# Computational Cultural Understanding Program Evaluation Plan for Eval3

**Last update: January 2, 2025**

**Audrey Tong, Jonathan Fiscus, Jennifer Yu, Kay Peterson**

**National Institute of Standards and Technology**

**Contact: nist_ccu@nist.gov**

## Revision History

- V1.1 January 2, 2025: Corrected typos in the dataset names of LC5 and LC6. They should be LDC2025E01-V1 for full LC5 and LDC2025E02-V1 for full LC6.
- V1.0 December 26, 2024: Updated for Eval3

## Disclaimer

Certain commercial equipment, instruments, software, or materials are identified in this document to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor necessarily the best available for the purpose. The descriptions and views contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NIST, DARPA, or the U.S. Government.

# 1. Introduction

The Computational Cultural Understanding (CCU) program is a 36-month research program from the Defense Advanced Research Projects Agency (DARPA) to create human language technologies that will provide effective dialogue assistance to monolingual operators in cross-cultural interactions.[1] CCU prescribes technology development and testing for two Technical Areas (TA): TA1 Sociocultural Analysis and TA2 Cross-Cultural Dialogue Assistance. TA1 technologies are component technologies supportive of the TA2 application and focus on sociocultural norms discovery, cross-cultural emotion recognition, and detection of impactful changes in sociocultural norms and emotions. TA2 is a framework for a sociocultural dialogue assistant to help monolingual operators to have successful interactions in cross-cultural settings.

The National Institute of Standards and Technology (NIST) is tasked to evaluate system performance on TA1 and TA2 research tasks. This document covers the evaluation methodology including evaluation task definitions, metrics, and file formats for the Eval3 evaluation. Section 1.1. summarizes changes implemented for the **Eval3**.

## 1.1. What's New (Eval3 in January 2025)

Eval3 closely follows MiniEval3. This section reviews the important points as well as the updated requirements for the upcoming Eval3.

- (same) Like MiniEval3, LDC will release segmentation information for the evaluation data. These segments are what the annotators saw when they performed the annotations.
- (same) For the valence diarization (VD), arousal diarization (AD), and emotion detection (ED) tasks, performers are required to make at least one submission using the provided LDC segmentation and optionally can make additional submissions without using the provided segments as contrastive submissions. Should TA1 performers choose to additionally process the data unsegmented, they should process the unsegmented data first and the segmented version after. Please refer to Table 1 for the submission requirements based on various system input and output combinations.
- (same) For the norm discovery (ND) and change point detection (CD) tasks, performers can choose to use the provided segments. However, the system output must use its own segmentation since these tasks are assessed and not aligned to any reference annotations. Please refer to Table 1 for the submission requirements based on various system input and output combinations.
- (same) The `<dataset>.system_input.index.tab` file will specify the start and end locations of the evaluated regions. LDC will still provide the entire source files, so TA1 performers can use the context if needed. However, TA1 performers will know exactly which regions are being evaluated if performers choose to also process the data without using the provided segments. See section 2.1.3.
- (same) Gold standard-based evaluation will be used to evaluate the VD, AD, and ED tasks. The system output will be compared against the gold-standard reference annotation. The gold standard annotation will have three independent passes. NIST will combine the three independent passes in various ways in scoring.
- (new) For system output processed from unsegmented evaluation data, NIST will experiment with various merging thresholds for both reference and system segments from no merging to covering the entire evaluated region. Scores will be reported for each threshold.

[1] https://www.darpa.mil/news-events/2021-05-03a

- (same) For system output processed from unsegmented evaluation data, the correct detection will be any amount of overlap; that is, for audio/video > 0 seconds and text > 0 characters. If more than one segment produced by the system overlaps with a reference segment, the segment produced by the system with the highest LLR will be selected as a match.
- (same) ND and CD tasks will be conducted through an assessment-based approach. Performers will process all the files, but only a subset of files will be chosen for assessment. This subset selection will be guided by LDC audit metadata, ensuring a diverse range of conditions are included in the assessment, rather than mirroring the overall distribution of the dataset. Each system-generated norm instance will undergo human review to verify whether it correctly matches the identified category. Although TA1 performers can submit multiple entries for ND and CD, only one submission—selected by the performer—will be evaluated. Additionally, LDC will annotate a small portion of the data, allowing automatic scores to be calculated on this subset for all submissions.
- (same) In addition to the known and hidden norms, performers will be allowed to choose up to 5 additional norms that their systems discovered for assessment. TA1 performers should coordinate with the LDC regarding the schedule. The steps are as follows:
    - LDC releases development data with annotations of known norms.
    - Performers select up to 5 norms (outside of the known norms) that they discovered during the development stage and send LDC the norm definitions. Performers may need to iterate with the LDC to get to the form that will work for their assessors. The amount of assessment is determined by time/resource/budget constraints so it is not guaranteed that all 5 additional norms will be assessed.
    - LDC releases the evaluation data.
    - Performers process the evaluation data and send the output to NIST.
    - LDC releases the identities of the hidden norms.
    - Performers identify which of their discovered norms map to the LDC hidden norms and which map to the additional norms that they previously defined for assessment.
    - NIST performs subselection for assessment.
    - LDC performs the assessment and returns the results to NIST.
    - NIST computes and releases the results.
- (same) We request that all norm IDs follow a 3-digit string format. System discovered norms and not from the LDC defined norms should start with 5 (e.g., 510, 511) so we can easily differentiate them from LDC-defined norms. The IDs should be globally unique across evaluation epochs. That is, TA1 teams should not reuse the same ID for different norms. See section 2.1.4.
- (same) For the ND task, the norm status will be evaluated.
- (same) For the CD task, in addition to the “file_id”, “timestamp” and “llr” fields, systems are required to output two additional fields “direction” and “impact_scalar”, and these fields will be assessed. See section 2.5.4.
- (same) Please note that the “llr” field in ND, ED, CD system output refers to the existence of the norm, emotion, or change point, respectively and nothing else (e.g., not norm status, not change point impact).

- (**new**) TA1 performers will reprocess the previous target languages which include LC1 (Mandarin), LC2 (Spanish), LC3 (Russian), and LC4 (Japanese) as well as process the new target languages which include LC5 (Korean) and LC6 (Turkish). Since performers already have the full LC1, LC2, LC3, and LC4 sets, the LDC will not re-release these datasets. Performers will submit for each language separately. Be sure to use the appropriate dataset name (below) in the submission ID. See "sub_id" in Table 6.
    - LDC2022E22-LDC2023E07-V1 for full LC1
    - LDC2023E24-LDC2024E03-V1 for full LC2
    - LDC2024E18-V1 for full LC3
    - LDC2024E27-V1 for full LC4
    - LDC2025E01-V1 for full LC5
    - LDC2025E02-V1 for full LC6
- (same) TA1 performers are to provide submissions for one of the following ASR/MT conditions on the TA1 data that the TA1 performers believe will yield the best TA1 results:
    - At least one submission using the same ASR/MT pipeline that TA2 SRI chose (<cond>="TA2SRI").

      OR
    - At least one submission using whatever ASR/MT pipeline TA1 performers want (<cond>="SELF")
- (same) The "sys" field of the TA1 submission ID now must contain the prefix seg<i><o> to let NIST know about the input and output segmentation, followed by a hyphen, followed by a short alphanumeric string. See "sub_id" in Table 6 where
    - <i> indicates the input segmentation and can have a value of "U" (unsegmented) or "L" (LDC segmentation)
    - <o> indicates the output segmentation and can have a value of "T" (team segmentation) or "L" (LDC segmentation)
    - For example: for a submission that didn't use any segmentation and output using their own segmentation
      `CCU_P2_segUT-mybestsys_ND_LCC_SELF_LDC1234-V1_20220719_110203`
- (same) TA2 evaluation protocol for the TA2 remains the same. TA2 sends their TA2 system to ARLIS at the date specified in the Eval3 schedule. ARLIS runs subjects through the system. Subjects will fill out a user questionnaire asking about their experience. ARLIS will annotate the recordings for various characteristics to inform about the performance of the TA2 system.

| | | ND / CD | | VD / AD / ED | |
|---|---|---|---|---|---|
| | | System Input | | System Input | |
| | | No segmentation | LDC's segmentation | No segmentation | LDC's segmentation |
| System Output | TA1's segmentation | TA1 performers select which segmentation for system input | | optional | optional |
| | LDC's segmentation | n/a | | n/a | required |

Table 1: Submission input and output requirements

# 2. Evaluation Tasks

## 2.1. TA1 Norm Discovery

The TA1 norm discovery (ND) task is structured as a combination of norm[2] detection and discovery. Systems will be evaluated on detecting known norms and discovering new norms in the data. The new norms can be among the hidden norms or additional norms that the systems found that are neither from the known nor hidden list of norms. Known norms are norms whose identities and exemplar annotations are disclosed in the development set. Hidden norms are norms that exist in the development set but whose identities and annotations are disclosed only after the evaluation period. Additional norms are norms that the system discovered during the development period but are not among the LDC list of known or hidden norm inventory.

