

# Experiences with the Introduction of AI-based Tools for Moderation Automation of Voice-based Participatory Media Forums

Aman Khullar  
aman.khullar@oniondev.com  
Gram Vaani

Abhishek Burnwal  
abhishekburnwal2@gmail.com  
IIT Delhi

Priyadarshi Hitesh  
priyadarshi.hitesh.mcs19@cse.iitd.ac.in  
IIT Delhi

Paramita Panjal  
paramita.panjal@gramvaani.org  
Gram Vaani

Prashit Raj  
cs1170359@cse.iitd.ac.in  
IIT Delhi

R Jayanth Reddy  
r.jayanth.reddy.cs116@cse.iitd.ac.in  
IIT Delhi

Aaditeshwar Seth  
aseth@cse.iitd.ac.in  
Gram Vaani, IIT Delhi

Rachit Pandey  
rachit.pandey@oniondev.com  
Gram Vaani

Ankit Akash Jha  
ankit.jha@alumni.iitd.ac.in  
IIT Delhi

Himanshu  
himanshurewar14@gmail.com  
IIT Delhi

## ABSTRACT

Voice-based discussion forums where users can record audio messages which are then published for other users to listen and comment, are often moderated to ensure that the published audios are of good quality, relevant, and adhere to editorial guidelines of the forum. There is room for the introduction of AI-based tools in the moderation process, such as to identify and filter out blank or noisy audios, use speech recognition to transcribe the voice messages in text, and use natural language processing techniques to extract relevant metadata from the audio transcripts. We design such tools and deploy them within a social enterprise working in India that runs several voice-based discussion forums. We present our findings in terms of the time and cost-savings made through the introduction of these tools, and describe the feedback of the moderators towards the acceptability of AI-based automation in their workflow. Our work forms a case-study in the use of AI for automation of several routine tasks, and can be especially relevant for other researchers and practitioners involved with the use of voice-based technologies in developing regions of the world.

## CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in collaborative and social computing**; • **Information systems** → **Decision support systems**; • **Computing methodologies** → **Artificial intelligence**.

## KEYWORDS

Interactive Voice Response systems, content moderation, artificial intelligence, automation

## 1 INTRODUCTION

Voice-based discussion forums using IVR (Interactive Voice Response) systems have emerged as a useful modality for information

sharing among low-income and less-literate populations in developing regions [24, 28, 31, 41]. These forums are accessible through ordinary phones, not smartphones, and do not require the internet. Users can call and listen to audio messages, and respond by recording their own message which can be published back on the forum. Voice forums have found wide application in citizen journalism [24, 25], social accountability and grievance redressal [2, 21], agricultural and livelihood information [28], behavior change communication [3], and cultural expression [17], among others.

Content moderation is a crucial element in the operation of voice forums. Most projects cited above employ a team of content moderators who listen to the recorded messages, and take action. They may reject the audio for poor audio quality or empty recordings, lightly edit the audio for publication on the platform, evaluate against editorial guidelines followed by the platform managers on aspects such as hate-speech or misinformation, annotate the recordings with tags or transcriptions, etc. Some of these steps are similar to content moderation done on internet-based participatory media platforms, where a combination of algorithmic and manual checks are used to filter inappropriate messages [12, 32]. Depending upon the platform design, peer users may be involved in a prior step to flag messages for further inspection [7], reasons behind the moderator actions may or may not be transparently disclosed to the users [6], and users may be offered guidance by the moderators to submit more acceptable messages [24].

We explore the feasibility of automation of some parts of the moderation process on voice-based discussion forums. We build machine learning based models to detect and reject empty audio recordings, determine the gender of the speaker, then use commercial ASR (Automatic Speech Recognition) APIs to obtain the transcript of the audio, and extract metadata from the transcript such as locations mentioned in the audio. In this study, we do not examine research topics on detection of hate-speech or fake news. We examine whether automation of the content operations listed above, can improve the efficiency of the content moderators so

that they can devote more time to complex editorial checks rather than some of the above mundane tasks. Our evaluation is done in the context of several voice-based discussion forums operated by the social enterprise Gram Vaani. Gram Vaani employs about fifteen full-time content moderators who among other tasks manage content moderation on 40+ discussion forums that cumulatively generate approximately 1000 voice recordings each day. Through our evaluation, we examine whether automation of some of the tasks was found to be acceptable and useful by the content moderators, whether the moderation process can be entirely automated, and changes that the automation tools brought about in the moderation practices.

Our main contributions are about interesting insights we gained in building machine learning models, a careful understanding about the introduction of AI-based methods in the moderation process, and a detailed time-cost analysis to evaluate any financial or time-saving benefits with automation of the moderation process. We find that with the current accuracy of speech recognition systems, moderation automation can reduce the workload of moderators but it certainly cannot replace them. This niche area of work presents an interesting case-study about AI-based methods and humans complementing each other. We have also open-sourced our models and made them available as a python library for the benefit of other researchers and practitioners working with voice-based discussion forums.

## 2 RELATED WORK

Content moderation is an integral part of voice-based participatory media platforms, with each platform having its own moderation policies to regulate permissible content on the platform. One such platform called Mobile Vaani [36], in whose context our research has been conducted, has fifteen full-time content moderators who review the audio content through a web-based content management system [33]. CGNet Swara [25] is another IVR platform with full-time moderators who check content quality and relevance, in line with the platform goals. Similarly, Avaaj Otalo [28] and IVR Junction [41] are voice forums which work for societal development and also have a set of dedicated moderators for content moderation.

As these platforms scale, a proportional increase in moderation resources is required. Increasing the size of the moderation team has its own limitations with management and funding constraints. Further, several tasks related to content moderation can be routine and mundane, drawing attention away from more crucial aspects that require human judgement. On the internet, community or distributed moderation options have been explored on platforms such as Reddit, Stack Overflow, and Slashdot. Several studies [4, 5, 18] have demonstrated the efficacy of these moderation policies and how they are in general a viable option for scaling of public forums. Sangeet Swara [37] and Gurgaon Idol [17] were among early voice-based forums to experiment with some aspects of community moderation as well, such as ranking of audio content based on its quality, and identifying the broad topic category of the content.

