

Energy Efficient Multimedia Message Sharing in Instant Messenger Apps

Pinial Khan Butt¹, Wang Yang^{1,*} and Guojun Wang^{2, 1}

¹*School of Information Science and Engineering, Central South University, Changsha, China, 410083*

²*School of Computer Science and Educational Software, Guangzhou University, Guangzhou, 510006, China*

pinial@yahoo.com, yangwang@csu.edu.cn, csgjwang@csu.edu.cn

Abstract

Multimedia messaging in instant messenger (IM) application (app) usage has become increasingly popular on smartphones today. Sharing of multimedia messages such as images, web links, videos and audio messages has become prevalent on smartphones. The IM app causes smartphone battery to drain owing to its intensive data upload / download operations when users share multimedia messages. This work focus on instant messenger app performs upload / download to share multimedia message impact on a smartphone battery. Our collected data shows that IM users share a large number of redundant multimedia messages with server side and also existing in local storage. A redundant network traffic directly can increase access wireless network (3G or Wi-Fi) usage and leading to high energy consumption in a smartphone. This paper proposes an energy efficient multimedia message sharing scheme to prevent redundant network traffic. The underlying principle of the proposed design is verifying hashes of shared multimedia messages prior to an instant messenger application uploading or downloading them. Our evaluation shows that our proposal saves up to 60 % of energy consumption.

Keywords: *multimedia message, smartphone, hash key, instant messenger, energy efficiency*

1. Introduction

Regular short-message service (SMS) has been replaced by an instant messenger (IM) application (app) due to cost effectiveness. In addition, IM app also provides several functionalities such as multimedia message sharing, user online presence awareness, and user typing notification status. A rapid development of more powerful hardware and more complex software in smartphones consume much more energy compared to feature phones. Smartphones generally obtain power from rechargeable batteries. In daily life, instant messenger is one of the most widely used mobile applications in a smartphone. Instant messenger users have probably shared a large number of photos, videos and audio notes that may have accumulated in smartphone storage. As a number of users are using multimedia messaging in instant messenger app, the importance of understanding IM app share multimedia messaging energy consumption become important. Wireless Network (such as Wi-Fi, 3G, and LTE) consumes the most energy in smartphones [1, 2]. A cache is software mechanism in smartphone that keep some data in RAM mobile applications such as instant messenger app, game app, entertainment app and utility app can access data from RAM before hit the local storage. In order to lower energy consumption by smartphone wireless network, mobile application may access a significant amount of data from RAM or local storage. Android manages files in several filesystem partition such as /data, /system and /cache. Android filesystem that mounts android partitions are YAFFS2, ext2, ext3 and ext4 [3]. Android stores IM app users related data in /data partition. The

*Corresponding author

ext4 filesystem used in the latest phones in an android system such as Samsung galaxy S6. In this paper, we investigate IM users share multimedia message energy inefficiencies. Instant messenger WeChat app has 650 million users which can easily share text, photos, and videos and forms in various chats [4]. It is clear that multimedia messaging has steady growing traffic in instant messenger apps. IM app facilitates to make a connection between known persons called “buddy list”. IM app lets users share instantly multimedia messages such as (images, web links, video and audio (voice notes or songs) with buddy list recipients. IM app performs upload / download which requires smartphone wireless network 3G cellular or Wi-Fi to be turned on. However, IM user share multimedia message with buddy list recipients which requires upload multimedia message to IM server. In addition, IM app user recipients can view or listen to multimedia message by downloading them to their smartphone local storage. IM app users share multimedia message to buddy list recipients impact on smartphone energy consumption is still a black box.

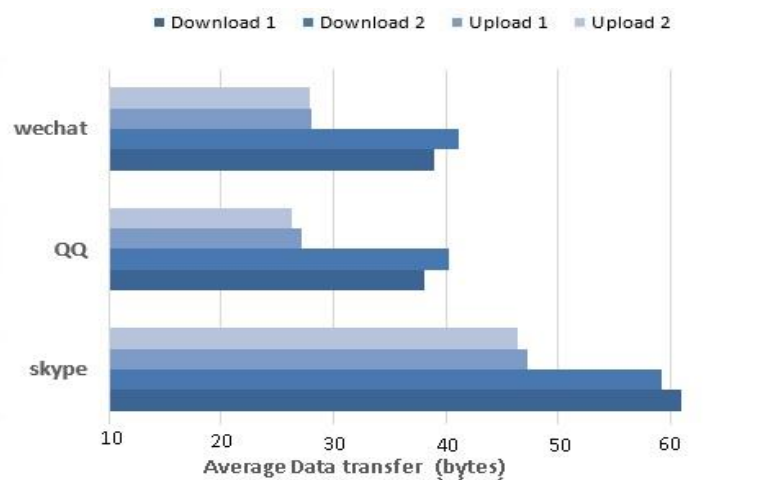


Figure 1(a). Shows Average Data Transfer, by Different Instant Messenger Apps Allow Performs Multi Upload / Multi Download To Share Redundant Multimedia Message Video Clip

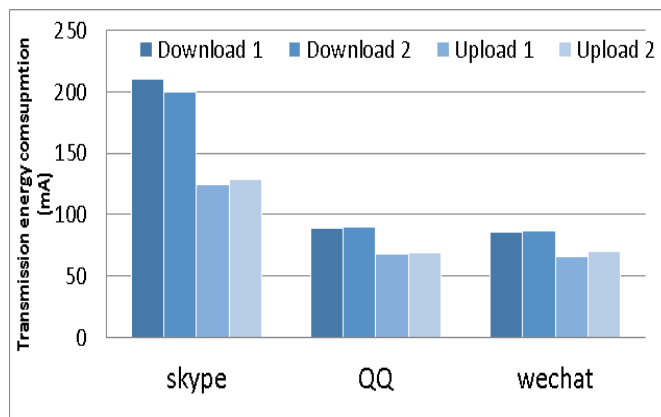


Figure 1(b). Shows Transmission Energy Consumption, by Different Instant Messenger Apps Allow Performs Multi Upload / Multi Download To Share Redundant Multimedia Message Video Clip