Additionally, prior to the evaluation period, performers can choose up to five additional norms that their systems discovered during the development phase and submit their norm definitions for these five norms for potential assessment. During the evaluation period, systems will process the evaluation collection, detecting both known norms and automatically discovering new norms. Performers will submit their system output to NIST. At the end of the evaluation period, the hidden norms along with their exemplar annotations from the development set will be disclosed to the performers. Performers will report a mapping between the performer-defined norms and the disclosed hidden norms.

### 2.1.1. Task Definition

Given a document set, an ND system automatically detects all instances of known norms and discovers new norms within the documents.

### 2.1.2. Evaluation Methodology and Metrics

A subset of the evaluation documents will be selected for assessment. LDC will also annotate a very small number of documents for comparison of the two evaluation models (assessment vs. gold standard references). Please note that this calibration set will only be annotated for known and hidden norms, not performer-defined norms.

#### 2.1.2.1. Assessment Portion

[2] Per Merriam Webster, a norm is a principle of right action binding upon the members of a group and serving to guide, control, or regulate proper and acceptable behavior. For this evaluation, the data provider will provide detailed definitions for the norms of interest. Throughout this document, norm is synonymous to norm category or norm type while a norm instance is an exemplar of a norm category.

For the assessment portion, the primary metric will be Precision (P) per norm category. The precision for each level of a given metadata factor may also be computed. Performance by factor combinations will not be computed because the data amount for each combination is too small.

#### 2.1.2.2. Gold Standard Portion

For the gold standard portion, NIST will experiment with various metrics including Precision (P), Recall (R), F1, Average Precision (AP), mean Average Precision (mAP) among others. The performance for known norms and hidden norms will be reported separately. The performance for each norm will also be reported.

If the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C are used, the distance function $d()$ for the ND task is given in Table 2. Consult Appendix A for further details.

| **Media type** | **Stream coordinates** | **Minimum overlap for correct detection** |
|---|---|---|
| text | character | $IoU_{character}(s_x, r_y) > 0$ |
| audio or video | time (in seconds) | $IoU_{time}(s_x, r_y) > 0$ |

Table 2: Parameters of the distance function for norm instance alignment.

### 2.1.3. System Input

The Linguistic Data Consortium (LDC) will release a package that includes the evaluation documents. These documents come from a wide variety of sources depicting conversational interactions in three modalities: text, audio, video. Performers will be provided with evaluated regions and segmentation information. Performers will only need to detect instances within the given boundaries but can use the entire documents for context if needed. The evaluated regions will be available in the system input index file while the segmentation information will be in the segment file. For details on the segment format, please refer to the README in the LDC data package.

TA1 performers will process the full LC1, LC2, LC3, LC4, LC5, and LC6 evaluation sets. Each language will have its own index file indicating the documents to be processed.

The index file is an ASCII, tab-separated value file with a header row and data row(s) that contains the elements listed in Table 3 and is named `<dataset>.system_input.index.tab`.

| **Field** | **Description** |
|---|---|
| file_id | (string) The ID of the input document to be processed |
| start | (numeric) The start location of the evaluated region, second offset (float) for audio and video and character offset (integer) for text. |
| end | (numeric) The end location of the evaluated region, second offset (float) for audio and video and character offset (integer) for text. |

Table 3: Elements in the system input index file.

Example of system input index file
`LDC2024E18-V1.system_input.index.tab`

```
file_id     start       end
M012345QD   35.6        335.6
M987654BY   500         800
M111111SP   200.9       500.8
```

### 2.1.4. System Output

An ND system will output the information identifying each system-produced norm instance. There should be one output file per input document. The output file is an ASCII, tab-separated value file with a required header row and data row(s) that contains the elements listed in Table 4.

If there is no output for a given input (e.g., failed to download a Tweet because its author had deleted it or nothing was detected from the input file), the system output file should include only the header row with no data rows.

Each output file should be named as:

**<file_id>.tab**

where <file_id> is the corresponding ID of the input document.

| **Field** | **Description** |
|---|---|
| file_id | (string) The ID of the document where the system has found an instance of a norm |
| norm | (string) 3-digit string indicating the norm ID. The ID can be one of the known norm IDs (released by LDC during the development period) or a system-generated ID for norms that the system believes are not among the known norms. System generated ID for norms must start with 5 (e.g., 510, 511) so we can easily differentiate them from LDC-defined norms. |
| start | (numeric) The start location in the document where the norm instance is found. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| end | (numeric) The end location in the document where the norm instance is found. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| status | (string) The indication if the norm was adhered or violated, one of “**adhere**” or “**violate**”. |

| Field | Description |
|---|---|
| llr | (float) A Log Likelihood Ratio (LLR) detection score is the log of the ratio of the probability of the observation being the norm and the probability of observation NOT being the norm. Please note the LLR refers to the existence of the norm, not the norm status. |

Table 4: The required elements in an ND system output file.

Example Norm Discovery System Output File

```
M012345QD.tab

file_id     norm  start end   status      llr
M012345QD   101   12.5  40.8  adhere      0.75
M012345QD   102   50.2  75.4  adhere      0.60
M012345QD   500   88.9  99.7  violate     0.60
```

### 2.1.5. System Output Index File

In addition to the system output files, performers are to include a system output index file to indicate the processing status of the input files. This is to let the scorer know how to differentiate between an input file that is no longer available at processing time (e.g., the user deleted his Tweet before the Tweet was downloaded by a performer) and one that was processed but had no output.

The system output index file is an ASCII, tab-separated value file with a header row and data row(s) that contains the elements listed in Table 5 and should be named as:

```
system_output.index.tab
```

| Field | Description |
|---|---|
| file_id | (string) The ID of the input document |
| is_processed | (boolean) The indication if the input file was processed (“**true**”) or not (“**false**”). |
| message | (text) An optional message to indicate the status of the processed file (e.g., failed to download). Please note that while the message is optional, the column is required. The column will be empty if no message. |
| file_path | (text) The file path pointing to where the system output file resides relatively within the submission file. |

Table 5: The required elements in the system output index file.

