to predict the location for caching contents. Further we evaluate the efficiency sharing contents in the network using latency and energy metrics.

3.3 Problem Statement

D2D content caching in 5G requires content to be cached nearer to the end users. This will reduce the latency and increase the number of hit ratios. Along with these parameters it should also be ensured that the technique consumes optimal amount of energy and the cost incurred is low. In the existing literature, the authors have proposed content caching techniques over D2D communication in 5G. In our work, we aim to locate the appropriate location for content caching in 5G such that it reduces latency, reduces the energy consumed and increases the accuracy when content sharing takes place over D2D communication. The prediction of cache locations is done on the basis of various parameters like, volume of the content requested, the storage size of the UE’s, mobility speed of the user, distance between the users. Moreover, for sharing content over D2D communication, the users need to connect directly with neighbouring users. The users prefer to connect with users whom they are friends with. So, another parameter trust is also considered in the prediction of the cache locations. The classification models used for predicting the cache locations are decision tree and random forest model.

4. Design and Implementation

4.1 Proposed Architecture

A typical framework for caching contents in 5G over D2D communication between the users comprises of smart mobile users like, mobile phones, vehicles, laptops, smart cities etc. The architecture is depicted in Fig. 8. It shows the three layers involved in the caching comprised of their respective functioning and peripheral network device. The lowest most layer, the base layer contains end users which forms clusters of various types and communicate through D2D communication. The middle layer called the edge layer consists of BS. They are responsible for caching contents at the edge of the network. Depending on the size of the BS, contents are cached. MBS's enables caching of those tasks which have caching and computation requirement at the edge of the network. SBS's cache tasks which have to be only stored and do not require any computation. The users can communicate with this layer using wireless communication depending on the distance between the user and the nearest BS. The uppermost layer is the cloud layer which is the core server which enables wired communication. Also called the core network server where all the contents were stored in the traditional cellular networks.

When there is a request by any user for accessing any content, the requested content is searched in the cache location of the neighbouring users through D2D communication, if they are in the range to form a D2D link. If a cache hit occurs, the content is shared and the communication is terminated. If a cache miss occurs, the requesting user's request is forwarded to the edge layer which consists of the BSs. The base stations are responsible for caching contents at the edge of the network and act as small data centres, for the users in the base layer, at the edge of the network. Further, when there is a cache miss at this layer as well, the user has to finally access the content from the core network server through wired communication. The core network server has the ability of both, caching and computing of the tasks.

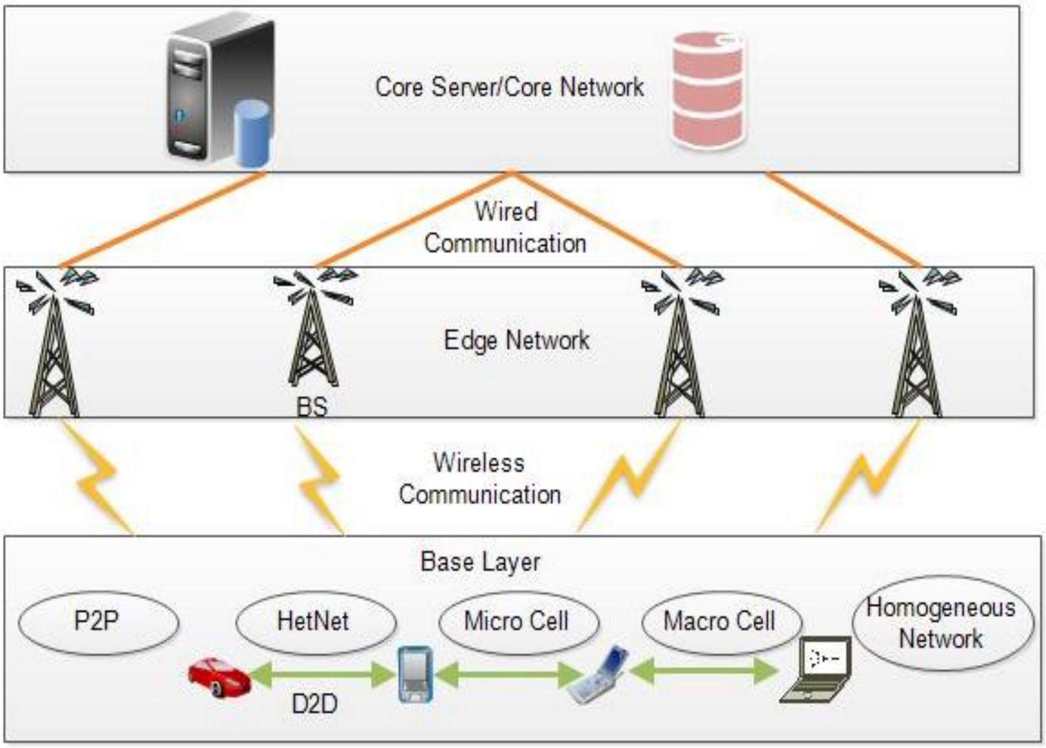

Figure 7: Proposed Architecture of caching in 5G

4.2 Proposed Approach

The caching of contents takes place on all the three layers as shown in the proposed architecture. The caching in the base layer is done on the UE's, the edge layer caches content at the BS and the cloud layer caches at the gateway or the core network server. We first identify the trust between two users. The trust is calculated by the distance between the users present in the network. The smaller the distance, the greater is the value of trust between the users. The users prefer to connect with the users whom they have the highest value of trust and are trustworthy. Then we use decision tree and random forest model to predict the cache locations for storing content. A comparative analysis is done between decision tree and random forest. The performance metrics used for evaluating the decision tree and random forest models are as:

1. Accuracy- Accuracy is the measure of the number of correct predictions to the total number of predictions made by the model.
2. Total Time- Total time is defined as the total time taken for building, training and testing the model and predicting the values.

3. Precision-It is the ratio of the relevant number of instances to the total number of instances retrieved. It is also called the positive predictive values.
4. Recall- Recall is the measure of actual positive values that are correctly predicted by the model. Also called as the sensitivity or true positive rate.
5. F1 score- It is harmonic mean between the recall and the precision, and is used for statistical evaluation to rate the performance of the model.