We selected three of the most popular instant messenger apps: Skype, WeChat and QQ, from the Play Store to illustrate redundant test results of different IM app to share original multimedia message which is an .mp4 video clip size of 2 MB. Figure 1 (a) (b) (c) shows test results includes average data transfer, transmission energy consumption and response time to share a video clip with 2 recipients requiring performing multi downloads (2 times named as download1, download2) and multi uploads (2 times named as upload1, upload2). We test different IM app which shares an original multimedia message in a homogeneous way such as skype to skype, QQ to QQ and WeChat to WeChat app running on a smartphone connected via Wi-Fi network. The measurement of IM app energy consumption computed by Monsoon meter [5] and tcpdump [6] to capture network traffic and our tools used have been described briefly in section 5.1. Figure 1 (a) (b) (c) shows that there is great diversity between skype and WeChat where as QQ IM app spends the amount of average data transfer, energy consumption spent and response time to complete multi uploads and multi downloads by different instant messenger apps.

IM app user share multimedia message with other IM recipient's performs an upload / download operations. In our finding based on experiment, IM app does not detect duplicate shared multimedia message. In this case, IM app allows to share a duplicate multimedia message with multi-recipients cause's multi-time redundant upload / download. IM app performs multi-time upload / download redundant multimedia message causes aggressive energy consumption on a smartphone battery. In our data collection described in subsection 3.2, we found IM users share large number of redundant multimedia message and copies in local storage database. This makes the important assumption that redundant multimedia message copies also exist in IM server storage. The result is that different IM applications allow redundant multimedia messages to upload and download which causes extra space occupied in local storage and leads to wireless network high energy consumption in a smartphone. In addition, it consumes the same amount of data transfer, energy transmission and time. However, WeChat and QQ app measurement values are significantly the same as shown in Figure 1 (a) (b) (c). Further, Skype app consume 48% more data transfer, consume 65 % more energy consumption and spend 56% more time than QQ and WeChat app. Skype employs larger packets than QQ and WeChat app but performs less and long transmission; this leads to high energy consumption. IM app performs upload / download for redundant multimedia message can be expensive energy consumption operation for smartphone battery. In addition, the upload and download of a multimedia message transfer speed available for these can result in long delays when sharing large video which are redundant multimedia message data files.

However, the famous website YouTube, does not allow the upload of redundant video files that already exist in their system server [7]. But IM apps are lacking in this verification data integrity for already existing redundant multimedia messages. IM apps perform uploads and downloads of redundant content causing extra burdens on battery power. In this paper, we focus solely on improving the energy consumption of instant messenger when IM app performs upload / download of users share multimedia messages. Due to current wastage of battery power practice by IM app, a redundant multimedia message in IM app storage to occupy extra space that could be used for unique content and IM app sent redundant multimedia message over the internet, this leads to wireless network of high energy consumption in a smartphone. Improvement in terms of IM app shares of redundant multimedia message can play a role in preventing redundant content in storage and lower wireless network energy consumption in a smartphone.

In summary, in this paper we make the following contributions:

- Present the data collected from WeChat, the most popular IM platform for sharing multimedia messages. It reveals IM users share a large number of redundant multimedia messages which cause a burden on storage space and leads to high wireless network

energy consumption in a smartphone.

Propose energy efficient multimedia message sharing design which verifies hashes and generate a hash key for a redundant multimedia message. Our approach to IM app access of redundant multimedia message uses hash key save energy up to 60% with 3G and Wi-Fi network.

The focus of research work is on IM app sharing multimedia messages. Therefore, we do not consider text conversations in this research paper. To our knowledge, we are the first contributors towards achieving energy efficient multimedia message sharing in instant messenger app.

The organization of this paper is as follows; section 2 presents work that has been done in the area of energy saving in smartphones, section 3, presents investigation of data collected from 8 IM users. In section 4, energy efficient multimedia messages sharing design. Section 5, evaluation to proposal, measurement tool, choosing hashing algorithm and testing proposal energy efficient multimedia message sharing design to get results about energy saving cost. Section 6 concludes the paper.

2. Related Work

The present research work on energy saving in a smartphone has attracted much attention. We divided related work into several categories such as storage, instant messenger apps, data redundancy, and multimedia message analysis research work.

Storage analysis. Recent research work focuses on optimizing and analyzing I/O storage. A tool was developed to capture android based smartphone cross layer storage I/O traces [8]. A study presents comparisons between different mobile web browsers application page browsing analyze I/O access SQLite database efficiency [9]. A recent study conveys that smartphone app, social network, games and web browsers SQLite database generate 80% I/O operations in android [10].

Instant messenger apps analysis. Ying Yang presents a forensic digital analysis of most popular instant messengers QQ, MSN, email and Foxmail storage file directories [11]. A measurement tool focusing on cross layer of smartphone app network traffic affect on 3G state transition energy consumption analysis [12]. The IM app presence updates have been studied [13] to delay text messages and [14] to IM users persistent presence update delay offered power saving. A previous study [15] reveals 5 famous instant messenger bandwidth utilization inefficiencies in conversion and background receiving messages. A previous study [16] investigated data collected from IM 51 users' single chat (one to one) text messages. IM end user typed multiple messages and bundled together to user end typing to reduce the energy cost of transmission packets.

Data redundancy analysis. Data duplication is a common problem in a rapid growth of multimedia content over the internet. To improve cloud backup efficiency deduplication detection performed within client and server based on an application [17]. A recent study prevents duplication efficiently by storing video application tradeoff between computer and storage to retain the copy with the highest quality [18]. To detect exact redundancy (duplication) hash approach (e.g., MD5 or SHA1 algorithm) is appropriate. However, hash approach to detect near redundant (*i.e.*; data contents which look similar to a human viewer) content could be inappropriate. Because hash approach detects near duplicate content as different. A Bellare *et al.* presented DupLESS, a solution to deduplication storage with secure encrypted [19]. Another similar research work but target duplication storage at client side [20].

