Learn-together a proposed Method that combines Classrooms with Computer Labs

[1]Shrey Purohit, [1]Rutul Sanglekar, [1]Dhrumit Patel, [2]Ms. Neha Minocha

[1]U.G. Student, Department of Computer Engineering, [2]Asst. Professor, Department of Computer Engineering

Department of Computer Engineering,
Sal Institute of Technology & Engineering Research (SALITER),
Ahmedabad- 380060, Gujarat, India

Abstract: This paper reviews a tool that will allow teachers to teach using a web-app Learn together, that will allow teachers to teach with screen sharing, video sharing and real-time chat. We also discuss variety of ways learn-together could be combined with other open source tools to enable high interaction while teaching and learning. This tool promotes hands on approach for learning which works lot better than passive lecture-based approaches that are used. This paper also looks at technology that makes this app possible "WebRTC".

I. INTRODUCTION

Technology is evolving at extremely rapid pace, traditional chalk and board teaching methods are decreasing in importance. These methods fail to satisfy needs of students as well as industry, then what could be purpose of such education. Not just this there is one more problem, computers used at universities are outdated and by no means useful for teaching or learning. So called labs have pentium 4(32 bit), 1 GB ram, no gpu, and outdated development environment with netbeans, oracle, notepad++ etc installed. Once you are in the lab you are supposed to write code as mentioned in some lab manual, then copy it down on paper. This leads a student nowhere. Learning must be active, where students themselves step ahead to solve problems that they've faced. In addition to that most commonly used libraries like tensorflow would not even install on these machines. Setting up development environment in one of the most time-consuming process for programmers, imagine doing that on pc of labs.

Hands on approach where students learn by doing, is need for present day, which is not possible with infrastructure most Indian colleges provides, infrastructure upgrade could solve many problems, but it's not something that would happen easily.This document suggests a small patch / workaround for solving these problems, in the most cost-effective way.Some parts of this document look at how WebRTC the technology used by learn-together (our webapp) works.

II. LITERATURE

Part of literature explains how WebRTC came into existence and evolved rapidly, other part suggests innovative way for teaching and learning.

Google released open source project in 2011 one year after it acquired Global IP Solutions which is company who made basic building blocks for WebRTC like codecs.[17] WebRTC open source code was standardized with efforts from IETF and W3C. [18] Google used it for hangouts this was the first cross browser WebRTC communication. [19] WebRTC can even run without browser in fact browser can be considered as thin layer over openWebRTC.[20]

WebRTC can be modified to optimize further for needs of application one such example is discord it handles about 0.5 million concurrent voice user using WebRTC they use lower level WebRTCAPI, so instead of SDP which can be around 10KB round trip, they perform joining voice channel in just 1000 bytes by sending only information they need Other optimizations such as not using ICE and directly using their own relay server are done. [21]

Some authors have illustrated how WebRTC could help open source education platform moodle, how it could enhance learning with media sharing provided by WebRTC. [11] Some authors have previously used WebRTC in context of STEM learning, these platforms must also be capable of supporting math notation to support STEM. Our solution is much similar but for computer science. [22] Other Alternative solution to problem is to bring entire development environment to could, several large companies like google, Microsoft, amazon have been doing this internally or externally. In fact, our virtual classroom is hosted on cloud 9 which offers cloud IDE. [23] Several universities are adopting methods which promotes self-learning without assistance from teachers. One such is École 42 university with no teachers by french billionaire Xavier Niel, to quote his words "we don't teach anything, Students create what they need all the time." [24] Initially we were inspired by such teaching methods, our learn together platform could be used for peer assisted teaching.

III. How Learn Together Works?

To Join a room, users visit room's URL, and type password if it was set. After which browser gets media stream from user, using getUserMedia(gum)[1] and emits join event which is relayed via signalling server to all other peers. Other peers create RTCPeerConnection[2], creates SDP[3] offer and emits offer event which is relayed to the user by signalling server. User creates an answer to received offer and sends it to other peer (SDP). After receiving offer, peer sets remote description on PeerConnection. Once session is established peer can communicate to any other peer directly without signalling server.

