The 2009 (RT-09) 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 2009 Rich Transcription (RT-09) Meeting Recognition Evaluation. This document (as well as additional documentation and data files pertaining to the RT-09 evaluation) are available from the NIST RT-09 website, http://nist.gov/speech/tests/rt/2009.

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 (STT) – convert spoken words into streams of text,
- Speaker Diarization (SPKR) – find the segments of time within a meeting in which each meeting participant is talking.
- Speaker Attributed Speech-To-Text – convert spoken words into streams of text with the speaker indicated for each word.

The RT-09 evaluation will be limited to English language meeting speech only.

1.1 MEETING TYPES: "CONFERENCE ROOM" MEETING SUB DOMAIN

This evaluation will a single type of meeting recordings: "conference room" meetings. The previously used "lecture room" sub domain will not be part of the evaluation.

1.2 PRIMARY VS. CONTRASTIVE SYSTEMS

Primary systems: Participants must submit output from exactly one *primary* system[2] for each task they participate in. The primary system must be run on the audio-input condition (see section 11) and can also be run on other conditions[3] specified in section 11. 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 conditions[4].

1.3 CHANGES FROM RT-07

The last meeting recognition evaluation was RT-07. This section briefly lists the differences between the RT-07 and RT-09 Meeting Recognition Evaluations.

- An all-new conference data test set will be used.
- There will be no Lecture test set.
- There will be a video and audio+video input condition defined for all tasks.
- Definition of average latency secondary metric.

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 evaluations will be easier for humans to read and can be processed in more useful ways by computers. While the RT-09 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), as well as a combined technology Speaker Attributed Speech-To-Text (SASTT). Future such evaluations may address additional metadata tasks and may make use of multi-media resources.

In previous RT evaluations, the research community has treated system runtime as a primary evaluation condition given the

[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 12.3.1.

[3] Those submissions will still be *primary*.

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

[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.

realization that reduced runtime systems often require different architectures. For the first time, the RT evaluations will address another factor of deployment performance 'sample processing latency' which is a measurement of the signal "look-ahead" time before emitting a system output. For the RT-09 evaluation, the effort will be focused on measuring system latency as a secondary measure and not a measure of performance.

The remainder of this document defines the tasks, metrics, corpora, annotations, input/output specifications, and schedule for the evaluation of these tasks.

3 RT-09 MEETING DOMAIN

The RT-09 evaluation will focus on a single Meeting Domain: the "conference room" sub domain, "**confmtg**". The confmtg data will be 180 minutes of data sampled ten meetings collected at two different sites: AMI and NIST. Excerpts will be selected from each meeting.

4 RT-09 SENSOR INPUT CONDITIONS

The RT-09 Evaluation has three classes of sensor input conditions: audio, video and audio+video. Section 11.1.5 explains each of these audio input conditions in detail. The audio conditions for the RT-09 evaluation are:

- Multiple distant microphones
- Single distant microphone
- Individual head microphone
- All Distant Microphones
- Multiple Mark III microphone arrays.

5 THE RT-09 SPEECH TO TEXT (STT) TASK

STT system output will be evaluated separately from SPKR output. Systems will output a word stream of lexical tokens with time locations within the recording, confidence scores and lexical type information. See the Evaluation Task/Evaluation Condition matrix for the definition of required and optional evaluation conditions in Section 11.

5.1 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 D. In order to simplify the evaluation, there will not be specific evaluation conditions for runtime speed thresholds. The only specified runtime speed evaluation condition is unlimited runtime (**sttul**).

5.2 SCOREABLE STT TOKENS

The same scoring conventions will be used as were implemented in the RT-03S, RT-04S, RT-05S, RT-06S, and RT-07 evaluations. RT-09 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-09 STT evaluation will include only English data. Non-English speech will be considered and treated as "foreign".

5.2.1 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.[7] 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.

5.3 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 scoring will use the same system output and lexical processing conventions as used for the RT-07 evaluation. The Word Error Rate (WER) will continue to be the primary evaluation metric with overlapping speech being included as testable material. The NIST Scoring Toolkit (SCTK) will be used to calculate the performance of systems. The remainder of this section describes the protocol for the primary metric unless otherwise explicitly stated.

