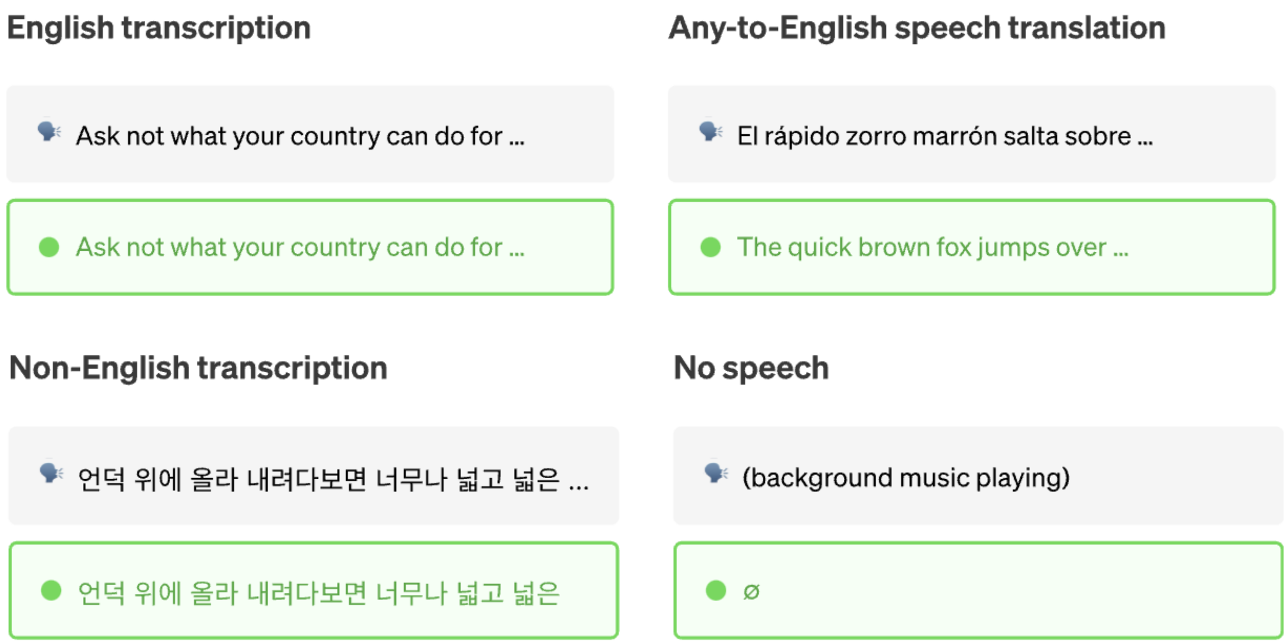
Whisper

It is a toolkit can do:



Overview

Context

Paper Title

"Robust Speech Recognition by Large Scale Weak Supervision"

Key Feature

- Generalization capabilities
- Transfer setting in zero-shot
- No need for fine-tuning