WebRTC plays a narrow role in the entire system. It's build on top of UDP/RTP[4] and allows user to transport media and data. The other parts such as service discovery, session establishment is left to the application. There are standardized ways other tasks. Signalling server which supports SDP can be used to send and receive offer and also used for other tasks such as controlling volume level.

HOW TO HANDLE NAT
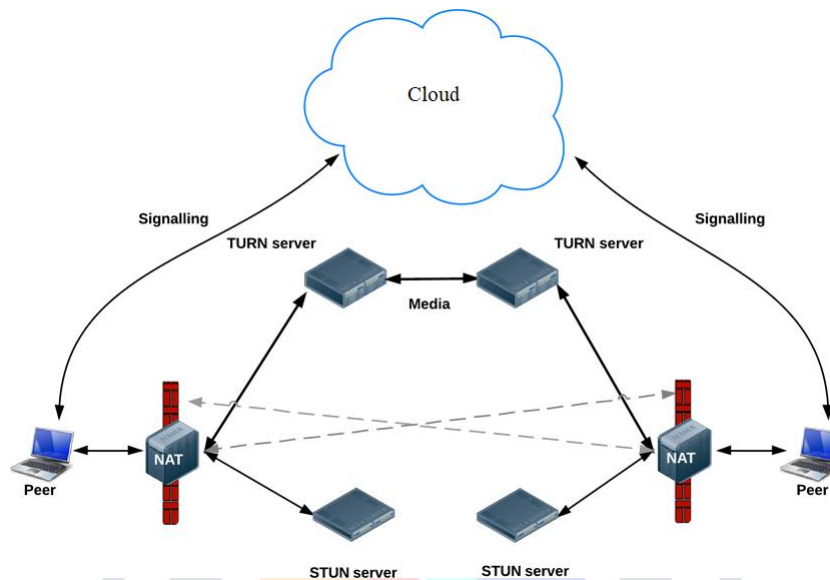
Figure 1: Servers involved in ICE(Interactive Connectivity Establishment)

In most cases users are behind NAT[5] which means they have private IP assigned by NAT which is not globally unique and cannot be used for communication. To tackle this problem ICE (Interactive Connection Establishment) is used[6], ICE tries to find most direct path for communication between two users it falls back to relay if no such path is possible. When user is behind asymmetric NAT, STUN (session traversal utilities for NAT) server gets users public IP which can be used for connection establishment. In case user's behind symmetric NAT, STUN server cannot be used and TURN (traversal using relay NAT) has to be used, TURN stays between users during entire session, media and data are relayed through TURN. Using turn servers makes system more reliable. But also has some added advantages, media could be recorded, or some computer vision algorithm could be applied etc.

There are lots of stun servers available that are free to use[16], but relatively few turn servers available. In our implementation we've used stun servers only we don't have any turn servers.
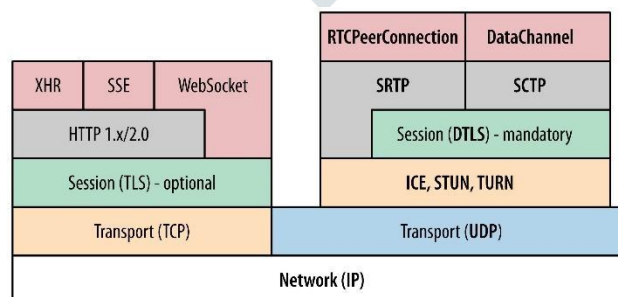
Figure 2:WebRTC Protocol Stack

Here's protocol stack for WebRTC, ICE[8], STUN, TURN are for finding the most direct connection between two network nodes. DTLS[7] provides encryption of streams. On top of it SRTP[9](Secure Real Time Protocol) and STCP(Stream Control Protocol Stack) are used for multiplexing streams, flow control. RTCPeerConnection[2] and DataChannel[10] are provided by WebRTCAPIs. Learn together currently does have any TURN server. In addition to these protocols. Some other important things are getUserMedia for getting media stream from user's device without installing any plugins which is part of HTML5[1].