Example System Output Index File

```
system_output.index.tab

file_id     is_processed      message                 file_path
M012345QD   true                                      ./M012345QD.tab
```

```
M987654BY    true              no output              ./M987654BY.tab
M111111SP    false             failed to download     ./M111111SP.tab
```

### 2.1.6. Hidden Norm Mapping Output

After the evaluation period has ended, the hidden norms will be disclosed to the performers. The performers will submit a mapping between their system-produced norms to the newly disclosed hidden norms and, if any, their own defined norms. The mapping file permits the most flexible form of mapping (e.g. many-to-one and one-to-many) to deal with possible discrepancy in norm granularity. The mapping file also allows norms previously detected by the system as known norms to be mapped to hidden or performer-defined norms (e.g., performers found that a norm they detected as known may fit better as a hidden norm). While the mapping file allows the known norms to be remapped, this has no effect on the norms that the systems have previously identified as known norms. The mapping file simply allows linking the system-produced norms to the hidden or performer-defined norms. System-produced norms not listed in the mapping file will not be scored, and therefore, the system will neither be penalized nor given credit.[3]

The mapping file is an ASCII, tab-separated value file with a header row and data row(s) that contains the elements listed in Table 6. The mapping file should be named as:

```
nd.map.tab
```

There should be one mapping file for each norm submission file. Please refer to section 3. for more information about the submission file. If more than one mapping file containing the same submission ID is submitted, the mapping file with the latest timestamp will be deemed the final version.

| Field | Description |
|---|---|
| sys_norm | (string) The ID of the system-generated norm |
| ref_norm | (string) The ID of the disclosed hidden or performer-defined norm |

[3] Performers are welcome to pool their results as a way to validate the norms that their systems discovered.

| Field | Description |
|---|---|
| sub_id | (string) The submission ID tells the scorer which norm submission to apply the mapping information. Since each mapping file is tied to one norm submission, each mapping file should contain only one submission ID. The submission ID must contain the following components in the form:<br><br>CCU_<phase>_<sys>_<task>_<team>_<cond>_<dataset>_<timestamp><br><br><phase> := P1 \| P2<br><sys> := seg<i><o> followed by hyphen and short alphanumeric (no underscore) description of the system, where:<br><i> indicates the input segmentation and can have a value of "U" (unsegmented) or "L" (LDC segmentation)<br><o> indicates the output segmentation and can have a value of "T" (team segmentation) or "L" (LDC segmentation)<br><task> := ND \| NDMAP \| VD \| AD \| ED \| CD<br><team> := COL \| PAR<br><cond> :== TA2SRI \| SELF<br><dataset> := <LDC catalog ID>-<version number> (e.g., LDC2022R17-V1)<br><timestamp> := YYYYMMDD_HHMMSS<br><br>For example:<br>CCU_P2_segUT-mybestsys_ND_LCC_SELF_LDC1234-V1_20220719_110203 |

Table 6: The required elements for a performer-provided norm mapping file.

Example Norm Discovery Mapping File

nd.map.tab

```
sys_norm    ref_norm    sub_id
501         110         CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203
502         112         CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203
```

## 2.2. TA1 Valence Diarization

The TA1 Valence Diarization (VD) task focuses on the system's ability to provide continuous valence changes in the document. Valence is used to indicate the polarity of the emotion. For this evaluation, the valence is defined to be an integer between 1 and 1000, with 1 being the most negative and 1000 being the most positive. The reference valence will be independently annotated three times at the segment level where the segmentation points are at the convenience of the annotation process. NIST will experiment with various ways of combining the N-way reference annotations.

The valence value is not a function of any emotion. This means that valence can be low (e.g., 100) or high (e.g., 900) even in the absence of an emotion label.

### 2.2.1. Task Definition

Given a document, a VD system automatically assigns the valence values with a range [1:1000] throughout the document as the emotion polarity changes. Systems must limit valence decisions to individual documents.

### 2.2.2. Evaluation Methodology and Metrics

The primary metric for VD will be the average Concordance Correlation Coefficient (CCC) using the Continuous Variable Diarization evaluation protocol in Appendix B: Continuous Variable Diarization. The performance will be assessed by comparing the system output to the reference. Prior to scoring, the reference and system-produced levels are normalized to z-scores.

The CCC metric evaluates continuous scale valence judgements. The valence values may also be binned into categorical levels to enable the use of agreement metrics such as Cohen's Kappa or other alternative methods if time permits.

### 2.2.3. System Input

Same as section 2.1.3.

### 2.2.4. System Output

A VD system will output valence levels for each document by labeling segments as having a homogeneous valence level. There should be no gap between segments, and the entirety of the document must be accounted for. The output file is an ASCII, tab-separated value file with a required header row and data row(s) that contains the elements listed in Table 7.

If the system could not process the input file (e.g., failed to download a Tweet because its author had deleted it), the system output file should include only the header row with no data rows. The NIST scorer will assign a default value of 500 for valence.

Each output file should be named as:

**<file_id>.tab**

where <file_id> is the corresponding ID of the input document.

| **Field** | **Description** |
|---|---|
| file_id | (string) The ID of the input document |
| start | (numeric) The location in the document where the valence level with a certain value begins. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| end | (numeric) The location in the document where the valence level with a certain value ends. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| valence_continuous | (integer) An integer between 1 and 1000 indicating the emotion polarity with 1 being the most negative and 1000 being the most positive. If the system fails to process the input document as indicated in the `system_output.index.tab`, the scorer will assign a value of 500. |

Table 7: The required elements in a VD system output file.

Example Valence Diarization System Output File
M012345QD.tab

| file_id | start | end | valence_continuous |
|---|---|---|---|
| M012345QD | 0.0 | 10.9 | 350 |
| M012345QD | 10.9 | 35.6 | 500 |
| M012345QD | 35.6 | 59.2 | 900 |
| M012345QD | 59.2 | 78.5 | 1000 |

#### 2.2.5. System Output Index File

Same as section 2.1.5.

### 2.3. TA1 Arousal Diarization

The TA1 Arousal Diarization (AD) task focuses on the system's ability to provide continuous arousal changes in the document. Arousal is used to indicate the intensity of the emotion. For this evaluation, the arousal is defined to be an integer between 1 and 1000, with 1 being the lowest and 1000 being the highest. The reference arousal will be independently annotated three times at the segment level where the segmentation points are at the convenience of the annotation process. NIST will experiment with various ways of combining the N-way reference annotations.

The arousal value is not a function of any emotion. This means that arousal can be low (e.g., 100) or high (e.g., 900) even in the absence of an emotion label.

#### 2.3.1. Task Definition

Given a document, an AD system automatically assigns the arousal values with a range [1:1000] throughout the document as the emotional intensity changes. Systems must limit arousal decisions to individual documents.

#### 2.3.2. Evaluation Methodology and Metrics

The primary metric for AD will be the average Concordance Correlation Coefficient (CCC) using the Continuous Variable Diarization evaluation protocol in Appendix B: Continuous Variable Diarization. The performance will be assessed by comparing the system output to the reference. Prior to scoring, the reference and system-produced levels are normalized to z-scores.

The CCC metric evaluates continuous scale arousal judgements. The arousal values may also be binned into categorical levels to enable the use of agreement metrics such as Cohen's Kappa or other alternative methods if time permits.

#### 2.3.3. System Input

Same as section 2.1.3.

#### 2.3.4. System Output

An AD system will output arousal levels for each document by labeling segments as having a homogeneous arousal level. There should be no gap between segments, and the entirety of the document must be accounted for. The output file an ASCII, tab-separated value file with a required header row and data row(s) that contains the elements listed in Table 8.

If the system could not process the input file (e.g., failed to download a Tweet because its author had deleted it), the system output file should include only the header row with no data rows. The NIST scorer will assign a default value of 1 for arousal.

Each output file should be named as:

**<file_id>.tab**

where <file_id> is the corresponding ID of the input document.

| Field | Description |
|---|---|
| file_id | (string) The ID of the input document |
| start | (numeric) The location in the document where the arousal level with a certain value begins. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| end | (numeric) The location in the document where the arousal level with a certain value ends. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| arousal_continuous | (integer) An integer between 1 and 1000 indicating the emotion intensity with 1 being the lowest and 1000 being the highest. If the system fails to process the input document as indicated in the `system_output.index.tab`, the scorer will assign a value of 1. |

Table 8: The required elements in an AD system output file.

Example Arousal Diarization System Output File

`M012345QD.tab`

```
file_id     start  end   arousal_continuous
M012345QD   0.0    10.9  350
M012345QD   10.9   35.6  500
M012345QD   35.6   59.2  900
M012345QD   59.2   78.5  1000
```

#### 2.3.5. System Output Index File

Same as section 2.1.5.

### 2.4. TA1 Emotion Detection

The TA1 emotion detection (ED) task focuses on the system's ability to detect expressions of emotion and label them as one of eight primary emotions as defined by Plutchik: joy, trust, fear, surprise, sadness, disgust, anger, anticipation.[4] For this evaluation, the emotion category label will be independently

[4] Plutchik, R. (1980). A general psychoevolutionary theory of emotion. In R. Plutchik & H. Kellerman (Eds.), Emotion: Theory, research and experience, Theories of emotion (Vol. 1, pp. 3–33). New York: Academic Press.

annotated three times at the segment level where the segmentation boundaries are at the convenience of the annotation process. NIST will experiment with various ways of combining the N-way reference annotations.

