ASPIRE CHALLENGE EVALUATION PLAN

Table of Contents

1 INTRODUCTION

This is the evaluation plan for the IARPA ASpIRE (Automatic Speech recognition In Reverberant Environments) Challenge. ASpIRE is a spin-off of the IARPA Babel program, which works to develop agile and robust technology that can be rapidly applied to any human language. Previous work has shown that automatic speech recognition (ASR) performance degrades on microphone recordings, especially when data used for training is mismatched with data used in testing. The ASpIRE Challenge asks the Solver to develop approaches to mitigate the effects of these conditions and create software that can function effectively in new acoustic environments and recording scenarios. Participants will have the opportunity to evaluate their techniques on a common set of challenging data that includes significant room noise and reverberation.

The Seeker (IARPA) asks Challenge Solvers to create innovative automatic speech recognition software that works in a variety of acoustic environments and recording scenarios without having access to matched training and development data. There are two evaluation conditions:

1. The *Single Microphone (single-mic) Condition* tests the ability to mitigate noise and reverberation given a single microphone recording (selected randomly) from speech recorded in several rooms with a variety of microphones.

2. The *Multiple Microphone (multi-mic) Condition* tests the ability to mitigate noise and reverberation given all of the microphone recordings of speech recorded in several rooms with a variety of microphones.

In both conditions, word error rate (WER) will be used as the objective measure of performance. Solvers can participate in either or both conditions.

This evaluation plan covers the data resources, Speech-To-Text task definition, file formats for both system inputs and outputs, evaluation metrics, scoring procedures, and the submission protocols for results.

2 DATA RESOURCES

Data resources used in this Challenge are described in this section.

Training Data:

Solvers are expected to train their algorithms using the following conversational telephone data sets:

1. LDC2004S13 Fisher English Training Part 1, Speech;
2. LDC2004T19 Fisher English Training Part 1, Transcripts;
3. LDC2005S13 Fisher English Training Part 2, Speech; and
4. LDC2005T19 Fisher English Training Part 2, Transcripts.

Only this data and algorithmic transformations of this data can be used in training systems. Solvers are also required to use only the above transcripts for the purposes of language modeling.

Development Data:

15 hours of development data (divided into 5 hour development and 10 hour development-test sets) of multi-mic recordings of conversational speech will be provided for each evaluation condition.

1. Single Microphone Condition:

i. ASpIRE_single_dev, Speech and Transcripts (5 hours): will be used for optimization, training-selection, and tuning purposes. This corpus contains speech recorded on a variety of far-field microphones in noisy, reverberant rooms. Only one microphone is provided for each recording, although the data will include speech recorded on different microphones.

ii. ASpIRE_single_dev_test, Speech (10 hours): will be used to evaluate systems using the online scoring tool and for posting scores on the Prodigy leaderboard. Like ASpIRE_single_dev, this corpus contains speech recorded on a variety of far-field microphones in noisy, reverberant rooms. Only one microphone is provided

for each recording, although the data will include speech recorded on different microphones.

2. **Multiple Microphone Condition:**
   i. ASpIRE_multi_dev, Speech and Transcripts (5 hours): will be used for optimization, training-selection, and tuning purposes. This corpus contains speech recorded on a variety of far-field microphones in noisy, reverberant rooms. Several microphone channels are provided for each recording.
   ii. ASpIRE_multi_dev_test, Speech (10 hours): will be used to evaluate systems using the online scoring tool. This corpus contains speech recorded on a variety of far-field microphones in noisy, reverberant rooms. Several microphone channels are provided for each recording.

The development data is designed to be different from the final evaluation-test set, but to provide a good representation of microphone recordings in real rooms with transcription conventions matching those of the evaluation set.

Please note that Solvers are not allowed to utilize development transcripts in training or language models. They may also not listen to or transcribe the development-test speech data. Single and multiple microphone development data sets should be considered separately, in order to best match the data available during evaluation. Additionally, Solvers are advised that multi-mic channel numbers should not be considered to represent a common microphone type or location between different files.

Evaluation-Test Data:

At the end of the development cycle, evaluation-test data that differs from the development and development-test data will be provided to Solvers for consecutive one week periods beginning with the single microphone condition. Solvers must submit system descriptions and system output by the end of the evaluation period for each condition to be eligible for the award associated with that condition.