Another measure called as Macro Averaged Metrics has been calculated. For multi-class classification evaluation we compute vectors that contain multiple values representing each class. For example, precision contains 3 values corresponding to the classes 1, 2, and 3. The macro averaged metrics evaluates the entire model by averaging the values of each class resulting into macro averaged precision, recall and F1. For decision tree and random forest multi-class classification we consider the dataset without the consideration of trust in the network and the prediction of the cache locations is done.

1. We consider a dataset without the values of trust to predict the locations of cache in the network.
2. Further, we consider a dataset that considers the value of trust while predicting the cache locations.

The locations where the contents can be cached are as Node, BS and Gateway. For simpler representation and ease of use, these classes have been symbolized as:

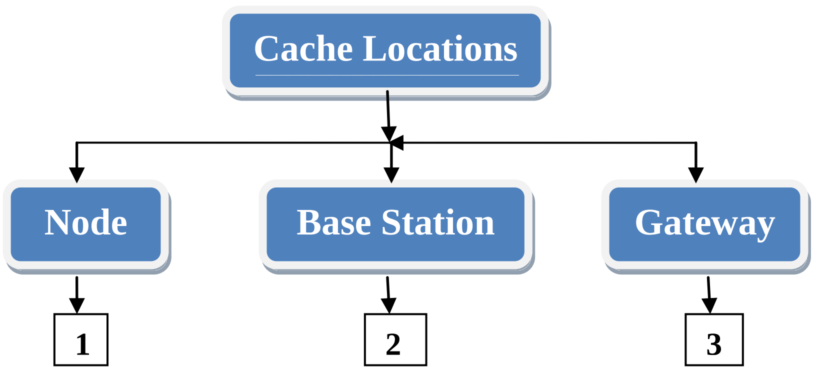

Figure 8: Cache Locations

The decision of where to cache contents depends on broadly five parameters as:

1. Volume of data- size of the requested content

2. Storage space at the UE- memory size of the UE’s
3. Distance- between the UE and the BS
4. Mobility speed of user- speed at which the user is travelling
5. Trust- the value of trust between two users will signify if they will connect to cache contents in the UE’s or not.

The size of the content being accessed by the user will be cached in one of the locations depending on how large the data is. The memory of the user’s device and the volume of the data being requested will yield that the data would be stored at the UE’s or not. Whenever the volume of data is large enough to be accommodated at the UE’s, the nearest BS is chosen for caching the content depending upon the speed of the user. But, if the user is not in the range of the nearest BS and is travelling at low-to-medium speed then the content is stored at the core network server. Moreover, if the user is travelling at a very high speed such that the contact time with the nearest neighbours and the BS is very less, then the content is cached at the core network server. The performance of network is evaluated by simulating the sharing of contents in a SDN environment. SDN is a framework which has aim of making an agile and flexible network. The aim of SDN is focus on improving the control of the network through enabling service providers and institutions to respond quickly towards the change in the requirements of business. A typical architecture of SDN architecture is comprised of three layers as: application layer, control layer and infrastructure layer. The application layer is comprised of network applications and the functions used by the organizations like, intrusion detection system, load balancing and firewalls. SDN uses appliances which make use of a controller to for managing the behaviour of data plane, whereas, a traditional network uses a specialized appliance, such as a firewall. The role of the control layer is to represents the centralized SDN controller software which acts as the brain of the SDN. This controller is located on a server and has the functionality of managing policies and the traffic flow throughout the network. The infrastructure layer is comprised of the physical switches in the network. The evaluation of the simulation is done on the following performance metrics:

1. Latency- Latency is defined as the time interval between the time when the user sends a request for accessing content and the time when the user receives the content. Also referred to as network latency which occurs due to the data communication in a network. It is measured in milliseconds.

2. Energy consumption- Energy consumption is the measure of the amount of energy required to transmit a data in the network using wireless communication. The energy required is the energy consumed from the mobile devices. It is measure in kilo joules (kJ).

4.3 Execution of Proposed Approach.

Initially we create a network of nodes and simulate the mobility, distance between the nodes to find the trust values between the users in the network using Matlab for obtaining a dataset for performing classification. The classification dataset is created which has 1000 instances, 3 classes and 5 attributes. We build a classification algorithm using random forest with multi-class.

The Decision Tree breaks down the dataset into subsets of data and develops an incremental tree. A final tree is obtained with the decision nodes and the leaf nodes. The nodes of the tree are built by splitting the parameters of the dataset using entropy and information gain. Entropy is defined as the amount of randomness in the information being computed and is calculated as:

$$Entropy\ (P) = \sum_{i=1}^{n} -p_i \log_2 p_i$$

Information gain is defined as the information gained about the randomness of parameter with respect to its attributes. The difference of the entropy of the target parameter and the parameter intended to split refers to the gained value. It is calculated as:

$$IG(T,P) = Entropy(T) - Entropy(T,P)$$

Here, IG refers to information gain, T refers to the target parameter, P refers to the parameter opted and p refers to probability of the attributes of the parameter. Random Forest builds a forest by ensembling the decision trees. It builds multiple decision trees and merges them to get a more accurate result. It uses the method of bagging for creating and merging different decision trees. For splitting the nodes, random forest does not search for the most important feature to split, but instead finds for the best feature for splitting. Bagging also called Bootstrap Aggregation is a simple technique used to ensemble the prediction from multiple classifying models. The only parameter considered while bagging decision trees is the number of trees to be considered. It does not focus on pruning the individual trees but lets the trees grow to greater depths. Large number of models might increase the execution of random forest algorithm but will not overfit the classification training dataset. It breaks the dataset into smaller subsets and builds an incremental decision tree.

The performance of the decision tree and random forest model, with and without considering trust, is compared on the basis of the latency and energy consumed while predicting the cache location.