Problem

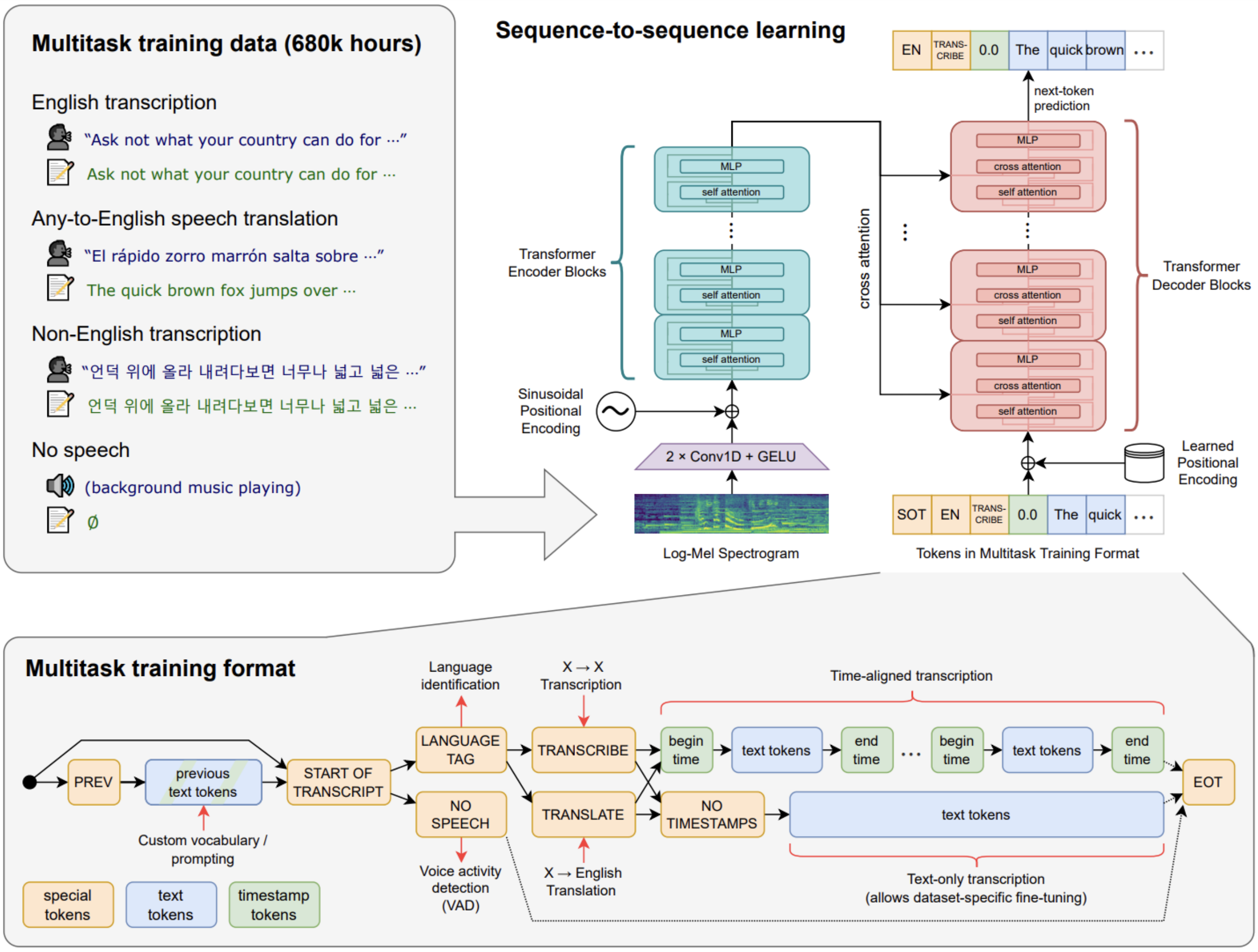
- Unsupervised models: need fine-tuning to be truly effective
- Supervised models: robust but not enough

Approach

- Scaling weakly supervised speech recognition the next order of magnitude to 680,000 hours of labeled audio data
- Without any need for fine-tuning
- The dataset is a global affair

Architecture

Overview of Architecture



Discussion Question #1

How does Whisper handle multitasking?

Discussion Question #2

Why does Whisper use basic Transformers for its model?

Model Variants

Model	Layers	Width	Heads	Parameters
Tiny	4	384	6	39M
Base	6	512	8	74M
Small	12	768	12	244M
Medium	24	1024	16	769M
Large	32	1280	20	1550M

Performance Metrics

1. Remove any phrases between matching brackets (`[`, `]`).
2. Remove any phrases between matching parentheses (`(`, `)`).
3. Remove any of the following words: `hmm`, `mm`, `mhm`, `mmm`, `uh`, `um`
4. Remove whitespace characters that comes before an apostrophe `'`
5. Convert standard or informal contracted forms of English into the original form.
6. Remove commas (`,`) between digits
7. Remove periods (`.`) not followed by numbers
8. Remove symbols as well as diacritics from the text, where symbols are the cl
starting with `M`, `S`, or `P`, except period, percent, and currency symbols that may b
9. Detect any numeric expressions of numbers and currencies and replace with a fo
thousand dollars" → "$10000".
10. Convert British spellings into American spellings.
11. Remove remaining symbols that are not part of any numeric expressions.
12. Replace any successive whitespace characters with a space.