Our work is with a similar goal to help scale voice-based forums. We however focus on the automation of routine tasks specific to voice-based discussion forums, such as the identification of empty

or noisy audio recordings, and metadata extraction from automatically transcribed speech recordings. We do not explore content-based moderation for aspects like hate-speech or misinformation, since this requires human judgement and is specifically why we want to free up time for the moderators to be able to focus on these aspects, as well as explore distributed moderation in the future where community volunteers can also participate in making these judgements. Our work is closer to platforms like BSpeak, ReCall and Respeak [38–40], which also used automatic speech recognition to achieve a reduced transcription cost and faster turnaround time for data collection tasks allocated to visually-impaired people. We evaluate the benefits that can be gained from recent advances in speech recognition and natural language processing for low-resource languages while acknowledging the risks of full automation in decision support systems [8]. For tasks like automatically rejecting blank or noisy audio, and gender identification, we build upon techniques in audio classification using signal processing [13, 20] and convolutional neural networks [14, 19]. For metadata extraction tasks, we operate on transcripts obtained through commercial speech recognition APIs and build topic classification and named entity recognition methods [26, 42]. We deployed these tools in the Gram Vaani [35] technology stack and ran several experiments and feedback studies to understand their impact on the moderation workflow.

## 3 MACHINE LEARNING MODELS

After several years of running voice-based discussion forums that were moderated manually, Gram Vaani has accumulated a large annotated dataset of audio recordings. We henceforth refer to these as *audio items* or simply *items*. The dataset has been annotated with the following metadata:

- **Item state:** Whether the audio recording was accepted for publication on the IVR or rejected. For most items, a reason for rejection is also marked of whether it was rejected because it was an empty audio recording, or it had too much background noise, or the recording was not articulate enough, or it was rejected for editorial reasons such as cyber bullying or hateful content.
- **Gender:** The gender of the speaker - Female, Male, Third gender, or a Group of speakers, identified by the moderators based on the voice of the speaker and any identification information they provided about themselves in the audio recording.
- **Location:** The State, District, Sub-district and the village of the speaker, based on the details provided by the speaker in their recording.
- **Tags:** Tags to identify a broad topic are marked by the moderators, out of a large set of approximately 150 standard tags that have emerged over several years at Gram Vaani. Short-lived event specific tags are also developed. These tags are helpful for project staff and researchers to search for items.
- **Rating:** A qualitative assessment of the quality of the item, with a value between 1 to 5. The quality largely depends on the content spoken in the recording, of whether it provides some interesting or novel information, or a new viewpoint.
- **Transcription:** This contains the transcription given by the moderators to the item, and may either be a full transcript or

**Table 1: Confusion matrix for the blank classifier**

Predicted Class → True Class ↓	Blank	Accepted
Blank	632	19
Accepted	33	2849

a short gist of what the user is speaking about. Moderation policies varying from project to project are used by the moderators to determine whether to give a gist or full transcript to the audio.

- Title: Each audio recording is also annotated with a title or headline given by the moderators.

We use a subset of this data to build and evaluate machine learning models, as described next.

### 3.1 Blank Classifier

The blank classifier attempts to classify whether a recording is empty or it has human speech. We used a subset of 17,000 audio recordings to build the train set, and 3,500 audio recordings to build the test set. We labeled an item as *accept* if its publication state was an accept and it had a rating of 4 or 5. Items were labeled as *reject* if their publication state was a reject and the reason for rejection was specified as blank. We ensured that we sampled these training and test-set audio recordings for both male and female speakers, and from across different geographic regions where Gram Vaani has been working, to provide diversity.

We build a feature set for an audio as follows. We first segment the given audio recording into chunks having a frame size of 50ms and a step size of 25ms, and then use the pyAudioAnalysis library [9] to extract a set of 34 audio features for each audio chunk. These include the mean amplitude, entropy of the amplitude, spectral spread, MFCCs and other signal processing features. We then find the distribution of these features across all the chunks in an audio, and use the 4 quartiles for each feature as the actual features that are fed into a machine learning classifier. Using these features, we trained several classifiers and found the Random Forest Classifier followed by recursive feature elimination and hyper-parameter tuning, to provide an accuracy of 98.5%, as shown in table 1, and a false negative rate of 1.15%.

### 3.2 Gender Classifier

We use the same methodology as above to train a gender classifier as well, that classifies based on the audio whether the speaker is a male or female. The classifier is intended to be applied only on non-blank audio that may be recommended for publication. To build

**Table 2: Confusion matrix for the gender classifier**

Predicted Class → Marked Class ↓	Male	Female
Male	552	68
Female	36	579

the training set, we used a subset of around 4,700 audio recordings, with a balanced split between the Male and the Female classes, sampled across the different geographies covered by Gram Vaani.

For feature extraction, we identified a subset of 136 features using recursive feature elimination from the feature-set distribution used in the blank classifier, which we then used to train our machine learning model. We finally trained an SVM classifier and fine-tuned the hyper-parameters to obtain a test set accuracy of 91.6% on around 1,200 audio recordings, as shown in table 2.

### 3.3 Transcription and Location

We used the Google ASR (Automatic Speech Recognition) API [11] to obtain transcripts for the audio recordings. These STT (Speech to Text) transcripts were then analyzed to extract location entities. We experimented with inbuilt entity extraction tools provided by Google’s Dialog Flow suite [10], but did not get a very high accuracy, likely due to the use of local names. We therefore built our own custom entity extraction tool.