Multimedia message analysis. An author's research on attitude study of student's adoption of WhatsApp app learning through multimedia messages perform better results [21]. Authors propose energy efficient multimedia transmission packet size which depends on channel availability and secondary user preference in cognitive radio [22]. The hash coding has been used for encoding images and documents for indexing and

quick searching [23, 24]. A survey on user recent adoption of instant messenger multimedia messaging due to lower cost and unlimited messages compared to SMS (Short Message Service) [25].

Aforementioned research work findings does not focus on redundant multimedia messages inefficiencies in instant messenger app. However, aforementioned research does focus on redundant use of hash function techniques verifies data integrity for error, duplication and fast indexing detection. IM app user shares redundant multimedia message that can utilize hash approach technique to verify hashes of a multimedia message can prevent redundant local or global storage and lead to wireless network lower energy consumption in smartphones.

3. Data Collection

Multimedia message service (MMS) provides more multimedia communication with entertainment effects than current text-based short message service (SMS). The focus of this work is on instant messenger multimedia message. We are interested into analyzing instant messenger when a user is sharing multimedia message which translates into smartphone energy consumption. This can reveal instant messenger multimedia message sharing inefficiencies in terms of energy consumption in smartphones. Our IM data collected consists of eight real user's single chat (one to one) and group chat (one to many) multimedia conversations that were collected between January 14, 2015 and December 14, 2015. The collected IM multimedia messages conversion was encrypted by 256-bit Advanced Encrypted Standard key. A parser was used for parsing multimedia messages from individual user smartphone. IM data collected logs period contained 8 WeChat instant messenger accounts and individual user device conversions. The collected multimedia messages types are images (.jpg format), video (.mp4 format), audio (.amr, .aac, .mp3 formats) and web links. Web links use comprehensive way to display and manage links to external websites contents text, images, video and audio. IM users can share Web links with another user. IM users can also share directly multimedia message files such as images, video and audio types to another user. WeChat IM app was launched by Tencent in 2011, but it has potential gain in user growth. According to 2015 statistics, WeChat has 600 million monthly active users [26]. Regarding ages of WeChat IM users data collected: 4 users were between 18 to 24 years, 2 users were between 25 to 30 and 2 users were between 31 to 40. Regarding WeChat IM users country, 4 users were from China, 1 user was from Pakistan, 1 user was from Russia, 1 user was from England and 1 user was from Kenya.

We are focusing on IM app user share multimedia message inefficiencies in term of energy consumption. IM diverse users data collected contains the classification of shared multimedia messages and IM app performs upload and download multimedia message sharing inefficiencies.

3.1. Classification of Shared Multimedia Messages

In this section, we describe the different types of multimedia messages shared by IM users. Instant messenger allows a user to share multimedia messages in, various media types: images, web links, video, and voice notes. IM users can use a smartphone camera to take a new picture (images) and videos *etc.* to share it, but the most multimedia messages shared is by others IM users. By energy prospective, it is interesting to know which user share more multimedia message ultimately consumes more energy in a smartphone. IM user share multimedia messages which requires IM app performs upload / download energy draining operation. Section 3.1.1 describe WeChat 8 users' collected shared multimedia messages.

3.1.1. WeChat Instant Messenger

Figure 2 shows WeChat IM data collected per user when they shared multimedia messages classification. WeChat IM has total 8 users; all of the users have shared a much larger proportion of video (48 %) compared to images (26%), web links (21%) and audio (2.8%). The standard deviation of images type is 5.5%, web links type is 5.7%, video type is 6% and audio type is 1.2%. According to standard deviation user 3 has the lowest images multimedia message and user 7 has the highest images multimedia messages. Looking into web links user 4 has the lowest web links multimedia messages and user 5 has the highest web links multimedia messages. Looking into all users messages video type multimedia has a much larger proportion. However, user 5 has the lowest video multimedia messages and user 1, 4 have the highest video multimedia messages. User 8 has the lowest audio multimedia messages and user 3 has the highest audio multimedia messages.

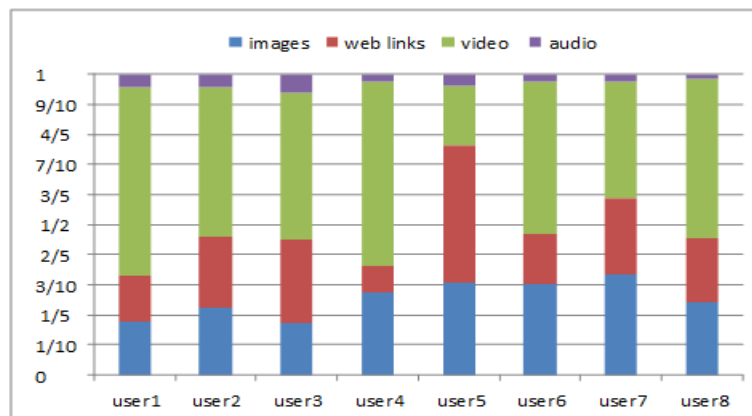


Figure 2. Classification of Shared Multimedia Messages by WeChat users

Table 1. Shows Percentage of Multimedia Messages Shared by WeChat Users

Content type shared	Percentage %
Images	26%
Web links	21%
Video	48%
Audio	2.8%

Table 1 shows WeChat users shared multimedia messages content type percentage. Video multimedia messages are the most common type of multimedia messages shared 48% against images 26%, web links 21% and audio 2.8%. There is a great diversity, a type of multimedia message shared by IM users. For example, few audio messages are shared and many video messages are shared by most users. Differentiating between multimedia message types shared by IM users reveals users smartphone energy consumption. As due to file size video content type 48% shared by IM users energy consumption cost obviously is more than audio 2.8% shared by IM users.