DISTANT LEARNING

Not everyone has access to good instructors to guide them. Distant learning allows everyone to have access to learning[11]. Using WebRTC offers several advantages over video conferencing apps like skype etc. First being nothing to install anyone with just a web browser can access WebRTC with nothing to install. Over pre-recorded lectures WebRTC offers advantage of interactivity, students can submit their assignment online and instructors could review them. Lower cost as ICE tries to find most direct way to communicate, between two users. The expenses of high-performance servers that handle media relay can be reduced significantly, but using media relay as fallback mechanism.

COMBINE WITH OTHER TOOLS

There are lots of other tools for interactive coding, such as JupyterNotebook[12], such tools reduce complexity of setting up development environment. Combine JupyterNotebooks with learn together you can get interactive data science/ machine learning/ python/ computer vision bootcamp, all that for zero cost[13]. Combine learn together with slides and you get a team presentation or office meeting. Take away all the UI you get direct file transfer without limitation on the file size[14]. While all of above use cases were possible before WebRTC, but WebRTC offers the same functionality with fraction of cost.

DISADVANTAGE USING WEBRTC

While WebRTC promises real time communication between browsers and other devices without need of installing any plugins WebRTC and the way it is used in learn-together creates some problems.

CPU memory and Network load (Mesh Topology):

In learn together, We do not have any TURN servers, which means every peer is connected to every other peer in mesh topology, This is common way of using WebRTC[15]. Of course, one can disable video streams from other users, but instructors cannot do it, let's consider an example of room where we have 20 students + 1 instructor has to send video, audio and data stream to 20 people and receive stream from 20 people, thus his computer has to handle 40 media and data streams simultaneously. In our experiments when we were able to handle max 5 users in peer to peer fashion that means each user handles 10 connections and displays 5 video streams. Of course, this could be handled by using relay servers, but that would increase the costs.

IV. RESULTS AND CONCLUSION

Originally learn together was designed to allow teachers bring their own development environment and teach students interactively, without limitations of what infrastructure institutes offers. Many institutes have hardware that is too restrictive to allow to teach anything useful. With learn together instructor creates virtual classroom called rooms, using their own hardware and students can join the room, to learn interactively, allowing instructor to teach and monitor their student's progress. But it can be useful in other scenarios too.

Picture illustrates scenario where user is teaching YOLO (deep learning algorithm of object detection) to students using learn together and google'scolab. Combination of learn together and colab allows to teach deep learning, in hands on way which would be impossible with infrastructure provided by university. This learning environment costs nothing.

WebRTC has potential to revolutionize teaching, it can make teaching and learning much more interactive than before, at the same time it could make education more accessible for all. It is the most cost-effective solution, which could be materialized by utilizing resources available for free. WebRTC is much more powerful when combined with other tools such as jupyter notebook or online coding playgrounds available for different languages.
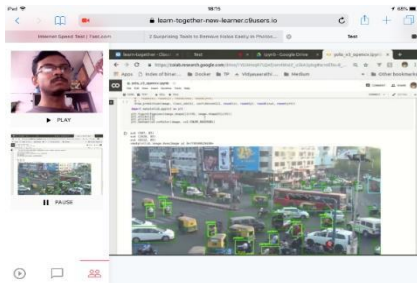

Screen Shot: 1

In start screen, Room name and user name are entered. User does not need to sign up or login.
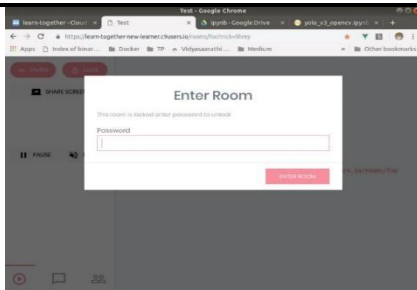
Screen Shot: 2

In password screen, Password is asked if it's set by instructor (it's optional).