Please note the emotion category is not a function of valence or arousal values. This means that an emotion label is not tied to any valence and arousal and can exist without either.

#### 2.4.1. Task Definition

Given a document, an ED system automatically detects all instances of the eight emotions (joy, trust, fear, surprise, sadness, disgust, anger, anticipation) in the document. Systems must limit emotion decisions to individual documents.

#### 2.4.2. Evaluation Methodology and Metrics

NIST will experiment with various metrics including precision (P), recall (R), F1, Average Precision (AP), mean Average Precision (mAP) among others. The performance for each emotion will also be reported.

If the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C are used, the distance function $d()$ for the ED task is as given in Table 9. Consult Appendix A for further details.

| **Media type** | **Stream coordinates** | **Minimum overlap for correct detection** |
|---|---|---|
| text | character | $IoU_{character}(s_x, r_y) > 0$ |
| audio or video | time (in seconds) | $IoU_{time}(s_x, r_y) > 0$ |

Table 9: Parameters of the distance function for emotion instance alignment.

#### 2.4.3. System Input

Same as section 2.1.3.

#### 2.4.4. System Output

An ED system will output the information identifying each system-produced norm instance. There should be one output file per input document unless the system fails to process the input document. The output file an ASCII, tab-separated value file with a required header row and data row(s) that contains the elements listed Table 10.

If there is no output for a given input (e.g., failed to download a Tweet because its author had deleted it or nothing was detected from the input file), the system output file should include only the header row with no data rows.

Each output file should be named as:

**<file_id>.tab**

.

where <file_id> is the corresponding ID of the input document.

| Field | Description |
|---|---|
| file_id | (string) The ID of the document where the system has found an instance of an emotion |
| emotion | (string) The emotion, one of eight Plutchik's primary emotions: "**anger**", "**fear**", "**sadness**", "**disgust**", "**surprise**", "**anticipation**", "**trust**", and "**joy**" |
| start | (numeric) The begin location in the document where the emotion instance is found. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| end | (numeric) The end location in the document where the emotion instance is found. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| llr | (float) A Log Likelihood Ratio (LLR) detection score is the log of the ratio of the probability of the observation being the emotion and the probability of observation NOT being the emotion. Please note the LLR refers to the existence of the emotion. |

Table 10: The required elements in an ED system output file.

Example Emotion Detection System Output File

```
M012345QD.tab

file_id     emotion     start end   llr
M012345QD   joy         12.9  40.6  0.75
M012345QD   trust       50.4  75.5  0.60
M012345QD   fear        88.1  99.3  0.60
```

#### 2.4.5. System Output Index File

Same as section 2.1.5.

### 2.5. TA1 Change Detection

The TA1 Change Detection (CD) task focuses on the system's ability to automatically detect impactful shifts of sociocultural norms and emotions. The determination of impactful change points is made independent of norm, emotion, valence, and arousal annotations.

#### 2.5.1. Task Definition

Given a document, the CD system automatically detects all points in the document where an "impactful" change occurs. An impactful change is defined to be a point in the document where a change in norm or emotional state can negatively or positively affect the outcome of the interaction. [This will mirror LDC's annotation definition.] Systems must limit change point decisions to individual documents.

### 2.5.2. Evaluation Methodology and Metrics

A subset of the evaluation documents will be selected for assessment using the LDC audit metadata. LDC will also annotate a very small number of documents for comparison of the two evaluation models (assessment vs. gold standard references).

#### 2.5.2.1. Assessment Portion

For the assessment portion, the primary metric will be Precision (P). The precision for each level of a given metadata factor may also be computed. Performance by factor combinations will not be computed because the data amount for each combination is too small.

#### 2.5.2.2. Gold Standard Portion

For the gold standard portion, NIST will experiment with various metrics including Precision (P), Recall (R), F1, Average Precision (AP) among others. The unit of detection is an 'instance' of a change point. In order for a change point to be determined to be correct, the system hypothesized change point must be within +/- a delta distance measured in characters ($\Delta CP_{character}$) for text and time (in seconds) ($\Delta CP_{time}$) for audio and video. Source documents will be triple annotated so the submission will be scored against each reference separately. Thus, the reported measures against each reference will be by signal type (text, audio, or video)[5]:

If the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C are used, the distance function $d()$ for the CD task is given in Table 11. Consult Appendix A: Streaming Instance Detection for further details.

| **Media type** | **Stream coordinates** | **Maximum distance for correct detection** |
|---|---|---|
| text | character | $\Delta CP_{character}(CP_{s_i}, CP_{r_y}) <= 100$ |
| audio or video | time (in seconds) | $\Delta CP_{time}(CP_{s_i}, CP_{r_y}) <= 10$ |

Table 11: Parameters of the distance function parameters for change point instance alignment.

### 2.5.3. System Input

Same as section 2.1.3.

### 2.5.4. System Output

A CD system will output the following information for each instance of an impactful change. There should be one output file per input document unless the system fails to process the input document. The output file an ASCII, tab-separated value file with a required header row and data row(s) that contains the elements listed in Table 12.

If there is no output for a given input (e.g., failed to download a Tweet because its author had deleted it or nothing was detected from the input file), the system output file should include only the header row with no data rows.

[5] CD performance by data type is used as the primary metric because the minimum agreement thresholds are not expected to be similar between the data types. The overlap used for ND and ED encompasses reference instance spans in the calculation so that the overlap thresholds are more or less similar across data types.

Each output file should be named as:

**<file_id>.tab**

where <file_id> is the corresponding ID of the input document.

| Field | Description |
|---|---|
| file_id | (string) The ID of the document |
| timestamp | (numeric) The point location in the document where the change is detected. If the document is text, the value is character offset (integer). If the document is audio or video, the value is time (in seconds) offset (float). |
| llr | (float) A Log Likelihood Ratio (LLR) detection score is the log of the ratio of the probability of the observation being the change point and the probability of observation NOT being the change point. Please note the LLR refers to the existence of the change point, not direction nor impact_scalar. |
| direction_impact | (string) The direction of the change point's potential impact on the conversation, one of "**positive**" or "**negativ**e". |
| strength_impact | (integer) The strength of the change point's potential impact on the conversation, one of **1**, **2**, **3**, or **4** with **1** very weak impact and **4** very strong impact. |

Table 12: The required elements in a CD system output file.

Example Change Detection System Output File

M012345QD.tab

```
file_id     timestamp   llr   direction_impact   strength_impact
M012345QD   12.2        0.75  negative    2
M012345QD   42.6        0.60  positive    3
M012345QD   56.8        0.60  positive    4
```

### 2.5.5. System Output Index File

Same as section 2.1.5.

## 2.6. TA2 Cross-Cultural Dialogue Assistance

TA2 is intended to be a framework for a sociocultural dialogue assistant utilizing component technologies developed in TA1 to assist monolingual operators to have successful interactions in cross-cultural settings. To evaluate this dialogue assistance framework, ARLIS will recruit subjects to interact with each other. Each interaction will be guided by a goal-directed scenario. During the interaction, the monolingual operator can receive assistance from a human cultural interpreter (Ceiling condition), a dialogue assistant/TA2 with machine translation (Operation condition 2), or a machine translation only (Baseline condition). As such, the evaluation of TA2 is structured as a comparison task across the three test conditions (see Table 13) with the goal of exceeding the Baseline condition performance and approaching the Ceiling condition performance.

| Test Condition | Monolingual Operator | | Assistant Role | Foreign Language Speaker |
|---|---|---|---|---|
| Ceiling | Culturally uninformed | Limited/no language ability | Human cultural interpreter | Native speaker |
| Operation 2 | | No language ability | TA2 with MT | |
| Baseline | | No language ability | MT only | |

Table 13: TA2 test conditions

#### 2.6.1. Task Definition

Per the BAA[6], the TA2 system monitors the conversation between two speakers in real-time and utilizes outputs from TA1 components to assist in the detection of misunderstandings and to suggest culturally and socially-appropriate conversational actions for remediation.

