# The Spoken Term Detection (STD) 2006 Evaluation Plan

## 1 INTRODUCTION

Information processing has become a primary economic activity in the world, and spoken communications is a major source of that information. This, coupled with growing computer-accessible volumes of audio data, has created an opportunity and a need for intelligent retrieval of information from archives of recorded speech. Recognizing this, NIST has created an evaluation initiative to facilitate research and development of technology for retrieving information from archives of speech data. This initiative is called Spoken Term Detection (STD) and is structured as a collaborative research activity that is intended to foster technical progress in STD, with the goals of

- exploring promising new ideas in spoken term detection,
- developing advanced technology incorporating these ideas,
- measuring the performance of this technology, and
- establishing a community for the exchange of research results and technical insights.

The NIST 2006 STD enterprise comprises two main activities for participants. These are a formal evaluation and a two-day follow-up workshop. The evaluation will be conducted in November and the workshop will be held in December. All participants must comply with the guidelines and evaluation rules set forth in this document. For more information, please consult the NIST web site http://www.nist.gov/speech/tests/std or e-mail queries to std-info@nist.gov.

## 2 THE TASK

The STD task is to find all of the occurrences of a specified "*term*" in a given corpus of speech data. For the STD task, a **term** is a sequence of one or more words.

The evaluation is intended to help develop technology for rapidly searching very large quantities of audio data. Although the evaluation actually uses only modest amounts of data, it is structured to simulate the very large data situation and to make it possible to extrapolate the speed measurements[1] to much larger data sets. Therefore, systems must be implemented in two phases: indexing and searching. In the indexing phase, the system must process the speech data without knowledge of the terms. In the searching phase, the system uses the terms, the index, and optionally the audio to detect term occurrences.

### 2.1 Terms

Terms will be specified only by their orthographic representation. Example terms are "grasshopper", "New York", "in terms of", "overly protective", "Albert Einstein", and "Giacomo Puccini".

Ideally, a term would have a single specific interpretation or meaning. In order to make the implementation of STD evaluation feasible, however, the occurrence of a term in the corpus will be judged solely on the orthographic transcription of the corpus. Thus, for example, "wind" (moving air) will match "wind" (twist), despite their distinctly different pronunciations and meanings. On the other hand, "grasshoppers" will not match "grasshopper" and "cat" will not match "catalog". This requirement that the term be an exact match to words in the transcription is judged as being a relatively minor distortion of the objective. Also, the requirement for an exact match corresponds to finding exactly what a user requested. (In real applications, plural forms could be sought simultaneously if that is what the user wishes.)

Terms will be presented in the language's native orthography: ASCII for English, UTF-8 for Arabic, and GBK for Mandarin Chinese. The Arabic terms will be diacritized for the evaluation. No terms will include more than five words.

Reference term occurrences will be found automatically by searching the reference transcript supplied in an RTTM file[2] using the following rules:

1. a term is a contiguous sequence of LEXEMEs of all LEXEME subtypes except 'fp' and 'frag'.
2. every word in a term must be from the same speaker.
3. every word in a term must be from the same file and channel.
4. the gap between adjacent words in a term must be <= 0.5 second
5. NON-LEX events are ignored

### 2.2 System Output

For each term supplied to the system, all of the occurrences of that term in the test corpus are to be found and statistics for each found occurrence are to be output. For each found occurrence of the given term, the system is to output a record that includes

- the location of the term in the audio recording
- a score indicating how likely the term exists with more positive values indicating more likely occurrences
- a hard (binary) decision as to whether the detection is correct

A system output will be considered correct if the term appears in the transcript as an exact match (disregarding case) and if the time of the occurrence corresponds to that of the matching transcript.

The score for each term occurrence can be of any scale (NIST recommends a log likelihood[3]). However, since the scores will be used to derive term-pooled Decision Error Tradeoff (DET) curves, scores across terms must be commensurate to ensure minimum DET curves

## 3 THE DATA

Two data sets will be provided. These are namely a development set ("DevSet") that participating sites may use to aid their research, and an evaluation set ("EvalSet") that will be supplied at the beginning of the formal evaluation. In addition, a third "pilot" evaluation data set will be created as a subset of the DevSet and provided to participants for the pilot ("Dry Run") evaluation. The Linguistic Data Consortium (LDC) will provide these data sets to

[1] It is understood that the extrapolation would be a best-case scenario. The engineering aspects of searching large data stores is significant and will not be accounted for.

[2] See Appendix D for a definition of RTTM files.

[3] The log likelihood, with base e, is suggested, so that the system may be evaluated in a variety of application scenarios that exhibit different prior probabilities

LDC members under membership agreements and to non-members under a limited use license.

Because abundant speech data are available from the LDC and other sources, no “training” data will be provided or specified. Participants may use whatever data they choose, except for the “Forbidden Data” described in Section 3.3.

There are two parts to each data set, namely a list of terms to search for and a speech corpus in which to search for them.