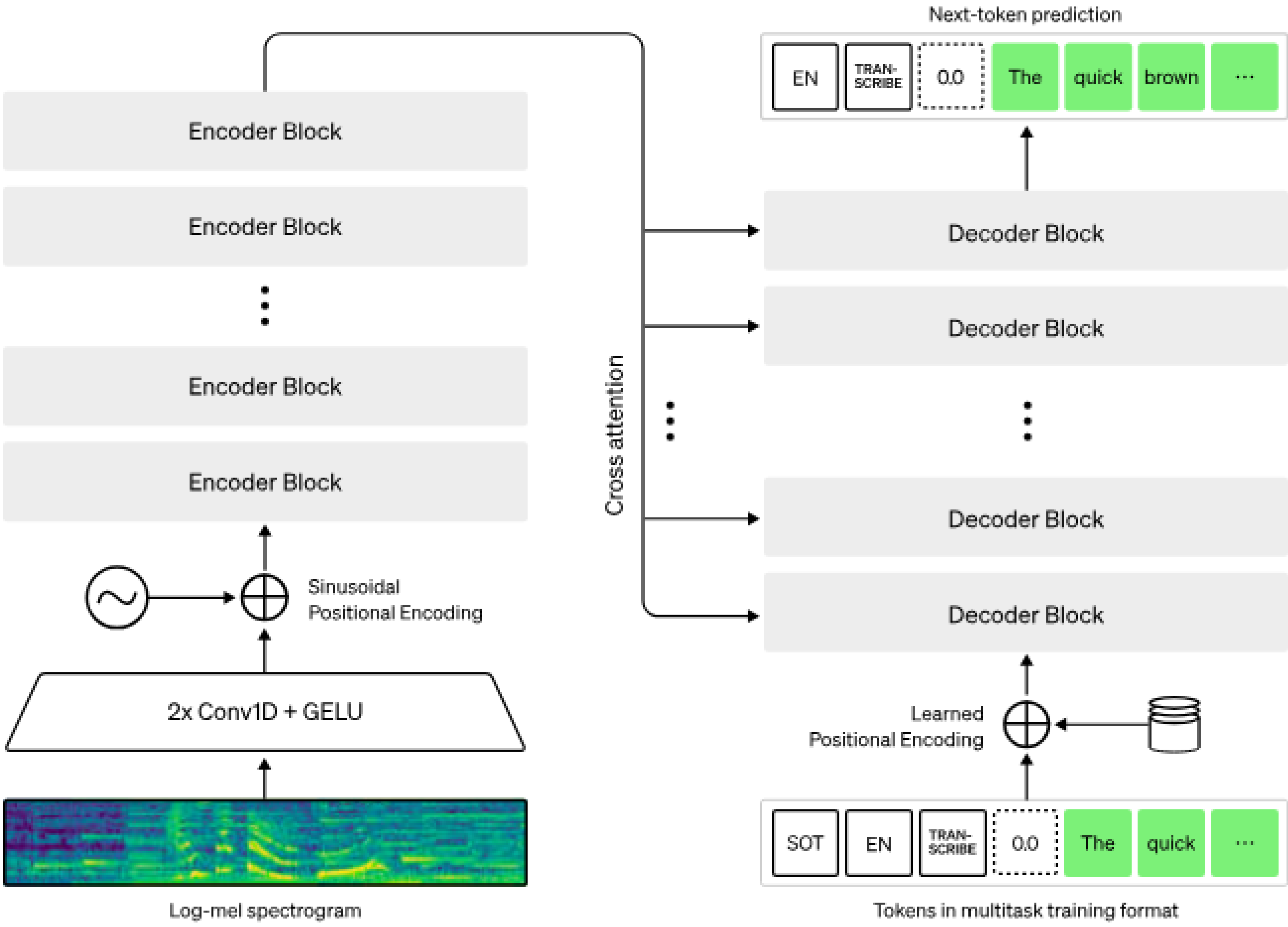
Pseudocode

Algorithm 1 Whisper

Class: Whisper
Parent Class: nn.Module
procedure INITIALIZATION(*dims* (ModelDimensions))
    Initialize *dims*
    Create *encoder* using AudioEncoder with:
        n_mels
        n_audio_ctx
        n_audio_state
        n_audio_head
        n_audio_layer
    Create *decoder* using TextDecoder with:
        n_vocab
        n_text_ctx
        n_text_state
        n_text_head
        n_text_layer
    Define *all_heads* as a tensor of zeros of shape (*n_text_layer*, *n_text_head*).
    Set the last half of *all_heads* to True.
    Register *alignment_heads* as a buffer with the sparse version of *all_heads*. Make it persistent.
end procedure

Algorithm 2 AudioEncoder

Class: AudioEncoder
Parent Class: nn.Module
procedure INITIALIZATION($n\_mels, n\_ctx, n\_state, n\_head, n\_layer$)
    Initialize parent class
    Create $conv1$ with parameters n_mels, n_state
    Create $conv2$ with parameters n_state, n_state
    Register buffer $positional\_embedding$ using sinusoids function
    Define $blocks$ as a list of ResidualAttentionBlock with length n_layer
    Define ln_post as LayerNorm with n_state
end procedure
procedure FORWARD(x (Tensor))   ▷ x is the mel spectrogram of the audio
    $x \leftarrow$ Apply GELU activation after passing through $conv1$
    $x \leftarrow$ Apply GELU activation after passing through $conv2$
    Permute dimensions of x
    Assert shape of x matches $positional\_embedding$