4.3.1 Decision tree and Random forest models considering trust.

Fig. 9 shows the data loading into R. The dataset is first shuffled and then loaded for correct prediction of locations and prevent overfitting. It contains the column of value of trust and shows the first six rows of the loaded dataset and the column names considered.

```
> head(dataset)    # Show Top 6 records
    Location Volume_of_data Storage_space Speed Distance Trust
934        3        9378.69      30368.40   121      171   0.5
264        3        9910.00      35151.40   129      208  0.75
575        1       13401.80      61361.00   147      314     1
797        1       10861.90      43273.00   112      231     1
30         2        8334.20       9499.81    97      130  0.75
679        2        9408.37      27442.00    91      158  0.75
> nrow(dataset)    # Show number of records
[1] 949
> names(dataset)   # Show fields names or columns names
[1] "Location"        "Volume_of_data" "Storage_space"   "Speed"
[5] "Distance"        "Trust"
```

Figure 9: Loading dataset for non-trust models

Then we divide the dataset into training and testing data. The loaded dataset is divided into training dataset for training the decision tree and random forest model and testing dataset to test the model trained. Fig. 10 displays the top six rows of the training dataset and total number of rows considered for training and testing the model.

```
> head(trainDataset)    # Show Top 6 records
    Volume_of_data Storage_space Speed Distance Trust Location
934        9378.69      30368.40   121      171   0.5        3
264        9910.00      35151.40   129      208  0.75        3
575       13401.80      61361.00   147      314     1        1
797       10861.90      43273.00   112      231     1        1
30         8334.20       9499.81    97      130  0.75        2
679        9408.37      27442.00    91      158  0.75        2
> nrow(trainDataset)    # Show number of train Dataset
[1] 569
>
```

```
> head(testDataset)
    Volume_of_data Storage_space Speed Distance Trust Location
352        9585.89       26095.7    27      174  0.75        2
251       12639.70       70542.2    70      262  0.75        3
288       18327.50      175790.0    45      430  0.75        1
78         9784.19       36750.7    41      196  0.75        3
932        9644.63       25580.3    19      174   0.5        3
238       24511.00       47472.0    45      483  0.75        2
> nrow(testDataset)
[1] 380
```

Figure 10: Creation of train and test datasets

The decision tree and random forest model are then trained and the predicted values of the cache locations are obtained. The models are evaluated on the basis of confusion matrix, accuracy, time taken, precision, recall and F1 measure. The obtained actual-predicted values and the evaluation metrics are saved in a csv file.

4.3.2 Decision Tree and Random Forest Model without considering trust

For evaluating the models without considering the trust between the users the users, we build a model for the same without considering the trust factor. The dataset is first shuffled and then loaded for correct prediction of locations and prevent overfitting. Fig. 11 shows the first six rows of the loaded dataset and the column names considered.

```
> head(dataset)    # Show Top 6 records
    Location Volume_of_data Storage_space Speed Distance
179        1        2399.10       28276.0   264      465
78         3        9784.19       36750.7    41      196
738        1        5092.22        4613.0     8       94
750        2       15570.60      116371.0    51      271
415        3        8704.70       11131.3    37      156
657        3       11421.10       27060.0    35      166
> nrow(dataset)    # Show number of records
[1] 949
> names(dataset)   # Show fields names or columns names
[1] "Location"       "Volume_of_data" "Storage_space"  "Speed"
[5] "Distance"
```

Figure 11: Loading dataset for non-trust models

Fig. 12 illustrates the training and testing dataset obtained by dividing the loaded dataset into train and test data. Train data is used for training the decision tree and random forest models. The test data is used to test the accuracy of the models for the obtained predicted values by comparing with actual values of the cache locations.

```
> head(trainDataset)     # Show Top 6 records
    Volume_of_data Storage_space Speed Distance Location
179        2399.10       28276.0   264      465        1
78         9784.19       36750.7    41      196        3
738        5092.22        4613.0     8       94        1
750       15570.60      116371.0    51      271        2
415        8704.70       11131.3    37      156        3
657       11421.10       27060.0    35      166        3
> nrow(trainDataset)     # Show number of train Dataset
[1] 569
```

```
> head(testDataset)
    Volume_of_data Storage_space Speed Distance Location
549        9868.87      35005.00    32      204        1
419        3974.61       2203.07    22       54        1
839       14553.70     147106.00   181      308        3
571        8122.78       5330.75    29       86        2
552        9625.67      29224.00    24      170        1
769        5260.89       4275.34    17       88        1
> nrow(testDataset)
[1] 380
```

Figure 12: Training and testing dataset for non-trust models

The decision tree and random forest model are then trained and the predicted values of the cache locations are obtained. The models build for non-trusted users are evaluated on the basis of confusion matrix, accuracy, time taken, precision, recall and F1 measure. The obtained actual-predicted values and the evaluation metrics are saved in a csv file. A comparative analysis is made between the four models on the basis of the evaluation metrics to analyze which model yields the best performance.

To obtain the results for the latency and the energy consumed for sending and receiving the contents, we simulate the network in iFogSim simulator. This is an open source toolbox having high-performance for fog computing, edge computing and IoT. They are used for modelling and simulating the networks of edge computing, IoT and fog computing. The simulation is done through in-built libraries of ifogsim which are written in Java. The ifogsim installation requires the Java Development Kit (JDK) for customising and working with the toolbox. It is executed on any Java based integrated development environment (IDE). For example, Eclipse, Netbeans, JCreator, JDeveloper, jGRASP, BlueJ, IntelliJ IDEA or Jbuilder. For this thesis we use Eclipse for executing the libraries and Software defined network (SDN) environment is used for simulating the network. The output of the execution is viewed in the console of the Eclipse IDE. We simulated the network to observe the amount the latency and the amount of energy of the UE's consumed.

A network is created and simulated to obtain the results for classification models in Matlab. The number of users in the network is represented as nodes and the connectivity between the nodes represents the wireless communication link between them. Fig. 13 shows the network created containing 500 nodes.

