Cloud Computing

Deadline: 19th April 2024 (Friday)

Form a Team of 1 to 3 members and select only **one single topic** below. Read the following.

1) Each team submits by email to Prof. Tan a **single Zip File with the Final Report and PPT Powerpoint Presentation Slides** with all names/student IDs.
2) NO Page Limit in Final Report– can be a few to hundreds of pages as you wish. A report *typically* has an introduction, problem statement, solution design, illustrative examples, conclusions and computer software snippets (if any), references. You may put your software on open-source repository, e.g., Github, for my reference.
3) Level of work is expected to commensurate with the team size. A project demonstrating originality and creativity has a high chance of getting full marks. Give your best!
4) We may invite teams to give short demos during Lecture to share with everyone. Note that the demo is completely optional.

Start early. Be Imaginative. Be Creative. Impress. Have Fun!

Topic 1: Software-as-a-Service in Final Frontier— Space and Mother Earth!

Cloud computing will be important to the final frontier🚀. For example, SpaceX solved a precision landing problem using CVXGEN, a SaaS created by Stanford researchers (https://ee.stanford.edu/news/2021/jan/stephen-boyd-cvxgen-guides-spacex-falcon. Another example is LEOcloud that enables public cloud service providers to offer their customers a hybrid cloud service in space. NSR- The Global Space Economy, remarked in March 2021 that: "Ranging from business models looking to commercialize crew and cargo missions to near-earth orbits, to emerging Earth Observation-based Data Analytics opportunities the "__ As-A-Service" business models have arrived for within the Space and Satellite Markets." Also, see the fireside chat on "The Future of Connectivity in Space" with Clint Crosier, director of Amazon Web Services' Aerospace and Satellite on satellite-based cloud computing: https://www.uschamber.com/on-demand/technology/cloud-computing-and-connectivity-in-space.

Explore a problem in the space industry that can be solved by cloud computing technologies and demonstrate your ideas with a SaaS. If you are interested in CVXGEN (https://cvxgen.com) and GPT for code translation, please talk to me.

Topic 2: PageRank-inspired Natural Language Generation

Natural Language Generation (NLG) technologies has always been inspired by advances in cloud computing. A well-known idea in NLG is Textrank that applies the Pagerank algorithm to natural language models: https://web.eecs.umich.edu/~mihalcea/papers/mihalcea.emnlp04.pdf.

This project will be an in-depth study of TextRank. You can implement TextRank using a single machine programming model or a distributed system programming model (e.g., MapReduce/Spark/serverless programming). Study how the parameters of the PageRank

algorithm affect the performance of NLG. Try to explore large state spaces and evaluate its performance. Can you design a SaaS or PaaS for NLG application using TextRank?

Topic 3: Conversational Chatbot-as-a-Service and Low-Code-No-Code Principle

Social messaging has become indispensable in human communications and the future cloud computing applications may become more conversational in nature, given the rise of Gen-AI like ChatGPT (https://openai.com/blog/chatgpt). Cloud companies like SAP are now developing "low-code-no-code" SaaS: https://ignitesap.com/sap-and-low-code-no-code-development. This ushers in the trend of citizen developers: https://github.com/antdimot/awesome-lowcode. Explore Chatbot-as-a-Service and low-code-no-code principle and build a software to demonstrate your ideas with illustrative examples. Here are some references:

1) How To Build a GPT3 AI Chat Bot: https://landbot.io/blog/build-a-chatbot-with-gpt
2) Cloudera's blog and Github on innovative cloud computing applications: https://blog.fastforwardlabs.com/2021/09/22/automatic-summarization-from-textrank-to-transformers.html. You can experiment with Cloudera's Streamlit application at: https://github.com/fastforwardlabs/summarize.
3) No-code data visualization: https://github.com/notY0rick/matplotlib_ai

Topic 4: Cloud Computation by Crowdsourcing— *Wisdom of the Crowd*!