Using a multilingual text processing library called Polyglot [30] which uses parts of speech rules for entity extraction, we first obtain candidate location entities identified by Polyglot’s Named Entity Recognition (NER) module. We then run direct and approximate string matching for these entities against the list of all Indian states, districts, and sub-districts, according to the 2011 Indian census [27]. We updated the census dataset with a few changes such as for the state of Andhra Pradesh which was split into two states, and similarly for a few districts that were split into smaller districts. We also added name variations such as *East/West* Champaran to the Hindi equivalent of *Purbi/Pashim* Champaran. In case a Hamming distance similarity of less than 10 (empirically determined threshold) was found between a location entity spotted by Polyglot and the census entries, we identified that entity as a valid location. We also backfilled information such that if a district name was matched then we auto-filled the name of the corresponding state, or if a sub-district was matched then we auto-filled the name of the district and state.

Table 3 reports the accuracy at the state and district levels, computed on a ground-truth of 425 datapoints. The accuracy provided by Dialog Flow was much lesser than the state level accuracy of 87.78% and district level accuracy of 75.57% obtained through our custom methods. These accuracies are computed on STT transcripts which contain meaningful location information. We found that between 25 to 30% of the STT transcripts were empty or with incorrect information due to ASR inaccuracies.

**Table 3: Accuracy of custom location entity extractor as compared to Google’s Dialog Flow suite**

Entity	Bad STT (%)	Accuracy of DF with good STT	Accuracy of custom module with good STT
Location			
State		51.81	87.78
District	28.71	51.15	75.57

## 4 TIME-COST ANALYSIS

We next report our findings from deploying the models described in the previous section. We analyze the impact of introducing these AI-based methods on potential time and cost savings, and how the moderators reacted to incorporating the tools in their workflow.

To build a study design, we first spoke to a few moderators to understand their daily workflow in detail. We found that moderators essentially perform four tasks. The first task involves a quick determination of whether an audio item may be publishable or not, based on its audio quality. Second, the moderators mark various metadata fields (such as the gender of the speaker, location, topic, title, and assign a qualitative rating between 1-5 based on the relevance of the item) associated with the item. To carry out this task, the moderators first listen to the audio and then mark the metadata. Third, the moderators listen to the item again to write its transcript. Here, the moderators determine whether they should just write a short gist or not even that, or should they give a full word-to-word transcription for the item. The moderators follow internal policies, such as to give a detailed transcript to interesting or unique items that may merit being shared with the wider project team, or for grievances related to government schemes which are often shared through social media channels to draw the attention of government administrators to the issue [2]. Finally, the moderators may lightly edit the item to normalize its volume or remove periods of silences before publishing it back on the the IVR platform.

In our study, we evaluate the impact of automation on the first three tasks only: namely, to identify and reject poor quality audio items, provide metadata annotation automatically to a few fields, and assist the moderators with providing an STT (Speech to Text) transcription obtained through the Google ASR APIs. We conduct this study in two rounds, with and without the automated tools being in place. In each round, we isolate the impact along two axes. First, we isolate the impact due to the three tasks of blank audio detection, metadata annotation, and transcription. It was not feasible to further breakdown the metadata annotation to different sub-parts because the moderators mark these fields in one go making it difficult to time their effort on each one separately. Second, for the transcription component, we separately analyze the impact depending upon whether the moderators gave a gist transcription or a full transcription.

The study was carried out with the help of 13 moderators, with at least four moderators participating in the experiment each day, for a span of 33 days. The activity was carried out for 6 hours each day in an instrumented manner. The moderators were asked to use a stopwatch to note down the time taken in carrying out each of these tasks for each item that they moderated. They also noted the item-id (a numeric six digit unique identifier for each item) so that we could link their notes with other item details such as its audio duration and the accept or reject action taken upon the item. The moderators counted the time they take to listen to the audio as part of the overall task duration. We next present the results to understand the impact of the blank classifier in reducing the number of items put up to the moderators for review, the impact of the gender classifier and location entity extraction on the time spent in metadata annotation, and time savings made by providing an STT transcription. The study was carried out during a time when

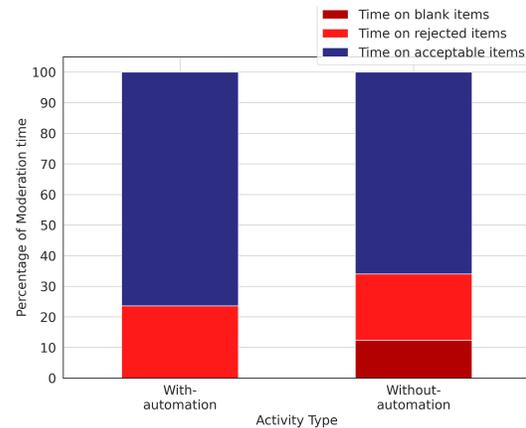


Figure 1: Overall time-saving offered by the Blank classifier

COVID-19 related topics had subsided in India and regular project activities had resumed, with no significant difference in the two study rounds of with and without automation.

### 4.1 Blank classifier

We first present the time-saving offered by the blank classifier. Out of 498 items received during the with-automation phase, the moderators rejected 174 (34.94%) items. During the without-automation phase, the moderators received 712 items and rejected 437 (61.38%) items.

Figure 1 shows the time-saved by the moderators: we found that the moderators spent 13% less time in sifting through rejected items when the blank classifier was deployed, which provided a saving of 11.74 seconds per accepted item and amounted to a time-saving of 17.07% in rejecting a blank item per accepted item. While the savings are modest, upon speaking with the moderators they reported that earlier they would do a quick check by waiting for the audio to be downloaded and then look at its waveform visualization to determine if it was blank or not, but with the automation they are able to save on the download wait time and move faster. Another interesting benefit they reported was that seeing less items for moderation helps reduce their anxiety. Especially during the COVID-19 lockdown in India, the IVR platforms were very heavily used, and a moderator reported:

*“During the early days of Coronavirus, there used to be 500 items each day and we got anxious seeing so many items to moderate, but after ML (machine learning) was deployed, we see much lesser number of items. During Covid, ML helped us very much.”* — Senior Moderator, Gram Vaani, Gurgaon.