    Add $positional\_embedding$ to x
    for each block in $blocks$ **do**
        $x \leftarrow$ Apply block to x
    end for
    $x \leftarrow$ Apply ln_post to x
    return x
end procedure



Algorithm 3 TextDecoder

Class: TextDecoder
Parent Class: nn.Module
procedure INITIALIZATION($n\_vocab, n\_ctx, n\_state, n\_head, n\_layer$)
    Initialize parent class
    Create $token\_embedding$ with parameters n_vocab, n_state
    Initialize $positional\_embedding$ with size $n\_ctx \times n\_state$
    Define $blocks$ as a list of ResidualAttentionBlock with cross attention, length n_layer
    Define ln as LayerNorm with n_state
    Create a mask $mask$ with size $n\_ctx \times n\_ctx$ and set upper triangle to $-\infty$
    Register buffer $mask$
end procedure
procedure FORWARD(x (Tensor), xa (Tensor), kv_cache (Optional[dict])) ▷ x is the text tokens, xa is the encoded audio features
    if kv_cache exists **then**
        $offset \leftarrow$ shape of the first value in kv_cache
    else
        $offset \leftarrow 0$
    end if
    Update x with $token\_embedding$ and $positional\_embedding$ based on offset
    Convert x to the same dtype as xa
    for each block in $blocks$ **do**
        $x \leftarrow$ Apply block to x, xa with $mask$ and kv_cache
    end for
    $x \leftarrow$ Apply ln to x
    Calculate $logits$ using x and the transpose of $token\_embedding$ weight
    Convert $logits$ to float type
    return $logits$
end procedure

