

Spring 2004 (RT-04S) Rich Transcription Meeting Recognition Evaluation Plan

1 INTRODUCTION

The goal of this document is to define the evaluation tasks, performance measures, and test corpora to support the 2004 Rich Transcription Spring (RT-04S) Meeting Recognition Evaluation. This evaluation has been offset in time from the RT-04 Fall Broadcast News and Conversational Telephone Speech evaluations to enable researchers to participate in both evaluations. As such, a separate evaluation plan is available for the RT-04F Evaluation. This document (as well as additional documentation and data files pertaining to the RT-04S evaluation) are available from the NIST RT-04S website, <http://nist.gov/speech/tests/rt/rt2004/spring/>.

Rich Transcription (RT) is broadly defined to be a fusion of speech-to-text (STT)¹ technology and metadata extraction technologies which will provide the basis for the generation of more usable transcriptions of human-human speech in meetings for both humans and machines. These evaluations are open to all interested volunteers. Broadly, this evaluation will include the following tasks in the meeting domain:

Speech-to-Text Transcription (STT) tasks

Microphone conditions:

- Multiple distant microphones
- Single distant microphone
- Individual head microphone

Processing time conditions:

- Unlimited time STT
- Less than or equal to twenty times realtime
- Less than or equal to ten times realtime
- Less than or equal to one times realtime

Diarization (SPKR) task (“who spoke when”)

Microphone conditions:

- Multiple distant microphones
- Single distant microphone

Input conditions:

- Speech input only
- Speech plus reference transcript input

Processing time conditions:

- Unlimited time
- Less than or equal to twenty times realtime
- Less than or equal to ten times realtime
- Less than or equal to one time realtime

The RT-04S evaluation will be limited to English language meeting speech only.

1.1 PRIMARY VS. CONTRASTIVE SYSTEMS

Primary systems: Participants must submit output from exactly one *primary* system² for each task they participate in. The primary system must be run on the speech-input condition (see section 7) and can also be run on other conditions³ specified in section 7. Only comparable (same condition) systems will be compared across sites.

Contrastive systems: Participants may submit output from additional *contrastive* systems, for tasks on which they have submitted output from a primary system. But each contrastive system must also be run on the required conditions⁴.

Changes from RT-02

The last meeting recognition evaluation was conducted as part of the Rich Transcription 2002 evaluation series. This section briefly lists the differences between the RT-02 and RT-04S Meeting Recognition Evaluations.

- An all-new test set will be used. However, the data will be from the same data collection sites as the RT-02 test set. The primary difference is that for RT-04S, all of the data for the “personal” mic condition will be from head-mounted noise canceling mics, not a mix of head and lapel mics that were used in RT-02.
- The development test set will use lapel mics for some meetings since the data will be the RT-02 evaluation test set. Unfortunately, an all-new development test set reflecting the evaluation test set could not be collected due to resource constraints.
- The primary condition for both STT and SPKR tasks will be the multiple distant microphone condition rather than the single distant mic condition chosen in RT-02 (although a single distant mic condition will continue to be supported as a contrast). Unlike RT-02, there will be no head/personal mic mix condition for RT-04.
- Certain side information is permitted for the RT-04S evaluation. See Section 8.1.2
- The rules for producing lexical tokens will follow those defined for the RT-03S STT conversational telephone speech task. These are similar to the rules employed for RT-02, but are more clearly defined.
- There will be no required processing speed task. However, sites will be expected to report their processing speed as described in Appendix B.
- An experimental scoring of STT performance during overlapping speech will be performed.

¹ formerly known as automatic speech recognition (ASR)

² That submission is to be designated as primary — see the description of the SYSID string in section 8.3.1.

³ Those submissions will still be *primary*.

⁴ That submission will still be *contrastive* not *primary*.

- Submissions for the speaker diarization evaluation (who spoke when) will be in RTTM (see Appendix A) format, rather than MDTM.
- The diarization task performance will be measured using the RT-03 speaker segmentation metrics – the only difference being that collar times will be employed to compensate for reference time error. This is because no forced alignments currently exist for this data.

2 BACKGROUND

While the traditional STT evaluations have provided a mechanism for evaluating word accuracy, it is clear that words alone are insufficient to formulate a transcription of speech that is maximally useful. A verbatim transcription of the speech stream into a string of lexical tokens yields a transcript that is often difficult to understand. This is because spoken language is much more than just a string of lexical tokens. It contains information about the speaker, prosodic cues to the speaker's intent, and much more. Spoken language also contains disfluencies, which speakers correct and which textual renderings should delete. All of this makes the task of rendering spoken language into text a great challenge, especially with less-than-perfect automatic speech recognition (ASR) performance.

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.⁵ The WER is defined as the sum of all ASR output token errors divided by the number of scoreable 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).⁶

Transcripts with the sorts of metadata called for by the RT-04S evaluations will be easier for humans to read and can be processed in more useful ways by computers. While the RT-4S evaluation does not seek to address all of the elements necessary to create maximally rich transcriptions of speech in meetings, it does address two crucial core technologies: Speech-to-Text Transcription (STT) and Speaker Diarization (SPKR). Future such evaluations may address additional metadata tasks and may make use of multi-media resources. The remainder of this document defines the tasks, metrics, corpora, annotations, input/output specifications, and schedule for the evaluation of these tasks.

⁵ 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, these terms may be treated more or less equivalently.

⁶ 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.

3 THE RT-04S SPEECH TO TEXT (STT) TASKS

STT system output will be evaluated separately from SPKR output. Three microphone type evaluation conditions are supported for the STT:

- **Primary condition: multiple distant microphones (sttmdm).** Sites will be provided all of the distant microphone recordings collected by the data collection sites for each meeting in the test set and may use any combination of them to generate a maximally-accurate transcription of the speech. Any combinations of these mics may be used and knowledge of the mic types and locations may be used to configure the recognition systems. However, the recordings may not be listened to or manually processed prior to, or during the recognition process. The data from these mics must be processed fully automatically.
- **Contrastive condition: Single distant microphone (sttsdm).** This is similar to the sttmdm condition. However, sites will be provided only a single centrally-located distant microphone recording from each meeting in the test set. Sites are encouraged to implement this condition as a contrast to the primary condition to examine the effectiveness of employing multiple distant microphones.
- **Contrastive condition: individual head microphone (sttihm).** Sites will be provided all of the head-mounted mic recordings collected by the data collection sites for each meeting in the test set. In processing this data, sites may also make use of the entire distant mic set. Sites are encouraged to implement this condition as a contrast to the primary condition to examine the degradation caused by distant microphones.

Participants are *strongly* encouraged to implement the sttmdm condition (which is considered primary for this evaluation). However, to make the evaluation as inclusive as possible, we will permit participants who cannot process the multiple distant microphone condition to run their system on the single distant microphone condition (sttsdm) only. All sites must also implement the sttihm close-talking microphone contrast.