5.3.1 SYSTEM OUTPUT GENERATION

The system output will be a CTM[8] file (see Appendix B). 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 Appendix B specific formatting requirements. The following describes each possible system output (CTM) token type[9]:

lex - a lexical token.

[7] 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.

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

[9] 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.

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.)[10].

misc - other annotations not covered in above.[11]

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.

5.3.2 Reference token processing

A Segment Time Marked (STM) scoring reference is generated from the human reference transcripts.[12] 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 types[9] 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 deletable[13] 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.[14]

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

5.3.3 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.

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.

5.3.4 Scoring

Once the pre-processing is complete, token alignment will be performed using a token-mediated alignment optimized for minimum word error rate. The primary metric will be all speech with 4 or less simultaneous speakers. The simultaneous speech alignment algorithm is explained in section 2.5 of “The Rich Transcription 2005 Spring Meeting Recognition Evaluation”[15].

The NIST Scoring Toolkit (SCTK)[16] version 2.1.1 or greater contains the necessary tools for scoring an STT system including the simultaneous speech. Once the tools are compiled and installed, the following command will perform the correct scoring.

```
hubscr.pl –o4 –a –h rt-stt –g <GLMFILE> -r <REFSTM> –l english <SYSCTM>
```

Note that the –o option controls the overlap factor to align, i.e., the number of simultaneous speakers.

5.4 STT Evaluation metrics

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

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

where

N_{Del} = the number of unmapped reference tokens,

N_{Ins} = the number of unmapped STT output to kens,

[10] 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.

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

[12] See ftp://jaguar.ncsl.nist.gov/current_docs/sctk/doc/infmts.htm

[13] 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 not of type 'lex'. These tokens also contribute to the count of reference tokens in the word error rate denominator.

[14] 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.

[15] http://www.nist.gov/speech/publications/papersrc/rt05sresults.pdf

[16] The latest version is available from the URL http://www.nist.gov/speech/tools/index.htm

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 tokens[17]

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]

6 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-09 metadata extraction task will be like the RT-03S, RT-05S, RT-06S, and RT-07 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 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]

The segment times used to distinguish speech activity from background noise will be constructed by force aligning the reference transcript with LIMSI tools to the head microphone signals and applying the 0.3 seconds smoothing parameter to the word segmentations. A forgivness collar of 0.25 seconds (both + and -) will not be scored around each boundary. This accounts for both the inconsistent annotation of segment times by humans and the philosophical argument of when speech begins for word-initial stop consonants.

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.

See the Evaluation Task/Evaluation Condition matrix for the definition of required and optional evaluation conditions in Section 11.

6.1 "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 overlap[20]. 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.
- Mapping of speakers will be computed separately for each speech data file.

Like the STT task, the primary metric for speaker detection systems will include all speech including overlapping speech.

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:

$$Error_{SpkrSeg} = \frac{\sum\limits_{\substack{\text{all}\\\text{segs}}} \{\mathrm{dur}(seg) \cdot (\max(\mathrm{N}_{\mathrm{Ref}}(seg), \mathrm{N}_{\mathrm{Sys}}(seg)) - \mathrm{N}_{\mathrm{Correct}}(seg))\}}{\sum\limits_{\substack{\text{all}\\\text{segs}}} \{\mathrm{dur}(seg) \cdot \mathrm{N}_{\mathrm{Ref}}(seg)\}}$$

where the speech data file is divided into contiguous segments at all speaker change points[21] and where, for each segment, *seg*:

$$\begin{aligned}
\mathrm{dur}(seg) &= \text{the duration of } seg, \\
\mathrm{N}_{\mathrm{Ref}}(seg) &= \text{the \# of reference speakers speaking in } seg, \\
\mathrm{N}_{\mathrm{Sys}}(seg) &= \text{the \# of system speakers speaking in } seg, \\
\mathrm{N}_{\mathrm{Correct}}(seg) &= \text{the \# of reference speakers speaking in } seg \\
&\quad \text{for whom their matching (mapped) system} \\
&\quad \text{speakers are also speaking in } seg.
\end{aligned}$$

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:

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

[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

[19] However, special scoring rules will apply to areas containing vocal noise. See Section 6.