#### 2.6.2. Evaluation Methodology and Metrics

The comparison of the conditions will be made relative to two general responses: performance and user satisfaction. The performance of the approach is a direct quantification of the success of the task, i.e., interactional effectiveness. To obtain an assessment of task performance, an independent observer (annotator) will determine to what degree the task was completed. As a secondary measure, an assessment of task performance and approach effectiveness from the users' perspectives will be obtained through a post-trial questionnaire.[4.]

#### 2.6.3. System Input

The TA2 evaluation is a live evaluation, compared to corpus-based evaluations for TA1; therefore, inputs to TA2 are specified by the BAA section I.B Technical Area 2.

#### 2.6.4. System Output

The TA2 evaluation is a live evaluation, compared to corpus-based evaluations for TA1; therefore, primary outputs from TA2 are specified by the BAA. In addition, TA2 systems are expected to log the interaction for post-collection analysis. The analysis of the log outputs will be diagnostic in nature to understand system behavior rather than a measure of performance.

# 3. Submission Protocol

## 3.1. TA1 Tasks

Evaluations of TA1 tasks will follow a "take home" protocol where the data provider (LDC) will send the test data to the performers who, in turn, will send their system output to the evaluator (NIST) for scoring. Please refer to the schedule that will be sent to performers about when the evaluation data will be released, when the system output will be due, and when the results will be reported.

For each task, performers are to package the system output files and system output index file or mapping file into a compressed, tar submission file <SUB_ID>.tgz as follows. Please refer to Table 6 for the components for <SUB_ID>.

[6] https://www.darpa.mil/attachments/HR001121S0024-Amendment02.pdf

```
% mkdir <SUB_ID>
% cp system_output.index.tab <SUB_ID>   ### ND, ED, VD, AD, CD submissions only
% cp <file_id>.tab <SUB_ID>             ### System output for ND, ED, VD, AD, CD submissions only
% cp nd.map.tab <SUB_ID>                ### NDMAP submissions only
% tar zcvf <SUB_ID>.tgz <SUB_ID>
```

Example of Packaging Submission

```
% mkdir CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203

% cp system_output.index.tab CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203

% cp M012345QD.tab CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203

% tar zcvf CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203.tgz CCU_P1_Magic_ND_LCC_SELF_LDC1234-V1_20220719_110203
```

The system output files, system output index file, and mapping file must contain the required information given previously in their respective section.

Performers are required to submit at least one and up to 5 submissions per task for those who wish to compare several versions of their systems on the same dataset. Please note that for ND and CD, only one submission will be assessed. If performers made several submissions for ND and CD tasks, they must tell NIST which submission should be used for assessment. No score feedback will be given except to indicate whether the submission was successfully scored. If a submission did not pass validation or couldn't be scored for any reason, it will be logged in the log file. Submissions that did not pass validation do not count toward the limit.

Performers will be assigned a folder in a Google team drive to deposit their submissions. The Google team drive has the following structure:

**CCU_performer_<team>/**

- **submissions/** — where performers will deposit their submissions
  - <SUB_ID>.tgz
- **logs/** — where submission status and/or error will be given
  - <SUB_ID>.status.txt — submission status
  - <SUB_ID>.validation.txt — validation command and any error messages
  - <SUB_ID>.score.txt — scoring command and any error messages
- **results/**
  - **<SUB_ID>**/ — where scores will be posted
    - validate.sh — validation command
    - score.sh — scoring command
    - instance_alignment.tab — alignment (ND, ED, CD only)
    - segment_diarization.tab — windowing (VD, AD only)
    - scores_by_class.tab — scores by class
    - scores_aggregated.tab — scores across classes

Please refer to Appendix F: Scoring Pipeline Data Flow for more information on the scoring data flow.

## 3.2. TA2

ARLIS will provide TA2 testing outputs to NIST for performance assessments. ARLIS and NIST will coordinate this effort separately.

# 4. References

[1] Steichen and Cox, “A note on the concordance correlation coefficient”, The Stata Journal (2002) 2, Number 2, pp. 183–189.

[2] "Lin’s Concordance Correlation Coefficient”, Real Statistics, https://www.real-statistics.com/reliability/interrater-reliability/lins-concordance-correlation-coefficient/

[3] “A Survey of Methods for Time Series Change Point Detection”, Aminikhanghahi and Cook. Knowl Inf Syst. 2017 May; 51(2): 339–367. doi: 10.1007/s10115-016-0987-z

[4] Post-scenario questionnaire, version August 9, 2022, posted to CCU Confluence T&E section CCU Evaluation page.

# Appendix A: Streaming Instance Detection

A Streaming Instance Detection (SID) system processes a data stream detecting all instances of the sought class within the stream. The stream of data could be video, audio, or text. For simplicity, this appendix will use 'time' within the stream to specify instance locations. The sought class could be a displayed emotion, a displayed cultural norm, a change point, a keyword, or a performed behavior/action. This appendix defines the protocol for evaluating SID systems because the steps are the same across tasks. Task-specific parameters are documented in the individual task sections.

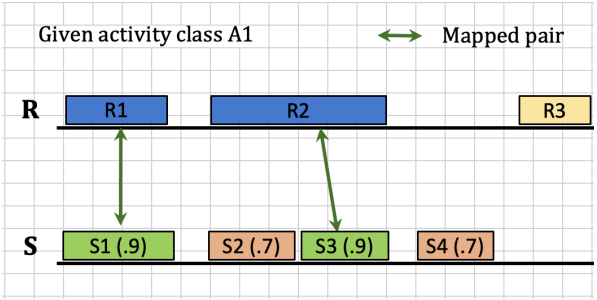


Figure 1: Depiction of finding correct system detections using IoU and detection score. In the system output (S), the first number indicates instance id and the second indicates detection score. For example, S1 (.9) represents the instance S1 with corresponding detection score 0.9. Green indicates TP instances, red for FP instances, and yellow for FN instances.

An instance of the sought class is defined by where it occurs in the stream, a detection score indicating how likely the instance is to have occurred, and other metadata describing the instance. The location within the stream could be a point-in-time (for change point detection), a range of time (for emotion detection), or even a 3-dimensional volume (spatial-temporal activity detection). The choice of coordinate system depends on the requirements of the evaluation task. Since an SID system operates on a stream of data, instances of a class can occur anywhere within the stream, e.g., at any time and for any duration; this considerably complicates the evaluation protocol because the evaluation tool must determine if a detected instance is correct. The 'detection score' can be a variety of measures, e.g., a rank, probability, or calibrated likelihood ratio. The primary role of the detection score within the evaluation is to define an ordering of detected class instances from most-likely to least-likely to have occurred. A common use of the detection score within an application is for a user to review detected instances starting with the highest detection score instance. Other metadata can be associated with the instance depending on the application, e.g., the sub-class.

Performance assessment for a SID system is a three-step process described below (see Figure 1 for an illustration):

- Step 1: Finding the set of correct instances.
- Step 2: Counting instance evaluation labels at specific detection score thresholds.
- Step 3: Computing performance metrics.

**Step 1 - Finding the set of correct instances:** A system-hypothesized instance is declared correct when the instance is sufficiently 'close' to one of the reference instances as determined by an $IoU$ threshold $\delta_{IoU}$. The mapping process results in a list of mapped pairs of system and reference instances that is constrained to a one-to-one mapping. The mapping method in pseudo code is:

- For each norm/emotion/change point:
    - Build a list of potentially mappable system/reference instance pairs:
        - For norms/emotions: If $IoU_{time}(s_x, r_y) >= \delta_{IOU}$
        - For change points: If $\Delta CP_{time}(s_x, r_y) <= \delta_{CP}$
    - Sort pairs by decreasing detection score (ignoring the pair's $IoU()$ or $\Delta CP()$).
    - While the pair list is not empty:
        a. The top pair $(s_x, r_y)$ is added to the correct detection pair list
        b. Remove unused pairs for $s_x$
        c. Remove unused pairs for $r_y$

The resulting pair list is for the entire system output regardless of the detection scores. The next step takes into consideration the detection scores.

