Finding Spoken Identifications: Using GPT-4 Annotation For An Efficient And Fast Dataset Creation Pipeline

Maliha Jahan[1], **Helin Wang**[1], **Thomas Thebaud**[1], **Yinglun Sun**[2], **Giang Le**[2], **Zsuzsanna Fagyal**[2], **Odette Scharenborg**[3], **Mark Hasegawa-Johnson**[2], **Laureano Moro-Velazquez**[1], **Najim Dehak**[1]

[1]Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD, USA
[2]University of Illinois Urbana-Champaign, Champaign, IL, USA
[3]Multimedia Computing Group, Delft University of Technology, the Netherlands

Abstract

The growing emphasis on fairness in speech-processing tasks requires datasets with speakers from diverse subgroups that allow training and evaluating fair speech technology systems. However, creating such datasets through manual annotation can be costly. To address this challenge, we present a semi-automated dataset creation pipeline that leverages large language models. We use this pipeline to generate a dataset of speakers identifying themself or another speaker as belonging to a particular race, ethnicity, or national origin group. We use OpenaAI's GPT-4 to perform two complex annotation tasks- separating files relevant to our intended dataset from the irrelevant ones (filtering) and finding and extracting information on identifications within a transcript (tagging). By evaluating GPT-4's performance using human annotations as ground truths, we show that it can reduce resources required by dataset annotation while barely losing any important information. For the filtering task, GPT-4 had a very low miss rate of **6.93%**. GPT-4's tagging performance showed a trade-off between precision and recall, where the latter got as high as **97%**, but precision never exceeded **45%**. Our approach reduces the time required for the filtering and tagging tasks by **95%** and **80%**, respectively. We also present an in-depth error analysis of GPT-4's performance.

Keywords: Large Language Model, GPT-4, OpenAI, ChatGPT, Fairness, Dataset, Self-identification, Annotation, Prompt

1. Introduction

Fair and inclusive speech technologies require training and evaluation datasets that include diverse types of speech, i.e., multiple ethnicities, dialects, accents, ages, genders, and people with atypical speech. However, current publicly available datasets tend to be limited to a few dialects, ethnicities, and age groups or are not annotated considering these factors. Consequently, developers cannot evaluate or improve the fairness of their approaches. The internet works as a vast reservoir of public domain information that could be leveraged to build large datasets. However, most of this information lacks annotations and assessment of relevant content that can require considerable investment in time and manual labor.

With recent advances in large language models (LLMs), researchers are seeing the potential to automate annotation. One such attempt came from (Gilardi et al., 2023), who used ChatGPT to perform various types of classification on a dataset of tweets that they collected. They showed that ChatGPT outperformed crowd-workers in terms of accuracy and intercoder reliability (krippendorff, 2004). In another work, (He et al., 2023) proposed a two-step approach called **Explain-then-annotate** to design better prompts for LLMs. They experimented with three tasks: query and keyword relevance assessment, BoolQ (Clark et al., 2019) which is yes/no question answering, and Word-in-Context (WiC) (Pilehvar and Camacho-Collados, 2019). They showed that GPT-3.5 achieves comparable, if not better, performance than crowdsource annotation. A big part of the research on LLM annotation revolves around prompting. In (Wei et al., 2023), a prompting technique called **Chain-of-Thought** was proposed where the prompt includes examples of questions, their answers, and the intermediate reasoning steps to arrive at the answers. Diao et al. (Diao et al., 2023) modified the CoT approach by proposing **Active-Prompting**, a technique to choose which examples to include in the CoT prompts. Among the manually prepared question-answer examples, they chose the ones with the most uncertainty, a metric that captures the 'difficulty' of a question. In another work, (Ding et al., 2023) used three techniques for prompting-regular prompting for annotation, labeled data generation, and external dictionary-guided labeled data generation. They trained models for specific tasks like NER, Relation extraction (RE), etc. The conventional approach worked well for sentiment analysis and sentiment triplet extraction, while the generation-based approaches worked best for NER and RE. In (Zhang et al., 2023), the authors used k-NN to select examples for the prompt. After generating labels using LLM, they used active acquisition techniques for efficient data selection and reweight-

This project was supported by NSF Award 2147350

LREC-COLING 2024, pages 7296–7306

ing to generate proper weights for the samples. A task-specific model is then trained for NER and RE tasks using the weighted data samples. Their approach outperformed the baseline approaches, namely- conventional prompting, supervised approach, and zero and few shot data generation.

In this article, we introduce a semi-automatic pipeline that leverages an LLM. It conducts complex annotation tasks on a newly collected audio dataset that features conversational speech from speakers of diverse racial, ethnic, and national origins. Our contributions are as follows:

- Introducing a unique speech dataset annotated with varying speaker group labels that can be used for training and evaluating fairness in speech processing systems;
- Proposing an innovative data creation approach;
- Exploring different prompting strategies; and
- Assessing the LLM's efficiency, accompanied by a detailed error analysis.

The pipeline's goal is to annotate the audio data with race, ethnicity, and/or national origin information of the speakers from the conversation. For the scope of this paper, we focused on Asian American speakers. The annotation process consisted of

- Filtering out transcripts that are not related to race, ethnicity, or national origin;
- Identifying instances where a speaker refers to their or another participant's race, ethnicity, or national origin. We only annotate speech segments where speakers acknowledge such information to ensure accurate, unbiased annotations and exclude external bias or prejudice.
- Extracting all this information to generate a new speech annotated dataset.