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

[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.

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

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

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

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

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

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 time 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.[22] This confidence measure will not, however, be evaluated.

6.2 SPEAKER-WEIGHTED DIARIZATION SCORES

The SpkrSegEval software also calculates a proposed speaker-weighted who-spoke-when diarization-error metric[23]. 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-06S.

6.3 SPEAKER DIARIZATION SYS TEM OUTPUT FILES

The RTTM format will be used for speaker diarization system output and reference files. See Appendix A for the format definition of RTTM files.

6.4 SPEAKER DIARIZATION TOOL USAGE

The RT-06S Speaker Diarization evaluation will use the md-eval version 18 software. The command line will be:

```
md-eval-18.pl -afc -c 0.25 -u <UEM> -r <SPKR_REFERENCE>.rttm -s <SYSTEM>.rttm
```

7 SPEAKER ATTRIBUTED SPEECH-TO-TEXT

The Speaker Attributed Speech-To-Text (SASTT) task is a joint technology development task that combines both Diarization "Who Spoke When" an Speech-To-Text technologies into a single, jointly optimized task. The goal of an SASTT system is to not only correctly transcribe the words spoken but also correctly identify the generically labeled speaker of the word.

7.1 SASTT SCORING

SASTT systems will be scored using a variety of methods since the systems are the joint combination of SPKR and STT systems. The primary metric, Speaker Attributed Word Error (SAWER) will be in line with the joint task and therefore be a modified version of the standard Word Error metric where there is an additional condition to the word being correct; namely the aligned word must be identified as being produced by the *correct* speaker. A correct speaker is determined using the speaker mapping determined during the Diarization Error computation.

An SASTT system will be scored using a multi-pass procedure:

- Pass 1: The Diarization Error is computed to find an optimal mapping between reference speakers and system speakers,
- Pass 2: System and reference transcriptions are aligned using a modified version of the multi-stream alignment as used for STT systems.
- Pass 3: Speaker Attributed Word Error is calculated from the alignments.

7.1.1 DIARIZATION ERROR COMPUTATION

Diarization Error will be computed in the same manner as SPKR systems. Along with the DER, which will be reported as a secondary performance statistic, the evaluation tool outputs a reference speaker-to-system speaker one-to-one mapping list. The mapping, specified below, will be used during the word alignment process to minimize the computed SAWER.

$$RSpkr(r) \Leftrightarrow SSpkr(s)$$

Where:

$RSpkr(r)$ = The r^{th} reference speaker

$SSpkr(s)$ = The s^{th} system speaker

Note that not all reference speakers will be mapped to a system speaker and not all system speakers will be mapped to a reference speaker.

7.1.2 WORD ALIGNMENTS FOR SASTT

Word alignment between the system and reference transcript will be performed using the same alignment engine as the STT task but the additional constraint of correct speaker attribution will be taken into account. Thus, for a system/reference word pair to be aligned as correct, the system word's speaker attribute and the reference word's speaker attribute must have been identified as equivalent during the diarization error calculation pass.

7.1.3 SPEAKER ATTRIBUTED WORD ERROR

Speaker attributed word error is the sum of 4 error types divided by the number of reference words:

$$Error_{SASTT} = \left(N_{\mathrm{Del}} + N_{\mathrm{Ins}} + N_{\mathrm{Subst}} + N_{\mathrm{SER}}\right) / N_{\mathrm{Ref}}$$

Where

N_{Del} = the number of unmapped reference tokens,

N_{Ins} = the number of unmapped STT output to kens,

N_{Subst} = the number of mapped STT output tokens with non-matching reference spelling per the token rules in Section 5,

N_{SER} = the number of mapped STT output tokens with matching reference spelling per the token rules in Section 5 but with non-matching speaker attribution,

[22] 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.