3.1 DEFINITION OF THE STT PROCESSING SPEED TASKS

Although sites are permitted to run their systems at any speed they wish, they are required to determine and report their processing speed as defined in Appendix B. Sites must then categorize their systems upon submission under one of the following processing speed categories:

- Unlimited time (sttul) – greater than twenty times realtime
- Less than or equal to twenty times realtime (stt20x)
- Less than or equal to ten times realtime (stt10x)
- Less than or equal to one times realtime (stt1x)

The processing speed is defined as the ratio of the wall-clock Total Processing Time (TPT) divided by the duration of the recorded audio input. TPT is defined (see Appendix B) as the

time it takes to process all channels of the recorded speech⁷ (including ALL I/O) on a single CPU. The TPT does not include time between batch processes or system start-up time (e.g., booting up and loading initial default models into memory). To elaborate, systems that are not completely pipelined should not count time in between batch processes.

The system description for each STT submission should include processing time information, calculated as described in Appendix B.

At a minimum, sites must run **all contrasts at the same approximate processing speed** as their primary system (same hardware and speed-versus-accuracy parameter settings).

3.2 SCOREABLE STT TOKENS

The same scoring conventions will be used as were implemented in the RT-03S evaluation. RT-04S will score lexical tokens and will not score non-lexical speaker sounds (cough, sneeze, breath, lipsmack, and laugh), or non-speech sounds (such as door slams and so forth).

The RT-04S STT evaluation will include only English data. Non-English speech will be considered and treated as “foreign”.

3.2.1 TOKEN STRING FORMATTING

A single standardized spelling is required for scoreable lexemes, and the STT system must output this spelling in order to be scored as correct.⁸ Homophones must be spelled correctly according to the given context in order to be considered correct. All tokens are to be generated 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.

⁷ For example, a 1-hour news broadcast processed in 10 hours is counted as 10 times realtime regardless of whether the broadcast is stereo or monaural. And a 5-minute telephone conversation processed in 50 minutes would also count as 10X realtime, whether the signal is a 4-wire/2-channel signal or a 2-wire/1-channel signal.

⁸ Token spelling is determined by NIST by first consulting an authoritative reference – e.g., the American Heritage Dictionary (AHD) for English. Lacking an authoritative reference, the www 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 previous years, 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 scoreable form.

3.3 STT EVALUATION FRAMEWORK

The STT task is similar to previous ASR “Hub-4” and “Hub-5” evaluations, but with additions to support the classification of output tokens, overlapping speech, and (optionally) speaker assignment. The primary form of scoring will use the same conventions as the RT-03S STT evaluation. A secondary experimental scoring will provide a first examination of performance during periods of overlapping speech. The remainder of this section describes the protocol for the primary metric unless otherwise explicitly stated.

The primary STT performance measure is essentially the same as the traditional NIST ASR WER measure using the NIST SCLITE software. The primary metric for the RT-04S STT evaluation will (as in RT-03S) be calculated over non-overlapping speech (i.e., omitting regions with multiple reference speakers in the same channel speaking simultaneously).⁹

3.3.1 SYSTEM OUTPUT GENERATION

The system output will be a CTM¹⁰ file (see section 8.2.3). A CTM file is token-based and is to include the following information for each recognized token: the name of the source file, the channel processed, the beginning time of the recognized token, the duration of the recognized token, the string representation of the recognized token, a confidence probability, a token type, and a speaker identifier. The speaker information is optional, but is included to support STT/MDE fusion experiments. If no speaker information is generated, a value of “unknown” should be used for lexical token types and “null” for non-lexical token types. See section 8.2.3 for specific formatting requirements. The following describes each possible system output (CTM) token type¹¹:

lex - a lexical token.

frag - a lexical fragment. Note: An optional hyphen may also be used in the token string to indicate the missing (unspoken) part of the token, but the **frag** type must also be used.

fp - a filled pause.

un-lex - an uncertain lexical token. This type tag is normally used only in the reference.

for-lex - a “foreign” lexical token. This type tag is normally used only in the reference.

non-lex - a non-lexical acoustic phenomenon (breath-noise, door-bang, etc.)¹².

misc - other annotations not covered in above.¹³

⁹ Note that anticipated upcoming domains in future evaluations, such as STT transcription of meetings, will include processing of overlapping speech.

¹⁰ The CTM file format is one of the immediate predecessors of the RTTM file format. The CTM and RTTM file formats *differ*.

¹¹ Note that in the RTTM format, some of what are token types in CTM and STM format data are instead subtypes of the RTTM *lexeme* type.

¹² RTTM (the reference data for the MDE evaluations) divides this category into non-speech (non-vocal noises) and non-lex (vocal noises). See Appendix A.

¹³ A system may give this tag to any token which is to be excluded from scoring – including tokens for which the more

Of the token types listed above, all types other than **lex** will be stripped from the system output prior to STT scoring, and in the reference they will be tagged as "optionally deletable". Therefore only tokens tagged as type **lex** in the system output will be aligned and scored, and all others (because stripped out) may be regarded as optional. Although systems aren't penalized (or rewarded) for outputting those optional types, we encourage their output to support metadata experiments.

3.3.2 REFERENCE TOKEN PROCESSING

A Segment Time Marked (STM) scoring reference is generated from the human reference transcripts.¹⁴ Contraction expansions are annotated in the human reference: the annotator will choose (and the STM file will contain) the single most likely expansion for each contraction. Non-scoreable regions (such as untranscribed and overlapping speech areas) are explicitly tagged in the STM file for exclusion from scoring (there will be no scoring UEM file for the STT evaluation). The tokens of the various STM token types¹¹ in the STM reference will be processed as follows:

lex – STM tokens of type **lex** are not specially tagged in the reference. As such, they are aligned and scored.

fp – STM tokens of this pause-filler type are tagged as optionally deletable¹⁵ in the reference. As the first step in scoring them, these tokens in the reference will be replaced by a generic internal **fp** token. Their orthography will be ignored.

frag – STM tokens of type **frag** are tagged in the reference both as optionally deletable and as fragments. They contribute to the WER denominator. Note: In addition, if a system output token of type **lex** aligns with a **frag** in the reference, it is counted as correct if the reference **frag** token string is a substring of the system output token string.¹⁶

un-lex, for-lex – Tokens of these types are tagged as optionally deletable in the reference. They contribute to the WER denominator.

non-lex and **misc** – These token types are removed from the reference

3.3.3 GLM PROCESSING

Prior to scoring, both the reference and system output token strings will be transformed using a global map file (GLM). The GLM is intended to ensure that reference and hypothesis tokens which do not differ semantically are scored as correct. This is accomplished by transforming the token strings in both the reference and system output via a set of mapping rules. The GLM applies a set of rules to the system output which expands contractions to all possible expanded forms.

specific CTM types exist. But where possible, sites are encouraged to use the supported more specific CTM types to enhance the usefulness of the data for MDE experiments.

¹⁴ See [ftp://jaguar.ncsl.nist.gov/current_docs/setk/doc/infmts.htm](http://jaguar.ncsl.nist.gov/current_docs/setk/doc/infmts.htm)

¹⁵ An "optionally deletable" token is a special token in the STT STM reference for which deletion errors are forgiven. For this evaluation, these are all STT CTM tokens of type 'lex'. These tokens also contribute to the count of reference tokens in the word error rate denominator.