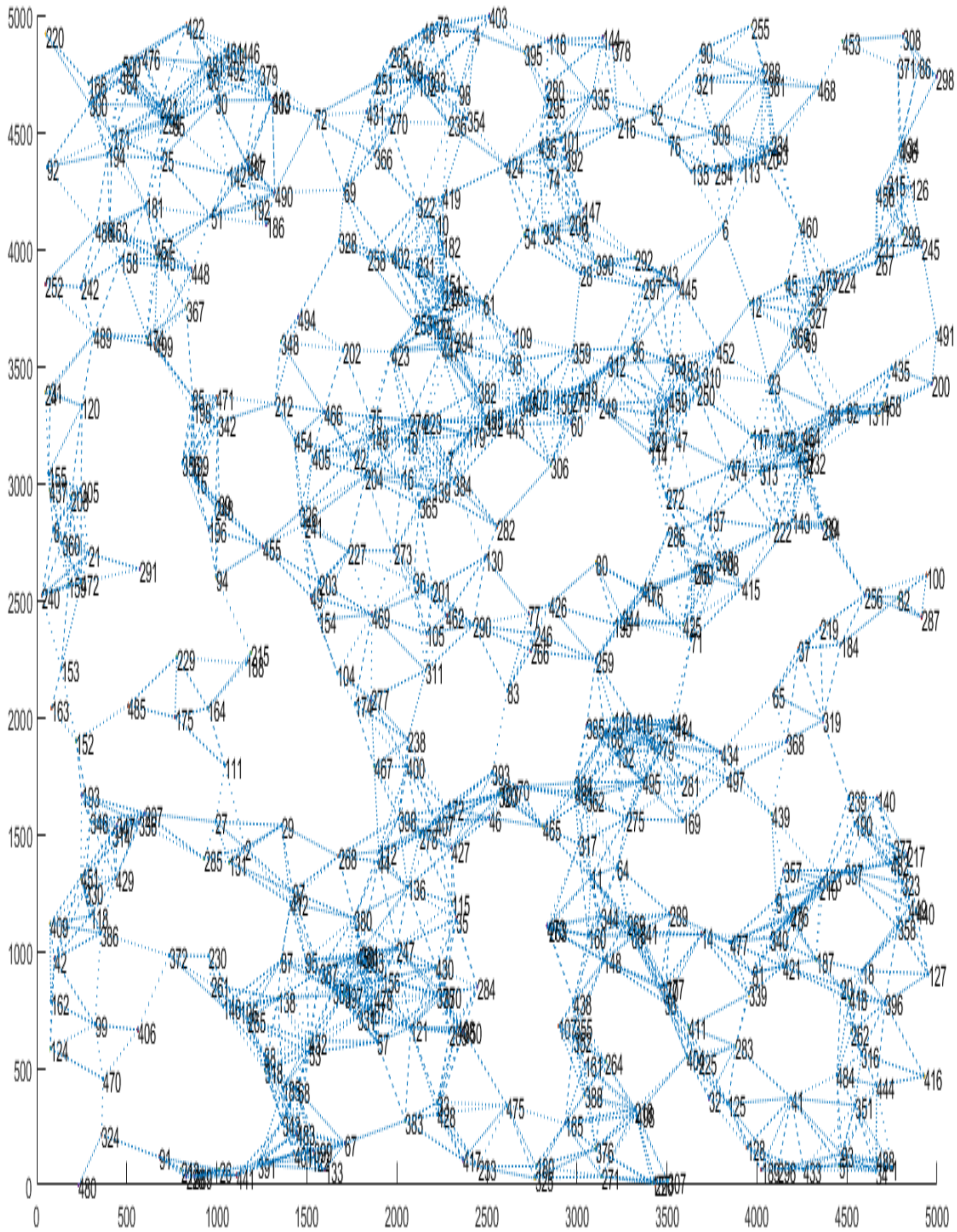

Figure 13: Created Network

According to the algorithms, the classification of the cache locations has been done. The classes in the predictions are Node, BS and Gateway. Now, the results of both the random forest models, with trust and without trust are compared. Fig. 14 and Fig. 15 shows the confusion matrix of decision tree and random forest model with trust and without trust, respectively.

```
> cm
      Predicted
Actual   1   2   3
     1  94  18   0
     2   4 135   6
     3   2  26  95
```

Confusion Matrix of trust Model

```
> cm
      Predicted
Actual   1   2   3
     1  32  79   6
     2   3 111  16
     3   3  76  54
```

Confusion Matrix of non-trust Model

Figure 14: Confusion Matrix of Decision Tree Model

```
> cm
      Predicted
Actual   1   2   3
     1  83  26   2
     2   3 128   6
     3   1  23 108
```

Confusion Matrix of trust Model

```
> cm
      Predicted
Actual   1   2   3
     1  33  73   5
     2  11 124   6
     3   4  77  47
```

Confusion Matrix of non-trust Model

Figure 15: Confusion Matrix of Random forest Model

The performance of the decision tree and random forest models are evaluated by using time taken, accuracy, precision, recall and F1 measure. Once the model is trained and the evaluation of the model is done, the obtained results are written and saved into a CSV file. The actual and predicted values of the cache locations are also written into a CSV file.

The results obtained from the decision tree shows that the accuracy achieved is of 47% and the accuracy tends become 78.68% from the trust based decision tree model. The results obtained from the decision tree model are shown in Fig. 16 shows the results obtained from the decision tree illustrates that the decision tree model tends to be very instable and is observed to over fit the tree by considering the too many splits and often makes wrong predictions.

```
> data.frame(precision, recall, f1)
  precision    recall        f1
1 0.8333333 0.7211538 0.7731959
2 0.6844920 0.8827586 0.7710843
3 0.9320388 0.7328244 0.8205128
>
> # Step 12.3: Macro-averaged metrics
> macroPrecision = mean(precision)
> macroRecall = mean(recall)
> macroF1 = mean(f1)
>
> data.frame(macroPrecision, macroRecall, macroF1)
  macroPrecision macroRecall   macroF1
1      0.8166214   0.7789123 0.7882643
>
> # Step 12.4: Accuracy
> accuracy <- round(mean(Actual==Predicted) *100,2)
> accuracy
[1] 78.68
>
>
> # Step 12.5: Total Time
> totalTime = proc.time()[3] - startTime
> totalTime
elapsed
   0.56
```

Results obtained from trust model

```
> data.frame(precision, recall, f1)
  precision    recall        f1
1 0.4690265 0.4568966 0.4628821
2 0.5217391 0.6000000 0.5581395
3 0.4150943 0.3548387 0.3826087
>
> # Step 12.3: Macro-averaged metrics
> macroPrecision = mean(precision)
> macroRecall = mean(recall)
> macroF1 = mean(f1)
>
> data.frame(macroPrecision, macroRecall, macroF1)
  macroPrecision macroRecall   macroF1
1        0.46862   0.4705784 0.4678768
>
> # Step 12.4: Accuracy
> accuracy <- round(mean(Actual==Predicted) *100,2)
> accuracy
[1] 47.63
>
>
> # Step 12.5: Total Time
> totalTime = proc.time()[3] - startTime
> totalTime
elapsed
   0.54
```