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

N_{Ref} = the maximum number of reference tokens

7.2 SASTT System Output files

SASTT systems will generate an RTTM (Appendix A) files with SPKR-INFO, SPEAKER, and LEXEME objects. The SPKR-INFO and SPEAKER objects must be generated identically to the SPKR task. The LEXEME objects will be identified as being produced by a speaker using the LEXEME's "name" attribute.

7.3 SASTT Evaluation Tool Usage

The NIST Scoring Toolkit (SCTK)[24] version 2.3.2 or greater contains the necessary tools for scoring an STT system including the simultaneous speech. Once the tools are compiled and installed, the following command will perform the correct scoring.

```
hubscr.pl –o4 –a –h sastt –f rttm –g <GLMFILE> -r <REFSTM> –l english <SYSRTTM>
```

Note that the –o option controls the overlap factor to align, i.e., the number of simultaneous speakers. The '-f rttm' option specifies the format of the system as an RTTM file which is required for SASTT scoring.

7.4 Additional SASTT Scoring

In addition to the Speaker Attributed Word Error, NIST will report the Word Error Rate of the system using the same procedure used for the STT systems. The difference between SAWER and WER is that for WER, speaker information is not taken into account during alignments for WER and therefore WER will be minimized.

8 Secondary metrics

Processing speed and latency are two fundamental parameters of system operation that need to be taken into account when deploying technologies. For the RT-09 evaluation the definitions are:

- **Processing speed** is the total number of processor seconds used per second of sensor data. See Appendix D for a description of the measure,
- **Sample Processing Latency** is maximum amount of sensor data an algorithm can process prior to emitting its final output for that sample. The time period is referred to as the "signal look ahead time" (`slat`). See Appendix E for the measures.

While the two properties are inter-dependent in deployed system, the RT evaluation separates them as two quantifiable measurements so that experiments can be conducted focusing on each system behavior separately or simultaneously. NIST will release a tool with the SCTK package that computes Average Sample Processing Latency for all of the evaluation tasks.

9 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-09 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.

9.1 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-06S, 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