¹⁶ But not the other way round. A complete word in the reference will never align to a frag in the system output because all frag's in the system output get stripped out before alignment occurs.

Note that GLM processing may result in the generation of several alternative token strings in the system output. It may also result in token strings being split into two or more strings. For example, contractions are mapped to their expanded form and compound words are split into their constituents. After GLM filtering, hyphens in both the system output and reference are transformed into token separators.

3.3.4 SCORING

Once the pre-processing is complete, token alignment will be performed using a token-mediated alignment optimized for minimum word error rate.

3.4 STT EVALUATION METRICS

An overall STT error score will be computed as the average number of token recognition errors per reference token:

$$Error_{STT} = \frac{(N_{Del} + N_{Ins} + N_{Subst})}{N_{Ref}}$$

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 per the token rules above, and

N_{Ref} = the maximum number of reference tokens¹⁷

As an additional optional performance measure, the confidence of a system in its transcription output will be evaluated. In order to do this, the system must attach a measure of confidence to each of its scoreable output tokens. 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 evaluated using the same normalized cross entropy score that NIST has been using in previous ASR evaluations.¹⁸

3.4.1 EXPERIMENTAL OVERLAPPING SPEECH SCORING

It is clear that meetings contain a large proportion of overlapping speech -- periods of time in which multiple speakers are talking. The existing (primary) scoring protocol necessarily ignores such speech. Unfortunately, therefore, a large portion of difficult-to-recognize speech in meetings has yet to be addressed or evaluated (and was not evaluated in RT-02).

For RT-04S, NIST is developing a protocol for multi-stream scoring of overlapping speech. The approach will map system output tokens from a single concatenated stream into multiple reference speaker streams using a WER minimization algorithm. Note that this approach does not evaluate the ability of STT systems to identify and separate different speaker streams. Whereas an operational system would be expected to fuse speaker diarization and STT capabilities, this evaluation seeks to decouple the tasks. So, this approach will ignore stream assignment errors. A significant advantage to this approach is that it also permits evaluation of system output without explicit

¹⁷ N_{Ref} includes all scoreable reference tokens (including optionally deletable tokens) and counts the maximum number of tokens (e.g., the expanded version of contractions). Note that N_{Ref} considers only the reference transcript and is not affected by tokens in the system output transcript, regardless of their type.

¹⁸ <http://www.nist.gov/speech/tests/rt/rt2003/doc/NCE.htm>

speaker/stream identification (although sites are encouraged to do this as a research task).

Such multi-stream STT scoring is currently a research task. As such, the results of this analysis will be viewed as experimental for RT-04S, but will lay the groundwork for better metrics for future evaluation of speech in meetings. The multi-stream scoring protocol will be described in more detail at a later date and this section will be amended accordingly.

4 DIARIZATION – “WHO SPOKE WHEN”

A transcript where the speakers are labeled, so that the reader can tell who spoke when, is more readily interpreted. This RT-04S metadata extraction task will be like the RT-03S speaker segmentation “who spoke when” evaluation.

Diarization is the process of annotating an input audio channel with information that attributes (possibly overlapping) temporal regions of signal energy to their specific sources. These sources can include particular speakers, music, background noise sources, and other signal source/channel characteristics.

For RT-04S, diarization will be limited to just the speaker segmentation “who spoke when” task. For the “who spoke when” task, small pauses in a speaker’s speech, of less than 0.3 seconds, are not considered to be segmentation breaks. Material containing no pauses of 0.3 seconds or more should be bridged into a single continuous segment. Although somewhat arbitrary, the cutoff value of 0.3 seconds has been determined to be a good approximation of the minimum duration for a pause in speech resulting in an utterance boundary. Systems should consider vocal noise (laugh, cough, sneeze, breath, lipsmack) to be silence in constructing segment boundaries.¹⁹

Although many systems perform the diarization task without transcribing the text, note that systems may make use of the output of a word/token recognizer (or any other form of automatic signal processing) in performing this task. The approach used should be clearly documented in the task system description.

4.1 “WHO SPOKE WHEN” DIARIZATION SCORING

In order to measure performance, an optimum one-to-one mapping of reference speaker IDs to system output speaker IDs will be computed. The measure of optimality will be the aggregation, over all reference speakers, of time that is jointly attributed to both the reference speaker and the (corresponding) system output speaker to which that reference speaker is mapped. This will always be computed over all speech, including regions of overlap²⁰. Mapping is subject to the following restrictions:

- Each reference speaker will map to at most one system output speaker, and each system output speaker will map to at most one reference speaker. If the system performance is perfect, this mapping will be one-to-one.
- Mapping of speakers will be computed separately for each speech data file.

Although the speaker mapping will take regions of overlapping speech into account, for consistency with the STT task, the primary metric will be based on non-overlapping speech only.

¹⁹ However, special scoring rules will apply to areas containing vocal noise. See Section 4.

²⁰ By “overlap” we mean regions where more than one reference speaker is speaking on the same audio channel.

However, performance over overlapping speech will also be reported.

Since segment times for this data will not have been created via a high-accuracy process like forced alignment, 250 millisecond time collars will be employed around each reference segment to forgive timing errors in the reference.

Speaker detection performance will be expressed in terms of the miss and false alarm rates that result from the mapping.

An overall time-based speaker diarization error score will be computed as the fraction of speaker time that is not attributed correctly to a speaker. This will be the **primary metric** for speaker segmentation diarization:

$$Error_{SpkrSeg} = \frac{\sum_{all\ segs} \{dur(seg) \cdot (\max(N_{Ref}(seg), N_{Sys}(seg)) - N_{Correct}(seg))\}}{\sum_{all\ segs} \{dur(seg) \cdot N_{Ref}(seg)\}}$$

where the speech data file is divided into contiguous segments at all speaker change points²¹ and where, for each segment, *seg*:

- $dur(seg)$ = the duration of *seg*,
- $N_{Ref}(seg)$ = the # of reference speakers speaking in *seg*,
- $N_{Sys}(seg)$ = the # of system speakers speaking in *seg*,
- $N_{Correct}(seg)$ = the # of reference speakers speaking in *seg* for whom their matching (mapped) system speakers are also speaking in *seg*.

The numerator of the overall diarization error score represents speaker diarization error time, and it can be decomposed into speaker time that is attributed to the wrong speaker, missed speaker time, and false alarm speaker time.

Speaker time that is attributed to the wrong speaker (called speaker error time) is the sum of the following over all segments:

$$dur(seg) * \{\min(N_{Ref}(seg), N_{Sys}(seg)) - N_{Correct}(seg)\}.$$

Missed speaker time is the sum of the following over only segments where more reference speakers than system speakers are speaking:

$$dur(seg) * (N_{Ref}(seg) - N_{Sys}(seg)).$$

False alarm speaker time is the sum of the following over only segments where more system speakers than reference speakers are speaking:

$$dur(seg) * (N_{Sys}(seg) - N_{Ref}(seg)).$$

No segment is both miss time and false-alarm time.