2. Dataset Description

The broader objective of this project is to compile a range of datasets containing speech from individuals of different racial, ethnic, and national origin groups, age brackets, and sexual orientations, among others, who self-identify or are identified as members of those groups. The goal is to allow the scientific community to assess the fairness of speech and language technologies using the speech of these groups and to design and train better LLMs. Fairness evaluation of various speech processing systems can be performed by comparing the performance of such systems on the different speaker groups available in the dataset. Table 1 shows some existing conversational speech datasets. On top of having rich metadata that includes race, ethnicity, and national origin labels, our dataset also offers a unique feature- the exact instances in the conversations where the speaker self-identifies or is identified by another speaker as belonging to any speaker group. Moreover, the dataset would be invaluable for linguistic research using naturalistic speech from diverse speaker groups.

For the scope of this paper, we created a dataset of interviews and conversations with Asian American speakers from the Internet Archives, a non-profit digital library offering free access, among others, to podcasts[1]. Speech samples deemed 'relevant' for content were transcribed using Whisper (Radford et al., 2023). In the context of our work, 'relevant' indicates being related to any of the racial, ethnic, or national origin groups of interest. To distinguish conversational participants and avoid overlapping speech, the speech samples were diarized. Speakers who explicitly or implicitly referred to their own or other participants' race, ethnicity, and national origins were labeled, and their speech was tagged for types of identification in the transcript where such identifications were found.

To select the identity categories of race, ethnicity, and national origin, we adhered to the classifications provided by the U.S. Census Bureau (Jensen et al., 2021) under the U.S. Office of Management and Budget (OMB) (fed). These census categories, although fluid and subject to ongoing social and political commentaries, provided comparable official terminology across different speaker groups. The categories used in the dataset were as follows.

- **Race:** Following official terminology, race was classified into six categories: White; Black or African American; Asian; American Indian or Alaska Native; Native Hawaiian or Other Pacific Islander; and Multiracial. When the individual's race could not be determined, the category 'Unknown' was used.
- **Ethnicity:** We classified it into three groups: Hispanic or Latinx, Non-Hispanic or Non-Latinx, and 'Unknown' for undetermined cases.
- **National Origin Group:** National origin groups were defined as groups of people sharing a common language, culture, ancestry, race, and/or other social characteristics (of Commerce), (Commission). Examples in the dataset included Korean, Vietnamese, Japanese, and so on.
- **Type of identification:** Instances when speakers identified themselves or another speaker in the conversation as belonging to a particular race, ethnicity, or national origin, were referred to as *"Self-identification"* and *"Other-person, in-interaction identification"*, respectively.

[1] https://archive.org/

Dataset	Metadata					Trans-cript	Publicly Available
	Age	Gender	Race	Ethnicity	National Origin		
CORAAL (Kendall and Farrington, 2023)	✓	✓	✓	x	x	✓	✓
People's Speech (Galvez et al., 2021b)	x	x	x	x	x	✓	✓
Casual Conversations (Hazirbas et al., 2022)	✓	✓	x	x	x	✓	✓
100,000 Podcasts (Clifton et al., 2020)	x	x	x	x	x	✓	x

Table 1: Existing conversational speech datasets.

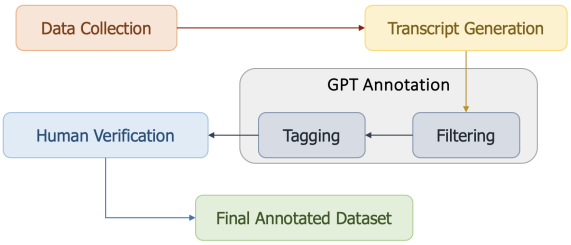


Figure 1: Proposed dataset creation pipeline

3. Dataset Creation Pipeline

The stages of our proposed pipeline are illustrated in Figure 1. Each step is detailed in one of the following subsections.

3.1. Data Collection

We collected the audio recordings and some associated metadata from the Internet Archive. Figure 2 shows the structure of our keywords[2]. Our focus was on content that featured Asian American speakers discussing topics related to their culture, history, literature, socioeconomic issues, and important public personalities. We used terms such as 'Podcast,' 'Conversation,' 'Discussion,' etc., in our keyword search to ensure that we find relevant content. Additionally, we used the names of specific nations to broaden the search for national origin groups. However, we are aware that the relationship between race and national origin is complex, so we carefully considered each label included in our search. We also used another process to find more keywords. Each file in the Internet Archive is associated with a list of keywords in the recording description and title. From the most frequent keywords of the dataset, we selected the set of keywords that would be associated with Asian American topics or speakers.

[2]https://github.com/Maliha-Jahan/GPT_Annotation/blob/main/Collection/keywords.txt

Figure 2: Example of keywords

Our crawler[3] then searched the Internet Archive for audio and video files that matched these keywords and downloaded them. Similar to (Galvez et al., 2021a), we have only acquired data that are licensed under CC-BY (Creative Commons Attribution), CC-BY-SA (Creative Commons Attribution-ShareAlike), and CC-BY-NC (Creative Commons Attribution-NonCommercial) because we intend to release our dataset publicly.