3.2. IM App Performs Upload / Download Multimedia Messages Sharing Inefficiencies

Sharing original multimedia messages to buddy list recipients is dependent on action of the IM app user. The IM app user shares an original multimedia message with multi users recipients are the most interesting value from energy consumption prospective. Since we

observed previously in Figure 1 (a) (b) (c), we find shortcomings where by when IM user share original multimedia message with multi users recipients causing upload and download of redundant multimedia messages. In our data collected from 8 IM users database. Users have shared a much larger proportion of video 48 % compared to images 26%, web links 21% and audio 2.8%. Further we find much of the percentage shared multimedia messages is redundant. We extract each shared multimedia message by SQL query to detect redundant records in their SQLite databases. The query returns a collection of redundant multimedia messages images, web links, video and audio type records illustrated in Figure 3 and Figure 4. We found a large number of IM users share a multimedia message with multi users recipients with redundant multimedia message records in a database. IM app user share with multi users recipients with redundant multimedia message burden on local (smartphone) and global (IM server) storage and leads to wireless network causing high energy consumption in smartphones. We describe multimedia message upload and download inefficiencies separately in subsections 3.2.1 and 3.2.2 respectively.

3.2.1. Upload Multimedia Messages Inefficiencies

IM app use'sr sharing a multimedia message with multi users recipients requires it to be uploaded at IM server. Figure 3 shows the percentage of different type of redundant multimedia messages when IM user shared with multi users. Further shared redundant multimedia messages have classes but in different proportion image, web links, video and audio. As shown in Figure 3, IM user's shared redundant multimedia messages with 3-4 users are predominant. The least redundant multimedia messages shared is with 9-10 users. For example redundant multimedia messages shared with 3-4 users have much larger proportion of redundant multimedia messages compared to when shared with 9-10 users. As shared with 3-4 users (25% images, 8% web links, video 60%, audio 5%) against shared with 9-10 users (5% images, 2% web links, 15% video and 1% audio). It means the most shared redundant multimedia messages are with 3-4 users. We can observe video is the most shared redundant multimedia messages among all shared. According to standard deviation images 7%, web links 4%, audio 2%. The highest redundant images shared are with 3-4 users and least redundant images shared are with 9-10 users. The most web links shared are with 1-2 users and least redundant web links shared is with 9-10 users. The most redundant audio shared are with 1-2, 3-4, and 6-7 users. The least redundant audio shared is with 9-10 users.

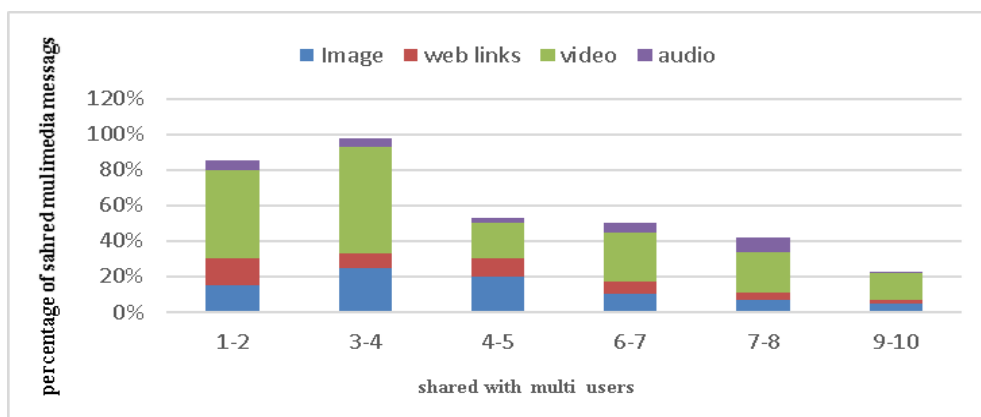


Figure 3. Shows Redundant Multimedia Messages Upload to Share With Multi Users

3.2.2. Download Multimedia Messages Inefficiencies

IM user recipients to view or listen to shared multimedia messages by other IM users which require to download it from IM server. According to IM users experience and also found in our data collected database repost already shared multimedia messages by other IM users shared at different time interval causes IM user to download redundant multimedia message to view or listen to them repeatedly. Figure 4 shows a percentage of different type of redundant multimedia messages downloaded and shared by other multi users. We found that many redundant multimedia messages exist in local storage and their related local path in IM app database. Further, downloading of redundant multimedia messages have classes but in different proportion image, web links, video and audio. In Figure 4, shared by 3-4 and 4-5 users has only images and video type's redundant multimedia messages. We observe that IM user download shared by 1-2 users redundant multimedia messages are predominant. The least time download of redundant multimedia messages are shared by 4-5 users. For example, 1-2 users compare to 4-5 users have download of redundant multimedia messages have much larger proportion (20 % images, 3% web links, video 8%, audio 2% against 5% images and 4% video). We can observe images are the most shared. Secondly, videos are most shared. Web links and audio multimedia messages have the least amount of sharing.

Current instant messenger applications may not be energy efficiently functioning, especially when IM user's share multimedia messages with multi recipients. Thus, results are for sharing an original multimedia message with multi users recipients IM app performs upload / download causes redundancy of multimedia messages in storage and transmission redundant data contents over wireless network which causes energy waste in smartphones battery. To overcome IM app user share redundant multimedia messages, we propose an energy efficient multimedia messaging sharing design in section 4.

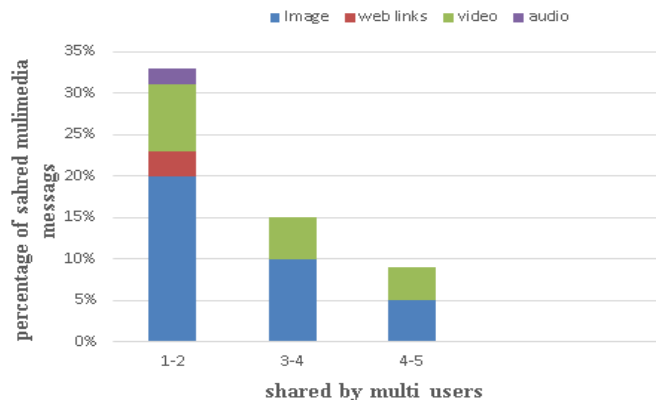


Figure 4. Shows Redundant Multimedia Messages download, Share by other Multi Users

4. Energy Efficient Multimedia Message Sharing Design

In this section, the objective is to obviate the problems with IM app allow perform upload / download redundant multimedia message and provide a simple and energy efficient multimedia message sharing. A well energy efficient multimedia message sharing should verify integrity of multimedia messages to prevent redundant file upload / download. We chose MD5 (Message Digest algorithm 5) algorithm to verify hashes and generate a hash key for redundant multimedia messages. In our proposal IM app upload /

download redundant multimedia message access by hash key. Figure 5 shows energy efficient multimedia message sharing design. Details are discussed below.