Word-based counterparts to the time-based speaker diarization error score, and to each of its three parts (speaker error time, missed speaker time, false-alarm speaker time), are also calculated and reported — by using word counts instead of time. These word-based versions count the number of reference words

²¹ A “speaker change point” occurs each time any reference speaker or system speaker starts speaking or stops speaking. Thus, the set of currently-speaking reference speakers and/or system speakers does not change during any segment.

covered by the segment (a word is covered by a segment if the word's midpoint time²² falls in the segment. (midpoint time is the start time of the word plus half its duration).

In areas of overlap (segments where more than one reference speaker is speaking), note that the duration of the segment is attributed to all the reference speakers who are speaking in the segment, thus counting the time more than once. But since the reference data tells us which speaker actually spoke each reference word, we can (and do) attribute each word to its actual speaker, and in areas of overlap this means the words are not counted more than once.

A system may, optionally, attach a measure of confidence to each of its output speaker segments. This confidence measure represents the system's estimate of the probability that the speaker of this segment is correctly assigned.²³ This confidence measure will not, however, be evaluated.

4.2 SPEAKER-WEIGHTED DIARIZATION SCORES

The SpkrSegEval software also calculates a proposed speaker-weighted who-spoke-when diarization-error metric²⁴. This metric will continue to be calculated in order to further explore the behavior of the proposed metric. It is not, however, part of the official metric set for RT-04S.

5 EVALUATION UN-PARTITIONED EVALUATIONS MAPS (UEM)

Un-partitioned evaluation maps (UEMs) are the mechanism the evaluation infrastructure uses to specify time regions within an audio recording. An *input* UEM file will be provided for all tasks (including STT), to indicate what audio data is to be processed by the systems. A *scoring* UEM file will be used to specify the time regions to be scored for the RT-4S diarization task. No scoring UEM files will be used in scoring the STT tasks. Rather, the STM files will be used to score the STT tasks).

5.1 UEM FILE STRUCTURE

The UEM file format is a concatenation of time mark records for a segment of audio in a speech waveform. The records are separated with a newline. Each record must have a file id, channel identifier [1 | 2], begin time, and end time. Each record follows this BNF format:

```
UEM ::= <F><SP><C><SP><BT><SP><ET>
```

where,

<SP> indicates a space (" ").

<F> indicates the file id, consisting of the path, filename, and extension of the waveform to be processed.

<C> indicates the waveform channel, which, for RT-04S, is always "1" since all speech waveform will be provided in separate files.

<BT> indicates the beginning time of the segment measured in seconds from the beginning of the file which is time 0.

<ET> indicates the ending time of the segment measured in seconds from the beginning of the file which is time 0.

For example:

```
audio/dev04s/english/meeting/NIST_20020214-1148_d05_NONE.sph 1 0 291.34
```

```
audio/dev04s/english/meeting/NIST_20020214-1148_d04_NONE.sph 1 0 291.34
```

...

5.2 SYSTEM INPUT UEM FILES

A UEM file is provided with the evaluation data to define the regions of the audio that the system must process. The boundaries specified by the UEM file will include the beginning and end of a meeting excerpt.

5.3 METADATA SCORING UEM FILES

An MDE scoring UEM file is provided with the reference transcripts that defines the scoreable regions of the audio file. In addition to the boundaries specified by the system input UEM, the MDE scoring UEM excludes extended regions of non-transcribed speech.

6 CORPORA RESOURCES

6.1 TRAINING DATA

While any publicly available data can be used for training, NIST has worked with the LDC, CMU and ICSI to put together meeting domain training and development resources for the evaluation. This data²⁵ consists of:

Corpus	Duration	Corpus #	Transcript #
CMU ISL Meeting Corpus	11 hours	LDC2004E04	LDC2004E05
ICSI Meeting Corpus	~72 hours (75 meetings)	LDC2004S02	LDC2004T04
NIST Pilot Meeting Corpus ²⁶	13 hours	LDC2004E01	LDC2004E02

²⁵ These data are currently available to RT-04S evaluation participants only. The LDC will make them available to the general public as they are able.

²⁶ The NIST data has been "quick transcribed" and made available quickly so that it can be used several weeks prior to the evaluation. If possible, it will be re-released at the beginning of the evaluation with additional quality control. See the NIST site RT-04S website, <http://nist.gov/speech/tests/rt/rt2004/spring/> for updates.

²² Midpoint time is the average of the start time and the end time.

²³ The confidence measure represents the confidence in speaker assignment only. It should exclude consideration of the correctness of other attributes such as speaker type and segment times.

²⁴ See message to MACEARS from Greg Sanders on June 24, 2003, which explains the proposed metric in detail.

6.2 DEVELOPMENT TEST DATA

The 80-minute test set used in the RT-02 Meeting Recognition Evaluation is the designated development test data for the RT-04 Meeting Recognition Evaluation. NIST has re-released this data with additional distant mics (if the data collection sites provided them). Although this data is comprised of 10-minute excerpts from the same data collection sites which will be represented in the RT-04 evaluation test set, it is not completely reflective of the evaluation test data since it contains lapel mics in lieu of head mics for the LDC and CMU data and some different distant mics for LDC data. Unfortunately, because of resource constraints, we were unable to provide an entirely new development test set for this evaluation. The transcripts and other files necessary for scoring recognition using this data are available from the RT-04S website, <http://nist.gov/speech/tests/rt/rt2004/spring/>.

7 EVALUATION CONDITIONS

There are many different conditions under which system performance may be evaluated. This section identifies those conditions for which performance will be computed and, of those, which are to be designated as the required evaluation conditions.

The following list of evaluation conditions (with submission codes in italics) apply to all RT-04S Evaluation tasks.

Data sets:

- multiple distant mic recordings (*eval04smdm*)
- single distant mic recordings (*eval04ssdm*)
- individual head-mic recordings (STT only) (*eval04sihm*)

Language:

- English (*eng*)

Domain:

- Meetings (*meeting*)

Input:

Speech-only input (spch): Any desired fully-automatic signal processing approaches may be employed (including the use of a site developed STT system). This is the required evaluation condition for Input for all RT-04S tasks. The “spch” (speech) condition is the primary condition of interest.

Speech plus the reference transcriptions (ref): The function of this evaluation condition (which only applies to the SPKR task) is to serve as a perfect-STT control condition. It is an optional contrast evaluation condition. The system inputs will be RTTM formatted files derived from the reference RTTM files and placed in the ‘input’ directory (described in Section 8.2.1 below) of the evaluation corpus.

Task/Processing Speed:

- STT >20 times realtime -- unlimited processing time (sttul)
- STT <=20 times realtime (stt20x)
- STT <=10 times realtime (stt10x)
- STT <=1 time realtime (stt1x)

- SPKR >20 times realtime -- unlimited processing time (spkrul)
- SPKR <=20 times realtime (spkr20x)

SPKR <=10 times realtime (spkr10x)

SPKR <=1 time realtime (spkr1x)

All participants must agree to completely process all of the data for at least one task (STT or SPKR) using at least one of the distant mic datasets (mdm or sdm). For all tasks, at a minimum, the speech-input-only (spch) processing condition must be implemented.