3.2. Transcription and Diarization

Once the audio and video files are downloaded, we generate the transcripts through the following five steps.

1. **Data cleaning**: Since the audio data from the Internet Archive contained many irrelevant and unwanted sounds, such as music and environmental noises, the first step was to clean the speech data. We used an audio tagging model, *i.e.* Panns[4] (Kong et al., 2020) to filter out the non-speech parts and noisy parts. More specifically, all the audio files were cut into continuous 10-second segments, and each clip was tagged by Panns. We kept the segments for which the probability of being a speech-related event is over 90%, and the probability of being other events was less than 5%.

2. **Voice activity detection**: We removed all the non-speech parts, e.g., silence, music, etc., with a voice activity detection (VAD) model[5].

[3]https://github.com/Maliha-Jahan/GPT_Annotation/blob/main/Collection

[4]https://github.com/qiuqiangkong/audioset_tagging_cnn

[5]https://huggingface.co/pyannote/voice-activity-detection

3. **Overlapping speech detection**: Overlapping speech can degrade the performance of speaker diarization and speech recognition. To combat that, we used an overlap detection model[6] to detect and remove the parts where more than one speaker talked simultaneously.

4. **Speaker diarization**: The clean data then served as input for an advanced speaker diarization system[7] (Bredin et al., 2020; Bredin and Laurent, 2021) to obtain the timestamps, while noting which speaker was speaking at each time-step. After diarization, each resulting segment was tagged with a speaker identifier, e.g. **Speaker_00**.

5. **Automatic speech recognition**: We used a medium-size whisper model [8] (Radford et al., 2023) to transcribe these audio files.

3.3. GPT Annotation

The annotation tasks performed by GPT-4 in this pipeline were multifaceted, as depicted in Figure 3. There were two main tasks: Filtering and Tagging.

3.3.1. Filtering

Out of all the files we downloaded in bulk, we expected a significant portion to be irrelevant to our purpose. Because some of these unfiltered files might have had wrong titles or descriptions or contained no Asian American speakers. Manual screening requires listening to each audio file and determining its relevance based on criteria such as being non-fictional, having coherence in dialogue or monologue, and having content pertaining to race, ethnicity, or national origin. This method could have taken months, which prompted us to streamline it using GPT-4. The filtering task was boolean, with two outcomes: 'Relevant' or 'Irrelevant.' We provided the GPT-4 model with a prompt including a set of instructions and the audio transcript, obtained as explained in Section 3.2. It responded with either 'Relevant' or 'Irrelevant.' For prompting and collecting responses, see section 3.3.3.

3.3.2. Tagging

After the filtering, the files that were deemed relevant were sent to GPT-4 for identification tagging. We used the term **'Identification Instance'** to indicate an instance (line/phrase) in the transcript where a speaker identified their own or someone else's race/ethnicity/national origin group. The tagging task involved finding such instances. An identification instance comprised the following fields.

–Identifier - Identified - Type - Race - National Origin - Ethnicity–

Identifier (Ir.) corresponded to the speaker uttering the line, **Identified (Id.)** was the speaker whose race/national origin/ethnicity was identified, and **Type**, **Race**, **National Origin (Nat. Or.)**, and **Ethnicity (Eth.)** were the target information as described in section 2. Except for **Identifier** and **Type**, any field could be empty. The following is an example of an identification instance:

–Speaker_00 - Speaker_03 - Other-person, in-interaction identification - Asian - Japanese - Non-Hispanic or Non-Latinx–

We ignore identification instances where the identified speaker is not present in the recording, e.g., public figures, fictional characters, etc. The tagging task has the following steps:

1. Finding the **'Identification Instances'**. These were places (lines/phrases) in the transcript where a speaker identified their own or someone else's race/ethnicity/national origin.

2. Finding the **Identifier** and the **Identified** speaker. For each speech segment, the transcript included the identification label of the speaker who pronounced it. Thus, finding the **Identifier** was not a challenge, but inferring who was being identified was.

3. Finding information about each field described in section 3.3 from each **Identification Instance**. Note that the fields for an **Identification** can only have information found in that specific instance (line/phrase) to avoid over-assumption from context.

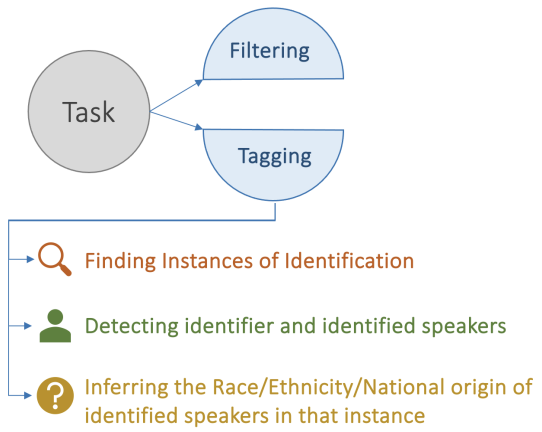
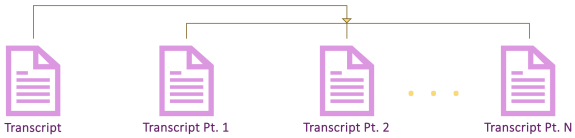


Figure 3: The outline of the complex tasks performed by GPT-4.