## 3.1 The Search Terms

There will be a wide variety of search terms. These will include single-word and multi-word terms, common and rare terms, and terms that are in the test corpus and those that are not. There will be a total of about 1000 terms, per language, in the DevSet and a total of about 1000 terms, per language, in the EvalSet. The terms in the DevSet will generally differ from those in the EvalSet, although some terms may appear in both.

## 3.2 The Speech Corpora

The development and evaluation corpora will include three languages and three source types.

- The three languages will be Arabic (Modern Standard and Levantine), Chinese (Mandarin) , and English (American).
- The three source types will be Conversational Telephone Speech (CTS), Broadcast News (BNews), and Conference Room (CONFMTG) meetings i.e., goal oriented, small group, roundtable meetings.[4]

Participants may choose the language(s) for which their systems are to be evaluated but must process all available source types. (Since results will be broken out by source type, systems need not try to perform well on all source types.)

Audio data will be distributed in excerpts of broadcasts, conversations, or meetings. The meeting audio data will be distributed as a single channel recording derived from the distantly placed microphones. The meeting audio was generated by ICSI-SRI Meeting Recognition team for the Spring 2006 Rich Transcription Meeting Recognition evaluation[5].

The DevSet and EvalSet corpora will be approximately the same size for each language/source type combination. The amount of audio material will very by language from 1 hour for Arabic and Chinese data to 2-3 hours for English data. The data will be selected from multiple recordings for each language/source type. Table 1 specifies the source types, languages, dialects, and audio durations for the DevSet and EvalSet.

Although the STD evaluation uses only a small amount of data, anticipated applications involve huge audio archives. The evaluation is designed (as indicated in Section 4.3) to ascertain the abilities of systems to process vast amounts of data. Anticipated applications also require users to make ad hoc searches. For this reason, systems may not make use of the term list during any audio indexing steps.

Table 1 Language/Source Type pairs to be tested and the durations of indexed audio for both the DevSet and EvalSet

| | **Arabic** | **Chinese** | **English** |
|---|---|---|---|
| **Broadcast News** | MSA<br>~1 hour | Mandarin<br>~1 hour | American<br>~ 3 hours |
| **Telephone Conversations** | Levantine<br>~1 hour | Mandarin<br>~1 hour | American<br>~3 hours |
| **Roundtable Meetings** | No | No | American<br>~2 hours |

## 3.3 Forbidden Data

The STD evaluation corpora will include speech from the following previously used RT evaluation corpora:

- The Fall 2004 BNews and CTS Rich Transcription Evaluation corpus
- The Spring 2006 Meeting Domain Rich Transcription evaluation corpus

Therefore, participants who possess these corpora must refrain from examining or using them in any way for lexicon building, system training, or development testing. If sites have previously studied or used this data, they must describe what they have done and take appropriate steps to minimize the problem.

Additionally, news-oriented material (audio, textual, etc.) generated after the beginning of the current test epoch (beginning December 1, 2003) or material (other than the RT03 eval data) from the DevSet epoch (February 2001) **may not be used in any way for system development or training.**

# 4 EVALUATION

Systems will be evaluated for both speed and detection accuracy. Speed and accuracy will be measured for a variety of conditions, for example as a function of term characteristics (such as frequency of usage and acoustical features) and corpus characteristics (such as source type and signal quality). Unlike many NIST evaluations of speech processing technology, this evaluation will not specify a single composite measure of goodness. Instead, system performance will always be broken down by source type and then by further distinctions.

## 4.1 Detection Error Tradeoff

Basic detection performance will be characterized in the usual way via standard detection error tradeoff (DET) curves of miss probability ($P_{Miss}$) versus false alarm probability ($P_{FA}$). Miss and false alarm probabilities are functions of the detection threshold, $\theta$, and will be computed separately for each search term:

$$P_{Miss}(term,\theta) = 1 - N_{correct}(term,\theta)/N_{true}(term)$$

$$P_{FA}(term,\theta) = N_{spurious}(term,\theta)/N_{NT}(term)$$

where:

$N_{correct}$(*term*, $\theta$) is the number of correct (true) detections of *term* with a score greater than or equal to $\theta$.

[4] The meeting data will be provided only for English.

[5] “The ICSI-SRI Spring 2006 Meeting Recognition System”, Adam Janin, Andreas Stolcke, Xavier Anguera, Kofi Boakye, Ozgur Cetin, Joe Frankel, and Jing Zheng, Proceedings of the Rich Transcription Spring 2006 Meeting Recognition Evaluation.

$N_{spurious}(term, \theta)$ is the number of spurious (incorrect) detections of *term* with a score greater than or equal to $\theta$.

$N_{true}(term)$ is the true number of occurrences of *term* in the corpus,

$N_{NT}(term)$ is the number of opportunities for incorrect detection of *term* in the corpus (= "Non-Target" *term* trials).

Since there is no discrete specification of "trials", the number of Non-Target trials for a term, $N_{NT}(term)$, will be defined somewhat arbitrarily to be proportional to the number of seconds of speech in the data under test. Specifically:

$$N_{NT}(term) = n_{tps} \cdot T_{speech} - N_{true}(term)$$

where:

$n_{tps}$ is the number of trials per second of speech
($n_{tps}$ will be set arbitrarily to 1), and

$T_{speech}$ is the total amount of speech in the test data
(in seconds).

