

End Research Paper

*Released: April 20th**Deadline: April 29th*

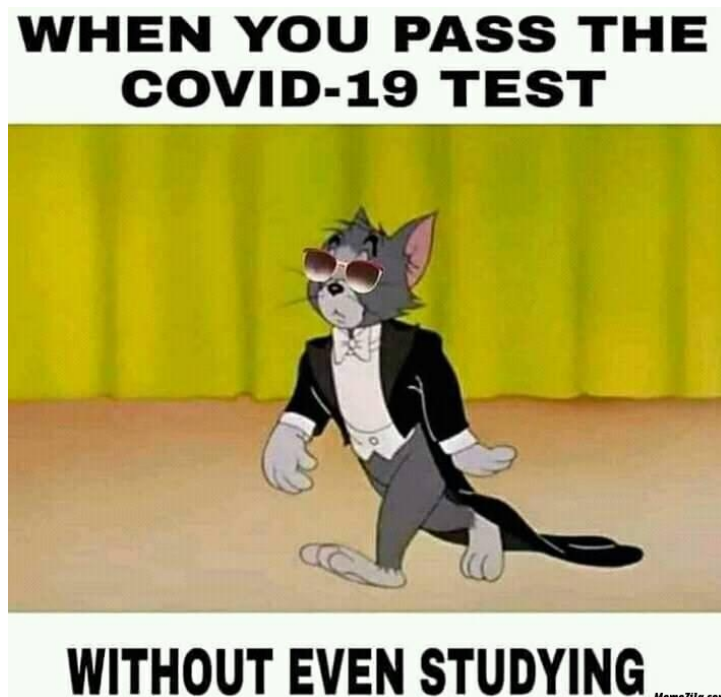
Instructions

- For the final component of the DASS course, you will be writing a research paper instead of the usual end-semester examination.
- This has been designed to give you a nice exposure to the world of research and help you get familiar with software systems research principles.
- You will be submitting a classic academia style research paper in a 2-column L^AT_EX document, named as rollnumber.pdf.
- This is an individual assignment and you are free to choose whatever topic you like. The deadline is April 29th, 23:55.
- Plagiarism detectors will be run on all submissions, so please do not copy from anyone. If found, you would be given a straight zero.

Introduction

We are living in truly unprecedented times. Deadly pandemics are often once-in-a-lifetime experiences, but when they do happen, they tend to cause havoc and kill thousands, even millions of people. A century ago, it was the ‘Spanish flu’ of 1918 and today, it is the Covid-19, disrupting so many lives around the globe.

Easily transmissible, cases of the virus have been exponentially growing in many places. Scientists are yet to discover an effective drug or medication against the virus, so in order to bring down exponential growth or ‘flatten the curve’ as it’s being commonly called, countries are adopting the age-old teaching of ‘**prevention is better than cure**’, as there still is no ‘cure’ to the virus. Prevention is in terms of complete lockdowns, travel bans, social distancing and so on, causing everyone to stay at home and causing the daily life schedule to go astray.



The current Covid-19 situation has truly changed the face of this planet, causing us to reflect and contemplate about mankind's harmful environmental practices that kill mother earth. Lots of problems and difficulties have arisen due to the lockdowns and travel-bans. Of course, there can be no recession in software, but many other professionals' economic lives are in grave peril. Various philosophical problems arise as well, like whether it is ethically right for us to lock down entire countries which could lead to a huge economic recession and impoverish many people. The rising amount of cases also place an extraordinary amount of stress on the healthcare infrastructure, thereby limiting the medical equipment and resources. So would it be morally wrong to deny medical care to some because the other is more 'critical'?

The setbacks of the current situation are incessant and researchers and scientists all over the world are working 24x7 not just on the vaccine, but also on solutions to the various troubles that the common people face.



What do you have to do?

We are giving you a bunch of problem statements which would capture the difficulties that people on our planet are facing due to these exceptional times. You are given 5 topics, you can choose whichever fascinates you the most.

You need to come up with a solution to this problem, from a software system perspective. You must **design and analyse the system** and convince the reader that it's an effective solution to the problem. It'd be great if your idea is novel, but it's alright if it isn't. We wouldn't cut marks if someone has had a similar idea before.

You can make use of all the AI/ML/NLP/Vision algorithms and networks out there in your design. What you really need to do is design the software system application around it, **keeping the ML stuff black-boxed**. But ensure that you cite the papers and include a short paragraph that explains the outer level functioning of the AI network.

You don't have to demonstrate that your idea can work, that is, **there is no need to implement or code anything up**. All we expect is an overall system design that explains the architecture of the software application you propose, using software design principles.