**Step 2 - Counting instance evaluation labels at detection thresholds:** The aligned system/reference instance pair list identifies correct detections if a threshold on the detection scores were set to the minimum detection score for the instances. While performance assessment for all instances has importance, in practice, evaluations assess performance at many different detection score thresholds. A detection score threshold (τ) is applied to the pair list to determine the 'evaluation label' for system and reference instances. The labels are:

- For each unique detection score threshold τ
    - 'Correct Detection at τ' ($CD_{\tau,\delta_{IoU}}$) System instances in the pair list with a system detection score >= τ using the IoU threshold $\delta_{IoU}$.
    - 'False Alarm at τ' ($FA_{\tau,\delta_{IoU}}$) System instances **not** in the pair list with a system detection score >= τ using the IoU threshold $\delta_{IoU}$.
    - 'Missed Detection at τ' ($MD_{\tau,\delta_{IoU}}$) Reference instances in the pair list with a system detection score < τ and reference instances not in the pair list using the IoU threshold $\delta_{IoU}$.
    - 'Correct Non-Detection ($CDN_{\tau,\delta_{IoU}}$)' For streaming instance detection systems, only target trials are annotated. Thus, non-target trials are implicit and are not countable. The various performance metrics account for this in different ways. However, for this evaluation, they are not evaluated.

**Step 3 - Computing Performance Metrics:** Using the evaluation labels ($CD_\tau$, $MD_\tau$, $FA_\tau$) assigned in Step 2, any number of performance assessment metrics can be used. Appendix C describes the metrics relevant to SID systems.

# Appendix B: Continuous Variable Diarization

A Continuous Variable Diarization (CVD) system fully partitions a document into segments with a homogeneous rating of the labeling variable. This process is referred to as 'diarization'.[7] The stream of data could be video, audio, or text. For simplicity, this appendix will use 'token' coordinates within the stream to specify locations. The labeled variable may be binary (e.g., speech/non-speech), multi-level ( e.g., nominal valence values or arousal values), or continuous (ratings from 1-100). This appendix defines the protocol for evaluating CVD systems because the steps are the same across evaluation tasks. Evaluation task-specific parameters are documented in the individual task sections.

For the CCU evaluations, there will be 3 independent, continuous label judgments per reference segment while systems will produce a single judgment for a system-defined segmentation. Both segmentations are purely chunking artifacts of the segmentation system used, and the consistency of the segmentations are not reflected in the CVD performance assessment.

CVD systems will be evaluated using the Concordance Correlation Coefficient (CCC) that is a correlation between two measurements of the same continuous variable. The evaluation protocol will explore various ways such as reference averaging to utilize the 3-way reference annotations to account for expected low inter-annotator agreement and segmentation differences between the reference and system output.

Figure 2 is an example 3-way valence reference annotation text file (rendered as light-gray points), the average reference (rendered as a blue point), and the hypothesized system output (rendered as orange points) on a single timeline. The vertical dashed lines are the reference segmentations, and they are included as landmarks.

Reference

| docid | token_start | token_end | rval1 | rval2 | rval3 | rval |
|---|---|---|---|---|---|---|
| doc1 | 1 | 7 | 428 | 701 | 639 | 589 |
| doc1 | 8 | 11 | 793 | 653 | 604 | 683 |
| doc1 | 12 | 18 | 815 | 865 | 929 | 869 |
| doc1 | 19 | 24 | 69 | 174 | 454 | 232 |
| doc1 | 25 | 31 | 280 | 364 | 363 | 335 |
| doc1 | 32 | 35 | 558 | 499 | 503 | 520 |
| doc1 | 36 | 42 | 303 | 457 | 160 | 306 |
| doc1 | 43 | 50 | 884 | 824 | 140 | 616 |

Hypothesis

| docid | token_start | token_end | hval |
|---|---|---|---|
| doc1 | 1 | 14 | 754 |
| doc1 | 15 | 20 | 974 |
| doc1 | 21 | 24 | 572 |
| doc1 | 25 | 31 | 219 |
| doc1 | 32 | 35 | 583 |
| doc1 | 36 | 50 | 158 |

Notes:

- 'rval1' the first reference judgement
- 'rval' is the numeric average of rval1,2,3
- X-axis [0,1] means "start==1"
- Examples use word counts for simplicity. This technique will be extended to time-based measures

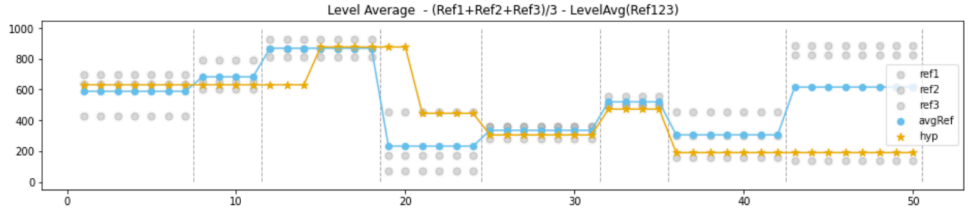


Figure 2: An illustrative example showing 3-way valence reference annotation, the average reference, and a hypothesized system output rendered on a single timeline.

[7] Diarization is derived from the root word /diary/ and is applied to stream processing technologies that label the entirety of the stream with a value for a specific variable. The term has been used in the Rich Transcription evaluations for speech/non-speech, speaker id, music/non-music detection technologies.

Each reference segment record includes a token span, the three judgments (rval1, rval2, and rval3), and the average of the three judgments (rval). The hypothesized system output uses a different segmentation and includes the single judgment per segment (hval).

Performance assessment for CVD systems is a three-step process:

- Step1: Convert the reference and hypothesized system output to discrete time series graphs (as in Figure 2).
- Step 2: Apply level normalizations and tolerance rules (if applicable) to ameliorate the measurement noise caused by low inter-annotator agreement and inconsistent temporal segmentation.
- Step 3: Compute performance metrics.

**Step 1 - Convert reference and hypothesized system output to discrete time-series graph:** The CCC metric, as well as other metrics, compare the label judgments from two sources for the same, single item. CVD systems work on unsegmented source material and developers have the option to report label judgments at a cadence best suited for the system. The same is true for the reference annotation provider. In order to use the contemplated metrics, the label judgments need to be 'discretized' to the same decision units. Text documents and audio/video documents will use different techniques.

**Text-based decision unit**: For text, judgments for characters or word tokens suffice for the decision unit. Thus, the evaluation metrics will use token-level, valence judgments for performance assessment. When the time interval intersects a segment with a 'no-speech' annotation, the decision unit is not evaluated.

**Time-based decision unit**: For audio and video, the continuous time nature requires both a common decision unit discretization step as well as a label level quantization or averaging step. For the decision unit discretization step, the CVD reference and CVD hypothesized segments ($S$) are separately queried at the same fixed cadence, for example, every 2 seconds. During the interval $(t_1, t_2]$, the $AverageLevel$ is computed as the time-averaged rating level during the time interval. This is calculated as a piecewise summation of the segment levels that intersect the time interval query divided by the interval duration. The $AverageLevel$ is then quantized to the nearest label level or used as is, depending on the performance metric used. The formula for The $AverageLevel$ is:

$$AverageLevel(S, t_1, t_2) = \frac{\sum_{i=0}^{\#S} Level(s_i) * TimeOverlap(s_i, t_1, t_2)}{t_2 - t_1}$$

where:

| | |
|---|---|
| $S$ | The set of segment judgments either the reference or the hypothesis |
| $s_i$ | The i[th] segment |
| $Level(s_i)$ | The variable level for $s_i$ |
| $TimeOverlap(s_i, t_1, t_2)$ | The temporal overlap of $s_i$ and the time range $(t_1, t_2]$ |

When the time interval intersects a segment with a 'no-speech' annotation, the decision unit is not evaluated.

