2011 TRECVID MULTIMEDIA EVENT DETECTION EVALUATION PLAN

1 Overview

This is the evaluation plan for Multimedia Event Detection (MED) track of the 2011 TRECVID evaluation. The multi-year goal of the MED track is to support the creation of *ad hoc* event detection technologies that will permit users to define their own complex events and to quickly and accurately search large collections of multimedia clips. Reprocessing large collections for each newly defined event is not a viable technology solution, especially given the deluge of multimedia clips generated each day.

A MED system is defined to have three separate phases:

1) **metadata generation** (video ingest and metadata store creation),
2) **event agent generation** (event definition ingest and event agent creation), and
3) **event agent execution** (search)

Searches will run one event at a time against the **metadata store**. The metadata store must be created prior to ingesting the event definition (and the search), thus the metadata store must be sufficiently rich such that it could be searched successfully by any *ad hoc* event. For MED 2011 (MED-11) the metadata store may be optimized with knowledge of the events to be evaluated, however, future MED evaluations will implement ad-hoc event tests for which the metadata store cannot be optimized.

For the purpose of MED, an event:

- is a complex activity occurring at a specific place and time;
- involves people interacting with other people and/or objects;
- consists of a number of human actions, processes, and activities that are loosely or tightly organized and that have significant temporal and semantic relationships to the overarching activity;
- is directly observable.

Participants may use any architecture to generate the metadata store. However, participants must use ONLY COTS standard personal computing platform(s) to run searches. The searches are to be performed locally and the output of the searches is to be submitted to NIST per the format provided below for official scoring and analysis. *Each* submission must include a system description which describes:

- the assembled components,
- how closely the system addresses the full MED goal of processing all test events,
- the hardware components used and computation times of the metadata generation phase,
- the implementation method (automatic or semi-automatic) for the event agent generation, and
- the hardware components used, computational times, and the results of the event agent execution

The MED evaluation is open to all that find the task of interest and who are willing to abide by the rules of the evaluation.

Jonathan Fiscus 7/25/11 10:20 PM
Deleted: generate event agents and

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

2 Task

The task is Multimedia Event Detection. The goal of the MED task is to build technologies capable of searching large collections of data using ad hoc events. Therefore, systems that process all ten test events will be referred to as **MEDFull**, while systems that process from one to nine events will be referred to as **MEDPart**.

Systems developed separately for *MEDFull* and *MEDPart* are not comparable and therefore comparison of such results will not be made by NIST. All participants must minimally build a *MEDPart* system but are encouraged to address the full challenge posed by *MEDFull*.

For each event search the system is to generate:

- **A Detection Threshold** for the event: A probability value between 0 and 1 - an estimation of the detection score at or above which the system will assert that the event is detected in the clip.

  A different threshold can be specified for each event. The threshold should be chosen to minimize errors based on an operating point with a 12.5:1 miss-to-false-alarm probability ratio. This ratio has been selected to support a typical search application where a user would not tolerate false alarms on the first page of output.

- **A Score** for each search collection clip: A probability value between 0 (low) and 1 (high) representing the system's confidence that the event is present in the clip.

The primary measure of performance will be based on a hard decision derived from the clip scores and event threshold.

2.1 System Inputs

System inputs will be specified through a set of three Comma Separated Value (CSV)[1] tables which will be provided for each collection of clips[2]:

- The Event Database - *_EventDB.csv files
  This two-column table defines the **EventID** and **EventNames**.

- The Clip Database - *_ClipMD.csv files
  This five-column table contains the metadata for each clip in the collection. The fields are **ClipID**, **MEDIA_FILE**, **CODEC**, **MD5SUM** and **DURATION**.

- The Trial Index Database - *_TrialIndex.csv files
  This three-column table specifies the detection trials a system must perform. A detection trial is an (**EventID**/**ClipID**) pair for which a system must provide output. Each trial is identified with a unique **TrialID**. This file contains the full matrix of event/clip trials.

Two additional CSV tables are provided for training and system building:

[1] See Appendix C for the CSV file format specification.
[2] These tables are the authoritative sources for system inputs. Participants should avoid using directory listings of the collections as inputs because the tables are an experimental control mechanism.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

- The Event Judgment Database - *_JudgementMD.csv files
  This multi-column file contains the clip-level event judgments (positive, near_miss, not_sure, and related) and additional annotations for (**EventID/ClipID**) pairs.

  The table does not contain the full matrix of pairs. Rather, it contains only the human judgments made during data collection.

- The Reference Database - *_Ref.csv files
  This two-column table specifies ground truth.

  *The file defines for each **TrialID** whether or not it is a "target" trial (contains an instance of the event) or a "non-target" trial (does not contain an instance of the event). The file contains the full matrix of* **EventID/ClipID** *pairs using information derived from the Event Judgment Database.*

2.2 System Outputs and Documentation

For each submitted run to be considered complete and valid, participants must provide the information requested in this section. The requested information is a crucial element required for the research community to properly interpret the performance results.

2.2.1 System Description

The purpose of the system description document is to provide a list of the resources and techniques used to build the MED system and it identifies the computing resources and time required to process the test set.

See Appendix B section B.1 for the template that covers the ***minimum*** requirements of the system description document.

2.2.2 Event Agent Execution Reporting

A MED system processes the metadata store detecting instances of each event independently. For each system submission, two output files must be created using Experiment Identifiers (**EXP-ID**) which describe the characteristics of the run (see Appendix B for the definition of *EXP-ID*):

1. <EXP-ID>.threshold.csv

   Each line will contain information pertaining to the processing of a single event. Events not processed should not be included in the file. The 3 fields in this file will be:

   - **EventID**: the Event ID processed - copied from the event database file.
   - **DetectionThreshold**: A floating point probability value between 0 and 1
   - **DetectionTPT**: A value indicating the number of hours used during the *event agent execution phase* for the event.

2. <EXP-ID>.detection.csv

   Each line will contain information pertaining to the processing of a single trial. The 2 fields in this file will be:

Jonathan Fiscus 7/25/11 10:27 PM
Deleted: and trial

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

1. **TrialID**: The Trial ID processed - copied from the input trial index file.
2. **Score**: A probability value between 0 (low) and 1 (high)

3 Data Resources

Internet multimedia data (i.e., video clips containing both audio and video) will be provided to registered MED participants. The data, collected by the Linguistic Data Consortium (LDC), consists of publically available, user-generated content posted to the Internet video hosting sites. The LDC will be the distribution point for the collection.

The video is provided in MPEG-4 formatted files. The video will be encoded to the H.264 standard. The audio will be encoded using MPEG-4's Advanced Audio Coding (AAC) standard.

See http://www.nist.gov/itl/iad/mig/med11.cfm for data licensing and acquisition instructions

Data will consist of *event kits*, *training resources*, and "blind" *testing resources*.

3.1 Event Kits

There will be 15 events in MED-11, 5 events will be designated as training events and 10 will be used as testing events. The event names and their designation are listed in Table 1.

Training Events	Testing Events
Attempting a board trick	Birthday Party
Feeding an animal	Changing a vehicle tire
Landing a fish	Flash mob gathering
Working on a woodworking project	Getting a vehicle unstuck
Wedding ceremony	Grooming an animal
	Making a sandwich
	Parade
	Parkour
	Repairing an appliance
	Working on a sewing project

Table 1: MED-11 Training and Testing Events

Events will each be defined via an **event kit** consisting of:

event name	A mnemonic title for the event.
event definition	A textual definition of the event.
event explication	An expression of some event domain-specific knowledge needed by humans to understand the event definition.
evidential description	A textual listing of some attributes that are often indicative of an event instance. The evidential description provides a notion of some potential types of visual and acoustic evidence indicating the event's existence but it is not an exhaustive list nor is it to be interpreted as required evidence.
illustrative video examples	Two forms of illustrative videos will be supplied: 1) "Positive": (100+) clips that contain at least 1 instance of the event 2) "Near_Miss": (<10) clips with content closely related to the event but lacking critical evidence for a human to declare the event occurred[3].

[3] For MED-11, the near miss clips will be identified in the clip collection, not in the event kits.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

	3) "Related": (<100) clips that contain one or more of the same or similar types of people, objects, locations, and/or actions associated with the target event, but does not meet the requirements to be a positive instance.

Table 2: Components of an event kit

Participants may use all resources supplied in the events kits, and any ancillary information provided in accompanying CVS tables, for system development and testing.

3.2 Training Resources

Two multimedia collections will be provided for training which participants may use for research, development testing, and error analysis of development testing. Both collections will include truth data. There are no MED '11 evaluation rules or restrictions governing how participants are to use the training resources in conjunction with the training and evaluation event kits.

3.2.1 Training Collections from Previous MED Evaluations

The first training collection is the MED-10 Pilot Evaluation data. This collection contains annotations for three events, "assembling_a_shelter", "batting_in_a_run", and "making_a_cake". The data set consists of 94, 102, and 94 positive examples of clips respectively and 3178 background clips (near_miss examples are annotated in an accompanying metadata table).

3.2.2 New Training Collections

The second training collection is the transparent development data collection (DEV-T). DEV-T[4] is expected to be a 350hr collection consisting of about 11K clips. The collection will contain positive clips for the training events, near_miss clips for both the training and testing events, and background clips.

3.3 Testing Resources

The test collection (MED11TEST) will be used for *blind testing*. MED11TEST is expected to be a 1000hr collection consisting of about 34K clips.

While MED11TEST will be used for *blind testing*, participants will have the data several months before the evaluation in order to debug their metadata generation process (prior to searching). Therefore, participants must adhere to the following rules:

- MED11TEST must be automatically processed to create the metadata store. Adaptation in the feature extraction process is permitted so long as the adaptation is fully automatic (***no human interaction***).
- Participants must not attempt to gain knowledge of MED11TEST properties or content by manually inspecting the video, clip metadata, output of the processing, or statistics developed during the processing[5].
- The MED11TEST metadata store must be frozen, (i.e., it may not be regenerated or augmented), before the first search occurs.
- MED11TEST may not be used to influence the event agent generation phase.

[4] DEV-T will be released in two parts; the first release will include all positive, all near_miss, and several background clips. The second release will contain additional background clips.

[5] An appropriate control method would be to have a non-team member process the MED11TEST data in preparation for the evaluation.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

4 Evaluation Measures

System output will be evaluated by how well the system detected MED events in MED11TEST and by the computing resources used to do so. The determination of correct detection will be at the clip level, i.e. systems will provide a response for each clip in MED11TEST. Each event will be scored independently.

MED system performance will be evaluated as a binary classification system by measuring performance of two error types: Missed Detection (MD) errors and False Alarm (FA) errors. NIST will report the primary performance measures for accuracy and processing speed, and a suite of diagnostic measures that may provide a deeper analysis of system performance.

4.1 Primary Measures

P_{MD} and P_{FA} for the event based on the *Detection Threshold*

NIST will report the following performance statistics by applying the event-specific *Detection Threshold* to the *Score* for each clip.

$P_{MD}(E,DT) = \#MD(E,DT) / \#Targets(E)$
$P_{FA}(E, DT) = \#FA(E, DT) / (\#TotalClips - \#Targets(E))$

Where

E	→ The event
DT	→ The detection threshold applied to the system's scores
#MD(E,DT)	→ The number of positive clips for event E < the Detection Threshold
#Targets(E)	→ The number of positive clips for event E.
#FA(E,DT)	→ The number of non-positive clips for event E >= the Detection Threshold
#TotalClips	→ The total number of clips in the testing collection

Metadata Generation Processing Speed

NIST will report the **real-time** factor to complete all steps necessary to build the metadata store. *Real-time* factor is the Total Processing Time for the process (as reported in the system description) divided by the number of hours of video in the test collection.

Event Agent Execution Processing Speed

NIST will report the *real-time* factor for each event processed during the event agent execution phase.

4.2 Diagnostic Measures

Detection Error Tradeoff Curves

Graphical performance assessment uses a Detection Error Tradeoff (DET) [2] curve that plots the system's Missed Detection probabilities (P_{MD}) and False Alarm probabilities (P_{FA}) over the full range of the system's decision scores. The resulting graph provides error performance characteristics for alternative operating points.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

Actual NDC Computation

The Normalized Detection Cost (***NDC***) function[6] is a weighted linear combination of the system's event specific missed detection and false alarm probabilities. ***ActualNDC*** is a specialized version of the ***NDC*** function in that the P_{MD} and P_{FA} are the probabilities based on the event *Detection Threshold* as reported. This measure may assist in forensic results analysis by computing a single-number measure of performance that is weighted by parameters that approximate the requirements of an application profile selected by NIST. NIST will use the following constants for the ***ActualNDC***, C_{MD}=80, C_{FA}=1, P_{Target}=0.001. Appendix A explains the formulation of the constants and the rationale for their selection.

NDC at the Target Error Ratio

NDC at the Target Error Ratio (***NDC @ TER***) is a diagnostic metric based on an analysis of the DET Curve. It is the location of the intersection between the system's DET Curve and the Target Error Ratio line. The measure compares system performance at the same operating point by ignoring the system's *Decision Threshold*.

Given the chosen parameters for the ***ActualNDC*** function, the ***TER*** for MED-11 is:

$$TER = \frac{Cost_{FA} * (1 - P_{Target})}{Cost_{MD} * P_{Target}}, \quad \text{where } Cost_{MD} = 80, Cost_{FA} = 1, \text{ and } P_{Target} = 0.001,$$

$$TER = \frac{1 * (1 - 0.001)}{80 * 0.001} = 12.4875$$

The Minimum ***NDC*** is the point on the system's DET Curve where the minimum ***NDC*** occurs using the same cost constants as ***ActualNDC***. The difference between the value of Minimum ***NDC*** and ***ActualNDC*** provides a quantitative indication of the benefit a system could have gained by selecting a better threshold based on the cost parameters.

4.3 Evaluation Tools and Command Line Example

NIST will use the Detection EVAluation (DEVA) tools within the NIST Framework for Detection Evaluation (F4DE) toolkit to score the evaluation submissions. The TRECVID MED '11 Scoring Primer document (DEVA/doc/TRECVid-MED11-ScoringPrimer.html within the F4DE release) contains instructions for how to use the scorer. NIST will use the following command line to evaluate MED systems.

```
% DEVA_cli\
--profile MED11\
--outdir <OUTPUTDIR> \
--refcsv <DATA>_Ref.csv \
--syscsv <EXP-ID>.detection.csv:detection \
--syscsv <EXP-ID>.threshold.csv:threshold \
<DATA>_TrialIndex.csv:TrialIndex \
<DATA>_ClipMD.csv:ClipMD \
<DATA>_JudgementMD.csv:JudgementMD \
<DATA>_EventDB.csv:EventDB
```

Where,

[6] NDC's derivation can be found in Appendix A

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

- <OUTPUTDIR> is the pre-existing output directory
- <EXP-ID> is the Experimental ID defined in Appendix B
- <DATA> is the data set as defined in Appendix B

5 Result Submission Instructions

Submissions will be made via ftp according to the instructions in Appendix B. In addition to the system output, a system description must be supplied for each submission.

Participants may submit up to four "runs"[7] of system results. One run must be designated as the primary submission and the rest as contrastive runs. The primary run is expected to yield the best performance on the blind test set based solely on experiments using the training resources. NIST will focus the cross-site analysis on the primary runs.

6 Schedule

For TRECVID related schedule information please consult the main schedule on the TREVID 2011 web site http://www-nlpir.nist.gov/projects/tv2011/#schedule.

7 References

[1] Harold W. Kuhn, "The Hungarian Method for the assignment problem", *Naval Research Logistic Quarterly*, **2**:83-97, 1955.

[2] Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M., "The DET Curve in Assessment of Detection Task Performance", Eurospeech 1997, pp 1895-1898.

[7] A run in this context is a particular configuration of technology components for an ensemble of MED events. For example, the inclusion or exclusion of acoustic features would be labeled as different runs but event-dependent parameter tuning would not. Participants should use their judgment as to whether or not technology configurations warrant differentiation.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

Appendix A: Normalized Detection Cost Definition and Derivation

Normalized Detection Cost ($\boldsymbol{NDC}$) is a weighted linear combination of the system's Missed Detection and False Alarm probabilities. *NDC* measures the performance of a detection system in the context of an application profile using error rate estimates calculated on a test set. The application profile is an arbitrarily selected use case for the technology. $\boldsymbol{NDC}$ is defined to be:

$$NDC(S,E) = \frac{Cost_{MD} * P_{MD}(S,E) * P_{\text{Target}} + Cost_{FA} * P_{FA}(S,E) * (1 - P_{\text{Target}})}{MINIMUM(COST_{MD} * P_{\text{Target}}, Cost_{FA} * (1 - P_{\text{Target}}))}$$

Where

- *S,* the system
- *E*, the event
- $P_{MD}(S,E)$, the probability of missed detection.
- $P_{FA}(S,E)$, the probability of a false alarm.
- P_{Target}, the probability of a target event occurring in the context of the application profile.
- $Cost_{MD}$, the detrimental cost to performance for a missed detection in the context of the application profile.
- $Cost_{FA}$, the detrimental cost to performance for a false alarm in the context of the application profile.

The first two values, $P_{MD}(S,E)$ and $P_{FA}(S,E)$, are calculated for the system using the test corpus[8]. The final three values, P_{Target}, $Cost_{MD}$, and $Cost_{FA}$, are constants that form a numeric representation of the application profile. They are supplied by the evaluation team after considering the application space for the evaluation. The chosen values communicate to evaluation participants an optimum error tradeoff point for system tuning.

$\boldsymbol{NDC}$ is derived beginning with the cost of missing an event ($Cost_{MD}$) and the cost of falsely detecting an event ($Cost_{FA}$). $N_{MD}(S,E)$ is the number of missed detections for system $\boldsymbol{S}$, event $\boldsymbol{E}$. $N_{FA}(S,E)$ is the number of false alarms for the same system and event.

$$DetectionCost(S,E) = Cost_{MD} * N_{MD}(S,E) + Cost_{FA} * N_{FA}(S,E)$$

To facilitate comparisons across systems and test sets, we divide Detection Cost by the number of video clips in the video collection ($\boldsymbol{N_{Trials}}$).

$$DetectionCost(S,E) = \frac{Cost_{MD} * N_{MD}(S,E) + Cost_{FA} * N_{FA}(S,E)}{N_{Trials}}; (1)$$

$$= Cost_{MD} * \frac{N_{MD}(S,E)}{N_{Trials}} + Cost_{FA} \frac{N_{FA}(S,E)}{N_{Trials}}$$

Both ratios, $N_{MD}(S,E)/N_{Trials}$ and $N_{MD}(S,E)/N_{Trials}$, convolve performance statistics with the richness of

[8] $P_{MD}(S,E)$ and $P_{FA}(S,E)$ can be calculated at any threshold on the decision score space. This appendix does not address the various rules for specifying the threshold.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

events in the test set. The two can be separated by dividing by unity ($N_{Target}(E)$, the number of targets for event *E* and $N_{NonTarget}(E)$ the number of non-target trials for event *E*).

$$DetectionCost(S,E) = Cost_{MD} * \frac{N_{MD}(S,E)}{N_{\text{Target}}(E)} * \frac{N_{\text{Target}}(E)}{N_{Trials}} + Cost_{FA} * \frac{N_{FA}(S,E)}{N_{NonTarget}(E)} * \frac{N_{\text{NonTarget}}(E)}{N_{Trials}}$$
$$= Cost_{MD} * P_{MD}(S,E) * P_{Target}(E) + Cost_{FA} * P_{FA}(S,E) * (1 - P_{Target}(S,E))$$

$P_{Target}(E)$ is the probability of a clip containing the event. This value is dependent on the event but providing this probability to a system for each event changes the definition of an event (i.e., to include probability in the event kit), which is not feasible. Instead, we replace the event-dependent prior with a single, global prior, P_{Target}, that in combination with the $Cost_{MD}$ and $Cost_{MD}$ reflects the characteristics of a single application profile. The modified formula becomes:

$$DetectionCost(S,E) = Cost_{MD} * P_{MD}(S,E) * P_{\text{Target}} + Cost_{FA} * P_{FA}(S,E) * (1 - P_{\text{Target}})$$

The range of the *DetectionCost(S,E)* measure is [0,∞). A second normalization scales the cost to be 0 for perfect performance and 1 to be the cost of a system that provides no output (either providing no output, $P_{MD} = 1 \text{ and } P_{FA} = 0$, or declaring every clip to be an instance $P_{MD} = 0 \text{ and } P_{FA} = 1$). The resulting formula is the Normalized Detection Cost of a system (***NDC***).

$$NormDetectionCost(S,E) = \frac{DetectionCostRate(S,E)}{MINIMUM(COST_{MD} * P_{\text{Target}}, Cost_{FA} * (1 - P_{\text{Target}}))}$$

In the NDC function, the units of $Cost_{MD}$ and $Cost_{FA}$ cancel out in the equation therefore only the relative values of $Cost_{MD}$ and $Cost_{FA}$ have an impact on *NDC*.

The cost model parameters define a line in DET Curve space where the two error types contribute equally to the measured *NDC*. The line formula can be calculated by setting the components of the NDC equal to each other and solving as follows.

$$\frac{Cost_{MD} * P_{MD} * P_{\text{Target}}}{MINIMUM(Cost_{MD} * P_{\text{Target}}, Cost_{FA} * (1 - P_{\text{Target}}))} = \frac{Cost_{FA} * P_{FA} * (1 - P_{\text{Target}})}{MINIMUM(Cost_{MD} * P_{\text{Target}}, Cost_{FA} * (1 - P_{\text{Target}}))}$$
$$Cost_{MD} * P_{MD} * P_{\text{Target}} = Cost_{FA} * P_{FA} * (1 - P_{\text{Target}})$$
$$P_{\text{MD}} = \frac{Cost_{FA} * (1 - P_{\text{Target}})}{Cost_{MD} * P_{\text{Target}}} * P_{FA}$$

The line formula can be converted to isolate the Target Error Ratio.

$$TER = \frac{P_{\text{MD}}}{P_{FA}} = \frac{Cost_{FA} * (1 - P_{\text{Target}})}{Cost_{MD} * P_{\text{Target}}}$$

Since all systems, regardless of their intrinsic performance level, are expected to be optimized for the

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

cost function parameters, the most consistent location in DET Curve space to compare systems is where the DET Curve crosses the ***TER*** line assuming thresholds further away from the optimized point are less well tuned. The intersection of the system's DET Curve and the ***TER*** line defines the P_{MD} and P_{FA} values used for the ***NDC*** at the Target Operating Ratio (***NDC@TER***) line.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

Appendix B: Submission Instructions

The packaging and file naming conventions for MED-11 relies on **Experiment Identifiers** (EXP-ID) to organize and identify the files for each evaluation condition and link the system inputs to system outputs. Since EXP-IDs may be used in multiple contexts, some fields contain default values. The following section describes the EXP-IDs to be used for the Development Transparent Subset (DEV-T) and the Development Opaque Subset (MED11TEST).

The following EBNF describes the EXP-ID structure:

EXP-ID ::= <TEAM>_MED11_<DATA>_<MEDTYPE>_<EAG>_<SYSID>_<VERSION>

where,

<TEAM> ::= your Short TRECVID Team Name

<DATA> ::= either "DEVT", "MED11TEST", or "DRYRUN"

<EAG> :== either "AutoEAG" or "SemiAutoEAG" specifying the event kit processing style as defined in Section 3.

<MEDTYPE> :== either "MEDFull" or "MEDPart" as defined in Section 3.

<SYSID> ::= a site-specified string (that does not contain underscores) designating the system used.

The SYSID string must be present. It is to begin with p- for a primary system (i.e., your single best system) or with c- for any contrastive systems. For example, this string could be p-baseline or c-contrast. This field is intended to differentiate between runs for the same evaluation condition. Therefore, a different SYSID should be created for runs where any changes were made to a system.

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

Note: there can be only one primary system in a given submission and only one submission per TEAM.

In order to facilitate transmission to NIST and subsequent scoring, submissions must be made using the following protocol, consisting of three steps: (1) preparing a system description, (2) packaging system outputs and system descriptions, and (3) transmitting the data to NIST.

B.1 System Descriptions

Documenting each system is vital to interpreting evaluation results. As such, each submitted system, (determined by unique experiment identifiers), must be accompanied by a system description with the following information:

Section 1 Experiment Identifier(s)

List all the experiment IDs for which system outputs were submitted. Experiment IDs are described in further detail above.

Section 2 System Description

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

Section 3 Metadata Generation System Hardware Description and Runtime Computation

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

Describe the hardware setup(s) to perform the metadata generation phase and the Total Processing Time (TPT). The phase may be broken down into sub-steps in which case the hardware and processing time of each sub-step must be documented.

A hardware setup is the aggregate of all computational components used to perform this phase. Examples of a system might be: a 16-node, Dual Quad Core 2.26 GHz Intel Xeon, 24GB RAM per node, with a 10TB Data Server.

TPT is the wall clock time (in hours) used to complete this phase, including I/O, from start to finish. The processing time for parallelized sub-steps adds to TPT as a single step. The processing time for metadata "shared" across sites, e.g., speech transcription, person tracking, etc. (including time used to incorporated data into the metadata store) adds to TPT as a sub-step.

Section 4 Event Agent Execution Hardware Description

Describes the computing hardware used for executing the event agent(s).

The hardware setup is the aggregate of all computational components used to perform this phase. This hardware platform must be limited to a COTS standard personal computing platform.

Section 5 Training data and knowledge sources

Lists the resources used for system training, development, and runtime knowledge sources beyond the provided MED corpora.

Section 6 References

A list of pertinent references.

B.2 Packaging Submissions

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

output/<EXP-ID>/<EXP-ID>.txt
output/<EXP-ID>/<EXP-ID>.detection.csv
output/<EXP-ID>/<EXP-ID>.threshold.csv

where,

EXP-ID is the experiment identifier as described in Section B.1,
<EXP-ID>.txt is the system description file as specified above (Section B.2),
<EXP-ID>.detection.csv is the CSV-formatted system output file containing the detection scores for each TrialID (see Section 2.2.2).
<EXP-ID>.threshold.csv is the CSV-formatted system output file containing the detection thresholds and processing speed measurements (see Section 2.2.2)

B.3 Validating the Submission

The F4DE distribution contains a submission checker that validates the submission both at a syntactic and semantic level. Participants should check their submission prior to sending it to NIST. NIST will reject submissions that do not pass validation. The TRECVID MED '11 Scoring Primer document (DEVA/doc/TRECVid-MED11-ScoringPrimer.html within the F4DE release) contains instructions for how to use the validator. NIST will use the following command line to validate MED submission files.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

```
%TV11MED-SubmissionChecker --TrialIndex <DATA>_TrialIndex.csv \
  MED11_testTEAM_DRYRUN_2.tar.bz2
```

B.4 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 prefer. 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 > MED11_<TEAM>_<DATA>_<SUB-NUM>.tgz

where,

<TEAM> and <DATA> and are the same as defined above.

<SUB-NUM> is an integer *1 to n*, where 1 identifies your first submission, 2 your second, etc.

Note that only the latest submission will be used for scoring, but a submission file can contain multiple EXPID as long as there is only one primary one. If there is only one EXPID, it must be the primary one.

This command creates a single tar/gzip file containing all of your results. After shipment to NIST (in the next step), NIST will validate your submission with a syntactic and semantic validator. The tool will be added to the evaluation tool suite so that sites can validate their submissions prior to shipment. More information about the submission validator will be supplied at a later date.

Next, ftp to jaguar.ncsl.nist.gov giving the username 'anonymous' and (if requested) 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 MED11_<TEAM>_<DATA>_ <SUB-NUM>.tgz
ftp> quit
```

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). Please note whether you get any error messages from the ftp process when you execute the ftp commands stated above and report them to NIST.

The last thing you need to do is send an e-mail message to brian.antonishek@nist.gov, jfiscus@nist.gov, and martial@nist.gov to notify NIST of your submission. The following information should be included in your email:

- the name of your submission file,
- the file size,
- a listing of each of your submitted experiment IDs.

Please submit your files in time for us to deal with any transmission errors that might occur well before the due date if possible. Note that submissions received after the stated due dates for any reason will be

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

marked late.

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan

Appendix C: Comma Separated Value File Format Specifications

The MED evaluation infrastructure uses Comma Separated Value (CSV) formatted files with an initial field header line as the data interchange format for all textual data. The EBNF structure the infrastructure uses is as follows:

CSVFILE :== <HEADER> <DATA>*

<HEADER> :== <VALUE> {“,” <VALUE> }* <NEWLINE>
<DATA> :== <VALUE> {“,” <VALUE> }* <NEWLINE>
<VALUE> :== <DOUBLEQUOTE><TEXT_STRING><DOUBLEQUOTE>

The first data record in the files is a header line. The header lines are required by the evaluation infrastructure and the field names for the trial index file and the system output file are dictated by Sections 4.1 and 4.2.

Each header and data record in the table is one line of the text file. Each field value is delimited by double quotes and is separated from the next value with a comma.

An example trial index is (*_TrialIndex.csv”):

```
"TrialID","ClipID","EventID"
"72.P001","72","P001"
"72.P002","72","P002"
"72.P003","72","P003"
"285.P001","285","P001"
"285.P002","285",”P002"
"285.P003","285","P003"
```

An example system output file is (“.detection.csv):

```
"TrialID","Score"
"72.P001","0.062712"
"72.P002","0.978791"
"72.P003","0.115392"
"285.P001","0.801007"
"285.P002","0.861036"
"285.P003","0.120700"
```

An example threshold system output file is (*.threshold.csv):

```
"EventID","DectectionThrehold",”DetectionTPT”
"P001","0.54",”5923.3”
"P002","0.74",”9204.3”
```

Jonathan Fiscus 8/1/11 8:30 PM
Formatted: Font:10 pt, Font color: Black, Do not check spelling or grammar

Jonathan Fiscus 8/1/11 8:29 PM
Deleted: Jonathan