### 4.2 Metadata annotation

Figure 2 shows the time-saving with metadata annotation deployed for gender marking and location entity extraction. In aggregate, moderators saved 5.97% time with automation, amounting to an average of 3.34 seconds saved per item out of 56.04 seconds taken on average for metadata annotation per item.

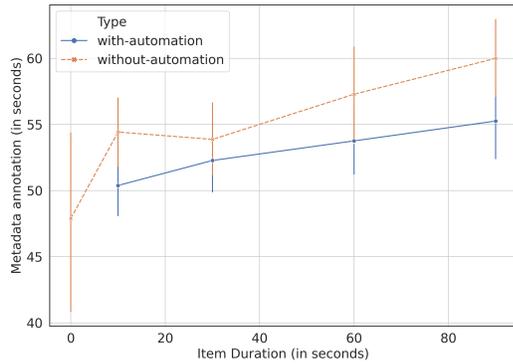


Figure 2: Time-saving offered by metadata annotation

### 4.3 Transcription

4.3.1 *Gist Transcription.* The moderators gave gist transcriptions to 186 items during the with-automation experiment, and to 139 items during the without-automation experiment. To understand variation with the duration of audio items, we create broad bins for the recordings and analyze the time saving within each bin for when the STT transcripts are available to the moderators and when they are not. Figure 3 and table 4 show the time savings which seem to become more significant for longer audio recordings. In aggregate,

Table 4: Duration-wise time-saving in gist transcription

Item Duration (sec)	Avg. time saved (s)	Avg. time taken (s)
10-20	-1.25	59.56
20-40	3.13	79.60
40-60	11.52	81.80
60-100	9.29	93.61
Greater than 100	10.61	85.04

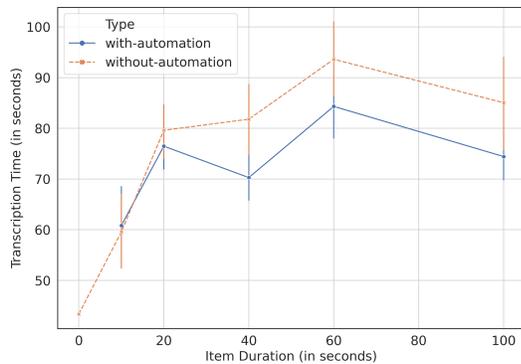


Figure 3: Time-saving offered by automated STT transcripts to write human-curated gist transcripts

moderators saved 8.6% time with automation, amounting to an average of 6.99 seconds saved per item out of 81.61 seconds taken on average for writing a gist transcription per item. The savings are modest though, and the moderators confirmed upon further discussion that the STT is useful in helping grab the name of the person or the location, but they prefer relying on well-practiced templates for writing gists. For example, if somebody records a song or poem, or appreciates a programme they have heard on Mobile Vaani, the moderators just write a line of the form: “[name] from [location] {talks about, sings, appreciates} [topic]”. Since the moderators anyway have to spend time in listening to the item, the STT does not offer substantial benefit in informing the moderators about the topic or offering text that can be copy-pasted to build the gist transcript. The STT is more useful if the moderators want to provide a full transcript, as described next.

4.3.2 *Full Transcription.* The moderators wrote full transcripts for 142 recordings during the with-automation experiment, and for 136 items during the without-automation experiment. Figure 4 and table 5 shows the time-savings for audios of different duration. In aggregate, moderators saved 17.77% time with automation, amounting to an average of 56.92 seconds saved per item out of 339.39 seconds taken on average for writing full transcription per item. These savings are much more than with the gist transcripts, as expected, but still quite modest. We were expecting that the moderators will be able to do a direct copy-paste of the STT transcripts but that was not the case. The moderators pointed out that the STT accuracy for some items could be very poor, while in other

Table 5: Duration-wise time-saving in full transcription

Item Duration (sec)	Avg. time saved (s)	Avg. time taken (s)
10-20	-0.57	135.26
20-40	-25.06	191.54
40-60	39.16	316.11
60-100	102.99	475.77
Greater than 100	208.00	608.40

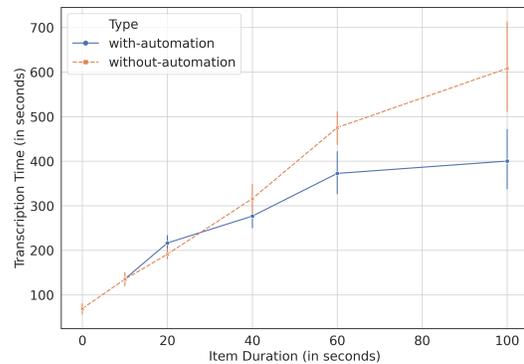
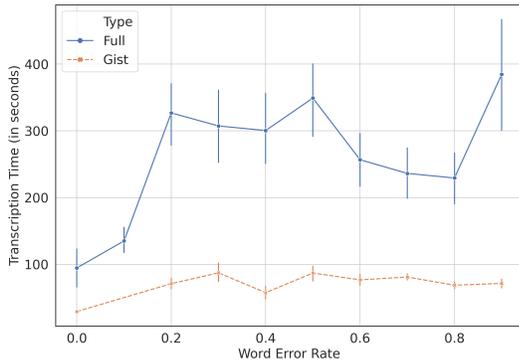


Figure 4: Time-saving offered by automated STT transcripts in writing human-curated full transcripts



**Figure 5: Affect of STT WER (Word Error Rate) on transcription time**

cases they did use the transcripts but had to correct it to fix the inaccuracies. The correction process itself is not straightforward, as one of the moderators explained:

*“We are not able to understand what the recording is talking about only with the help of ML and we need to listen to the audio and then pause the audio, remove the incorrect word, add the correct word and then play the audio again. All this takes a lot of time and we find it better to just sometimes write the word-to-word transcript ourselves”*  
 — Senior Moderator, Gram Vaani, Gurgaon.