**Step 2 - Apply level normalization and tolerance rules:** For categorical performance assessment metrics, such as Cohen's Kappa, there are several methods to infer categorical labels that NIST will explore and report. The variations are:

- 3-Point Valence Polarity Level Normalization - For valence, the measured range varies from negative valence to positive valence. For this normalization, the following rules, or variations, are applied to map the continuous [1-1000] scale to a 3-point scale ['negative', 'neutral', 'positive']. This effectively ignores low-negative and high-positive discrepancies.
    - 'negative' valence → 1-299
    - 'neutral' valence → 300-699
    - 'positive' valence → 700-1000
- 3-Point Arousal Intensity Level Normalization - For arousal, the measured range varies from low to high intensity. For this normalization, the following rules are applied to map the continuous [1-1000] arousal scale to a 3-point scale ['low', 'medium', 'high']. This effectively ignores low- and high-level discrepancies in both the reference and hypothesized system output.
    - 'low' arousal → 1-299
    - 'medium' arousal → 300-699
    - 'high' arousal → 700-1000

**Step 3 - Computing Performance Metrics:** After step 2, a list of decision unit tuples with the average reference and hypothesis judgment, and the quantized level for both the average reference and hypothesis judgments, depending on the level quantization approach. Appendix C describes the metrics relevant to CVD systems which includes Concordance Correlation Coefficient and Cohen's Kappa.

# Appendix C: Measurement Formulas and Performance Metrics

The performance assessment process documented in Appendix A and C consists of three steps (1) finding the set of correctly detected instances, (2) counting instance evaluation labels and (3) computing performance measures. The measurement formulas and performance metrics used for the three steps are documented in this appendix.

**C.1. Preliminary Definitions:**

C.1.1. Detection Score: A detection score is a numeric value for each instance indicating how strong the evidence supporting the system's assertion that the instance of the sought type exists. The value may have range $(-\infty, \infty)$ with more positive numbers indicating stronger evidence.

C.1.2. Confidence Detection Score: A confidence detection score is a detection score constrained to the range $[0, 1]$ with the value representing the posterior probability that the instance of the sought type exists.

C.1.3. Log Likelihood Ratio Detection Score: A Log Likelihood Ratio (LLR) detection score is a detection score based on the ratio of the probability of the observation with the hypothesis it is the sought type and the probability of the observation with the hypothesis it is NOT the sought type.

$$LLR = log\frac{p(observation|type=t)}{p(observation|\ type!=t)}$$

**C.2. Formulas for Finding Correctly Detected Instances**: These formulas are used to find correctly detected instances during the alignment process where comparisons are made for system instance $x$ of type $t$ ($s_{x,t}$) and reference instance $y$ of type $t$ ($r_{y,t}$) to determine if $s_{x,t}$ is a correct detection of. For the purpose of these formulas and metrics, the type is the sought item for the task, e.g., emotion, norm, etc.

C.2.1. Time-based Intersection over Union (IoU)

$$IoU_{Time}(s_{x,t}, r_{y,t}) = \frac{TimeSpan(s_{x,t}) \cap TimeSpan(r_{y,t})}{TimeSpan(s_{x,t}) \cup TimeSpan(r_{y,t})}$$

where:

$TimeSpan()$ = The time span of an instance.

C.2.2. Character-based Intersection over Union

$$IoU_{Character}(s_x, r_y) = \frac{CharSpan(s_{x,t}) \cap CharSpan(r_{y,t})}{CharSpan(s_{x,t}) \cup CharSpan(r_{y,t})}$$

where:

$CharSpan()$ = The character span of an instance. Note this follows the annotation precedent of character offsets as measured in the reference annotations.

C.2.3. Detection Score Congruence

$$DSC(s_{x,t}) = \frac{DS(s_{x,t})}{max(DS(S_t)) - min(DS(S_t))}$$

where:

| | |
|---|---|
| $DS(s_{x,t})$ | The detection score for system instance $x$ of type $t$. |
| $max(DS(s_t))$ | The maximum detection score for the system for instances of type $t$. |
| $min(DS(s_t))$ | The minimum detection score for the system for instances of type $t$. |

C.2.4. Change Point Delta. ($\Delta CP$)

For the Change Point evaluation, the system is required to find a change point within the temporal vicinity of the reference change point. Change Point Delta is the cartesian distance of the system and reference change points:

$$\Delta CP_{character}(CP_{s_i}, CP_{r_y}) = |\, CP_{s_i} - CP_{r_y} \,|$$

Where:

| | |
|---|---|
| $CP_{s_i}$ | The time of system change point $s_i$ |
| $CP_{r_y}$ | The time of reference change point $r_y$ |

**C.3. Formulas for Counting Instances**: The formulas below assume the following counts are available after alignment as described in Appendix A.

C.3.1. Correct Detections at System Detection Score τ

$CD_{\tau,\delta_{IoU}}$ The number of correct detections at the detection score threshold τ for a given $\delta_{IoU}$ instance overlap threshold. A correct detection is also called a 'True Positive'.

C.3.2. False Alarms at System Detection Score τ

$FA_{\tau,\delta_{IoU}}$ The number of false alarm system instances at the detection score threshold τ for a given $\delta_{IoU}$ instance overlap threshold. A False Alarm is also called a 'False Positive'.

C.3.3. Missed Detections at System Detection Score τ

$MD_{\tau,\delta_{IoU}}$ The number of missed detections in the reference annotations at the detection score threshold τ for a given $\delta_{IoU}$ instance overlap threshold. A missed detection is also called a 'False Negative'.

**C.4. Performance Measures**: The formulas below assess the performance of a system in terms of its ability to perform a given evaluation task.

C.4.1. Precision of instance type $t$ at system detection score τ and IoU threshold $\delta_{IoU}$: Precision is the fraction of correct detections (true positives) for instances with detection score >= τ and meeting the IoU threshold.

$$Precision(t, \tau, \delta_{IoU}) = \frac{CD_{t,\tau,\delta_{IoU}}}{CD_{t,\tau,\delta_{IoU}} + FA_{t,\tau,\delta_{IoU}}}$$

Precision is typically computed at retrieval rank depth rather than a function of the detection score. The detection score is used so that the comparisons to detection statistics (e.g., probability of missed detection) use the same thresholding mechanism.

C.4.2. Recall of instance type $t$ at system detection score τ and IoU threshold $\delta_{IoU}$: Recall is the fraction of correct detections (true positives) for instances with detection score >= τ and meeting the IoU threshold to the total number of true instances.

$$Recall(t, \tau, \delta_{IoU}) = \frac{CD_{t,\tau,\delta_{IoU}}}{CD_{t,\tau,\delta_{IoU}} + MD_{t,\tau,\delta_{IoU}}}$$

Recall is typically computed at retrieval rank depth rather than a function of the detection score. The detection score is used so that the comparisons to detection statistics (e.g., probability of missed detection) use the same thresholding mechanism.

C.4.3. Average Precision of instance type $t$ for IoU threshold $\delta_{IoU}$: Average Precision (AP) is defined to be the area under the precision-recall curve through the continuous variable formulation:

$$AP_{t,\delta_{IoU}} = \int_{\tau=max(DS_t)}^{min(DS_t)} Precision(t, \tau, \delta_{IoU}) * Recall(t, \tau, \delta_{IoU})$$

C.4.4. Mean Average Precision for IoU threshold $\delta_{IoU}$: Mean Average Precision (mAP) is the mean of the type Average Precision. The set of types include emotions, known norms, hidden norms, etc.

$$mAP_{\delta_{IoU}} = \frac{1}{N_{type}} \sum_{t=1}^{N_{type}} AP_{t,\delta_{IoU}}$$

C.4.5. Concordance Correlation Coefficient

Lin's Concordance Correlation Coefficient (CCC) is a method of comparing two measurements of the same variable. CCC has range [-1,1] with -1 being full anti-correlation and 1 being full correlation. See [1] and [2]. The sample version of CCC is $r_c$ and computed by two vectors $X$ and $Y$ of paired judgments.

$$r_c = \frac{2\, r\, s_x s_y}{(\bar{x}-\bar{y})^2 + s_x^2 + s_y^2}$$

where:

$r$ ; the correlation coefficient of vectors $X$ and $Y$

$s_x$ ; the sample standard deviation of $X$

$s_y$ ; the sample standard deviation of $Y$

$\bar{x}$ ; the sample standard mean of $X$

$\bar{y}$ ; the sample standard mean of Y

C.4.6. Cohen’s Kappa

Cohen's Kappa ($K$) measures the agreement between two raters who each classify N items into C mutually exclusive categories. The definition of $K$ is:

$$K = \frac{p_o - p_e}{1 - p_e}$$

where:

$p_o =$ rate of agreement

$p_e =$ rate of agreement due to chance

This section will be enhanced later because Cohen’s Kappa may be replaced by a more suitable metric.