$P_{Miss}$ and $P_{FA}$ will be computed separately for each term and then averaged over the selected terms, giving equal weight to each search term:

$$P_{Miss}(\theta) = \underset{term}{average}\{P_{Miss}(term,\theta)\}$$

$$P_{FA}(\theta) = \underset{term}{average}\{P_{FA}(term,\theta)\}$$

$P_{Miss}$ and $P_{FA}$ will be averaged over only those terms with a non-zero number of true occurrences in the test data, so that $P_{Miss}$ is defined.

DET curves will be computed as a function of language and source type as well as for various selections of data and terms.

## 4.2 System Detection Performance

Overall system detection performance will be measured in terms of an application model by assigning a value to each correct output and a cost (= negative value) to each incorrect output. Two definitions of overall system value will be used, these being namely an occurrence-weighted value and a term-weighted value.

- Occurrence-weighted value ($Value_O$) is computed by accumulating a value for each correct detection and subtracting from this a cost for each spurious detection, with each detection contributing equally, independent of term:

$$Value_O(\theta) = \frac{\sum_{term}\{V \cdot N_{correct}(term,\theta) - C \cdot N_{spurious}(term,\theta)\}}{\sum_{term}\{V \cdot N_{true}(term)\}}$$

- Term-weighted value ($Value_T$) is computed by first computing the miss and false alarm probabilities for each term separately, then using these and an (arbitrarily chosen) prior probability to compute term-specific values, and finally averaging these term-specific values over all terms to produce an overall system value:

$$Value_T(\theta) = 1 - \underset{term}{average}\{P_{Miss}(term,\theta) + \beta \cdot P_{FA}(term,\theta)\}$$

where:

$$\beta = \frac{C}{V} \cdot \left(Pr_{term}^{-1} - 1\right).$$

$\theta$ is the detection threshold.

For the current evaluation, the cost/value ratio, $C/V$, will be 0.1, and the prior probability of a term, $Pr_{term}$, will be $10^{-4}$.

The maximum possible Value is 100 percent, corresponding to "perfect" system output: no misses and no false alarms. Note that the Value of a system that outputs nothing is zero. Note also that negative Values are possible.

While occurrence-weighted Value models actual operational use, term-weighted Value has the advantage of being less susceptible to being biased toward frequently occurring terms and therefore having a lower sample variance, because it is an average over all terms rather than, in effect, an average over only a small number of frequently occurring terms. It also has the advantage of separating the effect of prior probability from basic detection performance. Note, however, that term-weighted Value by necessity excludes consideration of terms with no reference occurrences (i.e., the OOV terms).

System performance will be analyzed by computing the system's Value conditioned on source type and various term subsets. For example, Value will be computed separately for each source type, for each language processed by the system, and as a function of the number of syllables in the term.

So that NIST may perform a comprehensive analysis of system performance, systems will be required to output, for each search term, more putative occurrences than those which the system determines will maximize system output Value. System output must therefore also include a binary indication of detection, determined so as to maximize system output Value. This will enable NIST to measure system performance at various levels of output (for example, at levels of 1, 10, 100 and 1000 occurrences per term) in addition to the system's determination of the optimum number of occurrences needed to maximize Value. The maximum number of occurrences allowed per search term is 1000 (for each language/source pair).

## 4.3 Primary Evaluation Measures

For the purposes of the 2006 STD Evaluation, the primary evaluation measure will be the "Actual Term-Weighted Value" (ATWV) using the term-weighted value formulas as defined in Section 4.2. The ATWV is the detection value attained by the system as a result of the system output and the binary "YES/NO" decisions output for each putative occurrence. $\theta$, in this case, is superfluous.

NIST will also report the "Minimum Term-Weighted Value" (MTWV) based on the DET analysis. MTWV is the minimum term-weighted value found over the range of all possible values of $\theta$.

## 4.4 Processing Issues

The processing issues of speed and memory are to be reported separately for both preprocessing ("indexing") and for search. The indexing time and the size of the indexing database are to be reported for each corpus (language and source type) indexed. The search time is to be reported for each term and for each language/source type.

Processing times are to be reported in "CPU seconds" of elapsed time, which is the total aggregate time accumulated over all CPU's involved in the processing, as indicated in Appendix B.[6]

The computer(s) used to perform the processing must also be described in the system description as instructed in Appendix C. The computing hardware description will include a listing of key hardware components as well as the output of a computation speed calculation program supplied by NIST.

## 4.5 Evaluation Conditions

### 4.5.1 Occurrence judging:

Each output occurrence will be judged as correct or not according to whether it is close in time to a known occurrence of the search term. The system output occurrence will be judged as correct if the mid-point of the system output occurrence is less than or equal to 0.5 seconds from the time span of a known occurrence of the search term. Note, however, that mapping will be one-to-one. Therefore, if there are two output occurrences that are both permissible matches to only one known occurrence, then one of the output occurrences will be judged as incorrect. Similarly, if there are two known occurrences that are both permissible matches to only one system output, then that system output will be judged as correct for only one of the known occurrences. Within these constraints, the mapping will be performed so as to maximize the number of correct occurrences output by the system.

