



Disturbance Detection for MIM Apps.

The Project Proposal

Content.

- Introduction.
- Motivation.
- Innovation.
- Purpose of the project.
- Plan Of Action.
- Scope of the project.
- References.

Introduction.

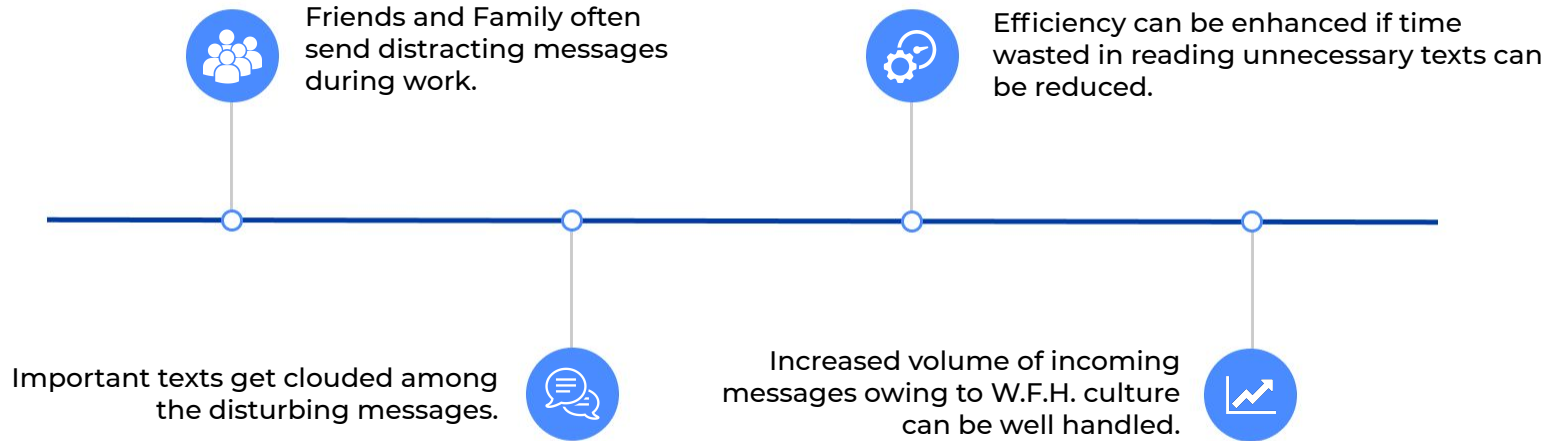
- Instant Messaging Apps have become a part of our day-to-day lives.
- 65% of work places have started adopting MIM apps for work related communication.
- Productivity is compromised owing to the huge time wasted in filtering out messages and information which is useful/relevant.
- No existing technology solves the above mentioned problem.

Abstract.

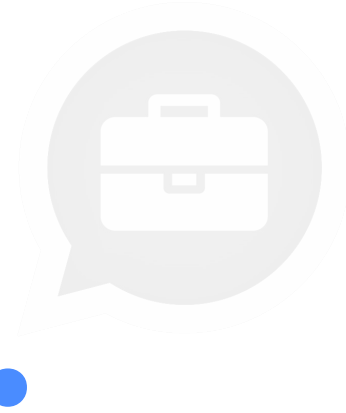
Instant messaging platforms have grown exponentially in popularity over the last few decades, especially after the Covid-19 Pandemic. This has led to a sudden influx in the number of messages received from important and unimportant sources. No existing technology enables the users to filter important content from the loads of messages received daily.

We aim to solve the above issue with the help of a multi-modal deep learning architecture coupled with a React Native MIM app that enables the users to filter distracting content according to their needs (based on if Work Mode is enabled).

Motivation.



The Innovation.