1. **Single Microphone Condition:**
   i. ASpIRE_single_eval_test, Speech (10 hours): contains speech recorded on several different far-field microphones in several noisy, reverberant rooms that are different than those used in the development and development-test sets. Only one microphone is provided for each recording, although the data will include speech recorded on different microphones.

2. **Multiple Microphone Condition:**
   i. ASpIRE_multi_eval_test, Speech (10 hours): contains speech recorded on several different far-field microphones in several noisy, reverberant rooms that are different than those used in the development and development-test sets. Several microphone channels are provided for each recording.

Please note that Solvers are not allowed to utilize evaluation-test data in training or language models. They must also not listen to or transcribe the evaluation-test speech data.

The single-mic condition evaluation will run for a week, followed by the multi-mic evaluation.

3 THE SPEECH TO TEXT TASK

The Speech-To-Text (STT) task will be used for the evaluation of Challenge submissions. The goal of the STT task is to produce a verbatim, case-insensitive transcript of uttered lexical items. Systems will output a stream of Conversation Time Marked (CTM) lexical tokens reporting the token's begin and end time within the recording. A CTM file is token-based and includes the following information for each recognized token: the name of the source file, the channel processed, the beginning time of the recognized token, and the duration of the recognized token (See Appendix B). Optionally, Solvers may include a confidence score value [0,1] indicating the system's confidence that the token is correct.

Beginning in the early 1980's, evaluation of ASR stabilized on the current performance measure of word error rate (WER). This measure scores ASR performance using a case-less lexicalized form of ASR output known as the Standard Normalized Orthographic Representation (SNOR) format.[1] The WER is defined as the sum of all ASR output token errors divided by the number of scorable tokens in a reference transcription of the test data. There are three types of errors: tokens that are missed (deletion errors), inserted (insertion errors), and incorrectly recognized (substitution errors).[2]

The STT performance measure is essentially the same as the traditional NIST ASR WER measure using the NIST SCLITE software.

STT performance will be measured as a function of deletion, insertion and substitution error types against reference transcription files. System evaluation will occur in three steps: (1) text normalization, (2) reference-to-system token alignment, and (3) performance metric computation.

3.1 Format of Reference Transcription Files

Segment Time Marked (STM) scoring reference files are created from the human reference transcripts. See Appendix A for a description of the format of an STM file. Data is transcribed to the listener's best ability and may contain errors.

The following rules define three types of tokens: Scored tokens (tokens that must be recognized), optionally deletable

[1] Since some languages' written forms are not word-based, this concept has been extended to cover lexemes — a representation of a written unit of meaning within a language. Thus, this document frequently refers to lexemes, lexical tokens, or tokens rather than words. For English, the language of this evaluation, these terms may be treated more or less equivalently.

[2] Underlying the tabulation of errors is a requirement to align the tokens in the system output transcript with the tokens in the reference transcript. Traditionally, this has been done using a dynamic programming algorithm that searches for an alignment that minimizes the WER.

tokens (tokens that may be omitted by the STT system without penalty), and non-scored tokens (tokens removed from both the reference and STT transcripts prior to scoring).

- Scored tokens
  - All words transcribed as specified in the transcription guidelines, based upon LDC's transcription guidelines (NQTR)3 adapted to English.
- Optionally deletable tokens
  - Fragments (marked with a -) in the reference transcript. System tokens with token-initial text matching the fragment's text will be scored as correct (e.g. /theory/ would be correct for fragment /th-/). The same test is applied to the obverse, token-final fragments /-tter/ matching /latter/.
- Non-scored tokens
  - Unintelligible speech tags [(())] and semi-intelligible speech [((text))].
  - Non-lexical punctuation, to include the ~ symbol.
  - Non-lexical, speaker-produced sounds (<lipsmack/>, <cough/>, <breath/>, <sneeze/>) as defined in LDC's transcription guidelines.
  - Laugh tags (<laugh> and </laugh>); any text resulting from speech during laughing is scored.
  - Background noise tags (<background> and </background> and any text between these tags
  - Tags indicating last names of conversation participants (<iname></iname>).
  - Regions transcribed as <silence>.

Non-scored tokens are simply deleted. Word hypotheses during transcribed <silence> regions will be counted as insertion errors. This approach will likely have the most significant effect on STT performance during inter-speech gaps. Participant systems should be robust to noise and other sounds occurring during long pauses in speech. For the development and development-test data only, due to the nature of the data collection, there may also be audible bleed of the other side of the conversation, into the microphone recording. Transcribers worked off of held-out recordings, where this bleed was not present. Participants should not focus on this audio bleed phenomenon as it is not prominent in the evaluation-test data.