### 4.5.2 Term interactions:

Each term must be processed separately and independently. The search results for each term are to be output prior to processing the next term.

### 4.5.3 Human interactions:

No manual or human interaction with the data is allowed, including both the evaluation corpus and the search terms. This means that the pronunciation model or other representation of the search terms must be created automatically by the STD system without human guidance.

### 4.5.4 Processing requirements:

Participants must process all of the data and all of the search terms for at least one language.

### 4.5.5 Additional Processing Rules

The following additional rules apply to the evaluation:

- During indexing, systems can perform any within recording adaptation. For the CTS data, this includes both channels.
- Systems must separately index the two CTS channels; however, the meeting data must be indexed as a single source channel.
- For the meeting data, systems must use only the single delayed sum recording provided in the STD corpora even though single-microphone data has been made available through separate means.

### 4.5.6 Primary vs. Contrastive Systems

NIST will accept multiple system submissions from each participant. Participants must designate one system per language as their "primary", or best, system. The rest of the systems submitted for a language will be considered "contrastive" systems. Participants will communicate their selection through the experiment ID as explained in Appendix A.

Participants should not submit "parameter sweeps" for evaluation. That would be considered a system development activity and not a set of contrastive systems.

NIST will restrict their cross-site analysis to the primary systems and use contrastive systems only for intra-site comparisons.

# 5 THE WORKSHOP

The follow-on technical workshop is a key part of the STD evaluation. The workshop is open only to evaluation participants and invited government representatives.

## 5.1 Workshop attendance:

Each evaluation participant is to be represented at the workshop by one or more individuals who participated actively in the evaluation who are knowledgeable of the technology and the work performed, and who will present a meaningful description of their site's system and their experimental results and observations.

# 6 PUBLICATION OF RESULTS:

In order to preserve the pre-competitive, research nature of this evaluation, participants will sign a participation agreement restricting the fair use of evaluation results, cross-site comparisons, and public disclosure of results. As part of the agreement, all system outputs and results will be made public subject to the same fair use agreement.

# 7 DATA STRUCTURES AND FORMATS

All data released for the evaluation will conform to the file formats and directory structures as specified in Appendix A. Systems will be given three information sources to process, the audio, a list of excerpts to index from the audio files, and a term list. The source audio data will be organized by language and source type and consist of NIST SPHERE-formatted[7], digital audio files. The excerpt list and term lists are XML-formatted text files.

System output will be stored in an XML-formatted text file for each language/source type combination. Systems will produce the following information:

1. the language
2. the source type
3. index generation time and index size measurements
4. the search terms with the following information per term: the search time and for each putative occurrence of a term the location, detection score, and binary decision of detection.

DevSet answer keys will be provided along with the speech and term data. The answer key will provide time information for all search terms in the CTM format, (as specified in Appendix A), one file per language/source type combination.

[6] For example, if 10 CPU's spent a total of 2 seconds each in executing a search, the search time would be 20 CPU seconds.

[7] http://www.nist.gov/speech/tools

# 8 SUBMISSION OF RESULTS

Submissions will be made via ftp. Appendix C explains the submission protocol. In addition to the system output results as specified above, a system description is also required for each language/source-type pair. This description must include a description of the hardware used to process the data, and a detailed description of the architecture and algorithms used in the system.

# 9 SCHEDULE

| Date | Event |
|---|---|
| **30 May** | Deadline for e-mail questions and teleconference agenda items |
| **June 2** | Teleconference: 11:00am EDT |
| **21 July** | NIST Distributes evaluation tool plus DevSet search terms, corpora, and answer keys |
| **13 September** | Deadline for signing up to participate |
| **22 September** | NIST provides dry-run test material |
| **29 September** | Sites submit dry-run test results |
| **3 November** | NIST distributes the EvalSet (both corpus and query terms) |
| **8 AM EST on 20 November** | Deadline for submission of evaluation results |
| **1 December** | Release of evaluation results |
| **8 AM EST on 8~~11~~ December** | Deadline for receiving workshop presentations |
| **14-15 December** | Follow-up evaluation workshop |

# Appendix A: Spoken Term Detection Evaluation Implementation Details

Figure 1~~Figure 1~~ shows the system input/output files and how they relate to system operation and evaluation. This appendix documents the file formats for each input and system output.

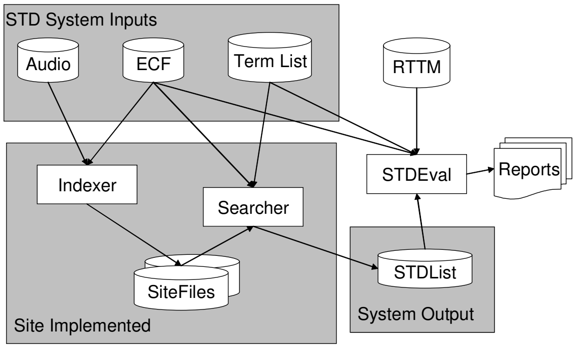

Figure 1: System and evaluation inputs and outputs

The three input files to the site-implemented STD system are as follows[8].