The moderators broadly stated that as long as word errors are in the range of 10-15%, they find the STT to be useful, but otherwise they prefer writing the transcript from scratch and at best just copy parts of the STT. This also bore out in our analysis, shown in Figure 5, where we plot the WER (Word Error Rate) for the STT transcripts compared with a manual transcription. We can see that as the WER climbs above 10%, the time taken by the moderators for writing a full transcript increases substantially, likely due to them resorting to transcribing the items themselves without relying on the STT transcripts. For gist transcripts on the other hand, since the moderators tend to write templated transcripts, the variation with WER is not very significant.

#### 4.4 Actual-run Experiment

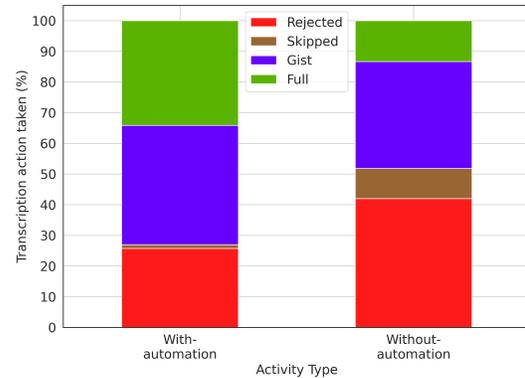
As we have seen so far, the use of automation does lead to time-savings, and we wanted to see how the moderators choose to use this time. We therefore ran an additional experiment with and without automation, as before, but this time we removed the restriction of moderating for 6 hours only. We wanted to create as natural an environment as possible so that the moderators could freely allocate their time to tasks they believed to be important. Additionally, to understand the reasons behind the actions taken by them especially on utilizing the STT transcripts, we asked them to also note the actions they took, as follows:

- Skipped transcription: Moderators skipped writing a transcript for the item
- Transcription type: Whether the item was given a gist or a full transcription

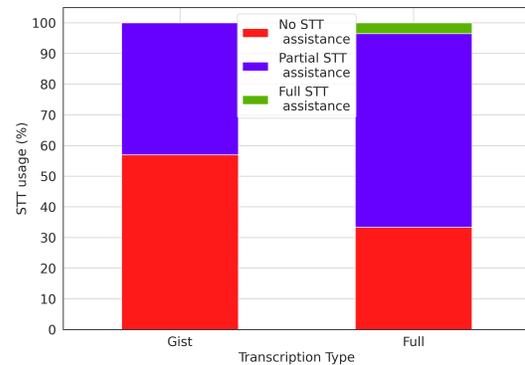
- No STT assistance: The STT had too many errors, or the information was not useful, to utilize it for a gist or full transcription
- Partial STT assistance: The moderators copied parts of the STT transcript and made edits
- Full STT assistance: The moderators copied the STT transcript as such

In this experiment, we saw a total of 422 items, out of which 167 (of which 124 were accepted for publication) were studied during the with-automation phase, and 255 (of which 148 were accepted for publication) were studied during the without-automation phase.

Figure 6a shows the differences in the actions taken by the moderators with and without automation. The most marked difference is that the time-saving during automation seems to have been used to provide more full transcripts and less of skipped items than without automation. We found this to be interesting that moderators felt that many more items deserved to be transcribed than what they were able to do without automation. This was confirmed during feedback sessions with the moderators, where many of them



**(a) Difference between the action taken by moderators with and without automation**



**(b) Use of STT to provide a gist or full transcription**

**Figure 6: Actual-run experiment**

**Table 6: Item duration wise average time-saving**

Item duration bins	% accepted items with gist transcript	Avg. time saved in writing gist transcript (s)	% accepted items with full transcript	Avg. time saved in writing full transcript (s)	Avg. time saved in transcription (s)
10-20	75.00	-1.25	25.00	-0.57	-1.08
20-40	60.00	3.13	28.00	-25.06	-5.14
40-60	75.00	11.52	20.00	39.16	16.47
60-100	59.62	9.29	30.77	102.99	37.50
100 or more	56.10	10.61	36.59	208.00	82.90

reported that they often faced a time crunch and were not able to give due attention to many useful items. In Figure 6b, we further analyzed the use of STT transcripts to provide a gist and full transcription. While *Full STT assistance* for full transcripts was rare, *Partial STT assistance* was taken actively by the moderators to write full transcripts, and also to write gist transcripts to some extent. In about 30% of the cases for full transcripts, *No STT assistance* was taken at all due to poor STT transcript quality. Overall, we found that the AI-based tools were accepted by the moderators and they did have an impact in terms of shifting the emphasis on various tasks done by the moderators.

#### 4.5 Aggregate Time-cost Savings

We use the results of the actual run experiment to calculate the aggregate time saving. Table 6 shows the percentage distribution of accepted items across the different bins, the average time-saving per item attributed to automated gist and full transcription respectively. Through this aggregate analysis, with the assumption that the arrival pattern of items remains invariant across the audio duration bins, blank and acceptable items in each bin, and those deserving gist or full transcription in each bin, we find that automation is able to provide a time-saving of approximately 40% per item (an approximate time-saving of 54 seconds per item out of the average time of 134 seconds taken to moderate an item) with the help of automation. This saving, as we saw, helps the moderators provide additional transcriptions, or can be used towards other useful tasks such as calling the users to seek feedback regarding their participation on the voice forums, or provide guidance to the users to record better audio messages.

Converting to cost terms, we use an average moderator salary as INR 20,000 per month, with 48 hours of work per week, 15 items moderated per hour, and an additional cost overhead of 30% to account for office space, utilities, etc. This leads to an average per-item moderation cost of INR 8.3. A 40% time-saving results in an improved per-item moderation cost of INR 4.98. However, to this we need to add the cost of the Google ASR APIs used to obtain the STT transcripts. The APIs cost INR 0.29 for a 15 seconds audio, and aggregating over the same distribution of audio duration gives an average STT transcript cost of INR 1.45 per item. We

inflate this by applying an overhead cost of 30% for additional technology management and other expenses like the use of cloud computing infrastructure. Adding it to the per-item moderation cost with automation, this gives us a final cost-saving of almost 17% with automation. The automation therefore provides us with both cost-saving as well as time-saving. The time-saving is more significant and can be used towards tasks which otherwise were getting ignored or sidelined in favour of various necessary routine tasks.