Local hash key: hash key which refers to already existing multimedia message in local IM app database record.

Global hash key: hash key which refers to already existing multimedia message in global IM server database record.

Global hash function: A MD5 algorithm hash function to generate a unique hash key for a multimedia message to verify data integrity at the server side.

System message: To carry SQL query between IM app and IM server to match local and global hash key.

Original multimedia message: This contains original file size of the multimedia message (images, web links, video and audio).

IM server: It is a middleware server that handles other IM distributed server authentication server, content delivery network and verifies data integrity.

Authentication server: An authentication server check and provides application users accounts, names and passwords.

Content delivery network: It handles contents such as multimedia messages upload / download based on IM user's geography.

Verify data integrity: A server side check to generate a hash key for a multimedia message. In addition, verifies the hashes and match local hash key with the global hash key of multimedia messages.

Current practice:

1. Establish a connection between smartphone IM app and IM server.
2. IM app performs upload / download to share original multimedia message with multi user recipients.

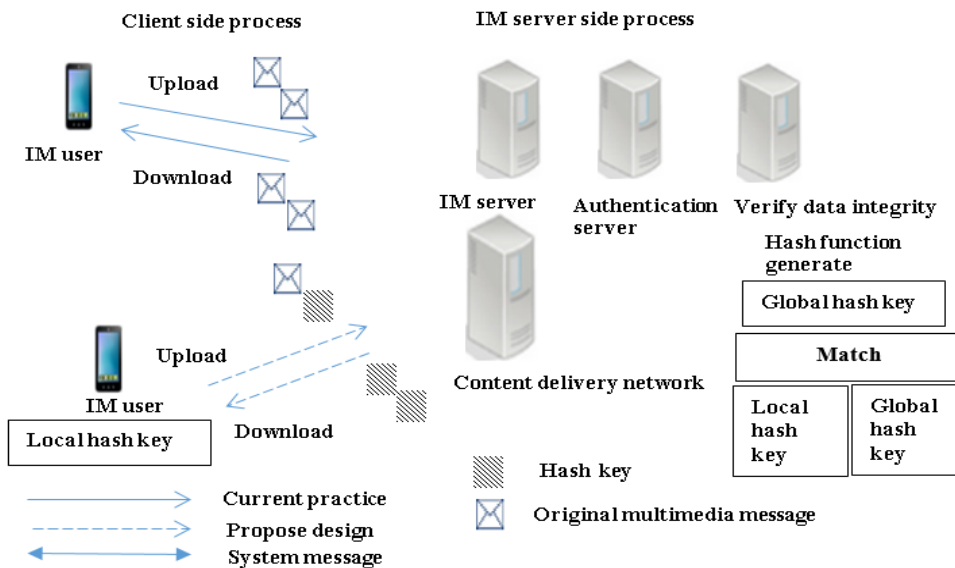


Figure 5. Energy Efficient Multimedia Message Sharing Design

Propose design:

1. Establish a connection between smartphone IM app and IM server.
2. Generate hash code key MD5 (message –digest algorithm) for original multimedia messages. In addition, prior upload / download verify hashes of the multimedia message.
3. IM perform upload / download to share redundant multimedia message with multi user recipients use hash key. Propose design upload and download details describe below.

Steps to upload multimedia message shown in Figure 6.

- Step 1. File to be upload.
- Step 2. System messages send SQL query match multimedia message local hash key with global hash key hashes at IM server.
- Step 3. If hash key is not identical than uploads the original multimedia message.
- Step 4. Generate a global hash key by hash function at IM server database.
- Step 5. If hash key identical. Share a hash key and recipients list to IM server.

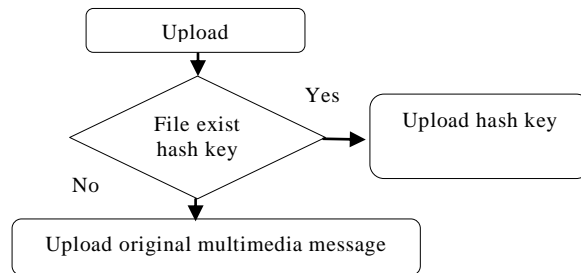


Figure 6. Flow Diagram Upload Multimedia Message

Steps to download multimedia message shown in Figure 7.

- Step 1. File to be downloaded.
- Step 2. System message SQL query match the global hash key with local key.
- Step 3. If local hash key is not identical.
- Step 4. Downloads original multimedia message with the global hash key.
- Step 5. If local hash key is identical.
- Step 6. Do not download a multimedia message from IM server. Open a multimedia message from IM local path database storage.
- Step 7. Store global hash key multimedia message as a local hash key in app local storage database.

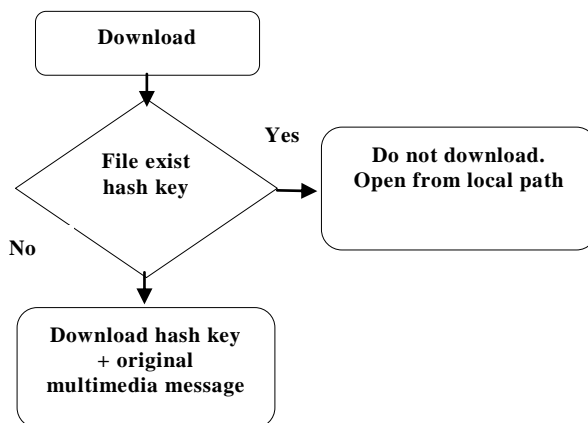


Figure 7. Flow Diagram Download Multimedia Message

4.1. Variable Definition

In this section, we present a variable definition use for energy efficient multimedia message sharing to prevent IM app to perform redundant upload or download.

$$M = \{m_1, m_2, m_3, \dots, m_n\} \text{ Original multimedia messages.}$$

$U = \{u_1, u_2, u_3, \dots, u_n\}$ IM app users.