8 PARTICIPATION INSTRUCTIONS

Participation is encouraged for all those who are interested in one or more of the RT-04S tasks. All participants must, however, agree to completely process all of the data for at least one task and must complete a required condition for that task. This means that, at a minimum, the speech-input-only processing condition must be implemented.

All participating teams are required to submit a primary system on the required task-specific evaluation condition. Each team may only submit one primary system for each task. Any contrastive system submissions must have a corresponding primary system submission.

As a condition of participation, all sites/teams must agree to make their submissions (system output, system description, and ancillary files) available for experimental use by other research sites. Further, submission of system output to NIST constitutes permission on the part of the site/team for NIST to publish scores and analyses for that data including explicit identification of the submitting site/team and system.

8.1 PROCESSING RULES

8.1.1 RULES THAT APPLY TO ALL EVALUATIONS

All developed systems must be fully automatic requiring no manual intervention to influence the system’s decision-making infrastructure when generating the system output. Manual intervention is allowed to shepherd system processes but not to change any parameter settings or processing steps in response to knowledge or intuition gained from processing the evaluation data.²⁷

The only exemption from the automatic processing restriction is for the reference text condition. Participants who use the reference text condition can manually add pronunciations to their dictionaries to enable forced alignment of the out-of-vocabulary items. Participants cannot use the lexical knowledge gained from the reference+speech-input system to modify their speech-input only system.

Systems will be provided with recorded SPHERE formatted waveform files and a UEM file specifying the speech files and regions within them to be processed. Each meeting speech test waveform will be provided in single-channel files, one per meeting excerpt and microphone.

Unlike the RT-03S evaluation, all of the distributed material (entire meeting recordings) may be used for automatic adaptation purposes. Therefore, material outside of the times specified in the UEM test index file may be used for automatic adaptation.

²⁷ For example, after processing one file and before processing the next file, shepherding does not include doing anything to exploit knowledge gained *by the researchers* as a result of processing that file.

However, recognition performance on this material will not be evaluated.

8.1.2 ADDITIONAL RULES FOR PROCESSING MEETING SPEECH

The data collection site, room configuration, microphone types, and microphone configurations can be 'known.' The number of subjects cannot be known a priori for the distant microphone conditions. However, the number of subjects will be permitted knowledge for the individual head microphone STT contrast condition. No other information about the subjects may be known a priori for any condition. NIST will provide the above info if it is available from the data collection sites. The data collection sites must provide this information to NIST prior to the start of the evaluation if they use it themselves in processing the evaluation data.

Participants are allowed to use whatever information can be automatically extracted from entire meetings for any particular test excerpt. However, only **fully automatic** processing of any material in the meetings in the test set is permitted.

8.2 DATA FORMATS

The test data formats and submission formats will be similar to those used in other NIST rich transcription evaluations.

8.2.1 AUDIO DATA AND OTHER CORRESPONDING INPUTS

For practicality, the recorded waveform files to be processed will be distributed on DVD-ROM and the corresponding indices, annotations, and transcripts will be made available via the Web or FTP using an identical directory structure. After the evaluation, system outputs will be released in this structure as well.

Directory	Description
indices/	Index files containing the list of files and times to be processed for particular experiments
audio/	Audio files
input/<EXP-ID>/	ancillary data including reference annotations for various experiments – must be used in accordance with instructions for that experiment
output/<EXP-ID>/	system output submissions – will be made available as received for integration tests
reference/	reference transcripts and annotations for post-evaluation scoring and analyses

Note: EXP-ID specifies a unique identifier for each experiment and is defined in section 8.3.1.

For clarity, the “audio/” and “reference/” directories are subdivided into <DATA>/<LANG>/<TYPE> subdirectories:

where,

<DATA> is either [dev04s|eval04s]

<LANG> is [english]

<TYPE> is [meeting]

The “indices/” directory contains a set of UEM test index files specifying the waveform data to be evaluated for each EXP-ID condition supported in this evaluation as described in 8.3.1 and these files are named <EXP-ID>.uem with the special site code “expt”. Separate UEM files, defined in section 5, will be provided for each experiment for each supported <DATA>, <LANG>, and <TYPE>. Corresponding ancillary data for some control conditions is given in the “input/” directory under subdirectories with the same EXP-ID.

8.2.2 MEETING FILE NAME CONVENTION

Each recorded meeting was assigned a consistent unique identifier. The naming convention uses a simple meeting identifier consisting of the collection site's name (<RECORDING_LOCATION>) and date and time of recording (<RECORDING_TIME>, in 24-hour format) as defined by the following BNF format:

```
<MEETINGID> ::=
<RECORDING_LOCATION>_<RECORDING_TIME>
<RECORDING_TIME> ::= <YYYYMMDD>-<HHMM>
```

where

<RECORDING_LOCATION> is either [CMU | ICSI | LDC | NIST]

Each recorded file pertaining to a given meeting contains a single recorded channel. Filenames are constructed by concatenating the meeting ID with a microphone type identifier along with the original site subject id. The audio file names are thus formatted as follows:

```
<MEETING_FILE> ::=
<MEETINGID>_<MIC_ID>_<SUBJECT_ID>.sph
```

where

<SUBJECT_ID> is the subject identifier as provided by the recording site. For distant microphones, no subject can be associated with the file. We therefore use the “NONE” value in this case.

.sph is the file extension (since all files are SPHERE-encoded).

<MIC_ID> is the microphone identifier defined as follows:

```
<MIC_ID> ::= <MIC_TYPE><MIC_NUM>
```

where

<MIC_TYPE> is the microphone type collapsed into a single character field and is either “h” for a head-worn microphone or “d” for any distant microphone,

<MIC_NUM> is a (0-padded) sequence number uniquely identifying the microphone in this meeting.

Example of a meeting recording name:

```
NIST_20020214-1148_d05_NONE.sph
```

8.2.3 STT OUTPUT FORMAT

The RT-04S STT output format will be the CTM format (.ctm filename extension), as in RT-03S. Each output file is to begin with two special comment lines specifying the experiment run and inputs used. These lines must appear at the beginning of the file and are to be formatted as follows:

The first line may be an optional special comment specifying the experiment ID as defined in section 8.3.1 (EXP-ID) and is of the form:

```
; ; EXP-ID: <EXP-ID>
```

For example,

```
; ;EXP-ID:  
icsi_04_stt10x_eval04_eng_meeting_spch_1
```

If present, this optional special comment line must begin with two semicolons “; ;”. Note that for purposes of scoring, all lines beginning with two semicolons are considered comments and are ignored. Blank lines are also ignored.

The header comments are followed by a list of CTM records. See the list below for the specific supported token types.

The CTM file format is a concatenation of time mark records for each output token in each channel of a waveform. The records are separated with a newline. Each field in a record is delimited with whitespace. Therefore, field values may not include whitespace characters. Each record follows the following BNF format:

```
CTM-RECORD ::= <SOURCE><SP><CHANNEL><SP>  
<BEG-TIME><SP><DURATION><SP><TOKEN><SP>  
<CONF><SP><TYPE><SP><SPEAKER><NEWLINE>
```

where

<SP> is whitespace.