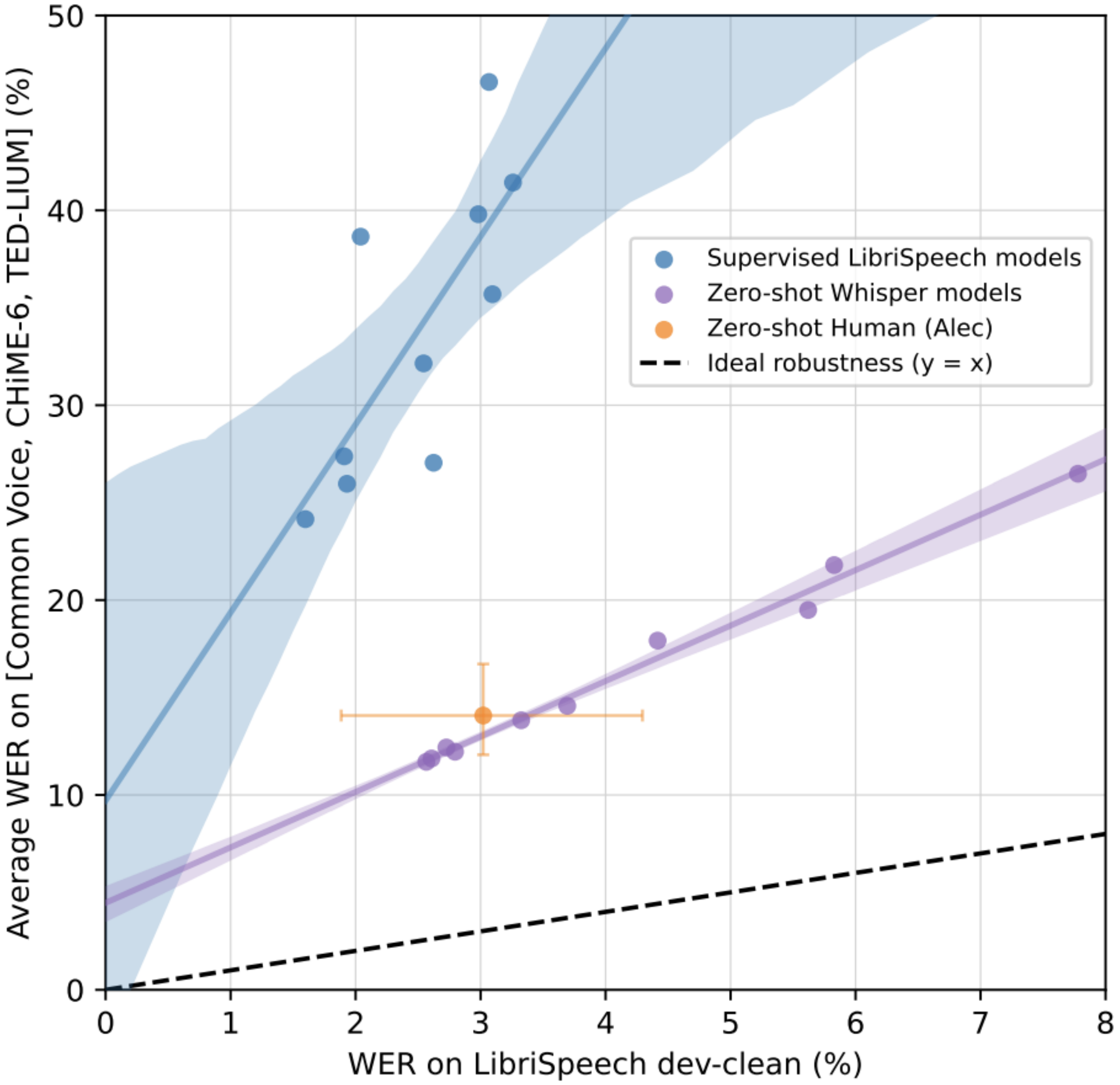
```
mask
tensor([[0., -inf, -inf,  ..., -inf, -inf, -inf],
        [0., 0., -inf,  ..., -inf, -inf, -inf],
        [0., 0., 0.,  ..., -inf, -inf, -inf],
        ...,
        [0., 0., 0.,  ..., 0., -inf, -inf],
        [0., 0., 0.,  ..., 0., 0., -inf],
        [0., 0., 0.,  ..., 0., 0., 0.]])
```