...
```

9.2 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.

9.3 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. For the RT-09 evaluation, the scoring UEM will be the system input UEM.

10 Corpora Resources

10.1 Training Data

While any publicly available data can be used for training, NIST has worked with the community to put together meeting domain training and development resources for the evaluation. See Appendix C for details.

11 Evaluation Conditions

There are many different conditions under which system performance may be evaluated. This section describes the conditions and links them to the submission code protocol (in **bold**). This list serves as a dictionary of data conditions and

[24] The latest version is available from the URL http://www.nist.gov/speech/tools/index.htm

Table 1 identifies the required[25] conditions for each task. Section 12 makes use of these conditions to specify how system submissions are to packaged and sent to NIST.

11.1 Evaluation conditions

11.1.1 Evaluation task and speeds:

There are three evaluation tasks and a single runtime speed threshold for all of the RT-09 tasks. Although the community general agrees that runtime speeds have a great impact on system design and effectiveness, specifying multiple runtime speeds greatly proliferates the number of supported evaluation conditions and greatly reduces the amount of comparable inter-system comparisons. For that reason, only the unlimited runtime speed condition will be specified for each of the evaluation tasks. Participants should still document their system's runtime factor in the system description.

The supported tasks and runtime speeds are as follows. No tasks are required.

- Unlimited runtime Speech-to-Text: (**sttul**)
- Unlimited runtime Speaker Diarization: (**spkrul**)
- Unlimited runtime Speaker Attributed Speech-To-Text: (**sasttul**)

11.1.2 Evaluation Data

The RT-09 evaluation corpus is the only corpus used in this evaluation. The experiment code element <DATA> is "**eval09**" for this data set.

11.1.3 Languages

The RT-09 evaluation will consist of English recordings only. The experiment code element "<LANG>" will be "**eng**"

11.1.4 Evaluation data Type

The RT-09 evaluation corpus includes only "conference room". The experiment code element <TYPE> will be "**confmtg**"

11.1.5 Signal input conditions

There are several input conditions for the RT-09 evaluation from two modalities, audio and video. Systems can use either or both modalities. The items below explain each audio/video input condition and provides the value for the experiment code element <INPUT> (on bold) and the sub-elements <VIDEO> and <AUDIO>.

Video Input Condition <VIDEO>:

- There is a single evaluation condition for video input which is all video (**av**) signals.

Audio Input Conditions <AUDIO>:

- Multiple distant microphones: (**mdm**) This evaluation condition includes the audio from at least 3 omni directional microphones placed on a table in between the meeting participants. The set of microphone recordings will include the microphone selected for the sdm condition.
- Single distant microphone: (**sdm**) This evaluation condition includes the audio of a single, centrally located omni directional microphone for each meeting. The microphone will be placed on a table in between the participants. This microphone's recording will be included in the multiple distant microphone condition explained above. Sites are encouraged to implement this condition as a contrast to the primary condition to examine the effectiveness of employing multiple distant microphones.
- All Distant Microphones: (**adm**) This evaluation conditions permits the use of all distant microphones for each meeting. This condition differs from the MDM condition in that the microphones are not restricted to the centrally located microphones and the Mark III arrays and Source Localization arrays can be used.
- Individual head microphone: (**ihm**) This evaluation condition includes the audio recordings collected from a head mounted microphone positioned very closely to each participants mouth. The microphones are typically cardioid or super cardioid microphones and therefore the best quality signal for each speaker. Since the ihm condition is a contrastive condition, systems can also use any of the microphones used for the mdm condition. Sites are encouraged to implement this condition as a contrast to the primary condition to examine the effectiveness of employing multiple distant microphones.
- Individual head microphones plus reference segmentations: (**ihm-refseg**) This evaluation condition includes the same audio as the ihm condition and systems will be given the additional resources of hand-marked reference speech segmentations.
- Multiple Mark III microphone arrays: (**mm3a**) This evaluation condition will include the audio from all the collected Mark III microphone arrays. The Mark III array is a digital 64-channel microphone, linear topology array. Some meeting spaces will have several arrays recording during the meetings.