3.2 Token Normalization

Text will be pre-filtered to appropriately handle the speech phenomena. A single standardized spelling is required for scorable lexemes, and the STT system must output this spelling in order to be scored as correct.[4] Homophones must be spelled correctly according to the given context in order to be considered correct. Systems are to generate all tokens according to Standard Normal Orthographic Representation (SNOR) rules:

- Whitespace-separated lexical tokens (for languages that use whitespace-defined words)
- Case insensitive alphabetic text (usually in all upper case)
- Spelled letters are represented with the letter followed by a period (e.g., "a. b. c.")
- No non-alphabetic characters (except apostrophes for contractions and possessives and hyphens for hyphenated words and fragments).

Note that in scoring, hyphenated words will be divided into their constituent parts. Thus, for scoring, a hyphen within a token will be treated as a token separator. A hyphen at either end of a token string indicates the missing part of a spoken fragment.

Prior to scoring, we will use a global map file (GLM) to transform both the reference and system output token strings via a set of rules. We do so to ensure that we score as correct hypothesis tokens that do not differ semantically from corresponding reference tokens. The GLM file used during ASpIRE scoring will be made available to participants.

The GLM rules expand contractions in the system output to all possible expanded forms, which may generate several alternative token strings in the system output.

The GLM rules may also split a token string into two or more strings. For example, compound words are split into their constituents.

After GLM filtering, hyphens in both the system output and reference are transformed into token separators.

3.3 Token Alignment

Scorable tokens, as defined in Section 3.1, are aligned using the Dynamic Programming solution to string alignments[5]. The weights used for substitutions, insertions, deletions, and correct recognition are 4, 3, 3, and 0 respectively.

[3]https://catalog.ldc.upenn.edu/docs/LDC2010S01/trans_guide_nqrt_span.doc

[4]Token spelling is determined by first consulting an authoritative reference — e.g., the American Heritage Dictionary (AHD) for English. Lacking an authoritative reference, the internet is searched to find the most common representation. If no single form is dominant, then two or more forms will be permitted via an orthographic map file. As in past speech evaluations, a transcription filter and orthographic map file will be used on both the reference and hypothesis transcripts to apply rules for mapping common alternate representations to a single scorable form.

[5]http://www.nist.gov/speech/publications/storage_paper/lrec06_v0_7.pdf

3.4 The Scoring Metric

An overall Word Error Rate (WER) will be computed as the fraction of token recognition errors per reference token:

$$WER = \left(N_{Del} + N_{Ins} + N_{Subst}\right) / N_{Ref}$$

where

N_{Del} = the number of unmapped reference tokens,
N_{Ins} = the number of unmapped STT output tokens,
N_{Subst} = the number of mapped STT output tokens with non-matching reference spelling, and
N_{Ref} = the maximum number of reference tokens[6]

As an additional optional performance measure, the confidence of a system in its transcription output may be included by participants. In order to do this, the STT system should attach a measure of confidence to each of its scorable output tokens in the CTM file. This confidence measure represents the system's estimate of the probability that the output token is correct and must have a value between 0 and 1 inclusive. The performance of this confidence measure will be calculated using the same normalized cross entropy score that NIST has been using in previous ASR evaluations.[7] See Appendix C for more details. System confidence will not be used to judge the official performance of submitted systems.

3.5 Scoring Procedures

Once the pre-processing is complete, we align the system and reference tokens, using a token-mediated alignment optimized for minimum word error rate (WER). The scorable lexical token sequences from the reference and system output are aligned (using Dynamic Programming) to minimize the Edit Distance[8] between the two token sequences (edit distance is usually called the Levenshtein Distance, after the paper[9] by V. I. Levenshtein that appears to have introduced the idea).

The NIST SCTK toolkit[10] will be used to evaluate the performance of STT systems. System-generated STT output must be in CTM format. (See Appendix B) Participants must remove non-scored tokens described in Section 3.1 from their system output file before submitting them for scoring. The following command will score a CTM-formatted file with hubscr using a segment time marked (STM) formatted file supplied with the reference. This invocation is the one we will use for scoring purposes.

```
% hubscr.pl-V -g glm_filename.glm –l english –h hub4
-r ref_filename.stm hyp_filename1.ctm
```

4 STT Evaluation Condition Requirements

ASpIRE Solvers in the single-mic track are required to submit STT system outputs for the single microphone evaluation set, along with a system description. ASpIRE Solvers in the multi-mic track are required to submit STT system outputs for the multi-mic evaluation set, along with a system description.