Empirical Results

Close the gap to human robustness

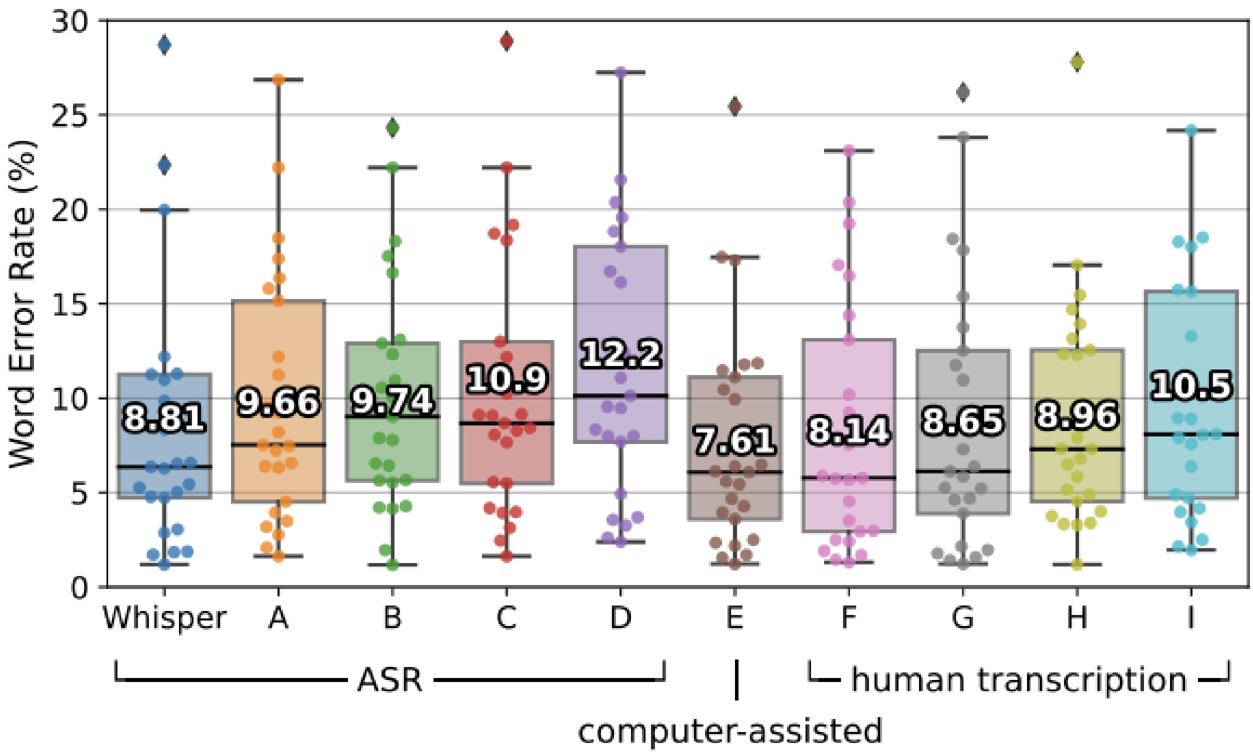
- Lack of Robustness in Supervised Models

- Human-Like Robustness



As good as human being?

Close to performance of professional human transcribers!



Code demonstration

https://colab.research.google.com/drive/1M8zNZ24lGcf05j-u53y73D-OhOv6z0l0?usp=drive_link#scrollTo=j9UgVYrod4SB

Critical Analysis

Were there any errors?

- The predictions may include texts that are not actually spoken in the audio input
- Lower accuracy on low-resource and/or low-discoverability languages or languages
- Prone to generating repetitive texts

Broader Impact

We hope Whisper's high accuracy and ease of use will allow developers to add voice interfaces to a much wider set of applications.
However,
it also raises dual-use concerns.