Note: This list categorizes the typical commercial-off-the-shelf microphones and experimental microphones placed in a meeting space. For some data collection rooms, experimental microphones may be recorded, for instance a KEMAR manikin was used in some AMI meetings. These types of microphones do not fall in the categories above because the frequency response and transfer functions are different than the typical cardioid and super cardioid microphones included in the above list.

11.2 Evaluation Condition Per Task

The following table outlines the evaluation conditions supported for each task. The evaluation conditions displayed in **bold** font are the required evaluation conditions for the tasks. Participants must run each system entered into the evaluation on the required evaluation condition for each task.

Table 1 RT-06S Evaluation Conditions

Evaluation Condition	**Evaluation Tasks**		
	STT	**SPKR**	**SASTT**
Speed	**ul**	**ul**	**ul**
Evaluation	**eval09**	**eval09**	**eval09**

[25] Required evaluation conditions are covered in Section 1.2.

Data			
Languages	**eng**	**eng**	**eng**
Data type	**confmtg**	**confmtg**	**confmtg**
Audio Input **(subject to availability in data set)**	**mdm** sdm adm ihm* mm3a[26]	**mdm** sdm adm mm3a	**mdm** sdm adm mm3a ihm

* The ihm condition for STT is a required contrast condition. While it is not the evaluation condition of primary interest, it is very similar to the conversational telephone speech domain and therefore a very important evaluation condition.

12 PARTICIPATION INSTRUCTIONS

Participation is encouraged for all those who are interested in one or more of the RT-09 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.

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.

12.1 PROCESSING RULES

12.1.1 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]

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. The waveforms will be in either single channel files for the head microphones, lapel microphones and the table microphones. Sensors like microphone arrays will be delivered in multi-channel, interleaved audio files.

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

12.1.2 ADDITIONAL RULES FOR PROCESSING MEETING SPEECH

The data collection site, room configuration, sensor types, collection date/time, and microphone configurations can be 'known' to the system.

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 and SPKR contrast conditions. 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.

12.2 DATA FORMATS

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

12.2.1 AUDIO DATA AND OTHER CORRESPONDING INPUTS

For practicality, the recorded waveform and video files to be processed will be distributed on either DVD-ROM or hard drives 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
video	Video 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 12.3.1.

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

where,

[26] To the extent possible, the mm3a condition will be supported for the conference room data. This only applies to the NIST meetings.

[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.

<DATA> is [eval09]

<LANG> is [english]

<TYPE> is [confmtg]

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 12.3.1 and these files are named <EXP-ID>.uem with the special site code "expt". Separate UEM files, defined in section 7, 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.

12.2.2 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 [AMI | CMU | ICSI | NIST | VT]

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 short character string the possible values are:

- l → Lapel microphones
- h → Head microphones worn by the participants
- d → Distant microphones with individual sensors placed in the center of the meeting
- sl → CHIL's 4-channel inverted "T" source localization arrays
- na → NIST's Mark III 64-channel linear microphone array
- ci → AMI's 8-channel circular microphone array
- ke → Audio recordings made from inside the head of a KEMAR mannequin.

<MIC_NUM> is a (0-padded) sequence number uniquely identifying the microphone in this meeting. The value may be 'sum' which indicates it is a summed version of all the MIC_TYPE channels. For instance "hsum" is sum of all the head microphones.

Example of a meeting recording name:

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

12.2.3 System Output Formats

Systems will generate a separate file for each meeting. Files will be encoded for in the following formats for each task:

- STT – CTM files as described in Appendix B. Each system output file must have a .ctm file extension.
- SPKR and SASTT – RTTM files as document in Appendix A. The output for each source file must have the extension .rttm.

The output files are to be named so as to be identical to the input file basenames with the appropriate 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.

See Section 12.3.2 which defines where the system outputs go in the submission directory structure

12.2.4 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 12.3.1)

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. System runtime:

[Compute the Total Processing Time (TPT), Source Signal Duration (SSD), and Speech Factor (SF) as specified in Appendix D. Report the numbers by including the following template in the system description:]