## 5 DISCUSSION

### 5.1 Other Opportunities

The benefits of AI-based automation both in terms of time and cost saving encourages us to find other opportunities for AI-based tools as well. A few such options include:

- *Detection of noisy items:* Just like the blank-items classifier, we are building a classifier to detect noisy items which can be automatically rejected. We however found that the same audio features we used for the blank and gender classifiers, are not adequate for the noisy classifier since many audios are noisy due to background human voice and a TSNE plot of the features showed that there is no clear separation between noisy and accepted items. Therefore, we developed a CNN (convolutional neural network) based classifier which used the mel scaled audio spectrogram [22] as input to the classifier and achieved 98.2% accuracy and a false negative rate of 3.6%. We are now in the process of deploying this model in production and are also trying to improve upon the model accuracy by augmenting our internal dataset with an external dataset containing 50 environment sounds [29].
- *Tag annotation:* Tags related to broad topics such health, agriculture, education, governance, etc. form a significant part of metadata annotation done by the moderators. We are building multi-label tag classifiers based on the STT transcripts, to pre-assign tags to the audio items. Simple keyword based classifiers using word vectors or TFIDF weighted bag-of-words approaches are not able to give a very good performance, and we are looking at other approaches that

can utilize underlying hierarchical structures in the tag assignments as well.

- *User interfaces for editing transcripts:* The STT transcripts obtained through the Google ASR APIs also provide confidence scores for each word in the transcript. When the STT transcripts are presented to the moderators, highlighting words predicted with low-confidence in the STT transcripts may make it easier for the moderators to spot errors in the transcripts and selectively fix these errors. If such improvements in the user interface make it easier for the moderators to utilize the STT transcripts then it is likely to lead to further time savings.
- *Continuous model improvements:* We evaluated the performance of our deployed machine learning models to observe the change in accuracy of such systems from a test environment to a production environment. We observed a performance dip of 3-4% among both the blank classifier as well as the gender classifier. We further examined if any gender bias had crept into the false negatives predicted by the blank classifier but found that the model performance was unbiased and false negative rate was equal among both the genders. To improve production accuracy, we are using a human-in-the-loop approach [43] to build an automated pipeline which incorporates the feedback of the moderators on a monthly basis and fine-tunes the trained models through hard negative mining. Such an approach should help in a continuous model improvement and prevent the models from drifting in the production environment.
- *Identification of erroneous predictions:* Our interaction with the Gram Vaani moderators revealed that they would benefit from seeing a confidence score against the prediction results of the various classifiers, since it would give them hints on which contributions to examine more carefully than others. This is in line with HCI best practices suggested for the incorporation of AI-based automation in risk assessment tool [8] and should help in identification and correction of erroneous classifier predictions.
- *Distributed moderation:* Centralized moderation stands the risk of making incorrect editorial decisions due to a lack of knowledge of the local context. This may include decisions on ranking and exposure given to different content, recognition of misinformation, acceptance/rejection of contentious contributions, etc. We have not studied such editorial decision making processes as yet among the content moderators, and the impact of AI-based automation on these processes. Since the Mobile Vaani platforms are supported by a team of offline community volunteers who are well placed to take more informed editorial decisions, as part of future work, we plan to also study editorial decision making in centralized and distributed moderation settings, with and without AI-based automation tools such as the ones described in this paper. Several interesting questions will be worth investigating, of how to arrive at a consensus in the editorial decisions taken by a group of volunteers, how do these compare with decisions made by the central team of moderators, time-saving among volunteers and moderators through the use of automation tools, and alternate activities for which time was

created such as to guide community members to make better contributions or to connect more deeply and frequently with the community members. We believe that AI-based tools can also be helpful to identify these community members who would benefit from guidance by the volunteers and reduce user dropout.

## 5.2 Use of Automated Tools During COVID-19

The Gram Vaani voice-forums were heavily used during the COVID-19 lockdown in India, which was announced towards the end of March 2020. An overnight order for suspension of factories and transport caused wide distress among migrant workers who were stranded in cities, rural populations who faced food shortages due to broken supply chains, and significant income loss due to stoppage of agricultural and daily wage work [1]. People called into the IVR platforms to report about problems they were facing, described their experiences, and asked for help. Over a million people used the platforms during the first few months of the COVID-19 lockdown, and contributed over 20,000 voice recordings [16]. The content moderators were naturally overwhelmed with this sudden spike and we conducted a rapid deployment of an initial version of some of the moderation automation tools we have presented in this paper. We first bifurcated content contributions in the IVR itself by asking people whether they wanted to seek emergency help or to contribute a voice report. The voice recordings of those asking for help was transcribed through the ASR APIs and the location entity extraction module was used to identify the place from which people were calling. In some geographies, these recordings were immediately passed on to the Gram Vaani volunteers from that location to provide assistance. In the Delhi National Capital Region where migrant workers were stranded and rendered out of cash and food, the volunteers were able to quickly connect them with humanitarian organizations working on the ground and managed to help 1,800 workers with food kits and cash transfers [23]. Several other organizations also requested for a rapid deployment of the IVR and were able to assist several thousand workers with transportation to their villages by registering their demand and using it to persuade governments to arrange trains for their safe transport

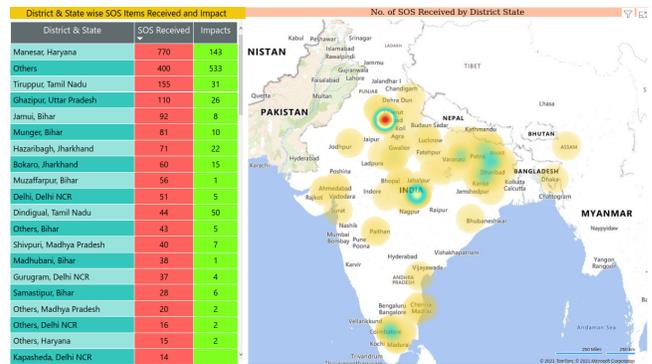


Figure 7: Location-wise grievance recordings as visible on Gram Vaani's live dashboard.