- Audio files: SPHERE formatted waveform files organized as originally distributed. (Appendix A.2)
- Experiment Control File (ECF): ECF files define the excerpts within audio files to be used for specific experiments and the language/source type of each file. (Appendix A.3)
- TermList: The term lists define the terms to search for in the indexed corpus. (Appendix A.4)

During the evaluation, the system inputs will be provided as an ensemble of files labeled with experiment IDs (EXP-ID) to associate the files together (Appendix A.1). Each experiment ID identifies the source type, language, and other salient information to describe the input supplied to the system.

Once the site's indexer and searcher completes processing the data, an STDList file (Appendix A.5) is generated and used by the evaluation code along with the reference RTTM (Appendix D) file to produce the performance analysis reports.

The remainder of this appendix first defines the experiment IDs since they tie together the inputs and outputs. Then, the directory structure of the audio corpora and STD results are described. Finally, the input and output file formats are described.

### A.1 Experiment IDs

Experiment IDs (EXP-ID) link together the system inputs and the system outputs. Since they are used in multiple contexts, some fields contain default fields. This regularization helps maintain corpus cohesion.

The following BNF describes how the experiment ID is structured:

EXP-ID ::= <SITE>_<YEAR>_<TASK>_<DATA>_<LANG>_<TYPE>_<INPUT>_<SYSID>_<*VERSION*>

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 ECF and TermList files.

<YEAR> ::= 06

<TASK> ::= std

<DATA> ::= eval06

<LANG> ::= eng | mand | arab

<TYPE> ::= all[9]

<INPUT> ::= spch

<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 runs for the same condition. Therefore, a different SYSID should be created for runs where any manual changes were made to a particular system.

The special SYSID code "expt" is used in the EXP-ID-based filename of the ECF and TermList files.

<VERSION> ::= 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.

The VERSION code "1" is used in the EXP-ID-based filename of the ECF and TermList files.)

Example EXP-IDs of systems:

cmu_06_std_dev06_eng_all_speech_p-phonetic_1

Example ECF EXP-ID:

expt_06_std_eval06_mand_all_spch_expt_1

### A.2 Corpora Directory Structure

The following directory structure will be used to distribute audio data on DVD-ROMs, distribute transcripts/input files via FTP, and for sites to submit their STD system results during the evaluation.

[8] The diagram is a stylized representation of site implemented system operation and developers are free to organize their systems at their discretion.

[9] Previously, this field specified the source type of the data. The source type designation is now contained in the ECF file.

After the evaluation, system outputs will be released in this structure as well.

| Directory | Description |
|---|---|
| indices/ | Index files (ECFs and IndexList files) for particular experiments |
| audio/ | Audio files |
| input/<EXP-ID>/ | ancillary data provided as additional side information. This data may have additional restrictions on use. |
| 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 Appendix A.1

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

where,

<DATA> is [ eval03 | eval04f | eval05s | eval06s | concatenated]

<LANG> is [ english | mandarin | arabic ]

<TYPE> is [ bnews | cts | confmtg ]

The "concatenated" directory will contain three RTTM files, one for each language, which contains all the RTTM files for the data set.

For the BNews and CTS data, the audio/transcripts will be placed in the <TYPE> directory. For the meeting data, a fourth directory "<MTGID>" is added to collect recordings for each individual meeting.

The "indices/" directory contains a set of ECF index files specifying the waveform data to be evaluated for each EXP-ID condition supported. The directory will also contain the TermList files that define the terms to be searched for. Each of these files is named with the special site code "expt". Separate ECF and TermList files will be provided for each supported language.

Corresponding ancillary data for some control conditions is given in the "input/" directory under subdirectories with the same EXP-ID.

The "output/" directory will contain the output generated by systems and submitted to NIST for the evaluation. Appendix C provides the submission instructions.

### A.2.1 Audio File Name Conventions

Audio corpora for this evaluation will be distributed in their original form. Included below are the corpora file naming conventions for each source type.

#### A.2.1.1 Broadcast News Data

Each BNews recording is a 1-channel, pcm-encoded, 16Khz, SPHERE formatted file. They were collected off of satellite, cable, or local television and radio broadcasts and licensed through the Linguistic Data Consortium. The file names are formatted as follows:

BNFILE :== <DAY>_<ST>_<END>_<NETW>_<SHOW>_ex<E>.sph

Where:

<DAY> :== The date of recording: YYYYMMDD

<ST> :== Start time of the recording on the 24 hour clock.

<END> :== End time of the recording on the 24 hour clock

<NETW> :== [ NBC | CNN | PRI | … ] The network broadcasting the show.

<SHOW> :== [ ARB | WNT | NBW | … ]

<E> :== [ A .. Z ] The excerpt counter.