```
  TPT = <FLOAT>

       SSD = <FLOAT>

SF = <FLOAT>
```

6. References:

[any pertinent references]

12.3 SUBMISSION INSTRUCTIONS

12.3.1 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>_<INPUT>_<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 ::= 09

TASK ::= sttul | spkrul | sastt

DATA ::= eval09

LANG ::= eng

TYPE ::= confmtg

INPUT ::= <AUDIO>|<VIDEO>|<AUDIO>-<VIDEO>

  VIDEO :== av

  AUDIO :== ihm | sdm | mdm | adm | mm3a |
ihm-refseg
```

`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_09_spkrul_eval09_ihm_eng_confmt_spch_p
-spkrsys_1

sri_09_sttul_eval09_sdm_eng_lectmtg_spch_c
-stttest3_1
```

12.3.2 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 12.2.3

`<EXP-ID>` is as defined in Section 12.3.1

`<OUTPUT-FILES>` are named as specified in Section 12.2.3.

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

12.3.3 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 12.3.1

`<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 Jerome Ajot at ajot@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_09_spkr_eval09_mdm_eng_confmtg_spch_p-
spkrsys_1 <NL>

cmu_09_spkr_eval09_mdm_eng_lectmtg_spch_c-
spkrsystest_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.

13 SCHEDULE

Consult the RT-09 web site for the schedule.

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.

14 UPDATES

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

http://www.nist.gov/speech/tests/rt/rt2009

15 WORKSHOP

This evaluation will be discussed at the NIST Rich Transcription 2009 Spring Meeting Recognition Evaluation being held May 28-29, 2009. See the RT 2009 website for registration details.

Appendix A: RTTM File Format Specification

Rich Transcription Time Marked (RTTM) files are space-separated text files that contain meta-data ‘Objects’ that annotate elements of the recording. Each line represents the annotation of 1 instance of an object. There are ten fields per line. They are:

Table A.1 RTTM Field Names

Field 1	2	3	4	5	6	7	8	9	10
Type	file	chnl	tbeg	tdur	ortho	stype	name	conf	Slat

Fields 1 and 7: Object types (*type*) and object subtypes (*stype*): There are four general object categories represented. They are STT objects, MDE objects, source (speaker) objects, and structural objects.[28] Each of these general categories may be represented by one or more types and subtypes, as shown in table A.2.

Table A.2 RTTM object types and subtypes

Categories	Type	Subtype values (as text strings)
Structural	**SEGMENT**	**eval**, or <NA>
	NOSCORE	<NA>
	NO_RT_METADATA	<NA>
STT	**LEXEME**	**lex, fp, frag, un-lex**[29]**, for-lex, alpha**[30]**, acronym**[30]**, interjection**[30]**, propernoun30**, and **other**
	NON-LEX	**laugh, breath, lipsmack, cough, sneeze**, and **other**
	NON-SPEECH	**noise, music**, and **other**
MDE	**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	<NA>
	SPEAKER	<NA>
Source Info	**SPKR-INFO**	**adult_male, adult_female, child**, and **unknown**

The STT, MDE and Source information objects are potential research targets. 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.)

Field 2: File name (*file*): The waveform file base name (i.e., without path names or extensions).

Field 3: Channel ID (*chnl*): The waveform channel (e.g., “**1**” or “**2**”).

Field 4: Beginning time (*tbeg*): The beginning time of the object, in seconds, measured from the start time of the file.[31] If there is no beginning time, use `tbeg` = ”**<NA>**”.

[28] 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.

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

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

[31] 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**).

Field 5: Duration (*tdur*): The duration of the object, in seconds[31] If there is no duration, use `tdur` = "**<NA>**".

Field 6: Othography field (*ortho*): The orthographic rendering (spelling) of the object for STT object types. If there is no orthographic representation, use `ortho` = "**<NA>**".

Field 8: 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 9: Confidence Score (*conf*): The confidence (probability) that the object information is correct. If `conf` is not available, use `conf` = "**<NA>**".

Field 10: Signal Look Ahead Time (*slat*): The "Signal Look Ahead Time" is the time of the last signal sample (either an image frame or audio sample) used in determining the values within the RTTM Object's fields. If the algorithm does not compute this statistic, `slat` = "<NA>".

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

Table A.3 Format specialization for specific object types

Field 1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Type	*File*	*Chnl*	*tbeg*	*tdur*	*Ortho*	*stype*	*Name*	*Conf*	*SLAT*
SEGMENT	File	chnl	tbeg	tdur	<NA>	eval or <NA>	name or <NA>	conf or <NA>	<NA>
NOSCORE	File	chnl	tbeg	tdur	<NA>	<NA>	<NA>	<NA>	<NA>
NO_RT_METADATA	File	chnl	tbeg	tdur	<NA>	<NA>	<NA>	<NA>	<NA>
LEXEME **NON-LEX**	File	chnl	tbeg	tdur	ortho or <NA>	stype	Name	conf or <NA>	slat or <NA>
NON-SPEECH	File	chnl	tbeg	tdur	<NA>	stype	<NA>	conf or <NA>	slat or <NA>
FILLER **EDIT** **SU**	File	chnl	tbeg	tdur	<NA>	stype	Name	conf or <NA>	slat or <NA>
IP **CB**	File	chnl	tbeg	<NA>	<NA>	stype	Name	conf or <NA>	slat or <NA>
A/P **SPEAKER**	File	Chnl	tbeg	tdur	<NA>	<NA>	Name	conf or <NA>	slat or <NA>
SPKR-INFO	File	Chnl	<NA>	<NA>	<NA>	stype	Name	conf or <NA>	<NA>

Appendix B: Conversation Time Mark (CTM) Format STT System Output

The RT-09 STT output format will be the CTM format (.ctm filename extension). 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 12.3.1 (`EXP-ID`) and is of the form:

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

For example,

```
;;EXP-ID: icsi_09_sttul_eval09_eng_confmtg_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 12.2.2 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. [32]

`<DURATION>` is the duration of the token. This time is a floating point number, expressed in seconds. [32]

`<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`" is used (in CTM format data) when no confidence is computed and in the reference data. [33]

`<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.

[32] 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.

[33] 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.