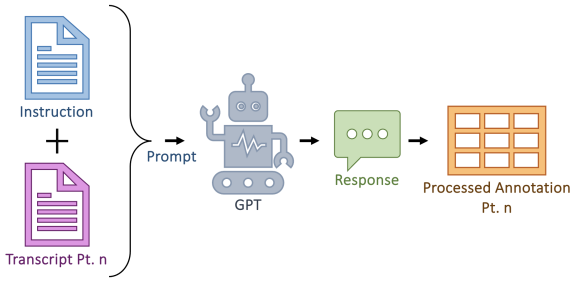
[6]https://huggingface.co/pyannote/overlapped-speech-detection

[7]https://huggingface.co/pyannote/speaker-diarization

[8]https://github.com/openai/whisper



(a) Large transcripts are split into multiple parts.



(b) Each part, with the prompt, is sent to GPT-4.

Figure 4: Generating annotation using GPT-4.

For tagging, we provided GPT-4 with instructions and a transcript. It responded with a list, where each item in the list corresponded to a found identification instance that included the previously mentioned fields (Identifier, Identified, Type, etc.). We call this task tagging as it is equivalent to human annotators tagging the identifications in a transcript.

3.3.3. Annotation Process

We used **OpenAI**'s API[9] for the GPT-4 models. GPT-4 responds to inputs, or "prompts," by generating text outputs. Thus, the annotation process for both filtering and tagging involved: **a)** Preparing a set of instructions that contained a clear and concise description of the task, definitions of relevant terms, the sequence of steps involved, and the required structure of the response, **b)** Sending the instructions along with a transcript to GPT-4 for annotation, and **c)** Processing the response. Filtering and tagging used different sets of instructions[10]. Figure 4 illustrates the process.

For GPT-4 to properly annotate a transcript, each transcript had to be accompanied by a set of instructions. The set of instructions and the transcript made up the 'prompt' in our case. We attached the same instruction with each transcript and generated a response for them through GPT-4, as shown in Figure 4. Often, the transcripts were quite large. Since the GPT models had limitations on the number of tokens, we needed to split a large transcript into multiple parts. Tokens are text units that can be as short as a character or as long as a word. We attached the set of instructions before each chunk and sent them separately, as shown in Figure 4 (a). If the transcript was split into N parts, we ended up with N responses for that transcript, which were then concatenated. If there were contradictory identifications, we kept all of them since the wrong ones would eventually get removed in the human verification step (section 3.4).

3.4. Manual Verification

The only purpose of the manual verification step is to create the final dataset after refining the GPT-4 annotations. This step is not involved in the evaluation process. In the experiments we performed in section 4, we evaluated the annotations generated by GPT-4 before this step.

Human annotators verified GPT-4's annotations to eliminate or correct potential 'false positives': identifications that GPT-4 flagged but were either wrong or not instances of identification at all. As will be discussed in section 4, GPT-4 occasionally over-interpreted, leading to the detection of identifications where none existed. Then, the annotators reviewed and refined the transcript, focusing solely on dialogues from speakers identified with any specific race, ethnicity, or national origin at some point in the transcript. For example, if a transcript had three speakers- Spk_00, Spk_01, and Spk_02, and Spk_01 was identified as Asian at any point in the transcript, then only the lines spoken by Spk_01 were manually checked and corrected. The correction involved amending all diarization and/or automated speech recognition errors.

4. Evaluation

Due to limited resources, we evaluated[11] our pipeline using randomly selected 440 files from our downloaded dataset. We used annotations by human annotators as ground truths. The evaluation was performed by comparing GPT-4's filtering and tagging results with the ground truths. We also experimented with GPT-3.5 for filtering and compared it with GPT-4. As mentioned in section 3.4, the annotations we evaluated are solely from the GPT models without applying any manual correction.

4.1. Filtering Experiment

4.1.1. Ground Truth

The human annotators separated the relevant and irrelevant files following the same filtering criteria as GPT-4: being coherent, being non-fictional conversation or monologue, and content relating to race, ethnicity, or national origin. Among the 440 files we selected for our experiments, 105 were deemed relevant by the human annotators. The human annotators used metadata such as the title

[9] https://platform.openai.com/docs/guides/gpt

[10] https://github.com/Maliha-Jahan/GPT_Annotation/tree/main/Prompts/

[11] https://github.com/Maliha-Jahan/GPT_Annotation/tree/main/annotate

and the description of the audio files to expedite the annotation process. It took a human annotator approximately 40 hours to filter all the files.

4.1.2. Experiment

We filtered the 440 files using the process in section 3.3.1. We treated the relevance of a file as a binary label. We compared the labels assigned to these files by human annotators and by GPT-4. We also performed the same experiment with GPT-3.5 Turbo and compared the two models' performances. Our prompt included an instruction describing the task and mentioning pointers to take note of. These pointers are specific instructions that we included after examining the responses of GPT-4 and finding out the error patterns.