Results obtained from non-trust model

Figure 16: Results obtained from Decision Tree Model

After evaluating the results of decision Tree Model, they were compared with the results obtained from the random forest model. The accuracy achieved from the non-trust model of the random forest is 53% which is higher as compared to the non-trust model of the decision tree. The trust model of the random forest yields an accuracy of 85% which is higher than the trust model of the decision tree. On making an overall comparative analysis on the basis of accuracy, the trust model achieves maximum accuracy of 85% which is higher than all the four models, that is, 47% and 53% of non-trust model of decision tree and random forest, and 78% of the trust model of the decision tree. On the basis of time taken for training the model, the decision tree model takes less time as compared to random forest, but has a disadvantage of creating overfit decision trees. Fig. 17 shows the evaluation results obtained from the random forest model.

```
> data.frame(precision, recall, f1)
  precision    recall        f1
1 0.9540230 0.7477477 0.8383838
2 0.7231638 0.9343066 0.8152866
3 0.9310345 0.8181818 0.8709677
>
> # Step 12.3: Macro-averaged metrics
> macroPrecision = mean(precision)
> macroRecall = mean(recall)
> macroF1 = mean(f1)
>
> data.frame(macroPrecision, macroRecall, macroF1)
  macroPrecision macroRecall  macroF1
1      0.8694071    0.833412 0.8415461
>
> # Step 12.4: Accuracy
> accuracy <- round(mean(Actual==Predicted) *100,2)
> accuracy
[1] 83.95
>
>
> # Step 12.5: Total Time
> totalTime = proc.time()[3] - startTime
> totalTime
elapsed
   2.98
```

Results obtained from trust model

```
> data.frame(precision, recall, f1)
  precision    recall        f1
1 0.6875000 0.2972973 0.4150943
2 0.4525547 0.8794326 0.5975904
3 0.8103448 0.3671875 0.5053763
>
> # Step 12.3: Macro-averaged metrics
> macroPrecision = mean(precision)
> macroRecall = mean(recall)
> macroF1 = mean(f1)
>
> data.frame(macroPrecision, macroRecall, macroF1)
  macroPrecision macroRecall  macroF1
1      0.6501332   0.5146391 0.5060203
>
> # Step 12.4: Accuracy
> accuracy <- round(mean(Actual==Predicted) *100,2)
> accuracy
[1] 53.68
>
>
> # Step 12.5: Total Time
> totalTime = proc.time()[3] - startTime
> totalTime
elapsed
   3.45
```

Results obtained from non-trust model

Figure 17: Results obtained from random forest model

The overall results obtained from all the models on the basis of total time taken, accuracy, macro averaged precision, macro-averaged recall and macro averaged F1 measure are depicted in Fig. 18, Fig. 19 and Fig. 20.