Seek for more?

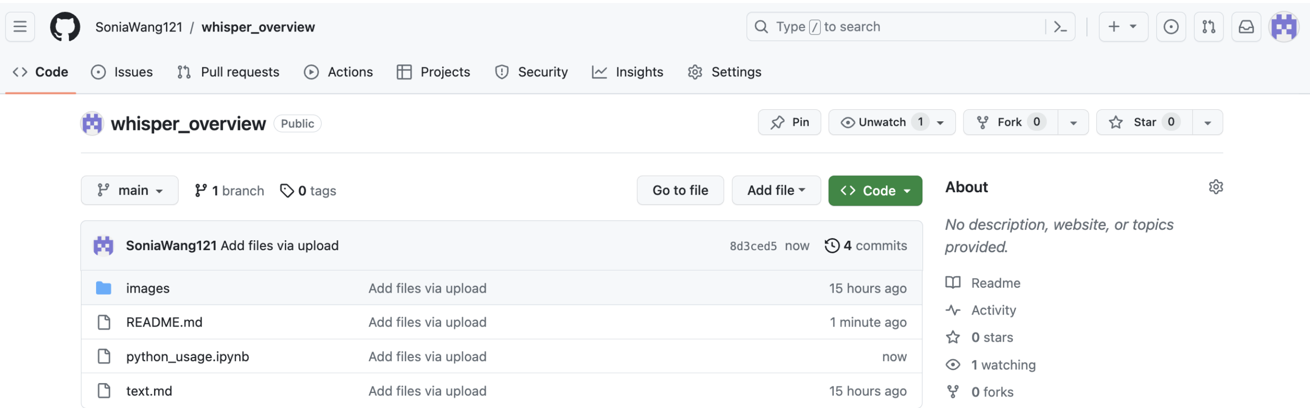
Research Index of OpenAI

OpenAI Menu

Mar 14, 2023	GPT-4	Read paper ↗
Sep 21, 2022	Introducing Whisper	Read paper ↗
Apr 13, 2022	Hierarchical text-conditional image generation with CLIP latents	Read paper ↗
Jan 27, 2022	Aligning language models to follow instructions	Read paper ↗
Sep 23, 2021	Summarizing books with human feedback	
Jul 7, 2021	Evaluating large language models trained on code	Read paper ↗
Mar 4, 2021	Multimodal neurons in artificial neural networks	Read paper ↗
Jan 5, 2021	DALL·E: Creating images from text	
Jan 5, 2021	CLIP: Connecting text and images	Read paper ↗
Sep 4, 2020	Learning to summarize with human feedback	Read paper ↗
Jun 17, 2020	Image GPT	Read paper ↗
May 28, 2020	Language models are few-shot learners	Read paper ↗
Apr 30, 2020	Jukebox	Read paper ↗
Oct 15, 2019	Solving Rubik's Cube with a robot hand	Read paper ↗
Sep 17, 2019	Emergent tool use from multi-agent interaction	Read paper ↗
Apr 25, 2019	MuseNet	

Link: https://openai.com/research?contentTypes=milestone

Repository



Resource Links

1. Paper: https://cdn.openai.com/papers/whisper.pdf
2. Code: https://github.com/openai/whisper
3. Model Card: https://github.com/openai/whisper/blob/main/model-card.md
4. Introducing Whisper: https://openai.com/research/whisper
5. Hugging Face Communityhttps://huggingface.co/openai/whisper-large

Citation For paper

- Chan et al. Simply mix all available speech recognition data to train one large neural network.
- Galvez et al. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage.
- Chen et al. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio.
- Baevski et al. wav2vec 2.0: A framework for self-supervised learning of speech representations.
- Baevski er al. Unsupervised speech recognition. Advances in Neural Information Processing Systems
- Zhang et al. BigSSL: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition.