Lab 1: Project Brainstorming

The goal of this lab is for you to work as a group to brainstorm project ideas. At the end of this exercise you should have an outline for your project proposal that you can share with others to get feedback. We will share these outlines for peer feedback in the next lab.

Group name: How Do You Turn This On

Group members present in lab today: Yi-Ting Yeh, Tin-Ray Chiang

1: Ideas

Write down 3-5 project ideas your group considered (a few sentences each). Depending on your group and process, the ideas may be very different, or they may be variations on one central idea.

1. Run visual navigation models such as Room2Room models on the device.
2. Perform ASR on the device with either Kaldi or other end-to-end models/toolkits.
3. Instead of ASR, focus on wakeup word detection.
4. Deploy state-of-the-art language models.
5. Run handwritten equation detection with the provided camera.

2: Narrowing

Choose two of the above ideas. For each one:

On device ASR

1. How would this project leverage the expertise of each member of your group?
    a. We both have experience in speech recognition and the Kaldi toolkit.
2. In completing this project, what new things would you learn about: (a) hardware (b) efficiency in machine learning (c) I/O modalities (d) other?
    a. We need to have a more comprehensive understanding of the whole pipeline of ASR.
    b. We need to learn different distillation/pruning/quantization techniques to deploy big ASR models to the device.
    c. To perform real ASR on the device, we should learn how to adjust the microphone and its recording quality. If necessary, we will learn how to finetune ASR models to different audio quality.

3. What potential roadblocks or challenges do you foresee in this project? How might you adjust the project scope in case these aspects present unsurmountable challenges?

   Challenges

   a. If we want to use Kaldi,
      i. The error messages might be extremely hard to understand.
      ii. We need to spend lots of time understanding how to customize each part in the pipeline.
      iii. Techniques designed to improve the inference speed of neural networks might not be directly applicable to HMM used in Kaldi.
   b. If we want to use end-to-end models or toolkits such as ESPnet:
      i. Training and fine-tuning models might take a huge amount of time
      ii. Without GPU, the model might be too large to run on Raspberry Pi.

   In the case it is too hard to run the entire ASR pipeline on the device, we might adjust the project scope to run only subsets of models. For example, we might use cloud servers to do feature extraction, and run the acoustic model and the language model on the device.

4. How could you potentially extend the scope of this project if you had e.g. one more month?
   a. We can extend the model to support real-time ASR.
   b. We can try to run TTS on the device.

Handwritten Equation to LaTex

1. How would this project leverage the expertise of each member of your group?
   a. We are familiar with the Seq2seq model.
2. In completing this project, what new things would you learn about: (a) hardware (b) efficiency in machine learning (c) I/O modalities (d) other?
   a. How to use PI camera
   b. How to use objection detection
3. What potential road blocks or challenges do you foresee in this project? How might you adjust the project scope in case these aspects present unsurmountable challenges?
   a. The accuracy of bounding box might be bottleneck
   b. The training data might be not sufficient
4. How could you potentially extend the scope of this project if you had e.g. one more month?
   a. Making the model real-time.
   b. Considering different angles of view.

3: Outline

Choose one of the ideas you've considered, and outline a project proposal for that idea. This outline will be shared with other groups next class (Tuesday) to get feedback.

Your outline should include:

- Motivation
    - Automatic speech recognition (ASR) has been a key component in different applications such as intelligent assistants. To facilitate on-device applications, on-device ASR is important. The ASR system needs to be run offline when no internet access is available. Having an offline ASR also preserves users' privacy. Therefore, in this project, we want to study how to perform ASR on the edge device.
- Hypotheses (key ideas)
    - End-to-end ASR models are suitable for edge devices, because we can use distillation/pruning/quantization techniques to minimize the model size.
- How you will test those hypotheses: datasets, baselines, ablations, and other experiments or analyses.
    - Dataset: Librespeech
    - Baselines: seq2seq ASR [1], CTC-based ASR [2], Kaldi.
    - Metric: word error rate (WER), latency, power consumption
- I/O: What are the inputs and output modalities? What existing tools will you use to convert device inputs (that are not a core part of your project) to a format readable by the model, and vice versa?
    - Input: raw wav file. Output: String
    - Recording Library: sounddevice
- Hardware, including any peripherals required, and reasoning for why that hardware was chosen for this project. (This is where you will request additional hardware and/or peripherals for your project!)
    - We will need a better microphone. While modern ASR models will use data augmentation to make model more robust to noise, usually a better microphone will lead to better performance.
- Potential challenges, and how you might adjust the project to adapt to those challenges.
    - Training time: The training time may be very long. To adapt to this challenge, we may use a pretrained model, and use only part of the training data for distillation.
    - Mismatch target distribution: The audio collected by the microphone may be very different from the audio in the training data. We may have to focus on the performance on the standard testing dataset. Or we may have to focus on making the model more robust to distribution shift due to the microphone.
- Potential extensions to the project.
    - Making the ASR model real time.

[1] Listen, Attend and Spell, W Chan *et al.*

[2] Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks, A Graves *et al*.