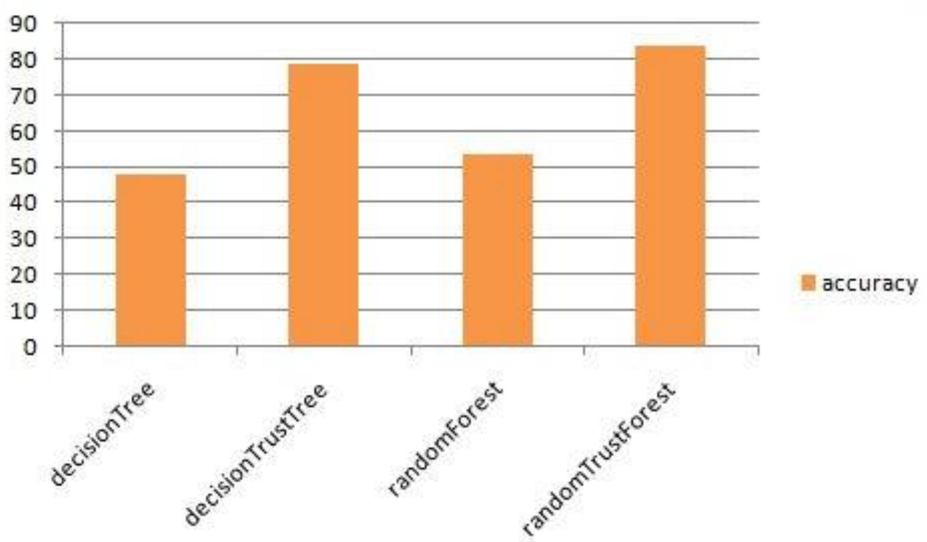

Figure 18: Accuracy

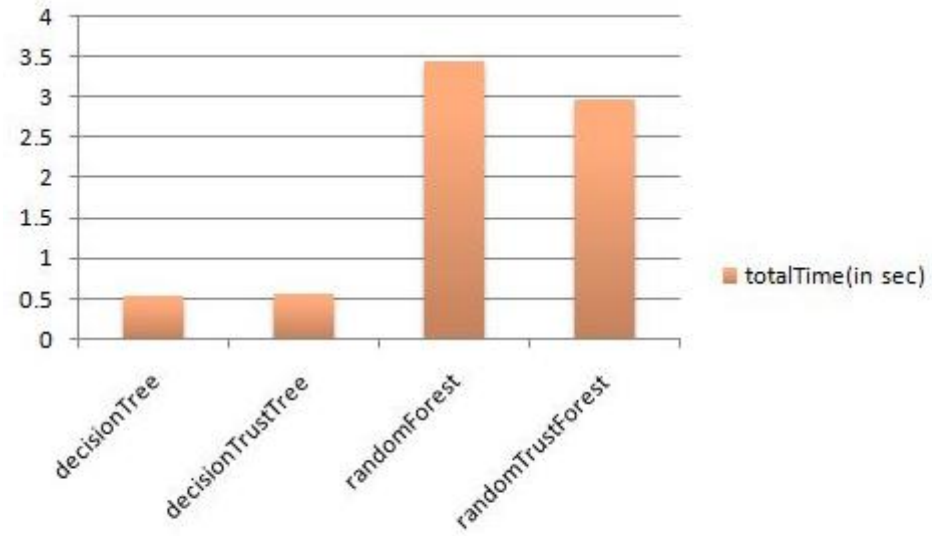

Figure 19: Time taken

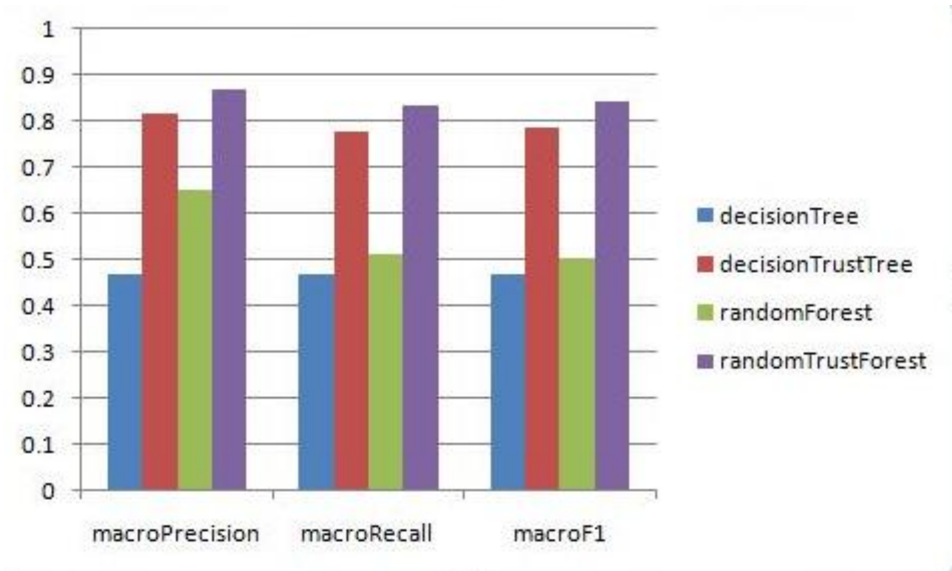

Figure 20: Macro-averaged Precision, Recall and F1

The performance metric considered for the network simulation using iFogSim are average access latency and the energy consumption of the UE's. The results for average access latency which is based on the factors like distance, channel condition etc, are obtained after simulating the network in a SDN environment for sending and receiving of the contents are illustrated in Fig. 21. The amount of energy consumption of the UE's is dependent upon the location from which the content is being accessed. On the basis of the locations, that is Node, BS and Gateway the energy consumption of the UE is shown in Fig. 22.

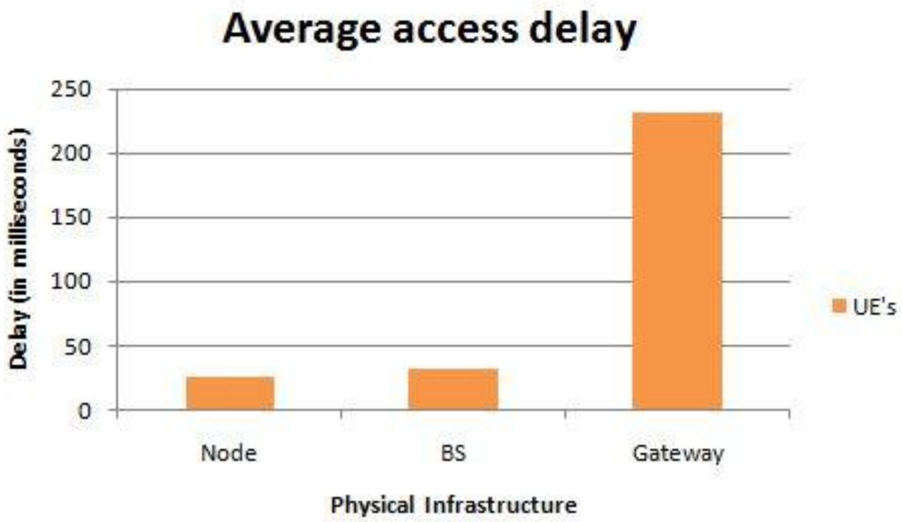

Figure 21: Average Latency

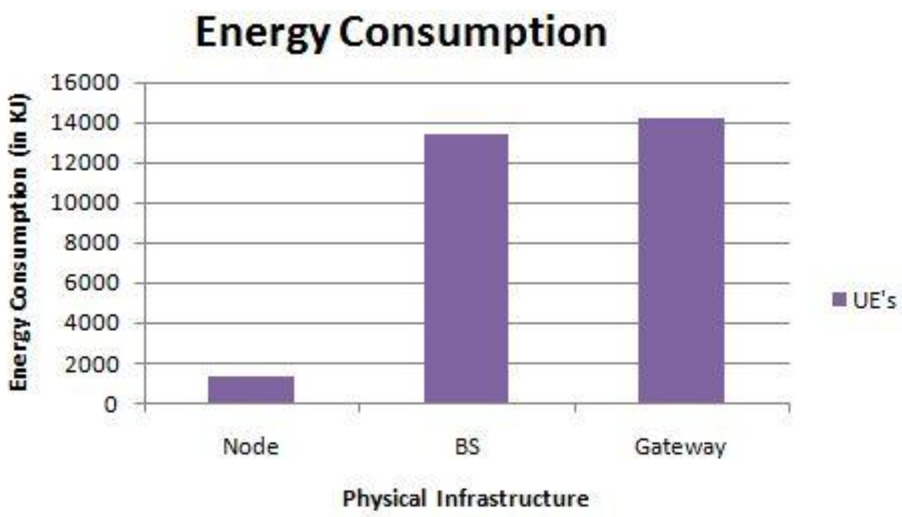

Figure 22: Energy Consumption

6. Conclusion and Future Scope

6.1 Conclusion

The main objective of caching in 5G using D2D communication is to minimize the delay experienced by the mobile users when sharing contents. We also provide a comprehensive comparison in the existing literature on the basis of connectivity and communication, storage, quality of service to its users and the trust management among the users. We have proposed an architecture which shows the method caching in 5G. Moreover, we form a network of users. Further, we propose a technique to decide the cache location using machine learning multi-class classification models. The classification models used are decision tree and random forest model. Both the classification models are used for predicting the cache locations under two condition, 1) considering trust as a parameter, and 2) not considering trust between the users. We trained the decision tree and random forest models and predicted the cache locations such that it yields an optimal solution. A comparative analysis is done amoung all the models and it is observed that random forest model with trust has the highest accuracy of 85% in all the models. Moreover, we simulate the sharing of contents between the users by simulating the network in iFogSim simulator. The performance metrics used is average access latency and energy consumption. The results obtained show that the energy consumption is minimum when the communication is with the neighbouring users and maximum when the content sharing is between users and gateway. The average access delay of the network is maximum when the accessed content is at the gateway and minimum when the content is accessed from a neighbouring user.