[34]. The moderators appreciated the benefit of this automated segregation and delegation, which reduced the workload on them:

“During Coronavirus, the Saajha Manch project received a huge number of news recordings and during that time ML helped us a lot. In a day, our team of 4 moderators were able to moderate 400 items each”.

We also built a simple word frequency based categorization to understand the nature of the problems being reported by the people, such as: *out of food, stuck in city, health emergency, out of cash, not received government cash transfers, bank not accessible, black marketing and price rise of food, gas relief not received, social distancing not being followed, issues at isolation centers, agriculture and other livelihood related issues*, etc. We then clubbed this category information with the output from the location entity extractor to map the issues on our own live dashboard (as shown in figure 7) and aggregators like *MapMyIndia* [15]. These dashboards helped portray the extent and intensity of the distress experienced by the people and were used in PILs (Public Interest Litigations) filed by activists to draw the government’s attention to the huge distress induced by the sudden and stringent lockdown.

## 6 CONCLUSIONS

Overall, we found that the AI-based tools we built were able to give modest cost-saving of 17% but substantial time-saving of 40% for content moderation in voice-based forums. These tools cannot replace human moderators but can augment their work. This nuance is different from oft-heard analysis of impending unemployment due to AI-based automation; our experience rather suggests that AI-based tools can improve the productivity of human workers rather than replace them, and the time thus saved can be dedicated for other important tasks that may have been ignored so far. Insights from the detailed instrumentation exercise conducted by us can be useful for other researchers and practitioners working with voice-based tools to automate their operations.

## ACKNOWLEDGMENTS

We would like to express our immense gratitude to the entire moderation team for helping us conduct this study and providing us with daily feedback and results for our time-cost analysis. We would also like to thank the entire Gram Vaani team who provided us with constant support in conducting this project. We would also like to thank the ACT4D lab at IIT Delhi for providing us with the initial compute infrastructure. Finally, we would like to thank the Bill & Melinda Gates foundation for funding this project.

## REFERENCES

[1] Vani Viswanathan Aaditeshwar Seth. 2020. “What Covid-19 Means To Us’ Voices from the Indian Hinterland. <https://www.theindiaforum.in/article/what-covid-19-means-us>

[2] Dipanjan Chakraborty, Mohd Sultan Ahmad, and Aaditeshwar Seth. 2017. Findings from a civil society mediated and technology assisted grievance redressal model in rural India. In *Proceedings of the Ninth International Conference on Information and Communication Technologies and Development*. 1–12.

[3] Dipanjan Chakraborty, Akshay Gupta, and Aaditeshwar Seth. 2019. Experiences from a mobile-based behaviour change campaign on maternal and child nutrition in rural India. In *Proceedings of the Tenth International Conference on Information and Communication Technologies and Development*. 1–11.

[4] Eshwar Chandrasekharan, Umashanthi Pavalanathan, Anirudh Srinivasan, Adam Glynn, Jacob Eisenstein, and Eric Gilbert. 2017. You can’t stay here: The efficacy

of reddit’s 2015 ban examined through hate speech. *Proceedings of the ACM on Human-Computer Interaction* 1, CSCW (2017), 1–22.

[5] Eshwar Chandrasekharan, Mattia Samory, Shagun Jhaver, Hunter Charvat, Amy Bruckman, Cliff Lampe, Jacob Eisenstein, and Eric Gilbert. 2018. The Internet’s hidden rules: An empirical study of Reddit norm violations at micro, meso, and macro scales. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–25.

[6] Kate Crawford. 2016. Can an algorithm be agonistic? Ten scenes from life in calculated publics. *Science, Technology, & Human Values* 41, 1 (2016), 77–92.

[7] Kate Crawford and Tarleton Gillespie. 2016. What is a flag for? Social media reporting tools and the vocabulary of complaint. *New Media & Society* 18, 3 (2016), 410–428.

[8] Maria De-Arteaga, Riccardo Fogliato, and Alexandra Chouldechova. 2020. A case for humans-in-the-loop: Decisions in the presence of erroneous algorithmic scores. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.

[9] Theodoros Giannakopoulos. 2015. pyaudioanalysis: An open-source python library for audio signal analysis. *PLoS one* 10, 12 (2015), e0144610.

[10] Google. 2021. *Dialog Flow*. <https://cloud.google.com/dialogflow>

[11] Google. 2021. *Speech To Text*. <https://cloud.google.com/speech-to-text>

[12] Robert Gorwa, Reuben Binns, and Christian Katzenbach. 2020. Algorithmic content moderation: Technical and political challenges in the automation of platform governance. *Big Data & Society* 7, 1 (2020), 2053951719897945.

[13] Guodong Guo and Stan Z Li. 2003. Content-based audio classification and retrieval by support vector machines. *IEEE transactions on Neural Networks* 14, 1 (2003), 209–215.

[14] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. 2017. CNN architectures for large-scale audio classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 131–135.

[15] Map My India. 2021. *Map My India*. <https://www.mapmyindia.com/>

[16] Mira Johri, Sumeet Agarwal, Aman Khullar, Dinesh Chandra, Vijay Sai Pratap, Aaditeshwar Seth, and the Gram Vaani Team. 2021. The first 100 days: how has COVID-19 affected poor and vulnerable groups in India? *Health Promotion International* (05 2021). <https://doi.org/10.1093/heapro/daab050> arXiv:<https://academic.oup.com/heapro/advance-article-pdf/doi/10.1093/heapro/daab050/37949360/daab050.pdf> daab050.

[17] Zahir Koradia, Piyush Aggarwal, Aaditeshwar Seth, and Gaurav Luthra. 2013. Gurgaon idol: A singing competition over community radio and IVRS. In *Proceedings of the 3rd ACM Symposium on Computing for Development*. 1–10.

