Alexander Khosrowshahi
William Stone
Saswata Majumder
Ross Williams
Prof. Ewing
CSCI1470
4/14/2025

Team Yell — Check-in #2

Project Title:
Team Yell: Adversarial Attacks on OpenAI's Whisper ASR Capabilities

Team Members (Name, Login):
Alexander Khosrowshahi (akhosrow)
William Stone (wlstone)
Saswata Majumder (smajum14)
Ross Williams (rjwillia)

NOTE: OUR PROJECT MAJORLY CHANGED BETWEEN THIS AND CHECKIN #3

Check-in Questions:

Introduction: What problem are you trying to solve and why?

We are attempting to poison speech-to-text models using adversarial learning. Our main motivation for this project is improving digital privacy by (hopefully) being able to provide a model that can generate an audio filter that tricks speech-to-text models into transcribing your speech incorrectly. To do this, we plan to introduce an adversarial perturbation model trained against OpenAI's Whisper, a leading open-source audio transcription model. Our motivation is in personal privacy—allowing secure audio communication protected from automated transcription, protecting intellectual property in orated art mediums, and generally giving users more discretion over how their speech is used in model training.

Related Work: Are you aware of any, or is there any prior work that you drew on to do your project?

Please read and briefly summarize (no more than one paragraph) at least one paper/article/blog relevant to your topic beyond the paper you are re-implementing/novel idea you are researching.

Muting Whisper: A Universal Acoustic Adversarial Attack on Speech Foundation Models introduces a poisoning attack exploiting whisper (and other ASR models)'s use of 'special tokens.' By prepending audio samples with a learned substitute for whisper's 'end of text' token, researchers were able to end Whisper's audio transcription early, completely obscuring any discerned information with more than 97% accuracy. This adversarial attack is similar to ours in that it seeks to perturb audio such that Whisper misinterprets it in a manner that damages transcription efficacy. However, it differs in that it relies on a white-box attack method due to the differences in 'eot' characters between models, along with targeting a specific character output rather than our goal of perturbing outputs by any large delta.

Data: What data are you using (if any)?

The plan is to use the public LibriSpeech dataset. The dataset contains 1000 hours of English audio plus transcriptions all derived from a public set of audiobooks, and is commonly used for benchmarking for audio transcribing models. The audio files already come pre-partitioned and sorted into "clean" and "not clean" sets. We anticipate that 1000 hours will be more than enough for our purposes, but if not, we can draw upon the much larger Mozilla Common Voice dataset. Because all the data comes prepackaged rather nicely, we do not anticipate needing to perform significant preprocessing. However, some standard reshaping and normalization may well be necessary.

Methodology: What is the architecture of your model?

How are you training the model?

We're taking two approaches to training our model—one that targets open-source models and one that targets closed-source black box models, with hopes of general extensibility to many speech-to-text models. Though we're training on an open-source model (OpenAI's Whisper), we'll treat it as a black box in approach 2.

Approach 1: Open-source stochastic gradient ascent/negative loss descent

We have two models: A perturbation generation model and a specific speech model (OpenAI Whisper)

1) Take some input x and feed it through the perturbation model to obtain $x + \delta$
2) Run both x and $x + \delta$ through the model
3) Obtain the output of the specific layer we're targeting
4) Get gradient from target layer
5) Backprop to apply gradients back to perturbation layer
6) Clip δ_{in} to within acceptable bounds
7) Repeat

Approach 2: Closed-source adversarial methods

A rough diagram of the "Black Box" attack model

We have two models: A perturbation generation model and a surrogate speech model (OpenAI Whisper)

1) Take some input x and feed it through the perturbation model to obtain $x + \delta$
2) Run both x and $x + \delta$ through the model
3) Compare the resulting outputs of x and $x + \delta$ with CTC loss
4) Take the negative of CTC loss and optimize perturbation model
5) Clip δ_{in} to within acceptable bounds
6) Repeat

Note that approach 2 is marginally simpler than approach 1, including the fact that CTC loss will have a direct mapping back to our perturbation model, while we might have to fudge Whisper's internal loss a bit for our combined adversarial function. We've chosen to take both approaches,

but will probably evaluate the efficacy of the black-box approach first to see if we can get any notable results with a simpler architecture.

If you are doing something new, justify your design. Also note some backup ideas you may have to experiment with if you run into issues.

Our design in approach 2 treats Whisper like a black-box, which should theoretically allow us to generalize to other similar models. Our main concern is the difference in deltas between the input (small change) and output (large change). Our choice of CTC loss in approach 2 is based on the seq2seq nature of text transcription and measuring sentence similarity, which we then flip to optimize for larger changes.

In approach 1, we may choose to use a compound loss function between CTC and whatever internal loss function Whisper uses on our targeted layer.