Model	FN Rate (%)	FP Rate (%)
GPT-4	6.93	55.09
GPT-3.5-turbo	11.00	29.04

Table 2: Performance of filtering using GPT

Table 2 shows the result of these experiments. FP and FN rates are false positive and false negative rates. A false positive is an irrelevant file marked as relevant by GPT. A false negative is the opposite. The higher the FP rate, the less precise the method is. Since filtering is the step before tagging in our pipeline, lower precision means more irrelevant files get passed to the tagging step, and more resources are required afterward. A higher FN rate indicates that the method is missing a lot of relevant files. As shown in table 2, GPT-4 misses very few files, less than 7%, but the FP rate is higher. While GPT 3.5 has a lower FP rate of 29%, the FN rate goes up. Since it is more important to not miss any identification than to have a few false alarms for the tagging task, we pick GPT-4 for it. But for filtering, GPT-3.5 appears to be a better option.

4.2. Tagging Experiment

4.2.1. Ground Truth

From the 105 files that were classified as relevant by human annotators, we randomly picked 76 files to perform the tagging experiment. To perform the human annotations for this task, we used Gecko as an interface(Golan Levy, 2019). The annotators first skimmed through the transcripts generated with Whisper for keywords indicating race, ethnicity, or national origin, then read through the transcript and tagged the portions of text containing keywords that indicated a speaker's race, ethnicity, or national origin. If the **Identified** speaker was a different speaker from the **Identifier**, the annotators also tagged the **Identified** speaker. The annotators were instructed to tag both white and non-white races and all relevant ethnic groups, even though this portion of the dataset intended to focus on Asian-American speakers. An identification instance for human annotation involved the same fields as discussed in section 3.3.

The annotators only tagged speakers' remarks about race, ethnicity, or national origin if they were explicit or could be solidly supported by the context of the transcript beyond doubt. Otherwise, the text was not to be tagged. **Explicit identification** is where the speaker directly attributes a description of race, ethnicity, or national origin to a specific speaker in the same interaction. For example, a speaker can identify their own national origin explicitly by uttering **"I am Korean."**. A speaker can also identify another speaker's national origin explicitly by uttering **"You are half Japanese."**. In both cases, the human annotator could use the context to unambiguously select the speaker of reference that the utterances were intended to address. **Implicit identification** corresponds to instances when a speaker identifies or eludes to race, ethnicity, or national origin in some indirect way. This can happen in several ways. Firstly, a speaker can identify the origins of their family members or ancestry, e.g., identifying their parents' origins by saying **"I was the second child of my parents who had emigrated from Vietnam."**, and thus attributing a national origin to themselves. Secondly, a speaker can also express their affiliation with a racial, ethnic, or national origin group by using phrases such as "as a (member of)". Similarly, the speaker can say **"I'm proud as Asian American."**. Lastly, attributions of race, ethnicity, or national origin can also be inferred from culturally specific context. For example, an utterance such as **"There wasn't a single line about my parents and grandparents being interned at Gila and Manzanar."** contains a reference to the internment of people of Japanese descent during World War II. It allows inference of Japanese ancestry from the speaker's "parents and grandparents" and, subsequently, an attribution of the speaker's own ancestry through family relations.

4.2.2. Experiment

We evaluated the race and national origin fields separately. An **identification** in a GPT-4 annotation matched one in the corresponding human annotation only if all the fields of interest matched. For instance, if **Identifier** (**Ir.**), **Identified** (**Id.**), and **Race** were the fields of interest, an identification [$Speaker\_00$, $Speaker\_01$, $White$] would not match with [$Speaker\_00$, $Unknown$, $White$] or [$Speaker\_00$, $Speaker\_02$, $White$], since not all the fields matched. Here, $Speaker\_n$ is n^{th} speaker.