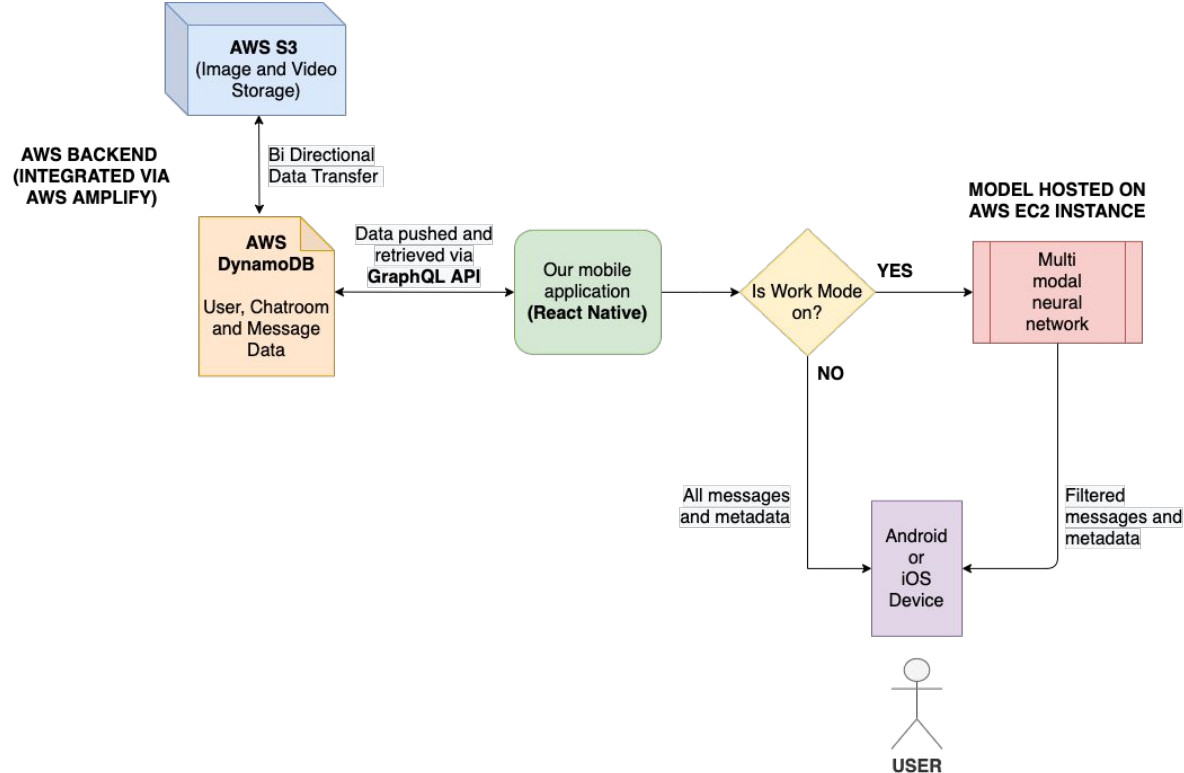


Innovation.

Mobile App Development:

- **React Native** (Developed by Facebook in 2015) will be used for the app development, which assures platform independence (for both Android and iOS).
- We plan to use **GraphQL** for API Querying which offers more flexibility and stability than the commonly used REST Apis.
- **AWS Amplify** will be used for integrating the mobile app with an AWS Backend.
- **Recoil** will be used for Global State Management.

Mobile App Architecture.



Innovation.

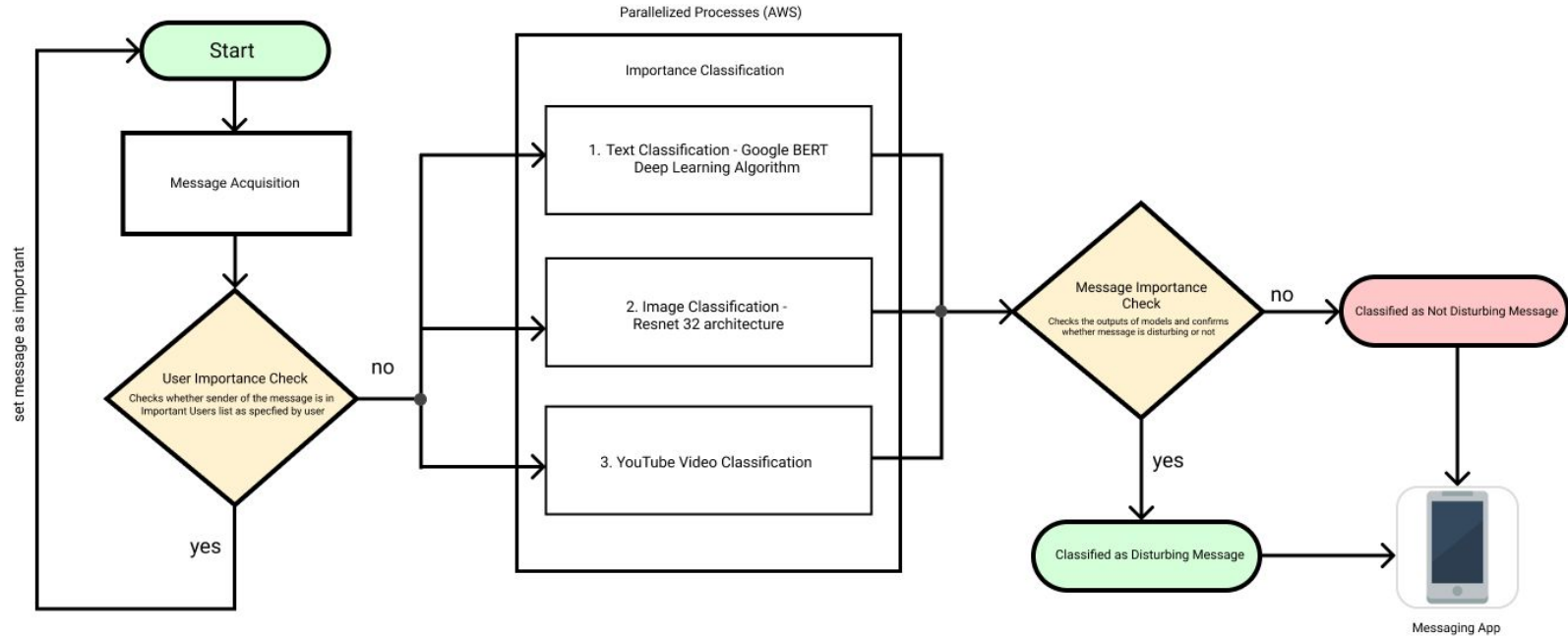
Deep Learning:

- We plan to use a **multi-modal deep learning** architecture for classifying disturbing messaging.
- We will be using **3 different models** for text classification, image classification and YouTube video classification using YouTube API.
- User can also set importance of messages within the app according to their needs.
- We will be using CI & CD procedures to accomplish bi-weekly retraining of our models on AWS EC2.

Deep Learning Architecture.



SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956



Purpose of The Project.

- Our project seeks to increase and improve work efficiency drastically.
- Handling huge influx of incoming unread text messages is a tedious task for one who is already burdened with workload and stress.
- Only the essential messages will be visible hence saving time and effort in having to scroll through a long conversation for a particular message.
- Our technology will help MIMs to be adopted by professionals who were so far refraining from it due to the previously mentioned drawbacks.

Plan of Action.

- **Review 1:**
 - App Dev - Frontend and App UI.
 - DL - Text message classification (BERT Algorithm) & image classification (Resnet 32 Algorithm).
- **Review 2:**
 - App Dev - AWS Backend integration via AWS Amplify.
 - DL - YouTube API, YouTube video classification & refactoring code for deployment.
- **Review 3:**
 - App Dev - Final Integration with ML model on AWS EC2.
 - ML - Model fine tuning, deployment and CI & CD.

Scope of the project.

- Our final deliverable will be a **mobile application** (both Android and iOS supported) that will be fully functional with an AWS Backend and DL models hosted on AWS EC2.
- The app will be equipped to handle disturbance detection for work based filtering of multiple data types like text, images and video links.
- The user will be given the provision to tailor the deep learning model to his/her preferences using CI & CD.
- The source code will be kept clean and refactored to enable upscaling of the product for market launch in the future.

References.

- Image Classification - [M. Abdullahi, A. D. Mohammed, S. A. Bashir and O. O. Abisoye, "A Review on Machine Learning Techniques for Image Based Spam Emails Detection," 2020 IEEE 2nd International Conference on Cyberspace \(CYBER NIGERIA\), 2021, pp. 59-65, doi: 10.1109/CYBERNIGERIA51635.2021.9428826.](#)
- Multi-Modal Approach - [Y. Gao, A. Choudhary and G. Hua, "A Comprehensive Approach to Image Spam Detection: From Server to Client Solution," in IEEE Transactions on Information Forensics and Security, vol. 5, no. 4, pp. 826-836, Dec. 2010, doi: 10.1109/TIFS.2010.2080267.](#)
- Spam Message Classification - [D. K. Renuka, T. Hamsapriya, M. R. Chakkaravarthi and P. L. Surya, "Spam Classification Based on Supervised Learning Using Machine Learning Techniques," 2011 International Conference on Process Automation, Control and Computing, 2011, pp. 1-7, doi: 10.1109/PACC.2011.5979035.](#)
- Spam Tweet Classification - [S. Sedhai and A. Sun, "Semi-Supervised Spam Detection in Twitter Stream," in IEEE Transactions on Computational Social Systems, vol. 5, no. 1, pp. 169-175, March 2018, doi: 10.1109/TCSS.2017.2773581.](#)
- Spam Email Classification - [S. Kaddoura, O. Alfandi and N. Dahmani, "A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach," 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises \(WETICE\), 2020, pp. 193-198, doi: 10.1109/WETICE49692.2020.00045.](#)
- Spam Classification Using Deep Learning - [Maariappan, Durga & Rao, Dhanush & Thirumoorthy, Sangeetha. \(2020\). A Competent Spam Prediction Technique by Supervised Deep Learning Classifiers. 80, in International Journal of Emerging Technology and Innovative Engineering Volume 6, Issue 02, February 2020 \(ISSN: 2394 – 6598\)](#)
- BERT Algorithm - [Rojas-Galeano, Sergio. \(2021\). Using BERT Encoding to Tackle the Mad-lib Attack in SMS Spam Detection in arxiv.org.](#)
- GraphQL - [Migrating to GraphQL: A Practical Assessment. Published By: IEEE on 18th March 2019](#)
- React Native - [Mobile development in Swift, Java and React Native: an experimental evaluation. Published By: IEEE on 15th July 2019.](#)
- AWS Amplify - <https://docs.amplify.aws/start/q/integration/react-native/>
- Whatsapp Clone - <https://github.com/Savinvadim1312/WhatsappClone>



Thank You.

By,
Jack Praveen Raj Ilango - RA1811026010004
Abhishek Paul - RA1811032010015