<SOURCE> is the waveform basename (no pathnames or extensions should be included). See Section 8.2.2 for more details on the file basenames.

<CHANNEL> is the waveform channel: “1”, “2”, etc. This value will always be “1” for single-channel files.

<BEG-TIME> is the beginning time of the token. This time is a floating point number, expressed in seconds, measured from the start time of the file.²⁸

<DURATION> is the duration of the token. This time is a floating point number, expressed in seconds.²⁸

<TOKEN> is the orthographic representation of the recognized word/lexeme or acoustic phenomena. For English, this is represented as a string of ASCII characters, but a token in the context of a non-English test might be represented in Unicode or some other special character set. Token strings are case insensitive and may contain only upper or lowercase alphabetic characters, hyphens (-), and apostrophes (’) only. No special characters are to be included in this field to indicate the type of token. Rather, the “TYPE” field is to be used to indicate the token type. Note however that a hyphen may be used for fragments to indicate the missing/unspoken portion of the fragment. However, the “frag” TYPE must still be used.

<CONF> is the confidence score, a floating point number between 0 (no confidence) and 1 (certainty). A value of “NA”

²⁸ A required time accuracy for BEG-TIME and DURATION is not defined, but these times must provide sufficient resolution for the evaluation software to align tags with the proper token in the reference when time-alignment-based scoring is used. This alignment can be problematic in the case of quickly-articulated adjoining words. Therefore, systems should produce time tags with as much resolution as is reasonably possible.

is used (in CTM format data) when no confidence is computed and in the reference data.²⁹

<TYPE> is the token type. The legal values of <TYPE> are “lex”, “frag”, “fp”, “un-lex”, “for-lex”, “non-lex”, “misc”, or “noscore”. See Section 3 for details on generation and scoring rules for each of these types.

lex is a lexical token.

frag is a lexical fragment. Note: A (optional) hyphen may also be used in the token string to indicate the missing (unspoken) part of the token, but the frag TYPE must also be used.

fp is a filled pause.

un-lex is an uncertain lexical token normally used only in the reference.

for-lex is a “foreign” lexical token normally used only in the reference.

non-lex is a non-lexical acoustic phenomenon (breath-noise, door-bang, etc.)

misc is other annotations not covered above.³⁰

noscore is a special tag used only in reference files for scoring to indicate tokens that should not be aligned or scored.

<SPEAKER> is a string identifier for the speaker who uttered the token. This should be “null” for non-speech tokens and “unknown” when the speaker has not been determined. This information is optional for this evaluation

Included below is an example of STT system output:

```
NIST_20020214-1148_d05_NONE 1 11.34 0.2  
YES 0.763 lex 1  
NIST_20020214-1148_d05_NONE 1 12.00 0.34  
YOU 0.384 lex 1  
NIST_20020214-1148_d05_NONE 1 13.30 0.5 C-  
0.806 frag 1  
NIST_20020214-1148_d05_NONE 1 17.50 0.2 AS  
0.537 lex 1
```

8.2.4 SPEAKER DIARIZATION OUTPUT FORMAT

The Rich Transcription Time Mark (RTTM) file format (with “.rttm” filename extension) will be used for both the system output and reference for the SPKR task. A separate RTTM file should be generated for each meeting in the test set.

The RTTM format supports markup of a variety of metadata tasks. However, for RT-04S, only the information required for the SPKR task should be provided. The RTTM file format provides two types of records related to the speakers: **SPKR-INFO** records and **SPKR** records. The **SPKR-INFO** record for a speaker is associated with all the SPKR records for that speaker

²⁹ STT systems are required to compute a confidence for each scoreable token output for this evaluation. The “NA” value may be used only for non-scoreable tokens.

³⁰ Any token which is to be excluded from scoring may be given this tag – including those for which specified types exist. However, where possible, sites are encouraged to use the supported types to enhance the usefulness of the data for MDE experiments.

by means of matching values in their name fields. RT-04S participants in the SPKR task will therefore need to output the following: one **SPKR-INFO** entry per unique speaker in each source file followed by a **SPEAKER** entry for each occurrence of a given speaker in the source file.

Participants in the RT-04S evaluation can run their systems on two distinct microphone conditions: multiple distant mics (`eval04smdm`) and single distant mic (`eval04ssdm`). The source file name (**file** field in the RTTM records) to be used is the name of the microphone recording file for the `eval04ssdm` condition. For the `eval04smdm` condition, however, a special **file** field value is to be used: a fictitious name is created by using the meeting identifier and using the “d*” microphone identifier and `NONE` for the subject id (e.g. `NIST_20020214-1148_d*_NONE.sph`). See Section 8.2.2 for more details on the meeting ids and file names. See below for examples.

The **SPKR-INFO** records (one per speaker) associate the speaker type (**adult_male**, **adult_female**, **child**, or **unknown**) in the **stype** field, with the speaker's name in the **name** field. There is only one **SPKR-INFO** record per speaker. The **SPKR-INFO** records are typically all put at the beginning of the RTTM file since they have no associated timestamp. For RT-04S, the speaker type will not be evaluated so there is no need for participants to provide a value other than “**unknown**” for the **stype** field.

The **SPKR** records give information about when a speaker is speaking. Each time the speaker (identified by its **name**) starts speaking, there is a **SPKR** record that states the time when the speaker began speaking (**tbeg**) and how long the speaker spoke (**tdur**).

See Appendix A for a description of the RTTM format.

Example for the `eval04ssdm` condition on a recording named `NIST_20020214-1148_d05_NONE.sph`:

```
SPKR-INFO NIST_20020214-1148_d05_NONE.sph 1
<NA> <NA> <NA> unknown speaker2 0.5

SPKR-INFO NIST_20020214-1148_d05_NONE.sph 1
<NA> <NA> <NA> unknown speaker5 <NA>

.

.

SPEAKER NIST_20020214-1148_d05_NONE.sph 1
4.57 8.70 <NA> <NA> speaker2 0.63

SPEAKER NIST_20020214-1148_d05_NONE.sph 1
8.54 9.03 <NA> <NA> speaker5 0.33
```

Example for the `eval04smdm` condition:

```
SPKR-INFO NIST_20020214-1148_d*_NONE.sph 1
<NA> <NA> <NA> unknown speaker3 <NA>

SPKR-INFO NIST_20020214-1148_d*_NONE.sph 1
<NA> <NA> <NA> unknown speaker8 <NA>

.

.

SPEAKER NIST_20020214-1148_d*_NONE.sph 1 0
3.70 <NA> <NA> speaker3 <NA>

SPEAKER NIST_20020214-1148_d*_NONE.sph 1
10.23 19.47 <NA> <NA> speaker8 0.86
```