Screen Shot: 3

Student screen shows screen of teacher's desktopand teacher's video.

V. FUTURE POTENTIAL

Learn together is open for lot of improvements that could be done in future, such as complete offline operation by using universities internal network. On the other hand, if it's used in online mode, it could be used to help virtual technological conferences. Learn together is just an example of how combining open source technologies could help tackle problems like poor infrastructure and theory focused passive teaching methods without heavy investments in infrastructure upgrade or restructuring education.

REFERENCES

[1] World Wide Web Consortium MediaCapture Stream www.w3.org/TR/mediacapture-streams/
[2] Mozilla Developer Network RTCPeerConnectiondeveloper.mozilla.org/en-US/docs/Web/API/RTCPeerConnection
[3] Internet Engineering Task Force -tools.ietf.org/html/rfc4566
[4] HTML 5 Rocks Webrtc Basics www.html5rocks.com/en/tutorials/webrtc/basics/
[5] How Stuff Works - Network address translation computer.howstuffworks.com/nat.htm
[6] Internet Engineering Task Force-Interactive Connectivity Establishment tools.ietf.org/html/rfc5245
[7] Internet Engineering Task Force-Datagram Transport Layer Security tools.ietf.org/html/rfc4347
[8] High Performance Browser Networking Webrtc Protocol Stack Image hpbn.co/
[9] Internet Engineering Task Force-Secure Real Time Protocoltools.ietf.org/html/rfc3711
[10] Mozilla Developer Network Data channels
developer.mozilla.org/en-US/docs/Web/API/WebRTC_API/Using_data_channels
[11] S. Ouya, K. Sylla, P. M. D. Faye, M. Y. Sow and C. Lishou, "Impact of integrating WebRTC in universities' e-learning platforms," *2015 5th World Congress on Information and Communication Technologies (WICT)*, Marrakech, 2015, pp. 13-17. ieeexplore.ieee.org/document/7489664/
[12]Jupyter Notebookjupyter.org/
[13] Google Research Colabcolab.research.google.com/
[14]Webrtc Direct File Transfer webrtc.github.io/samples/src/content/datachannel/filetransfer/
[15] Webrtcglosarry.com Webrtc Mesh topology webrtcglossary.com/mesh/
[16]Githubzziuni - List of free stun servers gist.github.com/zziuni/3741933/
[17]Gipscorp developed initial webrtcwww.gipscorp.com/
[18]Webrtc Working Group datatracker.ietf.org/wg/rtcweb/charter/
[19] Szymon Nowak webrtcpersentationszimek.github.io/presentation-meetjs-summit-2014-webrtc/
[20]Openwebrtcericsson research githubgithub.com/EricssonResearch/openwebrtc
[21] Discord blog on medium blog.discordapp.com/how-discord-handles-two-and-half-million-concurrent-voice-users-using-webrtc-ce01c3187429
[22] Mamadou Djidiack Faye, Pape &Gueye, Amadou&Lishou, Claude. (2018). Virtual Classroom Solution with WebRTC in a Collaborative Context in Mathematics Learning Situation. 10.1007/978-3-319-72965-7_6.
www.researchgate.net/publication/322679256_Virtual_Classroom_Solution_with_WebRTC_in_a_Collaborative_Context_in_Mathematics_Learning_Situation
[23] Cloud9 cloud IDE c9.io/
[24] QZ.com interview with Xavier Niel
qz.com/1054412/a-french-billionaires-free-teacher-less-university-is-designing-thousands-of-future-proof-employees/