5 Evaluation Rules

The following rules apply to all evaluation conditions.

(1) No manual or human interaction with the development-test and evaluation-test audio data is allowed. In general, human-in the loop on evaluations is not allowed. No listening to the audio, no crowdsourcing, etc. are allowed. When in doubt, contact InnoCentive for guidance.

(2) Only datasets provided may be used to train or adapt acoustic models. These datasets may be processed as the Solver wishes, for example with synthetically-generated room responses.

(3) Language models must be developed using only the Fisher training data.

(4) There is one exception to the dataset constraints. Solvers may use proprietary and publically-available speech-activity systems of their choice, trained on any data they choose other than the ASpIRE development, development-test, or evaluation-test data. Please include a description of the approach used on your speech activity detection and the data used for training it in the system description.

(5) Single- and multi-mic sets must only be used for their respective purpose in the Challenge. For example, Solvers are not allowed to use development or evaluation data for the single mic condition in the multi-mic condition, or vice versa.

(6) Single-mic divisions of the development, development-test, and evaluation-test data must not be used for training, adaptation, or other purposes on the multi-mic evaluation. Similarly, multi-mic

[6] N_{Ref} includes all scorable reference tokens (including optionally deletable tokens) and counts the maximum number of tokens. Note that N_{Ref} considers only the reference transcript and is not affected by scorable tokens in the system output transcript, regardless of their type.

[7] For a tutorial introduction to normalized cross-entropy as well as the ideas behind normalized cross-entropy and the information-theoretic idea of entropy, see: http://www.nist.gov/speech/tests/rt/rt2004/fall/docs/NCE.pdf

[8] Edit Distance is the minimum number of edits (insertions, deletions, and substitutions) necessary to convert one string into another. The three kinds of edits are simply counted (in effect, equally weighted). Each edit counts as an error in the WER.

[9] V. I. Levenshtein: "*Binary Codes Capable of Correcting Deletions, Insertions and Reversals*", in Soviet Physics Doklady, Vol. 10, Nr. 8, Feb. 1966, pp. 707 – 710.

[10] http://www.itl.nist.gov/iad/mig//tools/

divisions of the development, development-test, and evaluation-test data must not be used to for training, adaptation, or other purposes on the single-mic evaluation.

6 PUBLICATION OF RESULTS

Final evaluation results will be reported to the participant community once winning solutions have been validated. At that point, a result report will be made available to each Solver, and results can then be shared outside of the ASpIRE community.

Publication of vetted results is encouraged and should be in accordance with the evaluation agreement and the data license.

7 SCHEDULE

Consult the evaluation schedule on website:

https://www.innocentive.com/ar/challenge/9933624

Appendix A: STM File Format Specification

Segment Time Marked (STM) files are space-separated text files that contain reference transcriptions that are used by the speech recognition scoring code. They are derived from hand transcription of the audio data.

There are ten fields per STM line. They are:

Table A.1 STM Field Names

Field 1	2	3	4	5	6
`File`	`chnl`	`name`	`tbeg`	`tend`	`trans`

Field 1: File name (*file*): The waveform file base name (i.e., without path names or extensions).

Field 2: Channel ID (*chnl*): The waveform channel (e.g., "**1**" or "**2**").

Field 3: Speaker Name field (*name*): The name of the speaker. `name` must uniquely specify the speaker within the scope of the `file`. If `name` is not applicable or if no claim is being made as to the identity of the speaker, use `name` = "**<NA>**".

Field 4: Beginning time (*tbeg*): The beginning time of the object, in seconds, measured from the start time of the file. If there is no beginning time, use `tbeg` = "**<NA>**".

Field 5: Ending time (*tend*): The ending time of the object, in seconds, measured from the start time of the file. If there is no ending time, use `tend` = "**<NA>**".

Field 6: Transaction field (*trans*): A whitespace separated list of words.

Appendix B: CTM File Format Specification

Conversation Time Marked (CTM) files are space-separated text files that contain tokens output by the Speech-To-Text system. Each line represents a single token emitted by the system.

There are six fields per CTM line. They are:

Table B.1 CTM Field Names

Field 1	2	3	4	5	6
`File`	`chnl`	`tbeg`	`tdur`	`ortho`	`conf`

Field 1: File name (*file*): The waveform file base name (i.e., without path names or extensions).