8.2.5 SYSTEM DESCRIPTION

For each test run (for each unique `EXP-ID`), a description of the system (algorithms, data, configuration) used to produce the system output must be provided along with your system output. If multiple system runs are submitted for a particular experiment with different systems/configurations, explicitly designate one run as the primary system and the others as contrastive systems in the system description (as well as in the `SYSID` string in the submission filename). The system description information is to be provided in a file named:

`<EXP-ID>.txt`

(where `EXP-ID` is defined in Section 8.3.1)

and placed in the “output” directory alongside the similarly-named directories containing your system output. This file is to be formatted as follows:

1. `EXP-ID = <EXP-ID>`
2. Primary: `yes | no`
3. System Description:
 - [brief technical description of your system; if a contrastive test, contrast with primary system description]*
4. Training:
 - [list of resources used for training; for STT, be sure to address acoustic and LM training, and lexicon]*
5. References:
 - [any pertinent references]*

8.3 SUBMISSION INSTRUCTIONS

8.3.1 SUBMISSION EXPERIMENT CODES

The output of each submitted experiment must be identified by the following code as specified above.

```
EXP-ID ::=
<SITE>_<YEAR>_<TASK>_<DATA>_<LANG>_
<TYPE>_<COND >_<SYSID>_<RUN>
```

where,

```
SITE ::= expt | cmu | columbia | icsi |
sri | virage | isl | mitll | lia | uw |
panasonic | mqu | ...
```

(The special `SITE` code “`expt`” is used in the `EXP-ID`-based filename of the UEM test index files under the “indices/” directory to list the test material for a particular experiment and in the `EXP-ID`-based subdirectory name under the “input/” directory to indicate ancillary data to be used in certain control condition experiments.)

```
YEAR ::= 04s
```

```
TASK ::= sttul | stt20x | stt10x | sttlx
| spkrul | spkr20x | spkr10x | spkrlx |
data
```

`data` = a special `TASK` code used to provide a directory for ancillary data such as common CTM files used over many MDE experiments. Please make sure to use increasing run numbers for this special experiment ID when making multiple submissions so that your ancillary data from earlier submissions is not over-written here at NIST

```
DATA ::= eval04sihm | eval04smdm |
eval04ssdm

LANG ::= eng

TYPE ::= meeting

CONDITION ::= spch | ref

SYSID ::= site-named string designating the system used
```

The `SYSID` string must be present. It is to begin with `p-` for a primary system or with `c-` for any contrastive systems. For example, this string could be `p-wonderful` or `c-amazing`.

This field is intended to differentiate between contrastive runs for the same condition. Therefore, a different `SYSID` should be created for runs where any manual changes were made to a particular system.

```
RUN ::= 1..n (with values greater than 1
indicating multiple runs of the same experiment/system)
```

An incremental run number *must* be used for multiple submissions of any particular experiment with an identical configuration (due to a bug or runtime problem.) This should *not* be used to indicate contrastive runs. Instead, a different `SYSID` should be used. However, please note that *only* the first run will be considered "official" and be scored by NIST unless special arrangements are made with NIST.

Please also note that submissions which reuse identical experiment IDs/run numbers from previous submissions will be automatically rejected.

Example submission strings:

```
cmu_04s_spkr_eval04smdm_eng_meeting_ref_p-
spkrsys_1

sri_04s_sttlx_eval04ssdm_eng_meeting_spch_
c-sitttest3_1
```

8.3.2 SUBMISSION DIRECTORY STRUCTURE

All system output submissions must be formatted according to the following directory structure:

```
output/<SYSTEM-DESCRIPTION-FILES>
output/<EXP-ID>/ <OUTPUT-FILES>
```

where,

<SYSTEM-DESCRIPTION-FILES> one per <EXP-ID> as specified in 8.2.5

<EXP-ID> is as defined in Section 8.3.1

<OUTPUT-FILES> are as in sections 8.2.3, section 8.2.4, and section 8.2.5.

Note: one output file must be generated for EACH input file as specified in the test index for the experiment being run.

The output files are to be named so as to be identical to the input file basenames with the appropriate `.ctm` or `.rttm` filetype extension. For example, an STT output file for the speech waveform file `NIST_20020214-1148_d05_NONE.sph` must be named `NIST_20020214-1148_d05_NONE.ctm` and a SPKR output file must be named `NIST_20020214-1148_d05_NONE.rttm`.

When generated, these output files are to be placed under the appropriately-named `EXP-ID` directory on your system identifying the experiment run.

8.3.3 SUBMISSION PACKAGING AND UPLOADING

To prepare your submission, first create the previously- described file/directory structure. This structure may contain the output of multiple experiments, although you are free to submit one experiment at a time if you like. The following instructions assume that you are using the UNIX operating system. If you do not have access to UNIX utilities or ftp, please contact NIST to make alternate arrangements.

First change directory to the parent directory of your "output/" directory. Next, type the following command:

```
tar -cvf - ./output | gzip > <SITE>_<SUB-
NUM>.tgz
```

where,

<SITE> is the ID for your site as given in section 8.3.1

<SUB-NUM> is an integer 1 – n, where 1 identifies your first submission, 2 your second, and so forth.

This command creates a single tar file containing all of your results. Next, ftp to `jaguar.ncsl.nist.gov` giving the username 'anonymous' and your e-mail address as the password. After you are logged in, issue the following set of commands, (the prompt will be 'ftp>'):

```
ftp> cd incoming
ftp> binary
ftp> put <SITE>_<SUB-NUM>.tgz
ftp> quit
```

You've now submitted your recognition results to NIST. Note that because the "incoming" ftp directory (where you just ftp'd your submission) is write protected, you will not be able to overwrite any existing file by the same name (you will get an error message if you try) and you will not be able to list the incoming directory (i.e., with the "ls" or "dir" commands). So, pay attention to whether you get any error messages from the ftp process when you execute the ftp commands stated above.

The last thing you need to do is send an e-mail message to Chris Laprun at `chris.laprun@nist.gov` to notify NIST of your submission. The following information should be included in your email:

The name of your submission file

A listing of each of your submitted experiment IDs e.g.:

```
Submission: cmu_1 <NL>
Experiments: <NL>
cmu_04s_spkr_eval04smdm_eng_meeting_ref_p-
spkrsys_1 <NL>
cmu_04s_spkr_eval04smdm_eng_meeting_ref_c-
spkrstest_1 <NL>
```

Please submit your files in time for us to deal with any transmission/formatting problems that might occur — well before the due date if possible.

Note that submissions received after the stated due dates for any reason will be marked late.

9 SCHEDULE

Event	Date
Training Data available	February 2
Evaluation Spec available	February 18
Abstracts for non-evaluation papers due	March 1
Notification of acceptance of non-evaluation papers	March 15
Commitment to participate in evaluation	March 1
Evaluation begins	March 8
Evaluation system output due	March 22
Scored results available	March 26
Non-evaluation papers due	April 19
Evaluation papers due	April 27
Workshop (at ICASSP)	May 17

Please note that the stated dates are hard deadlines. Late submissions will be marked as such and given the tight schedule, severely late submissions may not be able to be scored prior to the workshop.