$S = \{s_1, s_2, s_3, \dots, s_n\}$ number of times multimedia message shared. Where $S > 0$.

$\lambda_G = \{\lambda_{G1}, \lambda_{G2}, \lambda_{G3}, \dots, \lambda_{Gn}\}$ global hash key is denoted as λ_G .

$\lambda_L = \{\lambda_{L1}, \lambda_{L2}, \lambda_{L3}, \dots, \lambda_{Ln}\}$ local hash key is denoted as λ_L .

$T = \{t_1, t_2, t_3, \dots, t_n\}$ is total time spend to upload / download transmit M multimedia messages and λ_L or λ_G binary hash key .

$\Phi P = \{\Phi P_1, \Phi P_2, \Phi P_3, \dots, \Phi P_n\}$ denote total energy consumption cost spend on

successful transmission of S sharing of M multimedia message and λ_L or λ_G hash key user U upload and download.

$X = \{x_1, x_2, x_3, \dots, x_n\}$ number of uploads.

$Y = \{y_1, y_2, y_3, \dots, y_n\}$ number of downloads.

4.2. Verify Multimedia Message Hashes

A frequently used solution to verify data integrity redundancy is to check hashes of original file that already exist in local (smartphone storage) or global (server storage) database. It is to calculate hashes of a file. In our design, a server side verifies data integrity and generate a hash key used to refer of original multimedia message which is stored in a local or global database. IM app performs upload / download to access redundant multimedia messages by using hash keys rather than the original multimedia message.

Our design has a direct positive impact on IM app preventing when user shares a redundant multimedia message. Our design benefits by preventing redundant content in storage and no redundant content transmission over wireless network in a smartphone. However the approach in our proposal to generate a global hash key by hash function at server side, would provide fast processing verifying redundancy check for IM app prior share multimedia message. System message provides querying about local / global hash key between IM app and IM server prior to IM app upload / download of multimedia message.

5. Evaluation

We evaluate our energy efficient multimedia message sharing design in this section. We are interested to get energy consumption results of access hash key and original multimedia message therefore we first describe the measurement tool, choosing hashing algorithm to check data integrity and test environment setup. Finally, show results in terms of energy efficient multimedia message sharing.

5.1. Measurement Tool

We recorded hash key and original multimedia messages energy consumption by Monsoon meter and network traffic by tcpdump in Samsung I9300 Galaxy S III smartphone. Figure 8 shows a Dell desktop computer used to log power monitor traces console. Monsoon meter supplies a stable 4.2 voltage to the smartphone and tcpdump ran on a smartphone to capture network traffic. We conducted a test on 3G cellular internet service from CMCC (China mobile Communication Corporation) 3G cellular network data connection and setting a Wi-Fi access point in Dell desktop computer in our computer laboratory LAN to access Wi-Fi internet service in Samsung I9300 Galaxy S III

smartphone. To limit others application network traffic” ES Task Manager (Task Killer)” app was used to kill additional apps running in a smartphone.

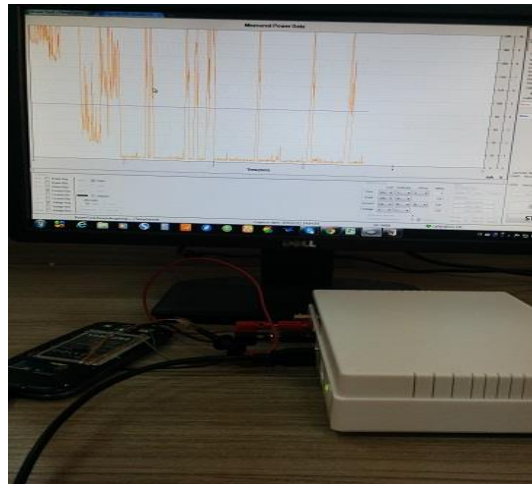


Figure 8. Experimental Setup to Measure Network Traffic and Power Measurement

5.2. Choosing Hashing Algorithm

Many commonly used type hash algorithm like MD5, SHA-1, and SHA-256 are available. We preferred to use MD5 (message –digest algorithm) 128 bit (32 hexadecimal digits) having digest hash key size smaller than SHA-1 (Secure Hash Algorithm 1) 160 bit and SHA-256 (Secure Hash Algorithm 2) 256 bits. Table 2 shows the use of MD5 algorithm [27] online tool available to generate hash key for our multimedia message 2 MB .mp4 video clip file. MD5 online tool return a 32 character digit. It is shown in Table 2 used video clip multimedia message file attributes includes type of algorithm use, format type, name, size and hash key.

Table 2. Shows Using MD5 Generate a Hash Key for Video Clip

Hash algorithm type	MD5
multimedia message format type	.mp4
Video clip name	abc
Size	2097152
Hash key 32 digits	d8c2eafd90c266e19ab9dcacc479f8af

5.3. Test Environment Setup

We used hash key and original multimedia message in our test environment as shown in Table 2. We evaluate our hash key using Android Push Notification [27] androidpn (an open source code project used for push notification messaging). Androidpn use to send a hash key between smartphone (act as IM app) andriodpn server (act as IM server). However, androidpn does not support multimedia messages. Further, we use WeChat app user accounts to upload / download original multimedia message to get energy consumption results.

Android Push Notification is an open source code project that pushes messages notification which supports android. It supports XMPP (Extensible Messaging and Presence Protocol) the most famous instant messenger protocol used by Gtalk, Facebook chat *etc.* The androidpn supports XMPP protocol which support asynchronous

bidirectional messages sent and received between smartphone and server. It means not only server sends messages to the smartphone, but a smartphone can also send messages back to the server. Monsoon meter and tcpdump tool were used to metered results on a smartphone.