Field 2: Channel ID (*chnl*): The waveform channel (e.g., "**1**").

Field 3: Beginning time (*tbeg*): The beginning time of the object, in seconds, measured from the start time of the file.

Field 4: Duration (*tdur*): The duration of the object, in seconds.

Field 5: Orthography field (*ortho*): The orthographic rendering (spelling) of the token.

Field 6: Confidence Score (*conf*): The confidence (probability with a range [0:1]) that the token is correct. If `conf` is not available, omit the column.

Appendix C: An Auxiliary STT Performance Metric

The primary STT measure WER (Section 3.4) estimates performance of all space-separated tokens; however, it is useful to understand system performance using a second metric:

Confidence Score Normalized Cross Entropy

As an additional performance measure, the quality of the token confidence scores will optionally be evaluated. The confidence score represents the system's estimate of the probability that the output token is correct and must have a value between 0 and 1 inclusive.

The performance of this confidence measure will be evaluated using Normalized Cross Entropy (NCE). It is assumed that the role of the confidence score is to distribute the probability mass of a correct recognition (i.e. the percent correct) across all the system transcribed words.

$$NCE = \left\{ H_{\max} + \sum_{w=1}^{CorrectWord} \log_2(\hat{p}(w)) + \sum_{w=1}^{IncorrWord} \log_2(1-\hat{p}(w)) \right\} \Big/ H_{\max}$$

Where:

$H_{\max} = -n\log_2(p_c) - (N-n)\log_2(1-p_c)$

$n =$ the number of correct system words

$N =$ the total number of system words

$p_c = n/N;$ the average prob. that a system word is correct

$\hat{p}(w) =$ the confidence of system word w

Appendix D: System Descriptions and Auxiliary Condition Reporting

System descriptions are expected to be of sufficient detail for a fellow researcher to both understand the approach and the data/computational resources used to train and run the system.

The proposed structure of the system description includes the following six sections:

Section 1: Abstract

Section 2: Notable Highlights

Section 3: Data Resources

Section 4: Description

Section 5: Hardware Description

Section 6: Timing

Each system description section is covered below in a separate sub-section of this appendix. The scoring server will require system descriptions to be included along with the system output. The author should expect the reader is already familiar with the ASpIRE project, this evaluation plan, and speech recognition in general.

Section 1: Abstract (<300 words)

A few paragraphs describing the system at the highest level. This should help orient the reader to the type of system being described and how the components fit together.

Section 2: Notable Highlights (<500 words)

A few paragraphs on the major differences between this system and a "conventional" system. Questions often answered are: How is this system approaching the ASpIRE Challenge? What is unique?

Section 3: Training Resources (as many words as needed)

This section describes how training data resources were used by the system and for which major components.

Section 4: Description (<2500 words for a full system description)

Sufficient detail should be provided for each component of the system such that a practitioner in the field can understand how each phase was implemented. You should be very brief on components that are standard in the field.

For system combinations, there should be a section for each subsystem.

For each subsystem, there should be subsections for each major phase. They may also refer to other subsystems or referent system descriptions if they share components.

Suggested Subsections:

- Signal processing - e.g., enhancement, noise removal, crosstalk detection/removal.
- Low level features - e.g., PLP, Gabor filterbank.
- Speech/Nonspeech – e.g., speech activity detection algorithm and data used
- Learned features – e.g., MLP tandem features, DNN bottleneck features, etc.
- Acoustic Models – e.g., DNN, GMM/HMM, RNN, etc.
- Language Models – e.g., training, data used, etc.
- Adaptation – e.g., speaker, channel, etc. Specify how much of the evaluation data was used as well as the computational costs (memory and time).
- Normalization - Normalizations not covered in other sections
- Lexicon – methods used to update
- Decoding – e.g., Single pass, multipass, contexts, etc.
- OOV handling – e.g., Grapheme, syllable, phoneme, etc.
- System combination methods

- Other methods that are unique to your solution

Section 5: Hardware description

Requirements on the description of architecture will be here. Reporting of the following environment elements relate directly to the reporting of time and memory requirements.

- OS (type, version, 32- vs 64-bit, etc.)
- Total number of used CPUs
- Descriptions of used CPUs (model, speed, number of cores)
- Total number of used GPUs
- Descriptions of used GPUs (model, number of cores, memory)
- Total available RAM
- RAM per CPU
- Used Disk Storage (Temporary & Output)

Section 6: References

This section contains references to research papers and other system descriptions submitted for the evaluation.