10 UPDATES

Updates, errata and ancillary files can also be found on the evaluation website at:

<http://www.nist.gov/speech/tests/rt/rt2004/spring/>

11 WORKSHOP

This evaluation will be discussed at the ICASSP 2004 Meeting Recognition Workshop in Montreal on May 17. Participation in this workshop will be by invitation only. Evaluation participants will have an automatic slot at the workshop and will be expected to contribute a workshop paper and presentation.

See the ICASSP 2004 Meeting Recognition Workshop page (http://www.nist.gov/speech/test_beds/mr_proj/workshops.html#icassp04) for more details.

Information regarding workshop logistics and registration will be distributed in email.

Appendix A: RTTM File Format Specification

We have renamed **propersname** to **propernoun** and renamed **lip-smack** to **lipsmack**, to correspond to actual practice and actual reference data. There are four general object categories to be represented. They are STT objects, MDE objects, source (speaker) objects, and structural objects.³¹ Each of these general categories may be represented by one or more types and subtypes, as shown in table 1.

Table 1 Rich Text object types and subtypes

Type	Subtypes
Structural types:	
SEGMENT	eval , or (none)
NOSCORE	(none)
NO_RT_METADATA	(none)
STT types:	
LEXEME	lex , fp , frag , un-lex ³² , for-lex , alpha ³³ , acronym ³³ , interjection ³³ , propernoun33 , and other
NON-LEX	laugh , breath , lipsmack , cough , sneeze , and other
NON-SPEECH	noise , music , and other
MDE types:	
FILLER	filled_pause , discourse_marker , explicit_editing_term , and other
EDIT	repetition , restart , revision , simple , complex , and other
IP	edit , filler , edit&filler , and other
SU	statement , backchannel , question , incomplete , unannotated , and other
CB	coordinating , clausal , and other
A/P	(none)
SPEAKER	(none)
Source information:	
SPKR-INFO	adult_male , adult_female , child , and unknown

The STT, MDE and Source information objects are potential research target. And, except for the static speaker information object [**SPKR-INFO**], each object exhibits a temporal extent with a beginning time and a duration. (The duration of interruption points [**IP**] and clausal boundaries [**CB**] is zero by definition.)

These objects are represented individually, one object per record, using a flat record format with object attributes stored in white-space separated fields. The format is shown in table 2.

³¹ Structural objects are important because they are produced by LDC to provide a modicum of temporal organization in the annotation and identify non-evaluable regions.

³² Un-lex tags lexemes whose identity is uncertain and is also used to tag words that are infected with or affected by laughter.

³³ This subtype is an optional addition to the previous set of lexeme subtypes which is provided to supplement the interpretation of some lexemes.

Table 2 Object record format for EARS objects

Field 1	2	3	4	5	6	7	8	9
type	file	chnl	tbeg	tdur	ortho	stype	name	conf

where

file is the waveform file base name (i.e., without path names or extensions).

chnl is the waveform channel (e.g., “1” or “2”).

tbeg is the beginning time of the object, in seconds, measured from the start time of the file.³⁴ If there is no beginning time, use tbeg = “<NA>”.

tdur is the duration of the object, in seconds.⁴ If there is no duration, use tdur = “<NA>”.

stype is the subtype of the object. If there is no subtype, use stype = “<NA>”.

ortho is the orthographic rendering (spelling) of the object for STT object types. If there is no orthographic representation, use ortho = “<NA>”.

name is 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>”.

conf is the confidence (probability) that the object information is correct. If conf is not available, use conf = “<NA>”.

This format, when specialized for the various object types, results in the different field patterns shown in table 3.

Table 3 Format specialization for specific object types

Field 1	2	3	4	5	6	7	8	9
<i>Type</i>	<i>File</i>	<i>chnl</i>	<i>tbeg</i>	<i>tdur</i>	<i>ortho</i>	<i>stype</i>	<i>name</i>	<i>conf</i>
SEGMENT	File	chnl	tbeg	tdur	<NA>	eval or <NA>	name or <NA>	conf or <NA>
NOSCORE	File	chnl	tbeg	tdur	<NA>	<NA>	<NA>	<NA>
NO_RT_METADATA	File	chnl	tbeg	tdur	<NA>	<NA>	<NA>	<NA>
LEXEME NON-LEX	File	chnl	tbeg	tdur	ortho or <NA>	stype	name	conf or <NA>
NON-SPEECH	File	chnl	tbeg	tdur	<NA>	stype	<NA>	conf or <NA>
FILLER EDIT SU	File	chnl	tbeg	tdur	<NA>	stype	name	conf or <NA>
IP CB	File	chnl	tbeg	<NA>	<NA>	stype	name	conf or <NA>
A/P SPEAKER	File	chnl	tbeg	tdur	<NA>	<NA>	name	conf or <NA>
SPKR-INFO	File	chnl	<NA>	<NA>	<NA>	stype	name	conf or <NA>

³⁴ If tbeg and tdur are “fake” times that serve only to synchronize events in time and that do not represent actual times, then these times should be tagged with a trailing asterisk (e.g., tbeg = **12.34*** rather than **12.34**).

Appendix B: Processing Time Calculation for System Descriptions

1. CTS Echo Cancellation

To keep the playing field level, you need not count echo cancellation in your realtime calculation. If you run it during recognition processing, the "official" realtime calculation you report should be (your total processing time, minus your echo cancellation processing time) divided by the recording duration.

2. RT-03S Processing Speed Computation — Total Processing Time (TPT):

For this and future RT evaluations, the time to be reported is the Total Processing Time (TPT) that it takes to process all channels of the recorded speech (including ALL I/O) on a single CPU.

TPT represents the time a system would take to process the recorded audio input and produce lexical token output as measured by a stopwatch.

So that research systems that aren't completely pipelined aren't penalized, the "stopwatch" may be stopped between (batch) processes.

Note that TPT should exclude time to implement CTS echo cancellation. This is so that sites using the Mississippi State Echo Cancellation Software, which was not optimized for speed or integration, are not penalized.

TPT may also exclude time to "warm up" the system prior to loading the test recordings (e.g., loading models into memory.)

Source Signal Duration (SSD):

In order to calculate the realtime factor, the duration of the source signal recording must be determined. The source signal duration (SSD) is the actual recording time for the audio used in the experiment as specified in the experiment's UEM files. This time is channel-independent and should be calculated across all channels for multi-channel recordings.

Speed Factor (SF) Computation:

The speed factor (SF) (also known as "X" and "times-realtime") is calculated as follows:

$$SF = TPT/SSD$$

For example, a 1-hour news broadcast processed in 10 hours would have a SF of 10 (regardless of whether the broadcast is stereo or monaural). And a 5-minute telephone conversation processed in 50 minutes would also have an SF of 10 (regardless of whether the signal is a 4-wire/2-channel signal or a 2-wire/1-channel signal).

Reporting Your Processing Speed Information:

Although we encourage you to break out your processing time components into as much detail as you like, you should minimally report the above information in the system description for each of your submitted experiments in the form:

TPT = <FLOAT>

SSD = <FLOAT>

SF = <FLOAT>