5.4. Energy Saving Cost Sharing Multimedia Message

In this section, the effectiveness of our energy efficient multimedia message sharing will be determined. Table 3 shows energy consumption test results sharing multimedia messages current practice and proposed design with recipient. But in our proposed design share with recipient, original multimedia message and already existing redundant multimedia message share access use global (IM server) hash key and local (IM app) hash key.

Table 3. Energy Consumption Comparison Results between User Upload and Download Of Original Multimedia Message And Use Hash Key To Access Redundant Multimedia Message

	Messages shared $S=\{s1, s2\}$	Recipients / sender	Download1 y_1	Download2 y_2	Upload 1 x_1	Upload2 x_2	Energy Consumption ϕP Wi-Fi	Power Consumption ϕP 3G
Current practice	m_1	u_1, u_2, u_3	m_1	m_1	m_1	m_1	309 mA	418 mA
Propose design	m_1	u_1, u_2, u_3	$m_1 + \lambda_G$	N/A	λ_L	λ_L	124 mA	167 mA

As shown in Table 3 we can see in current practice share original multimedia message note energy consumption is higher than propose design. The current practice, while user u_1 recipient during total time $T = \{t_1, t_2\}$ shared $S = \{S_1, S_2\}$ by users u_1, u_2 requires download $Y = \{y_1, y_2\}$ and u_1 share $S = \{S_1, S_2\}$ with u_1, u_2 requires upload $X = \{x_1, x_2\}$ use original message m_1 . However, our propose design user u_1 total time $T = \{t_1, t_2\}$ first time shared $S = \{S_1\}$ by u_1 download $Y = \{y_1\}$ requires original message m_1 plus global hash key λ_G and 2 time shared $S = \{S_2\}$ by u_2 $Y = \{y_2\}$ do not allow download redundant m_1 . However user can open m_1 from local storage path in smartphone. Since m_1 already exist in local storage. However, u_1 share $S = \{S_1, S_2\}$ with u_1, u_2 upload $X = \{x_1, x_2\}$ sharing use local hash key to refer redundant multimedia message in IM server. Further, we calculate energy consumption current practice and propose design using 3G, Wi-Fi separately describe below.

Current practice 3G energy consumption

$$\phi P_1 = y_1(m_1) = 110 \text{ mA}$$

$$\phi P_2 = y_2(m_1) = 113 \text{ mA}$$

$$\phi P_3 = x_1(m_1) = 98 \text{ mA}$$

$$\phi P_4 = x_2(m_1) = 97 \text{ mA}$$

$$\Phi P = \Phi P_1 + \Phi P_2 + \Phi P_3 + \Phi P_4 = 418 \text{ mA}$$

Current practice Wi-Fi energy consumption

$$\phi P_1 = y_1(m_1) = 86 \text{ mA}$$

$$\phi P_2 = y_2(m_1) = 87 \text{ mA}$$

$$\phi P_3 = x_1(m_1) = 66 \text{ mA}$$

$$\phi P_4 = x_2(m_1) = 70 \text{ mA}$$

$$\Phi P = \Phi P_1 + \Phi P_2 + \Phi P_3 + \Phi P_4 = 309 \text{ mA}$$

Total energy consumption in current practice download / upload 3G consume 418 mA and using Wi-Fi consume 309 mA.

Propose practice 3G energy consumption

$$\phi P_1 = y_1(m_1 + \lambda_G) = 125 \text{ mA}$$

$$\phi P_2 = y_2 = \text{N/A}$$

$$\phi P_3 = x_1(\lambda_L) = 20 \text{ mA}$$

$$\phi P_4 = x_2(\lambda_L) = 22 \text{ mA}$$

$$\Phi P = \Phi P_1 + \Phi P_2 + \Phi P_3 + \Phi P_4 = 167 \text{ mA}$$

Propose practice Wi-Fi energy consumption

$$\phi P_1 = y_1(m_1 + \lambda_G) = 95 \text{ mA}$$

$$\phi P_2 = y_2 = \text{N/A}$$

$$\phi P_3 = x_1(\lambda_L) = 15 \text{ mA}$$

$$\phi P_4 = x_2(\lambda_L) = 14 \text{ mA}$$

$$\Phi P = \Phi P_1 + \Phi P_2 + \Phi P_3 + \Phi P_4 = 124 \text{ mA}$$

Total energy consumption in propose design download / upload 3G consume 167 mA and using Wi-Fi consume 124 mA.

Based on results above, propose design sharing multimedia message consume 60% less energy consumption than current practice. In summary, use a hash key in IM app perform redundant upload / download cost of transmitting over wireless network is lower in 3G and Wi-Fi. It is fair enough to use hash key to access already existing redundant multimedia messages in local or global storage database. Hence, IM app shares multimedia message use hash key to access already existing redundant multimedia message save a lot of power.

6. Conclusions

Redundant multimedia messages in a large number shared by instant message app have a potential impact on server storage, end user device storage, and energy consumption. A challenging first step is to determine an accurate measurement of redundant network traffic generated by IM app. In this paper, we performed an experiment with real user's data collection to better understand multimedia messages shared by IM users. We observe IM app allows a user to share a redundant multimedia message which obviously causes redundant network traffic. We provided an energy inefficiencies analysis of sharing multimedia message by instant messenger app. To conclude findings, IM app user share with recipient's redundant multimedia message which causes extra content copies in database storage and redundant content transmission leads to a wireless network having high energy consumption in a smartphone.

We proposed to verify data integrity hashes of shared multimedia message before IM app performs upload or download. It helps prevent redundant content in storage and

lowering wireless network energy consumption in a smartphone. In our design, multimedia message sharing energy results shows saved energy consumption up to 60%.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 61309025, 61402543 and 61472451, Hunan Provincial Natural Science Foundation of China under Grant No. 13JJ4016 and International Science and Technology Cooperation Programme of China under Grant No. 2013DFB10070.