[18] Cliff Lampe and Paul Resnick. 2004. Slash (dot) and burn: distributed moderation in a large online conversation space. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 543–550.

[19] Honglak Lee, Peter Pham, Yan Largin, and Andrew Ng. 2009. Unsupervised feature learning for audio classification using convolutional deep belief networks. *Advances in neural information processing systems* 22 (2009), 1096–1104.

[20] Lie Lu, Hong-Jiang Zhang, and Hao Jiang. 2002. Content analysis for audio classification and segmentation. *IEEE Transactions on speech and audio processing* 10, 7 (2002), 504–516.

[21] Meghana Marathe, Jacki O’Neill, Paromita Pain, and William Thies. 2015. Revisiting CGNet Swara and its impact in rural India. In *Proceedings of the Seventh International Conference on Information and Communication Technologies and Development*. 1–10.

[22] Brian McFee, Alexandros Metsai, Matt McVicar, Stefan Balke, Carl Thomé, Colin Raffel, Frank Zalkow, Ayoub Malek, Dana, Kyungyun Lee, Oriol Nieto, Dan Ellis, Jack Mason, Eric Battenberg, Scott Seyfarth, Ryuichi Yamamoto, viktorandreevichmorozov, Keunwoo Choi, Josh Moore, Rachel Bittner, Shunsuke Hidaka, Ziyao Wei, nullmightybofo, Dario Hereñú, Fabian-Robert Stöter, Pius Friesch, Adam Weiss, Matt Vollrath, Taewoon Kim, and Thassilo. 2021. *librosa/librosa: 0.8.1rc2*. <https://doi.org/10.5281/zenodo.4792298>

[23] Gram Vaani Community Media. 2020. *Lockdown Chronicle: The story of a Migrant workers’ platform across India’s lockdown*. <https://drive.google.com/file/d/1ViL56UUX5g-AGddriF4N2bXAj4LHXNM/view>

[24] Aparna Moitra, Vishnupriya Das, Gram Vaani, Archana Kumar, and Aaditeshwar Seth. 2016. Design lessons from creating a mobile-based community media platform in Rural India. In *Proceedings of the Eighth International Conference on Information and Communication Technologies and Development*. 1–11.

[25] Preeti Mudliar, Jonathan Donner, and William Thies. 2012. Emergent practices around CGNet Swara, voice forum for citizen journalism in rural India. In *Proceedings of the Fifth International Conference on Information and Communication Technologies and Development*. 159–168.

[26] David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Linguisticae Investigationes* 30, 1 (2007), 3–26.

[27] Government of India. 2011. *Census Data*. <https://censusindia.gov.in/2011-common/censusdata2011.html>

[28] Neil Patel, Deepti Chittamuru, Anupam Jain, Paresh Dave, and Tapan S Parikh. 2010. Avaj otalo: a field study of an interactive voice forum for small farmers

- in rural india. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 733–742.
- [29] Karol J. Piczak. [n.d.]. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia* (Brisbane, Australia, 2015-10-13). ACM Press, 1015–1018. <https://doi.org/10.1145/2733373.2806390>
- [30] Polyglot. 2021. *Polyglot*. <https://polyglot.readthedocs.io/en/latest/>
- [31] Agha Ali Raza, Mansoor Pervaiz, Christina Milo, Samia Razaq, Guy Alster, Jahanzeb Sherwani, Umar Saif, and Roni Rosenfeld. 2012. Viral entertainment as a vehicle for disseminating speech-based services to low-literate users. In *Proceedings of the Fifth International Conference on Information and Communication Technologies and Development*. 350–359.
- [32] Marietje Schaake and Rob Reich. 2021. *Election 2020: Content Moderation and Accountability*. [https://fsi-live.s3.us-west-1.amazonaws.com/s3fs-public/hai\\_cyberpolicy\\_election\\_3\\_v1.pdf](https://fsi-live.s3.us-west-1.amazonaws.com/s3fs-public/hai_cyberpolicy_election_3_v1.pdf)
- [33] A Seth, A Gupta, A Moitra, D Kumar, D Chakraborty, L Enoch, O Ruthven, P Panjal, RA Siddiqi, R Singh, et al. 2020. Reflections from Practical Experiences of Managing Participatory Media Platforms for Development. In *Proceedings of the 2020 International Conference on Information and Communication Technologies and Development*. 1–15.
- [34] Gram Vaani. 2021. *COVID-19 response services*. <https://gramvaani.org/?p=3631>
- [35] Gram Vaani. 2021. *Gram Vaani*. <https://gramvaani.org/>
- [36] Gram Vaani. 2021. *Mobile Vaani*. <http://mobilevaani.in>
- [37] Aditya Vashistha, Edward Cutrell, Gaetano Borriello, and William Thies. 2015. Sangeet swara: A community-moderated voice forum in rural india. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 417–426.
- [38] Aditya Vashistha, Abhinav Garg, and Richard Anderson. 2019. Recall: Crowdsourcing on basic phones to financially sustain voice forums. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.
- [39] Aditya Vashistha, Pooja Sethi, and Richard Anderson. 2017. Respeak: A voice-based, crowd-powered speech transcription system. In *Proceedings of the 2017 CHI conference on human factors in computing systems*. 1855–1866.
- [40] Aditya Vashistha, Pooja Sethi, and Richard Anderson. 2018. BSpeak: An accessible voice-based crowdsourcing marketplace for low-income blind people. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.
- [41] Aditya Vashistha and William Thies. 2012. {IVR} Junction: Building Scalable and Distributed Voice Forums in the Developing World. In *6th USENIX/ACM Workshop on Networked Systems for Developing Regions ({NSDR} 12)*.
- [42] Sida I Wang and Christopher D Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 90–94.
- [43] Zijie J Wang, Dongjin Choi, Shenyu Xu, and Diyi Yang. 2021. Putting humans in the natural language processing loop: A survey. *arXiv preprint arXiv:2103.04044* (2021).