We tested different versions of instructions to

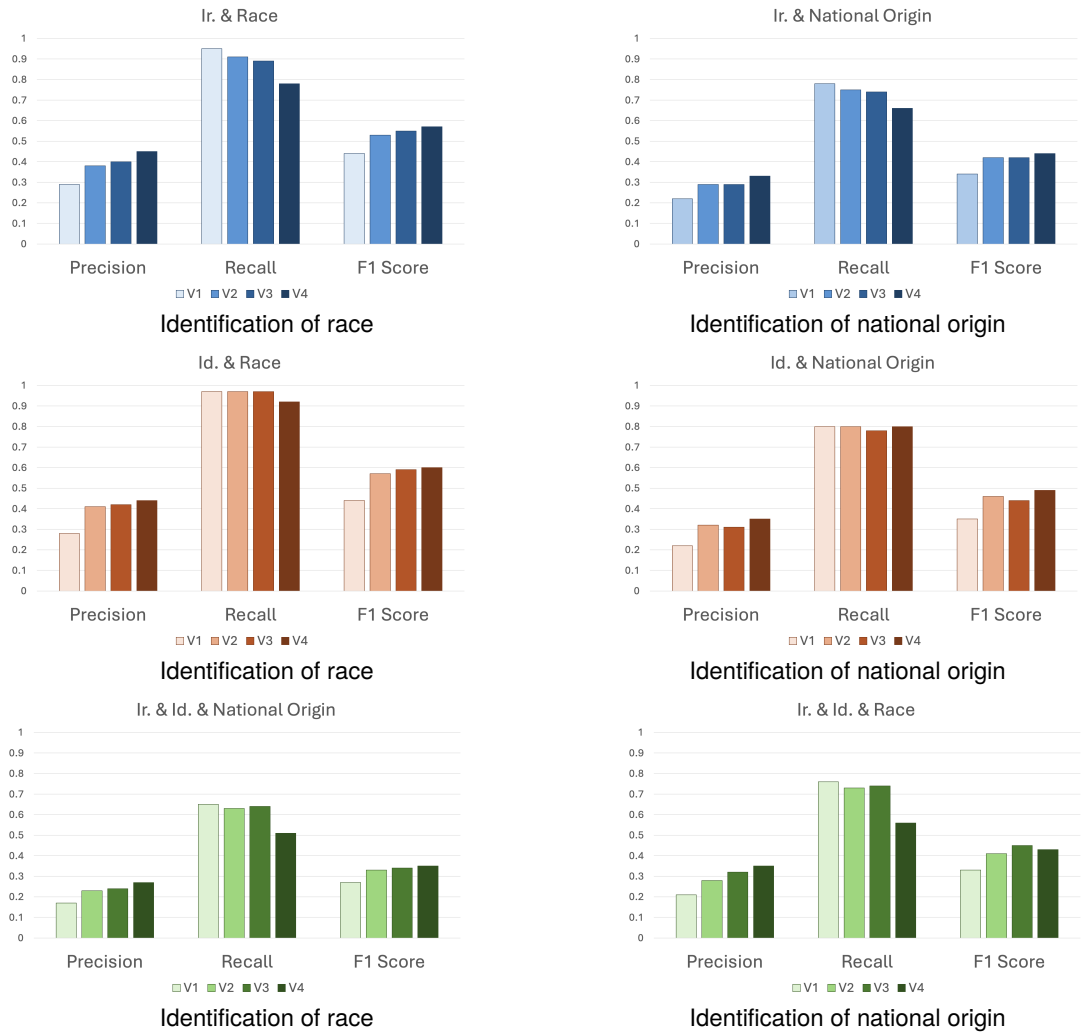


Figure 5: Performance of Annotation using GPT-4

see how changes in the prompt affected GPT-4's performance.

- **Version 1 (v1)** includes definitions of all the fields (section 3.3.2), a step-by-step description and breakdown of the task, and the required structure of the response. In the task steps, We used words like 'imply' and 'infer' and directly suggested the use of context.
- **Version 2 (v2)** is the same as Version 1, with the only difference being one new line stating a warning not to over-interpret or include unsupported information
- **Version 3 (v3)** is very similar to Version 2, with one big difference. We removed all usage of words like 'imply' and 'infer' and removed the suggestion about using context. We also add a 'pointer' saying that a mere mention of race/ethnicity/national origin is not of interest unless it is related to a speaker's identity.
- **Version 4 (v4)** has quite a few differences compared to the other three. This one only focuses on self-identifications. We add an extra step in the breakdown of the task that asks to decide if the transcript has any identification instance at all before finding and listing the specific instances. We also include examples of lines containing instances of identifications, making it a few-shot task.

Figure 5 shows GPT-4's performance with the instructions v1, v2, v3, and v4. It is obvious from the figure that precision and recall have a trade-off. When the instruction focuses on not missing any identification, GPT-4 may end up with more false positives, while if the focus is on minimizing false positives, GPT-4 may miss some identifications. A false positive means that GPT-4 finds an identification instance not present in the human annotation.

The figure shows that, with one line of difference, **Version 2** increased the precision scores by 37% on average compared to Version 1. But recall decreased, though not always, by around 2.5% on average. The performance did not change much from **Version 2** to **Version 3**, implying that the one warning line about not over-interpreting nullified the

mentions of implying and suggestions of context usage, so removing them made no big difference. The precision increased by about 11% on average, while the recall dropped by around 12% on average when going from **Version 3** to **Version 4**. So, the few-shot case demonstrated lower false positives as the examples provided outlines of how lines containing identifications look like. But that limits the scope of 'possible identification' space and results in higher misses.

So, version 2 or 3 is the best option if it is important to detect as many identifications correctly as possible. However, if saving resources is of the most importance, then version 4 is the best as it yields much fewer false positives. Fewer false positives mean the manual reviewer will have to go through fewer lines, resulting in a decreased resource requirement. On average, the total word count of all lines identified by GPT-4 was only 3% of the total word count of the transcript, suggesting that if human annotators only review the lines annotated by GPT-4 instead of the entire transcript, they would only need to examine 3% of it.

5. Analysis

We performed a time comparison to see how much GPT can speed up the annotation process. While it took about 40 hours for a human annotator to filter 440 files, the GPT models (4 and 3.5 Turbo for filtering) only took about 2 hours on average. For tagging, human annotation typically takes time equivalent to the length of the audio file (*human time*). However, GPT-4 tagging combined with manual verification of the tags took about 0.2x *human time*.