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

misc is other annotations not covered above.[34]

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
```

[34] 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.

Appendix C: Data Resources

This Appendix identifies the corpora available to system developers for the 2009 NIST Rich Transcription Evaluation (RT-09). These resources are licensed through one of the following: the Augmented Multiparty Interaction (AMI) Program, the Evaluations and Language resources Distribution Agency (ELDA), or the Linguistic Data Consortium (LDC). Participants should request these corpora by contacting NIST and signing all appropriate licensing agreements.

Publicly available meeting resources:

- ICSI Meeting Speech: LDC catalog number LDC2004S02
- ICSI Meeting Speech: LDC catalog number LDC2004T04
- ISL Meeting Speech Part1: LDC catalog number LDC2004S05
- ISL Meeting Transcripts Part 1: LDC catalog number LDC2004T10
- NIST Meeting Pilot Corpus Speech: LDC catalog number LDC2004S09
- NIST Meeting Pilot Corpus Transcripts and Metadata: LDC catalog number LD2004T13
- Rich Transcription 2004 Spring (RT-04S) Development & Evaluation Data
- RT-04S Dev-Eval Meeting Room Data (speech+transcripts) LDC2005S09
- Effective, Affordable, Reusable, Speech-To-Text (EARS) RT-04 Broadcast News training corpus distributed to non-EARS partners as a resource for developing RT-06S systems.
    - Topic Detection and Tracking Phase 4 (TDT4) Multilingual Broadcast News Speech Corpus: LDC2005S11
    - TDT4 Multilingual Text and Annotations: LDC2005T16
- Effective, Affordable, Reusable, Speech-To-Text (EARS) RT-04 Conversational Telephone Speech training corpus distributed to non-EARS partners as a resource for developing RT-06S systems:
    - Fisher English Training Speech Part 1 Speech: LDC catalog number LDC2004S13 (5850 two sided telephone conversations)
    - Fisher English Training Speech Part 1, Transcripts: LDC catalog number LDC2004T19 (5850 transcribed two sided telephone conversations)
    - Fisher English Training Speech Part 2: LDC catalog number LDC2005S13
    - Fisher English Training Speech Part 2, Transcripts: LDC catalog number LDC2005T19

Non-publicly available corpora offered to the RT-06S evaluation participants:

The corpora listed in this section have been produced by several non-affiliated programs. A data sharing agreement has been reached whereby sites not affiliated with each corpus' producer are granted a non-transferable evaluation license to the data. Sites are allowed to retain and use the data for research purposes.

- Augmented Multiparty Interaction (AMI) development test set: a twelve meeting data set collected by the AMI project and distributed to non-AMI partners as a resource for developing RT-06S systems.

Appendix D: 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>
```

Appendix E: Sample Processing Latency Calculations

The SCTK toolkit includes a script,'slatreport.pl', to calculate measures related to the "signal look ahead time(s)" (slat) recorded in RTTM files. The program computes a three version of Sample Processing Latency (SPL):

- SPL$_b$ – the SPL measuring latency from the beginning time of the object to the SLAT time,

$$SPL_b = \frac{\sum_{i=1}^{N_{obj}} SLAT_i - Tbeg_i}{N_{obj}}$$

- SPL$_m$ – the SPL measuring latency from the midpoint time of the object to the SLAT time,

$$SPL_m = \frac{\sum_{i=1}^{N_{obj}} SLAT_i - (Tbeg_i + Tdur_i/2)}{N_{obj}}$$

- SPL$_e$ – the SPL measuring latency from the ending time of the object to the SLAT time

$$SPL_e = \frac{\sum_{i=1}^{N_{obj}} SLAT_i - Tend_i}{N_{obj}}$$

Where:

$$\begin{aligned} N_{obj} &= the\ number\ of\ RTTM\ objects\ used\ for\ SPL\ calculation \\ SLAT_i &= the\ SLAT\ time\ of\ the\ i^{th}\ RTTM\ record \\ Tbeg_i &= the\ beginning\ time\ of\ the\ i^{th}\ RTTM\ record \\ Tdur_i &= the\ duration\ time\ of\ the\ i^{th}\ RTTM\ record \\ Tend_i &= the\ ending\ time\ of\ the\ i^{th}\ RTTM\ record \end{aligned}$$