For example:

20050603_0800_0900_NPR_TWD_exA.sph

#### A.2.1.2 Conversational Telephone Speech Data

Each CTS recording is a 2-channel, μ-law encoded, 8 Khz, SPHERE formatted file. These Linguistic Data Consortium data were collected using T1 digital telephony links using two different methods: the Switchboard corpus and the Fisher corpus. The filenames are formatted as follows:

CTSFILE :== <CORPUSID>_<RECNUM>_ex<E>.sph

Where:

<CORPUSID> :== fsh

<RECNUM> :== An integer number for the conversation number

<E> :== [ A..Z ] The excerpt counter

For example:

fsh_00210.sph and sw_00401.sph

#### A.2.1.3 Meeting Data

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. Each meeting ID identifies the data collection site and recording time. The audio file names are formatted as follows:

<MTG_FILE>:== <REC_LOC>_<REC_TIME>_ex<E>.sph

where

<REC_LOC> :== [ AMI | CMU | ICSI | NIST | VT]

<REC_TIME> :== <YYYYMMDD>-<HHMM>

<E> is the excerpt identifier and varies from A..Z.

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

Example of a meeting recording name:

NIST_20020214-1148_exA.sph

### A.3 Experiment Control Files

Experiment Control Files (ECF)s are the mechanism the evaluation infrastructure uses to specify time regions within an audio recording, the language, and the source type specified for the experimental condition. A *system input ECF* file will be provided for all tasks to indicate what audio data is to be indexed and searched by the system. The evaluation code also uses an ECF file to determine the range of data to evaluate the system on. In the event a problem is discovered with the data, a special *scoring ECF* file will be used to specify the time regions to be scored.

The ECF file is an XML-formatted text file.

**ECF File Naming**

ECF file names use the relevant EXP-ID and end with the '.ecf.xml' extension.

**ECF File Format Description**

The ECF consists of two hierarchically organized XML nodes: "ecf", and "excerpt". The XML scheme for a ECF file can be found on the STD web site. The scheme is the authoritative source: below is the conceptual description.

The "ecf" node contains a list of "excerpt" nodes. The "ecf" node has the following attributes:

- source_signal_duration : a floating point number indicating the total duration in seconds of recorded speech specified by the excerpts as described in appendix C.1.2
- version : A version identifier for the ECF file

Each "excerpt" tag is a non-spanning node which specifies the excerpt from a recording that is part of the evaluation. The "excerpt" has the following attributes:

- audio_filename : The attribute indicates the file id, consisting of the path, filename, and extension of the waveform to be processed. For meeting corpora, the meeting ID <MTGID> is supplied rather than all the audio file names.
- language : language of the original source material either "arabic", "english", or "mandarin".
- source_type : the source type of the recording either "bnews", "cts", or "confmtg"
- channel : The channel in the waveform to be processed.
- tbeg : The beginning time of the segment to processes. The time is measured in seconds from the beginning of the recording which is time 0.0.
- dur : The duration of the excerpt measured in seconds.

For example:

```
<ecf source_signal_duration=”340.00”
     version=”20060618_1400”>
<excerpt
     audio_filename=”audio/dev04s/english/confmtg/NIST_2
     0020214-1148” channel=”1” tbeg=“0.0” dur=”291.34”
     language=”english” source_type=”confmtg”/>
<excerpt
     audio_filename=”audio/eval03/english/bnews/ABC_WN
     N_20020214_1148.sph” channel=”1” tbeg=”0.0”
     dur=”291.34” language=”english”
     source_type=”bnews”/>
…
</ecf>
```

**A.4. TermList Files**

A TermList file is an XML-formatted text file that defines the search terms to be processed by an STD system. Each term is identified by a TERMID which is used to track terms through the evaluation process[10].

**TermList File Naming**

Since TermLists apply to specific experiments, they are named for the experiment ID they are associated with plus an additionally identifier for the term list. The filenames are structured as follows:

TERMLIST-NAME :== <EXP-ID>_<TERMLIST-ID>.tlist.xml

Where:

<TERMLIST-ID> is a free formed text string indicating the use of the termlist.

For example:

expt_06_std_dev06_mand_all_spch_expt_1_monosyl.tlist.xml

**TermList File Format Description**

The TermList file consists of three hierarchically organized XML nodes: "termlist", "term", and potentially several nodes under "term" to specify the term. The XML scheme for a TermList file can be found on the STD website. The scheme is the authoritative source: below is the conceptual description.

The "termlist" node contains a list of "term" nodes. The "termlist" has the following attributes:

- ecf_filename : The basename[11] of the ECF file associated with this TermList file.
- version: A version identifier for the file.
- language : language of the original source material either "arabic", "english", or "mandarin".

Each "term" node is a spanning XML tag that contains a set of additional XML nodes to specify the term. At present, the "term" tag contains a single sub-node "termtext" which is a spanning tag than contains the CDATA (character) string for the term. For processing, leading and trailing white space of the termtext string is NOT considered part of the search term while single term-internal white spaces are. "term" nodes have a single attribute, specified below, while "termtext" has no attributes:

- termid : a string identifying the search term. Search terms may or may not be unique across all corpora.