The paper MUST BE WRITTEN IN L^AT_EX. You can use any of the hundreds of research paper templates on overleaf. Here's one research template you could use: CVPR 2018

The Topics

Here are the problem statements, feel free to choose whatever you deem to be engrossing enough.

Problem-1: Test, Test, Test

For this project we want you to propose a pipeline for fighting COVID, harnessing the power of Medical Imaging and Machine Learning. "Test, test and test" WHO Chief Dr. Tedros Adhanom had said, in response to a question asking for optimal strategy to defeat COVID-19. However, as you would know, no country is testing enough, especially India. As engineers we want to help the doctors and the medical community which are already under stress, overworking and fighting a war. For this one should look how we can help them by providing them a streamlined detection/preventive process or develop any other tool that you think would help. As you are capable software engineers, we would want you to think about the whole flow which makes the solution plausible. You should think about how different institutes can add data to this sample, how will this be monitored, how would you ensure that the reliability of this crowd sourced data is maintained, since the amount of data at present would be quite less and so we want your solution to seamlessly take in new data. Moreover, you should think about how your proposed solution would gain acceptance, that is your flow should account for the same.

Below are a few potential ideas you can have a look at but don't limit yourselves to it.

- Alternative test methods like screening of COVID-19 through Chest X-Rays or CT Scans.
- Finding people with preexisting conditions like identification of people with lung diseases, as most people who have lost their lives to COVID-19 had finally lost it to pneumonia so if we identify people with who already have these diseases we can take proactive efforts to isolate them and make sure that the virus does not find an easy path to them.
- Assistance in RT-PCR testing either in testing itself or to manage the batch testing which are now gaining popularity. One idea can be to assist in the formation of batches, like people with high risk should get priority. Moreover, there might be some who could make the whole batch positive which is not exactly optimal - ideally we would like to have similarly risked samples in a batch.

You can refer to this Qantib article and this Sciencemag article. You will get to explore the beautiful world of medical vision, but remember that your objective is to design an efficient software system!

Problem-2: Lockdown

There is a colossal amount of research happening today in the world in order to be able to find a cure to the virus in the form of drugs and vaccines. Until then, many countries have imposed lockdowns to maximise social distancing in the hope that they would flatten the curve and mitigate the virus from their country. Then until a cure is found, the country would remain internationally locked so that all its citizens are safe. (Of course, some may disagree with the idea of a lockdown. It's true that it reduces the mortality rate, but it also gravely affects the livelihoods and financial conditions of people. But that's a discussion for another day).

The Prime Minister of India, Narendra Modi, announced the largest lockdown that history has ever seen, in order to alleviate the virus. States like Goa and Kerala have been successful, but the virus still looms in the other states. People have been given strict orders to stay at home and not go out anywhere, unless for essential groceries or immediate healthcare. Unfortunately, many people in India are breaking these rules and going out on the streets carelessly. The police and law enforcement officials wish to catch hold

of these people and penalise them for breaking the law. However, the police themselves want to avoid going out in the open as it defeats the purpose of a lockdown. Can you help the law officials to remotely catch hold of the people breaking these laws and penalise them?

You could come up with lots of captivating ideas for this and explore the world of computer vision (and maybe, robotics) as you do so. You could check this, this and maybe even this out. Remember that all this would be black-boxed and your primary objective is to design the software system around it which would store useful data, keep track of the people breaking the rules, notify the officials and penalise them accordingly.

Problem-3: Hotspots

The coronavirus pandemic has caused multiple issues for governments all over the world. From tracking cases to managing resources, the current situation has posed multiple challenges to the government. With the lockdown in place, our economy is suffering a huge blow. Instead of a complete lockdown in place, our government has resorted into placing lockdown only on the capable coronavirus hotspots. However the hotspots are by virtue dynamic in nature and varies with the changing social interaction and needs of the people. Thus predicting regions of the probable hotspots is a challenging task.

Recently, a company that makes internet-connected thermometers has proposed a solution to detect infection hotspots to track the virus down (check this out!). What we would like you to do is to design a software system that tracks down coronavirus hotspots with various techniques (including, but not limited only to the solution using thermometers) and alert nearby hospitals and health centers about potential case spurts in their area, and notify them the quantity (based on the number of cases in the neighbourhood) of safety and medical equipment (such as ventilators, masks etc.) and medication to stock up.

Your system could also take into account the various aspects and research involved in the prediction of the hotspots (check this out too!) and keep the people informed about the current lockdown regions (assuming it changes daily or weekly). They could also be notified if they are moving near a lockdown region and the authorities could keep track of the same.

Problem-4: Contact Tracing

Apart from the lockdown, one of the most important keys into fighting the coronavirus outbreak is extensive testing and contact tracing. Currently one of the best tools experts have for contact tracing is the old-school detective work, that is finding every coronic patient and then figuring out who they recently interacted with.