6.2 Future Scope

- In the future the proposed approach can be extended to further cache contents in a network consisting of users.
- The proposed approach can be used to further minimize the energy consumption of the UE’s which is a challenge for the 5G cellular networks.
- The proposed network can be simulated considering more number of parameters and under more realistic environmental conditions.

REFERENCES

[1] “Trend increase kernel description,” https://www.statista.com/statistics/278414/ number-of-worldwide-social-network-users/, accessed: 2019-05-21.

[2] “Trend increase kernel description,” https://blog.hootsuite.com/youtube-stats-marketers/, accessed: 2019-05-21.

[3] Z. Wu, V. D. Park, J. Li, Enabling device to device broadcast for lte cellular networks, IEEE Journal on Selected Areas in Communications 34 (1) (2016) 58–70.

[4] D. Tsolkas, N. Passas, L. Merakos, Device discovery in lte networks: A radio access perspective, Computer Networks 106 (2016) 245–259.

[5] K. Sharmila, V. Mohan, C. Ramesh, S. P. Munda, Proximity services based device-to-device framework design for direct discovery, in: 2016 2nd international conference on advances in electrical, electronics, information, communication and bio-informatics (AEEICB), IEEE, 2016, pp. 499–502.

[6] “Trend increase kernel description,” https://www.edgefx.in/different-types-wireless-communication-technologies/, accessed: 2019-05-21.

[7] “Trend increase kernel description,” https://www.rohm.com/electronics-basics/wireless/short-range-wireless-communication, accessed: 2019-05-21.

[8] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, W. Wang, A survey on mobile edge networks: Convergence of computing, caching and communications, IEEE Access 5 (2017) 6757–6779.

[9] Y. Mao, C. You, J. Zhang, K. Huang, K. B. Letaief, A survey on mobile edge com- puting: The communication perspective, IEEE Communications Surveys & Tutorials 19 (4) (2017) 2322–2358.

[10] Z. Zhang, S. Li, A review of computational offloading in smart mobile devices for mobile cloud computing, in: 2016 2nd IEEE International Conference on Computer and Communications (ICCC), IEEE, 2016, pp. 2738–2743.

[11] M. Anugraha, S. Krishnaveni, Recent survey on efficient trust management in mobile ad hoc networks, in: 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), IEEE, 2016, pp. 1–4.

[12] M. Ji, G. Caire, A. F. Molisch, Wireless device-to-device caching networks: Basic principles and system performance, IEEE Journal on Selected Areas in Communications 34 (1) (2016) 176–189.

[13] W. Li, E. Chan, D. Chen, Energy-efficient cache replacement policies for cooperative caching in mobile ad hoc network, in: 2007 IEEE Wireless Communications and Networking Conference, IEEE, 2007, pp. 3347–3352.

[14] C. Wang, C. Liang, F. R. Yu, Q. Chen, L. Tang, Computation offloading and resource allocation in wireless cellular networks with mobile edge computing, IEEE Transactions on Wireless Communications 16 (8) (2017) 4924–4938.

[15] H. Wang, R. Li, L. Fan, H. Zhang, Joint computation offloading and data caching with delay optimization in mobile-edge computing systems, in: 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), IEEE, 2017, pp. 1–6.

[16] M. Liu, F. R. Yu, Y. Teng, V. C. Leung, M. Song, Computation offloading and content caching in wireless blockchain networks with mobile edge computing, IEEE Transactions on Vehicular Technology 67 (11) (2018) 11008–11021.

[17] H. Zhao, Y. Wang, R. Sun, Task proactive caching based computation offloading and resource allocation in mobile-edge computing systems, in: 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), IEEE, 2018, pp. 232–237.

[18] G. S. Park, H. Song, Cooperative base station caching and x2 link traffic offloading system for video streaming over sdn-enabled 5g networks, IEEE Transactions on Mobile Computing.

[19] D. Wang, H. Li, S. Wang, J. Yang, X. Liu, X. Zhang, Mobility aware caching incentive scheme for d2d cellular networks, in: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), IEEE, 2017, pp. 1–6.

[20] C. Zhan, G. Yao, Optimizing caching placement for mobile users in heterogeneous wireless network, in: 2017 IEEE 42nd Conference on Local Computer Networks (LCN), IEEE, 2017, pp. 175–178.

[21] N. Naderializadeh, D. T. Kao, A. S. Avestimehr, How to utilize caching to improve spectral efficiency in device-to-device wireless networks, in: 2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE, 2014, pp. 415–422.

[22] Q. Xu, Z. Su, Q. Zheng, M. Luo, B. Dong, Secure content delivery with edge nodes to save caching resources for mobile users in green cities, IEEE Transactions on Industrial Informatics 14 (6) (2018) 2550–2559.

[23] S. Sivashankari, M. M. Raseen, A framework of trust management on location privacy and minimising the error rate in wireless sensor networks, in: 2013 International Conference on Optical Imaging Sensor and Security (ICOSS), IEEE, 2013, pp. 1–6.

[24] T. Zhang, J. He, Y. Zhang, Trust based secure localization in wireless sensor networks, in: 2011 2nd International Symposium on Intelligence Information Processing and Trusted Computing, IEEE, 2011, pp. 55–58.

[25] Y. Luo, J. Wang, W. Gui, A distributed algorithm for building energy-efficient group- shared multicast tree in ad hoc networks, in: 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing, Vol. 1, IEEE, 2010, pp. 341–344.

[26] S. Deb, P. Linga, R. Rastogi, A. Srinivasan, Accelerating lookups in p2p systems using peer caching, in: 2008 IEEE 24th International Conference on Data Engineering, IEEE, 2008, pp. 1003–1012.

[27] N. Naderializadeh, D. T. Kao, A. S. Avestimehr, How to utilize caching to improve spectral efficiency in device-to-device wireless networks, in: 2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE, 2014, pp. 415–422.

[28] E. Datsika, A. Antonopoulos, N. Zorba, C. Verikoukis, Green cooperative device–to-device communication: A social–aware perspective, IEEE Access 4 (2016) 3697–3707.

[29] Q. Jia, R. Xie, T. Huang, J. Liu, Y. Liu, Energy-efficient cooperative coded caching for heterogeneous small cell networks, in: 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2017, pp. 468–473.

[30] C.-F. Chao, Y.-H. Wang, J. Chen, Y.-C. Lin, A cache sharing interface for data access in mobile ad hoc networks, in: ITRE 2005. 3rd International Conference on Information Technology: Research and Education, 2005., IEEE, 2005, pp. 78–82.

[31] D. Malak, M. Al-Shalash, Optimal caching for device-to-device content distribution in 5g networks, in: 2014 IEEE Globecom Workshops (GC Wkshps), IEEE, 2014, pp. 863–868.

[32] X. Huang, Z. Zhao, H. Zhang, Cooperate caching with multicast for mobile edge computing in 5g networks, in: 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), IEEE, 2017, pp. 1–6.

[33] K. Zhang, S. Leng, Y. He, S. Maharjan, Y. Zhang, Cooperative content caching in 5g networks with mobile edge computing, IEEE Wireless Communications 25 (3) (2018) 80–87.

[34] S. Mehamel, K. Slimani, S. Bouzefrane, M. Daoui, Energy-efficient hardware caching decision using fuzzy logic in mobile edge computing, in: 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), IEEE, 2018, pp. 237–242.

[35] Y. Cui, W. He, C. Ni, C. Guo, Z. Liu, Energy-efficient resource allocation for cache- assisted mobile edge computing, in: 2017 IEEE 42nd Conference on Local Computer Networks (LCN), IEEE, 2017, pp. 640–648.

[36] C. Liang, Y. He, F. R. Yu, N. Zhao, Energy-efficient resource allocation in software- defined mobile networks with mobile edge computing and caching, in: 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2017, pp. 121–126.

[37] Y. Hao, M. Chen, L. Hu, M. S. Hossain, A. Ghoneim, Energy efficient task caching and offloading for mobile edge computing, IEEE Access 6 (2018) 11365–11373.

[38] S. Yu, R. Langar, X. Fu, L. Wang, Z. Han, Computation offloading with data caching enhancement for mobile edge computing, IEEE Transactions on Vehicular Technology 67 (11) (2018) 11098–11112.

[39] M. K. H. Yeung, Y.-K. Kwok, A game theoretic approach to energy efficient cooperative cache maintenance in manets, in: 2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, Vol. 3, IEEE, 2005, pp. 1500–1504.

[40] J. Liu, J. Xu, X. Chu, Fine-grained scalable video caching for heterogeneous clients, IEEE transactions on multimedia 8 (5) (2006) 1011–1020.

[41] Z. Naor, Efficient wireless access to video-on-demand services, in: 2011 Ninth Annual Communication Networks and Services Research Conference, IEEE, 2011, pp. 278–283.

[42] J. Li, S. Gu, Y. Wang, Q. Zhang, Double replication mds codes for wireless d2d distributed storage networks, in: 2018 10th International Conference on Wireless Communications and Signal Processing (WCSP), IEEE, 2018, pp. 1–6.

[43] J. Guo, J. Yuan, J. Zhang, An achievable throughput scaling law of wireless device-to-device caching networks with distributed mimo and hierarchical cooperations, IEEE Transactions on Wireless Communications 17 (1) (2018) 492–505.

[44] W. Han, A. Liu, W. Yu, V. K. Lau, Joint frequency reuse and cache optimization in backhaul-limited small-cell wireless networks, IEEE Transactions on Wireless Communications 17 (10) (2018) 6917–6930.

[45] J. Wu, C. Yuen, B. Cheng, Y. Yang, M. Wang, J. Chen, Bandwidth-efficient multipath transport protocol for quality-guaranteed real-time video over heterogeneous wireless networks, IEEE Transactions on Communications 64 (6) (2016) 2477–2493.

[46] M. Sinnwell, G. Weikum, A cost-model-based online method for distributed caching, in: Proceedings 13th International Conference on Data Engineering, IEEE, 1997, pp. 532–541.

[47] L. B. Pinho, C. L. Amorim, E. Ishikawa, Glove: A distributed environment for low cost scalable vod systems, in: 14th Symposium on Computer Architecture and High Performance Computing, 2002. Proceedings., IEEE, 2002, pp. 117–124.

[48] A. Araldo, M. Mangili, F. Martignon, D. Rossi, Cost-aware caching: optimizing cache provisioning and object placement in icn, in: 2014 IEEE Global Communications Conference, IEEE, 2014, pp. 1108–1113.

[49] M. Sutaone, P. Mukherj, S. Paranjape, Trust-based cluster head validation and outlier detection technique for mobile wireless sensor networks, in: 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), IEEE, 2016, pp. 2066–2070.

[50] N. Alsaedi, F. Hashim, A. Sali, Energy trust system for detecting sybil attack in clus- tered wireless sensor networks, in: 2015 IEEE 12th Malaysia International Conference on Communications (MICC), IEEE, 2015, pp. 91–95.

[51] Y. Ding, Q. Li, F. Li, A novel method for evaluating trustworthiness between strangers in large, dynamic ad hoc networks, in: 2009 Second International Workshop on Knowledge Discovery and Data Mining, IEEE, 2009, pp. 280–283.

[52] M. M. Abuzaid, M. A. Azer, Janitor, certificate and jury (jcj) trust scheme for wireless ad-hoc networks, in: 2012 Seventh International Conference on Availability, Reliability and Security, IEEE, 2012, pp. 383–387.

[53] R. Saini, R. K. Gautam, Establishment of dynamic trust among nodes in mobile ad- hoc network, in: 2011 International Conference on Computational Intelligence and Communication Networks, IEEE, 2011, pp. 346–349.

[54] A. GowriDurga, B. Prakash, A zigbee sms alert system with trust mechanism in wireless sensor networks, in: 2013 International Conference on Information Communication and Embedded Systems (ICICES), IEEE, 2013, pp. 1010–1014.

LIST OF PUBLICATIONS

Divya Prerna and Rajkumar Tekchandani, “Machine Learning based Energy-Efficient Cache Localization Technique for Device-to-Device Communication in IoT Environment,” Communicated for publication in International Conference on Computing, Analytics and Networking.