References

- [1] G. Maier, F. Schneider and A. Feldmann, "A first look at mobile hand-held device traffic", in *Passive and Active Measurement*, Springer, (2010), pp. 161–170.
- [2] S. K. Baghel, K. Keshav and V. R. Manepalli, "An investigation into traffic analysis for diverse data applications on smartphones", in *Communications (NCC), 2012 National Conference on, IEEE*, (2012), pp. 1–5.
- [3] D. T. Nguyen, G. Zhou, X. Qi, G. Peng, J. Zhao, T. Nguyen, and D. Le, "Storage-aware smartphone energy savings," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, ACM, (2013), pp. 677–686.
- [4] B. public service broadcaster, <http://www.bbc.com/news/business-34773936>, vol. 11, (2015)..
- [5] I. Monsoon Solutions, www.msoon.com/labequipment/powermonitor/, vol. 03, (2014).
- [6] S. McCanne, C. Leres, and V. Jacobson, "Tcpdump and libpcap," (2012).
- [7] Google, <https://productforums.google.com/forum/#!topic/youtube/yc7hhqwtbx0>, vol. 2, (2016).
- [8] S. Jeong, K. Lee, J. Hwang, S. Lee and Y. Won, "Androstep: Android storage performance analysis tool", in *Software Engineering (Workshops)*, vol. 13, (2013), pp. 327–340.
- [9] M. Kim, S. Lee and Y. Won, "Comparative study on i/o characteristics of mobile web browsers".
- [10] K. Lee and Y. Won, "Smart layers and dumb result: Io characterization of an android-based smartphone", in *Proceedings of the tenth ACM international conference on Embedded software*, ACM, (2012), pp. 23–32.
- [11] Y. Yang, K.-P. Chow, L. Hui, C. Wang, L. Chen, Z. Chen and J. Chen, "Forensic analysis of popular chinese internet applications", in *Advances in Digital Forensics VI*, Springer, (2010), pp. 285–295.
- [12] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen and O. Spatscheck, "Profiling resource usage for mobile applications: a cross-layer approach", in *Proceedings of the 9th international conference on Mobile systems, applications, and services*, ACM, (2011), pp. 321–334.
- [13] E. J. V. Alonso, S. Andersson and S. Nadjm-Tehrani, "When mice consume like elephants: Instant messaging applications".
- [14] L.-S. Meng, D.-s. Shiu, P.-C. Yeh, K.-C. Chen and H.-Y. Lo, "Low power consumption solutions for mobile instant messaging", *Mobile Computing, IEEE Transactions on*, vol. 11, no. 6, (2012), pp. 896–904.
- [15] L. Zhang, C. Xu, P. H. Pathak and P. Mohapatra, "Characterizing instant messaging apps on smartphones", in *Passive and Active Measurement*, Springer, (2015), pp. 83–95.
- [16] E. J. Vergara Alonso, "Exploiting energy awareness in mobile communication", (2013).
- [17] Y. Fu, H. Jiang, N. Xiao, L. Tian, F. Liu and L. Xu, "Application-aware local-global source deduplication for cloud backup services of personal storage", *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 5, (2014), pp. 1155–1165.
- [18] A. Kathpal, M. Kulkarni and A. Bakre, "Analyzing compute vs. storage tradeoff for video-aware storage efficiency", in *HotStorage*, (2012).
- [19] S. Keelveedhi, M. Bellare, and T. Ristenpart, "Dupless: server-aided encryption for deduplicated storage," in *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*, (2013), pp. 179–194.
- [20] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data deduplication scheme for cloud storage," in *International Conference on Financial Cryptography and Data Security*, Springer, (2014), pp. 99–118.
- [21] S. So, "Mobile instant messaging support for teaching and learning in higher education," *The Internet and Higher Education*, vol. 31, (2016), pp. 32 – 42.
- [22] X.-L. Huang, G. Wang, F. Hu, S. Kumar and Y. Zhang, "Optimal packet size design for multimedia transmissions in cognitive radio networks", in *Communications and Networking in China (CHINACOM), 2011 6th International ICST Conference on, IEEE*, (2011), pp. 827–830.
- [23] F. Shen, C. Shen, Q. Shi, A. van den Hengel, Z. Tang and H. T. Shen, "Hashing on nonlinear manifolds", *Image Processing, IEEE Transactions on*, vol. 24, no. 6, (2015), pp. 1839–1851.
- [24] X. Zhu, Z. Huang, H. Cheng, J. Cui and H. T. Shen, "Sparse hashing for fast multimedia search", *ACM Transactions on Information Systems (TOIS)*, vol. 31, no. 2, (2013), pp. 9.

- [25] K. Church and R. de Oliveira, "What's up with whatsapp?: comparing mobile instant messaging behaviors with traditional sms", in Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services, ACM, (2013), pp. 352–361.
- [26] S. Millward, <https://www.techinasia.com/wechat-438-million-active-users-q2-2014/>, vol. 10, (2014).
- [27] <https://md5file.com/calculator>, vol. 2, (2016).

Authors



Pinial Khan Butt, He received master's degree in Telecommunication Technology from University of Sindh, Pakistan in 2002. He is currently a Doctoral Degree Candidate at the School of Information Science and Engineering, Central South University, Changsha, Hunan China. His research interests include E-Commerce , Telecommunication Technology and Energy Efficient Mobile Computing.



Wang Yang, He received B.Sc. in Computer Science and technology from National University of Defense Technology, China in 2004, and Ph.D. in Computer Science and technology from Tsinghua University, China in 2011. He is now an Assistant Professor in School of Information Science and Engineering, Central South University, China. His research interests include mobile computing, sustainable computing and information centric networking.



Guojun Wang, He received his B.Sc. in Geophysics, M.Sc. in Computer Science, and Ph.D in Computer Science from Central South University, China. He is currently a Professor of Guangzhou University and Central South University. He has been an Adjunct Professor at Temple University, USA; a Visiting Scholar at Florida Atlantic University, USA; a Visiting Researcher at the University of Aizu, Japan; and a Research Fellow at the Hong Kong Polytechnic University. His research interests include network and information security, Internet of things, and cloud computing. He is a distinguished member of CCF, and a member of IEEE, ACM, and IEICE.