Error analysis on the false positives and false negatives was conducted to investigate patterns of errors made by GPT-4 as well as any other discrepancies between machine and human annotation. The error analysis was made on two features: race and national origin. We found that GPT-4 was effective at utilizing long context, such as context from several lines before or after, but that backfired sometimes. The annotators avoided making assumptions and only confirmed the presence of identification if supported by textual evidence in the transcript. Sometimes GPT-4 used broad contextual information, which may not have matched the human annotations. This could result in false positives, even though the identification itself was not necessarily wrong. Rather, the difference lied in the degree to which the annotators and GPT-4 allowed the interpretation of implicit information.

5.1. False Positives - Race

Over-detection of identification by GPT-4 could be seen in 138 examples, or 60.5% of the total number of lines examined. Reasons for over-detection were the presence of triggering words, inference from speakers' names, or the presence of outside-interaction identification.

"I wanted an Asian-American band." was labeled by GPT-4 as having an identification of race, but in this instance, "Asian-American" was only modifying a group entity. Similarly, GPT-4 would confuse words referring to language as indicators of race identification: ***"I did a Cambodian language school."*** Generic mentions of racial groups also had a similar effect, for example, ***"a practice that was instilled in me by the Vietnamese women who raised me"***. In another example, and others with the same pattern, GPT-4 concluded from proper names that they were instances of identification of the speaker's race: ***"That's JD reading from her memoir..."*** While this technique could yield reasonable results for Asian-Americans, it could be less conclusive for African-Americans. For this reason, we decided not to rely on names for finding identifications.

GPT-4 also had a tendency to pick out the outside-interaction identification, as in ***"JD was African-American"***. Since we were only interested in identifications of race, ethnicity, and national origin of speakers participating in the interactions, this example and similar lines counted as false positives. GPT-4 also tended to make generalizations about race from places of birth, while the human annotation guidelines would only allow inferences of national origin from such information. For example, ***"I was born in Guam in 1966."*** was annotated by GPT-4 as having 'native Hawaiian or other Pacific islander' racial identification. Thus, race and national origin were often confused in the GPT-4 annotations.

Analyzing the false positives has revealed 46 instances (20.2%) of race identification that the human annotators overlooked. One type of missed identifications involved speakers revealing their racial origins indirectly, for instance by discussing their ancestry or family background: ***"he went from Hong Kong came to America and i think they denied him entry. Because remember you had the Chinese Exclusion Act."*** In this instance, the speaker was talking about their grandfather who was denied entry for being Chinese. GPT-4 was able to correctly infer the speaker's racial background that the human annotators missed. In light of such errors, the human annotation guidelines were revised with additional clarifications about the different ways in which self-identifications can be expressed indirectly (e.g., by ancestry, family, heritage, group identity, etc.). In another instance,

GPT-4 utilized contextual knowledge and correctly inferred that the speaker was of Japanese descent via family affiliation: ***"fortunately they didn't go through the whole internment camp experience which I think I'm very fortunate that my immediate family was spared that whole ordeal during the war"***.

5.2. False Positives - National Origin

Over-detection by GPT-4 of national origin identification (45 examples or 40.9%) showed patterns similar to those identified previously, such as modifiers related to a nation or a language: ***"I've lived in China."*** Outside-interaction identification often triggered GPT-4 to create a false positive. The following line, for instance, was irrelevant since the speaker's identity was not discussed, but GPT-4 deemed it relevant based on a previous mention of the speaker's Chinese origin: ***"We're hoping to do it actually here next year in January of 1991 again as part of the Chinese New Year"***.

Several false positives overlooked by the human annotators were ones where the national origin was identified to be "American". For example, ***"I grew up in Chicago although I was born in North Carolina."***. Such an implicit treatment of "American" as the default for national origins was also corrected in the revised annotation guidelines.

5.3. False Negatives

Compared to false positives, there were far fewer false negatives to examine, as GPT-4 tended to be more loose in capturing the identification compared to humans. Nonetheless, a few patterns could be established. GPT-4 seemed to have difficulties resolving identification when a complex syntactic structure with subordinate clauses was involved (see square brackets): ***"it is incontestable [that the applicant, [or me], is of Asian descent and part of an Asian band]"***. Reported speech also posed a challenge: ***"when we asked them why that was the case, the trademark office said it's because you're too Asian"***. Multiple national origins appeared to cause identifications to fail as well, such as in the false negative example: ***I'm fifth-generation Japanese American and sixth-generation Chinese American***."

Discourse-level complexity could also be confusing. When a speaker introduced a second speaker in the third person, GPT-4 mistook that the second speaker was not present, thus missing the following second-person identification: ***"Some of the Americans in Hawaii at that time were of Japanese ancestry. They included JD, who's now a physician in Sacramento, but at that time, he was a nine-year-old boy. He's in our studio."***

6. Conclusion

In this article, we introduced a novel semi-automatic pipeline that utilizes GPT-4 to tackle the challenging task of annotating a unique audio dataset with speakers from diverse racial, ethnic, and national origin groups. This pipeline opens up opportunities for rapid and efficient annotation, reducing the burden of finding resources for manual annotation. The pipeline consists of automatic audio data collection from the Internet Archive, SOTA ASR and diarization systems, an automated filtering of the pertinent files, and annotation of the speakers' identifications to various groups.