In the Lecture on Crowdsourcing, we learnt about SaaS Security, particularly CAPTCHA, to defend against spambots. We also learnt about the power of collective human-assisted computation to solve interesting problems, e.g., Amazon's AWS Mechanical Turk for machine learning classification, ReCAPTCHA (acquired by Google and now repackaged as Google Crowdsource at https://crowdsource.google.com) and scalable autograding. Notably, "*Play Kasparov*" in 1999 must have been perhaps the first SaaS to pit the whole world against Kasparov, the reigning world champion. Footages of "*Play Kasparaov*" are viewable at: http://www.aparchive.com/metadata/youtube/f683dd980efa3d3492de6c747c168a94.
In 2022, the debut of ChatGPT SaaS makes conversational AI chess a reality: https://medium.com/@mmabini/i-beat-chatgpt-in-chess-92ec0c757463. All these have taken Aleksandr Kronrod's phrase "*Chess is the drosophila of artificial intelligence*" to dizzying heights.

Study a problem that can be addressed by crowdsourcing techniques and cloud computing technologies (e.g., Google Crowdsource). Demonstrate your solution with illustrative examples (and, if possible, make your own SaaS). Analyze its effectiveness with respect to the size of the crowd. Examples could be solving an AI problem, finding logical bugs in software or to enhance SaaS security by detecting spambots?

Topic 5: Cloud Security

This project will explore cloud security and large language models (LLMs). Microsoft is already rolling out a SaaS called Security Copilot for cybersecurity. Explore questions like: how would large language models shape cloud security? How would cloud security shape the future development of LLMs? For example, can watermarking technologies be leveraged for security? How will AI-assisted programming affect cloud security since it can be easier to create malicious code by anyone that go undetected. Some relevant readings include:

1. "A Double-Edged Sword for Cloud Security? Weighing the Benefits and Risks of Large Language Models" by Kathryn Shih at Google Cloud Security: https://cloud.withgoogle.com/cloudsecurity/podcast/ep144-llms-a-double-edged-sword-for-cloud-security-weighing-the-benefits-and-risks-of-large-language-models/
2. Disrupting malicious uses of AI by state-affiliated threat actors: https://openai.com/blog/disrupting-malicious-uses-of-ai-by-state-affiliated-threat-actors
3. Microsoft Security Copilot: https://www.microsoft.com/en-us/security/blog/2023/11/15/microsoft-unveils-expansion-of-ai-for-security-and-security-for-ai-at-microsoft-ignite/
4. Stanford's DetectGPT: https://ericmitchell.ai/detectgpt/ & SpamGPT watermarking large language models: https://fastml.com/spamgpt-watermarking-large-language-models/
5. Is AI-assisted Programming secure: https://blog.codacy.com/is-coding-with-ai-secure

Topic 6: LLM Apps using Nemobot SaaS

Vertical SaaS is becoming important in years to come due to the advent of ChatGPT and LLM: https://techcrunch.com/2023/06/27/vertical-ai-the-next-logical-iteration-of-vertical-saas/ and the development of LLM Apps will be a key focus. This project explores the development of LLM Apps using the Nemobot platform. You may explore in depth new concepts like "LLM functions", OpenAI GPT programming and Retrieval Augmented Generation in the development of LLM Apps. A group of brilliant undergraduates at NTU recently demonstrated building innovative LLM Apps. You may read their interview in the NTU student magazine article: https://www.hey.ntu.edu.sg/2024/02/04/we-used-ai-to-create-a-digital-tour-guide/ & demos:

1) Tour bot: https://www.youtube.com/watch?v=ov6FnseF4Lw
2) Medical bot: https://www.youtube.com/watch?v=6hGqYoiRwWs

Please talk to me if you are interested to develop LLM Apps using Nemobot.

Topic 7: Interesting X-as-a-Service

Explore a challenging problem that interests you and consider how it can be addressed and solved using a Software-as-a-Service or a Platform-as-a-Service. Provide some illustrative examples to your problem formulation and its solution.

You may consult https://en.wikipedia.org/wiki/As_a_service for ideas, and we list down a few instances below for reference.

For instance, there may be malicious adversary who uses a malicious *Disinformation-as-a-Service (DaaS)* to spread viral misinformation and disinformation. Can a *Factchecker-as-a-Service (FaaS)* counter against DaaS?

For example, there were widespread anti-vaccine rumors during the COVID-19 pandemic and how can your FaaS help the World Health Organization? Likewise, rumors generated by *DaaS or Propaganda-as-a-Service* can influence real-world events like election outcomes.