The following is an example TermList file:

```
<termlist
     ecf_filename=”expt_06_std_eval06_mand_all_spch_expt
     _1” version =”20060511-0900”
      language=”english”>
<term termid=”dev06-0001”>
```

[10] TERMIDs simplify the term tracking process by enabling multi-word search terms and non-ASCII search terms.

[11] The basename of a file excludes the directory names and extensions. For example the basename of "the/directory/file.txt" is "file".

```
        <termtext>find</termtext></term>
    <term termid=”dev06-0002”>
        <termtext>many items</termtext></term>
    </termlist>
```

### A.5 STDList Files

The STDList file is an XML-formatted file produced during the search phase of a system. It contains all the runtime information as well as the search term output generated by the system.

**STDList File Naming**

Since STDLists are produced by an STD system, they apply to a particular ECF and TermList. When system outputs are submitted to NIST, the following naming convention must be used enable NIST to understand the submissions. Internally, sites may use a file naming scheme of their choosing.

The submitted STDList filenames will be as follows:

STDLIST-NAME :== <EXP-ID>_<TERMLIST-ID>.stdlist.xml

Where:

<EXP-ID> is the experiment ID for the data processed by the system. It should be derived from the system input TermList's experiment ID and tailored to the system generating the results.

<TERMLIST-ID> is the TermList ID from the TermList used to produce the system output.

For example

cmu_06_std_eval06_mand_all_spch_p-phonetic_1_monosyl.tlist.xml

**STDList File Format Description**

An STDList file is an XML file with three hierarchically organized XML nodes: “stdlist”, “detected_termlist”, and “term”. The “stdlist” records the system inputs and parameters used to generated the results. The “detected_termlist” is a collection “term” nodes which are the putative detected terms. The XML scheme for an STDLList file can be found on the STD website. The scheme is the authoritative source: below is a content description of the XML nodes and attributes.

The “stdlist” node contains a set of “detected_termlist” nodes: one for each search term. The “stdlist” node contains the five attributes:

- termlist_filename : The name of the TermList file for used to generate this system output.
- indexing_time : The “indexing time” (IT) in seconds of the audio corpus as specified by Appendix B.
- index_size : The byte size of the index built during the indexing phase of system operation. This includes all information sources used during the search phase of system operation to build the system's SDTList output file.
- language : language of the original source material either “arabic”, “english”, or “mandarin”.
- system_id : A text field supplied by the participant to describe the system.

Each “detected_termlist” node contains the system output for a single term. It consists of a set of “term” nodes; each “term” node specifying the location of single detected term. The three attributes of a “detected_termlist” are:

- termid : The termid from the TermList file used to generate the detected terms
- term_search_time : A floating point number indicating the number of CPU seconds spent searching the corpus for this particular term as specified in Appendix B.
- oov_term_count : An integer reporting the number of words in the search term that are Out-Of-Vocabulary (OOV) for the system. If the system does not use a word dictionary, the value should be “NA”.

Each “term” node is a non-spanning XML node that contains the location and detection score for each detected term. The six attributes are:

- file : The basename of the audio file as specified in the ECF file.
- channel : the channel of the audio file were the term was found. For meeting data, the channel is always 1.
- tbeg : The beginning time of the term expressed in seconds with 0.0 being the beginning of the audio recording.
- dur : The duration of the term in seconds.
- score : The detection score indicating the likelihood of the detected term.
- decision : [ YES | NO ] The binary decision of whether or not the term should have been detected to make the optimal score.

An example STDList file is:

```
<stdlist
        termlist_filename=”expt_06_std_eval06_mand_all_spch_
        expt_1_Dev06.tlist.xml”
        indexing_time=”74390.00”
        index_size=”32959402”
        language=”english”
        system_id=”Phonetic subword lattice search”>
<detected_termlist termid=”dev06-0001”
                term_search_time=”24.3” oov_term_count=”0”>
    <term file=”NIST_20020214-1148_d05_NONE”
                channel=”1” tbeg=”6.956” dur=”0.53”
        score=”4.115” decision=“YES“/>
    <term file=”NIST_20020214-1148_d05_NONE”
                channel=”1” tbeg=”45.5” dur=”0.3” score=”4.65”
        decision=“NO“/>
</detected_termlist>
</stdlist>
```

# Appendix B: STD Processing Time Calculations

Spoken Term Detection systems are expected to have two phases of operation to deliver the required system output: indexing and searching. This appendix describes how the calculations are to be computed and reported as part of the system description and published material. Systems should make these calculations for each source type and language separately.

### C.1 Indexing Speed Factor

Indexing Speed Factor (ISF) expresses systems "real-time" factor of the system. It is broken down into to components: indexing time and source signal duration.

### C.1.1 Indexing Time

The Indexing Time (IT) is the number of seconds it takes to process all channels of the recorded speech (including all I/O) on a single CPU. IT represents the time a system would take to process the recorded audio and produce a searchable representation on disk for an operational, batch oriented, STD system as measured by a stopwatch. Processes split over multiple CPUs must be accumulated to compute IT.

IT includes all audio pre-processing steps, intermediate I/O for non-pipelined processing steps, and writing the searchable index to disk. IT specifically does not include a system "warm up" time since an operational system is expected to pre-load computational resources before processing audio data.

### C.1.2 Source Signal Duration

In order to calculate ISF, 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 by the experiment's ECF file. This time is channel-independent and should be calculated across all channels for multi-channel recordings. For example, even though meeting recordings include several table microphones, the SSD measures the wall clock length of the meeting.

### C.1.3 Indexing Speed Factor

Indexing Speed Factor (ISF) is the ratio of "Indexing Time" (IT) to the "Source Signal Duration" (SSD).

$$ISF = IT / SSD$$

The ratio expresses systems "real-time" factor of the system.

### C.1.4 Reporting Indexing Speed

Indexing speed will be reported as an integral part of the STDList system output file. The indexing time must be recorded in the "stdlist" node's "indexing_time" attribute as specified in Appendix A.5.

### C.2 Term Search Speed

Term Search Speed (TSS) measures an operational system's response time for a given term query in terms of the number of single-CPU seconds. Processes spread over multiple CPUs must be accumulated for each search. TSS includes all pre-processing steps for each term but specifically does not include a system "warm up" time since an operational system is expected to pre-load computational resources before processing any term data.

Term search speed will be reported as an integral part of the STDList system output file. The search time must be recorded for each term in the "ddetected_termlist" node's "term_search_time" attribute as specified in Appendix A.5.

# Appendix C: Results Submission Instructions

Each participant will deliver their system outputs to NIST according to the evaluation schedule. In order to facilitate transmission to NIST and subsequent scoring, submissions must be made using the following protocol.

The protocol consists of three steps: (1) preparing a system description, (2) packaging system outputs and system descriptions, and (3) transmitting the data to NIST.

### C.1 System Descriptions

Documenting the system is a vital resource for interpreting system results. As such, each submitted system, (determined by unique experiment IDs), must be accompanied by a system description with the following items included:

Section 1. Experiment ID(s)

List all the experiment IDs for which this system produced submitted results

Section 2. System Description

A brief technical description of your system; if a contrastive test, contrast with primary system description.

Section 3. Training:

A list of resources used for training and development.

Section 4. Computer Resources

This section will describe the computing resources used to produce the system output. The description will include two parts: a listing of the computing hardware and the output of a computational speed measurement program supplied by NIST.

The computing hardware description will include the following: Computer brand, CPU model and clock speed, RAM capacity, and Operating system.

The computational speed analysis program will be provided by NIST as a standalone C program to which will calculate the integer and floating point computation speed of computer. The output of the program must be cut and pasted into the system description.

Section 5. References:

List all pertinent references.

### C.2 Packaging Submissions

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

```
output/<EXP-ID>.txt
output/<EXP-ID>.stdlist.xml
```

where,

`<EXP-ID>.txt is the system description file` as specified in C.1

`<EXP-ID>.stdlist.txt is the STDList generated by the STD system as` defined in Appendix A.5.

### C.3 Transmitting Submissions

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 Appendix A.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 Jonathan Fiscus at jfiscus@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
Experiments:
cmu_06_std_dev06_eng_all_spch_p-sys1_1
cmu_06_std_dev06_eng_all_spch_c-sys2_1
```

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.***

# Appendix D: RTTM File Format Specification

The Rich Transcription Time Mark (RTTM) file format (with ".rttm" filename extension) will be used to represent the reference transcription. The file contains a set of object records, each record contained on a single line with nine, white-space separated text fields. The fields are listed in Table 2. Table 3 contains the list of permissible subtypes "stype" for each "type" field. Table 4 lists the fields used for each "type" object.

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.[12] 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 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**[13], **for-lex, alpha**[14], **acronym**[14]~~[15]~~, **interjection**[14]~~[15]~~, **propernoun14~~15~~**, and **other** |
| NON-LEX | **laugh, breath, lipsmack, cough, sneeze**, and **other** |

[12] 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**).

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

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

| | |
|---|---|
| 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** |

Table 4  Format specialization for specific object types

| **Field 1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
|---|---|---|---|---|---|---|---|---|
| *Type* | *File* | *Chnl* | *tbeg* | *Tdur* | *Ortho* | *stype* | *name* | *conf* |
| **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**<br>**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**<br>**EDIT**<br>**SU** | File | Chnl | tbeg | tdur | <NA> | stype | name | conf or <NA> |
| **IP**<br>**CB** | File | chnl | tbeg | <NA> | <NA> | stype | name | conf or <NA> |
| **A/P**<br>**SPEAKER** | File | chnl | tbeg | tdur | <NA> | <NA> | name | conf or <NA> |
| **SPKR-INFO** | File | chnl | <NA> | <NA> | <NA> | stype | name | conf or <NA> |