We first plan on training against Whisper's tiny model before we move to larger models. Approach 1 is, in a way, our backup, as adversarial training using gradient ascent is well-studied. We could also attempt to run gradient ascent on a smaller open-source model, which may be less complex.

Metrics: What constitutes "success?"

What experiments do you plan to run?

Our main method of experimentation will be by comparing Whisper's output on perturbed inputs vs. unperturbed inputs. In an ideal world, we'd want outputs to become nonsensical while looking somewhat normal, but this would likely require some notion of "correctness" and "grammar," which are difficult to determine without human input—see examples of normal sounding, but ungrammatical sentences like "The horse raced past the barn fell.", among others. Evidently, we'll need some deterministic way to experiment on this.

In order to do this, we'll use a corpus of clear speech with correct transcriptions (from librespeech) or elsewhere. We'll then have Whisper transcribe them unperturbed and compare against the perturbed outputs. If they are dissimilar enough from *both* the unperturbed and actual data (it's important we don't just reproduce Whisper's inaccuracies, but actually distance ourselves from the input data), we'll constitute that a success.

For most of our assignments, we have looked at the accuracy of the model. Does the notion of "accuracy" apply for your project, or is some other metric more appropriate?

For our project, “inaccuracy” is honestly a better metric. We want our perturbation generation models to minimize accuracy on the actual speech-to-text model. We’ll do this by comparing the speech corpus sample and its verified transcription with Whisper’s unperturbed transcription, then compare the perturbed transcription with both the verified transcription and unperturbed transcription. The less accurate we are, the better.

If you are doing something new, explain how you will assess your model’s performance.

Our intended goal is to compare and contrast attack architectures to see which scale best, and which can apply to most general speech-to-text models. We will assess our model’s performance by comparing accuracy on our poisoned input as compared to accuracy on the clean input, and calculating a score of how different the two outputs are. The more different they are, the better our poison filter has performed. We also really want to prioritize maintaining intelligibility of the audio being passed through our model, and we’ll ensure that by manually listening to our model’s output and seeing if it messed up the audio beyond recognition. We can also do spectral analysis on the outputs to determine interpretability, but it will likely require some human touch ultimately.

What are your base, target, and stretch goals?

Base Goal: Get the “black box” approach to work on Whisper’s tiny model

Target Goal: Get the “black box” and “open source” approach to work on Whisper’s tiny model

Stretch Goal: Scale up to bigger Whisper models, and also run a bitcoin miner on OSCAR

Ethics: (Remember that there’s not necessarily an ethical/unethical binary; rather, we want to encourage you to think critically about your problem setup.)

What broader societal issues are relevant to your chosen problem space?

With improvements in automatic data processing methods come increasing data privacy nightmares. Models can (and will) be deployed for all sorts of surveillance tactics, such as measuring compliance of members of a political party, or detecting specific topics like unionization or mobilization, which can have far-reaching effects. Adversarial attacks are one possible example in favor of data privacy.

Who are the major “stakeholders” in this problem, and what are the consequences of mistakes made by your algorithm?

There are three major stakeholders in this issue that we identify. Firstly, the consumers who wish to use the tool to protect their digital privacy—whom we are chiefly concerned with. Secondly, corporate entities and AI developers/researchers, who we classify as not inherently malicious or

“threatening,” but need to be considered in our threat model for such a project. Third, we consider malicious actors—whom can fall into the second category along with the third, but are our main adversaries with this tool.

While the second category might not be malicious by nature, we essentially treat them as an “adversary” when evaluating this problem—as there is no actionable distinction between malicious use and research use unless explicit consent is given to the usage of audio by party one. A researcher may not intend any harm in training off of, say, protected IP in the form of an audiobook, but this distinction is challenging to draw between the researcher and, for example, a malicious actor using ASR to steal protected information and IP. For this reason, we consider all parties attempting to run models or train on audio data not approved by party one an adversary.

Since this model would be an additional model that aims to degrade performance, mistakes would only amount to communication lines remaining insecure, since it would be improbable for our model to instead improve the victim model. A concern of importance is that, as with any deep learning model, there is systematic uncertainty in the efficacy of our model in assuring security—which could lead to users having unfounded notions of absolute security. This can be handled with disclosing that our model will likely not achieve perfect security, but cannot defend the consumer against self-imposed risk if they choose to use it despite this.

Division of labor: Briefly outline who will be responsible for which part(s) of the project.

William:

- Procurement of data/data normalization, preprocessing, and cleansing
- Working on initial explorations of white-box attack architecture/training with Alex
- Literature review

Alex:

- Build tools/tooling
- Initial explorations of white-box attack architecture/training with William
- Perturbation model architecture research
- Literature review

Sas:

- Data processing and procurement
- Model verification and testing
- Working on explorations of black-box attack methodology/training with Ross

Ross:

- Black-box attack methodology/training
- Figuring out compute feasibility
- Resident hyperparameter tuner