While Apple and Google have recently collaborated to offer a technical solution to this problem (link) using the bluetooth technology to contain the spread, we would like you to design a software based solution which could use any similar technology or resources to do the same. Taking into account the various aspects (like the social transmission probability, the density of social network at an area, location of established hotspots, NBDA etc) you may even assign an individual with the probability of him being infected. With the limited testing facility, this could even help the health agencies in filtering the potential carriers to give them a priority in testing!

So yes, contact tracing is a great way to fight the virus, especially when there's limited testing. Use your creativity to come up with a nice software application that would help society in the war against the coronavirus using the points mentioned above.

Problem-5: Stay Home!

Due to COVID-19 spreading so expeditiously, even the symptoms of cold and sneeze frighten us and people around. Even though the government is trying its best, we don't have the health resources to cover up the screening of all citizens. Moreover visiting a doctor for Corona testing involves the risk of getting it even if you don't have it in the first place - such is the irony!

How about getting the tests done at the comforts of your home – interesting isn't it? Seeking the potential in you as a software developer, we expect you to come up with an effective software solution which offers coronavirus screening through an AI-powered platform that behaves like a 'smart' doctor. How can this be achieved? How can this solution be scaled to meet the demands of the world population (the technological aspects obviously!)? We need your help to answer these questions. You can explore the amazing world of AI and NLP, but remember to use all that as a blackbox and focus primarily on the system design. Helpful Material: [this](#), [this](#), and [this](#).

Major Sections

Your research paper must comprise of 6 sections, namely the Abstract, Introduction, Literature Review, System Architecture, Conclusion and References. We honestly don't care about how much you write, you should design your system well and convince the evaluator that you've done your research. 2.5 to 3 pages is a decent amount of content. Even if you write 5 pages, but if you don't design your system efficiently with a good architecture, then you'll lose marks.

Abstract

Just a 3-4 sentence description of your solution to the problem in hand. Something like 'This is a major problem that people face and we propose XYZ, a novel software system application that will take as input yada yada yada'.

Introduction

Explain briefly about your problem and the difficulties that people face because of it. You must convince the reader that the problem is indeed a major one and that coming up with a solution is paramount.

Literature Review

This is where you have to do some research and read papers and scientific articles online. You must talk about the ongoing research in the field of your problem statement and the existing technologies out there that try to solve a similar problem that you're working on. You should explore existing products in the solution space, maybe study the existing data and convince the reader that you are well versed about the ongoing research related to your problem.

System Architecture

This is the significant part of your research paper, where you design the software system and explain its overall architecture. You should explain the pipeline of your system, about the various interactions between the model, view and the controller, how the database is being updated etc.

We expect a neat design that eloquently explains your system. Use classic software design principles for this, listing down the use cases and making use of UML sequence, state and class diagrams.

You should present a nice blueprint of your system and convince the reader that it's an effective system and would have a high probability of succeeding in the real world. You don't have to code it up, maybe you can do that over the summer!

Conclusion and Future Work

Conclude well and summarise your solution in 3-4 sentences. Talk about future work and how things can be improved in the coming times.

References

Whatever you read online - research papers, scientific journal articles etc. - please cite them.

Problem-6: Project Based (For JK people)

This is strictly only for those who have serious internet connectivity issues and have taken prior permission from Prof. Ramesh. You must talk about the product that you have developed as part of your DASS Project for the semester.

- The first three sections remain the same - Abstract, Introduction and Literature Review. Introduce the problem, the existing solutions to it and relevant work done in the field to solve that problem.
- System Architecture: You've already done this! So formalise it well, explain the overall design and the pipeline of your product. Employ UML design diagrams to convince us that your design is flawless. Along with that, also talk about the design patterns used in your code.
- System Functionality: Demo your product! Attach pictures of your app/website and take the reader on a trip through all the use cases of your product.
- Testing: You need to explain about how you tested your product and whether it passed all the test cases or not. You can include screenshots of your product too. Talk about the different types of software testing methods that you used at different levels. You must explain all the testing methods and analyse which one was most beneficial for your project.
- Conclusion and Future Work - same as before. And if you have any references, include that.

Deliverable

A zip file, {rollnumber}.zip containing the pdf file and the original .tex file. For instance, 20171010.zip when extracted must generate a folder with the name '20171010'. Inside that folder, your pdf and .tex files must be present. You can include any other files/diagrams if you wish. Please ensure to follow the submission format and submit ONLY .zip files.

Plagiarism is a strict NO, and we will be calling Special Forces Agent Puru for moss services! So please do not copy from anyone. We'd be forced to give you a zero :(.

Hope you have fun and explore some new things while working on this paper. All the very best, and may the Force be with you!