We have evaluated the performance of our data collection and annotation pipeline by preparing human annotations of all tasks on a subset of the files and comparing them to the automated results. The automatic filtering with GPT models showed as low as a 7% False Negative Rate for up to 55% False Positive Rate, ensuring a very high recall for a fraction of the human filtering time. By experimenting with various prompts, we showed high recall of up to 97% and 80%, respectively, across the three variations of identification tasks of Race and National origin, which is useful for capturing most of the identifications. By comparison with the human annotation on the same task, and analysis of the false positives, GPT-4 found up to 20.2% of identifications missed by humans.

Although false positive rates were high in both filtering and annotation tasks, the proposed method can be used for pre-dataset creation. This pre-dataset will have to be reviewed by humans to remove false positives, but the time and human resources employed to obtain the final dataset will be much lower than building a dataset without using LLM annotations.

Examining the false positives and false negatives served as a basis for revising both the prompt to GPT-4 and the annotation guidelines for humans. Many false positives and relatively much fewer false negatives were reflected in generally high recall and lower precision scores of the GPT-4 annotations (Figure 5). GPT-4 appeared to make reasonably good inferences based on context, which might make it a good tool for bootstrap human annotation and large-scale dataset creation. These results, overall, indicated a reasonably good performance for our purposes: using GPT-4 for pre-annotation with a follow-up human review.

Our future works will include the improvement of the annotation precision. We are also pursuing the application of this pipeline to new demographic groups in the USA, such as African Americans and Latinx Americans, in the hope to support more inclusive systems, with the use of more variability in the training datasets available.

7. References

Revisions to the standards for the classification of federal data on race and ethnicity. 62 Fed. Reg. 210 (Oct. 30, 1997).

Hervé Bredin and Antoine Laurent. 2021. End-to-end speaker segmentation for overlap-aware resegmentation. *Proc. Interspeech 2021*, pages 3111–3115.

Hervé Bredin, Ruiqing Yin, Juan Manuel Coria, Gregory Gelly, Pavel Korshunov, Marvin Lavechin, Diego Fustes, Hadrien Titeux, Wassim Bouaziz, and Marie-Philippe Gill. 2020. Pyannote. audio: neural building blocks for speaker diarization. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7124–7128. IEEE.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2924–2936, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, and Rosie Jones. 2020. 100,000 podcasts: A spoken English document corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5903–5917, Barcelona, Spain (Online). International Committee on Computational Linguistics.

U.S. Equal Employement Opportunity Commission. Eeoc enforcement guidance on national origin discrimination. https://www.eeoc.gov/laws/guidance/\eeoc-enforcement-guidance-national-\origin-discrimination.

Shizhe Diao, Pengcheng Wang, Yong Lin, and Tong Zhang. 2023. Active prompting with chain-of-thought for large language models.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. Is GPT-3 a good data annotator? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada. Association for Computational Linguistics.

Daniel Galvez, Greg Diamos, Juan Ciro, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Maximilian Lam, Mark Mazumder, and Vijay Janapa Reddi. 2021a. The People's Speech: A Large-Scale Diverse English Speech Recognition Dataset for Commercial Usage. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1 (NeurIPS Datasets and Benchmarks 2021)*.

Daniel Galvez, Greg Diamos, Juan Torres, Keith Achorn, Juan Cerón, Anjali Gopi, David Kanter, Max Lam, Mark Mazumder, and Vijay Janapa Reddi. 2021b. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30).

Ido Amir Eduard Golshtein Ran Mochary Eilon Reshef Reichart Omri Allouche Golan Levy, Raquel Sitman. 2019. Gecko - a tool for effective annotation of human conversations. In *20th Annual Conference of the International Speech Communication Association, Interspeech 2019*, Herzliya, Israel.

Caner Hazirbas, Joanna Bitton, Brian Dolhansky, Jacqueline Pan, Albert Gordo, and Cristian Canton Ferrer. 2022. Towards measuring fairness in ai: The casual conversations dataset. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 4(3):324–332.

Xingwei He, Zhenghao Lin, Yeyun Gong, A-Long Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2023. Annollm: Making large language models to be better crowdsourced annotators.

Eric Jensen, Nicholas Jones, Kimberly Orozco, Lauren Medina, Marc Perry, Ben Bolender, and Karen Battle. 2021. Measuring racial and ethnic diversity for the 2020 census. https://www.census.gov/newsroom/blogs/random-samplings/2021/08/measuring-racial-ethnic-diversity-\2020-census.html.

Matthijs Kalmijn and Frank Van Tubergen. 2010. A comparative perspective on intermarriage: Explaining differences among national-origin groups in the United States. *Demography*, 47(2):459–479.

Tyler Kendall and Charlie Farrington. 2023. The corpus of regional african american language. The Online Resources for African American Language Project.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. 2020. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894.

Klaus krippendorff. 2004. Measuring the Reliability of Qualitative Text Analysis Data. *Quality & Quantity*, 38(6):787–800.

U.S. Department of Commerce. National origin discrimination. https://www.commerce.gov/\cr/reports-and-resources/discrimination-quick-facts/\national-origin-discrimination.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1267–1273, Stroudsburg, PA, USA. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Upcounsel. National origin: Everything you need to know. https://www.upcounsel.com/national-origin.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. LLMaAA: Making large language models as active annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103, Singapore. Association for Computational Linguistics.