# Appendix D: Reference Data Preprocessing

The reference segments for the norms are singly annotated while the segments for emotions are triply annotated. The LDC will select regions of each text/video/audio document to annotate. Regions not annotated by the LDC will be treated as no-score regions for all tasks.

Reference annotations will be transformed from segment-based annotations to instance-based annotations where each expression (of a norm or emotion) is represented as one item to detect. Several reference merging and judgment collapsing techniques will be implemented during the course of the evaluations. Below are initially planned methods. Systems are expected to produce instance detections therefore merging is not necessary.

**Instance Merging** - This method merges adjacent segments if they have the same category (e.g., norm category) or type (e.g., emotion label) and the gap between segments (end of previous segment and start of next segment) is less than some threshold (e.g., 16 seconds for audio/video documents or less than or equal to 160 characters for text documents). Segments annotated as 'no-annot' are interpreted as non-annotated regions, and therefore for the purpose of instance merging, these segments break category continuity.

**Judgment Collapsing by Majority Voting** - This method is for references that have more than one independent judgment with *categorical* values and require collapsing these categorical values into one. For emotion label annotation, the majority emotion decision is determined by applying the following rules to each segment.

Vote Threshold >= 3:

- For 3-way annotation: The emotion is present if three annotators agree. Otherwise we deem the emotion not present and the segment is treated as none.
- For 2-way annotation (if a 3rd is missing): Translate as no-score region
- For 1-way annotation (if a 2nd and 3rd are missing): Translate as no-score region

Vote Threshold >= 2

- For 3-way annotation: The emotion is present if at least two of the three annotators agree. Otherwise we deem the emotion not present and the segment is treated as none.
- For 2-way annotation (if a 3rd is missing): The emotion is present if two annotators agree. Otherwise we deem the emotion not present and the segment is treated as none.
- For 1-way annotation (if a 2nd and 3rd are missing): Translate as a no-score region.

Vote Threshold = 1

- For 3-way annotation: The emotion is present if at least one annotator labels an emotion. Otherwise we deem the emotion not present and the segment is treated as none.
- For 2-way annotation: The emotion is present if at least one annotator labels an emotion. Otherwise we deem the emotion not present and the segment is treated as none.
- For 1-way annotation: The emotion is present if at least one annotator labels an emotion. Otherwise we deem the emotion not present and the segment is treated as none.
- The annotation is used as is

**Judgment Averaging** - This method is for references that have more than one independent judgment with *continuous* values and require collapsing these continuous values into one when there is a common segmentation. For valence/arousal annotation, the final valence/arousal value is determined by:

- For 3-way annotation: Average the three judgments

- For 2-way annotation (if a 3rd is missing): Average the two judgments
- For 1-way annotation (if a 2nd and 3rd are missing): Translate as a no-score region

Example Norm Discovery Reference Preprocessing

For ND, since there is only one annotation pass, only “Instance Merging” will be performed.

Reference Segment Norm Annotations

```
Seg1  0s-10s greeting
Seg2 10s-15s greeting, criticize
Seg3 15s-18s greeting
```

Reference Norm Instances (after applying Instance Merging)

```
 0s-18s greeting
10s-15s criticize
```

Example Emotion Detection Reference Preprocessing

For ED, since there are more than one annotation passes (up to 3), “Judgment Collapsing by Majority Voting” will be applied followed by “Instance Merging”.

Reference Segment Emotion Annotations

```
Seg1  0s-10s sad,   sad+happy, angry      => 2-sad,   1-happy, 1-angry
Seg2 10s-15s sad,   happy+sad, happy+sad  => 3-sad,   2-happy
Seg3 15s-18s angry, angry+joy, angry+joy  => 3-angry, 2-joy
Seg4 18s-23s none,  angry,     joy        => 1-angry, 1-joy, 1-none
Seg5 23s-33s joy,   joy,       joy        => 3-joy
Seg6 33s-43s joy,   joy+angry,            => 2-joy, 1-angry
Seg7 43s-53s joy,      ,                  => 1-joy
Seg8 53s-63s joy+angry,      ,            => 1-angry, 1-joy
Seg9 63s-73s none,      ,                 => 1-none
Seg10 73s-83s noann,      ,                => noann
```

Reference Emotion Instances (after applying Judgment Collapsing by Majority Voting)

```
Seg1  0s-10s sad
Seg2 10s-15s sad+happy
Seg3 15s-18s angry+joy
Seg4 18s-23s none
Seg5 23s-33s joy
Seg6 33s-43s joy
Seg7 43s-53s noann
Seg8 53s-63s noann
Seg9 63s-73s noann
Seg10 73s-83s noann
```

Reference Emotion Instances (after applying Instance Merging)

```
 0s-15s sad
10s-15s happy
15s-18s angry
```

```
15s-18s joy
23s-33s joy
```

Example Valence/Arousal Diarization Reference Preprocessing

For VD and AD, since there are more than one annotation passes (up to 3), “Judgment Averaging” will be applied followed by converting it into a time series as described in Appendix B Step 1.

Reference Segment Valence Annotations:

```
Seg1  0s-10s 156, 178, 165
Seg2 10s-15s 259, 281, 301
Seg3 15s-17.5s 978, 899, 950
Seg4 18s-27.5s 600, 800,
Seg5 28s-38s 978,    ,
```

Reference Valence Annotations (after applying Judgment Averaging)

```
 0s-10s 166.3
10s-15s 280.3
15s-17.5s 942.3
18s-27.5s 700
28s-38s noann
```

Reference Valence Annotations after extending gaps

```
 0s-10s 166.3
10s-15s 280.3
15s-18s 942.3
18s-28s 700
28s-38s noann
```

# Appendix E: Scoring Constants

For quick reference, Table 14 summarizes all the constants used in automatic scoring that are mentioned throughout this document.

| Type | Task | Text | Audio/Video |
|---|---|---|---|
| Agreement Threshold | Norm/ Emotion | Overlap(ref,sys) > 0 characters | Overlap(ref,sys) > 0 seconds |
| | Changepoint | Distance(ref,sys) <= 100 characters | Distance(ref,sys) <= 10 seconds |
| Reference/System Norm Instance Merging (no splitting of reference or system instances) | Norm (same category) | Will be sweeped at some TBD increment from no merging to full file | Will be sweeped at some TBD increment from no merging to full file |
| Reference/System Emotion Instance Merging (no splitting of reference or system instances) | Emotion (same category) | Will be sweeped at some TBD increment from no merging to full file | Will be sweeped at some TBD increment from no merging to full file |
| Reference Diarization Segment Extension (no extension will be done for system segments) | Valence/ Arousal | < 10 characters | < 1 second |
| | | If the above condition is met, the gap is deemed non-consequential and has the value of the first segment. If the above condition is not met, the gap becomes a no-score region. | |
| Judgment Collapsing by Majority Voting | Emotion | • For 3-way annotation: The value chosen is the same for at least two of the three annotators, else it is a no-score region.<br>• For 2-way annotation (if a 3rd is missing): The value chosen is the same for two annotators, else it is a no-score region.<br>• For 1-way annotation (if a 2nd and 3rd are missing): The value is a no-score region. | |
| Judgment Averaging | Valence/ Arousal | • For 3-way annotation: Average the three<br>• For 2-way annotation (if a 3rd is missing): Average the two<br>• For 1-way annotation (if a 2nd and 3rd are missing): Translate as a no-score region | |
| Diarization Window Size | Valence/ Arousal | 1 character | 2 seconds |
| Scaled Metric at MinLLR Collar | Norm/Emotion | <=150 characters | <=15 seconds |

Table 14: Summary of constants used in the scoring process for the current evaluation. This table will be updated if the constants change from evaluation to evaluation.

# Appendix F: Scoring Pipeline Data Flow

Please note that the scoring pipeline will activate when NIST receives the reference data. Figure 3 shows the sequence the submission passes through the scoring pipeline.

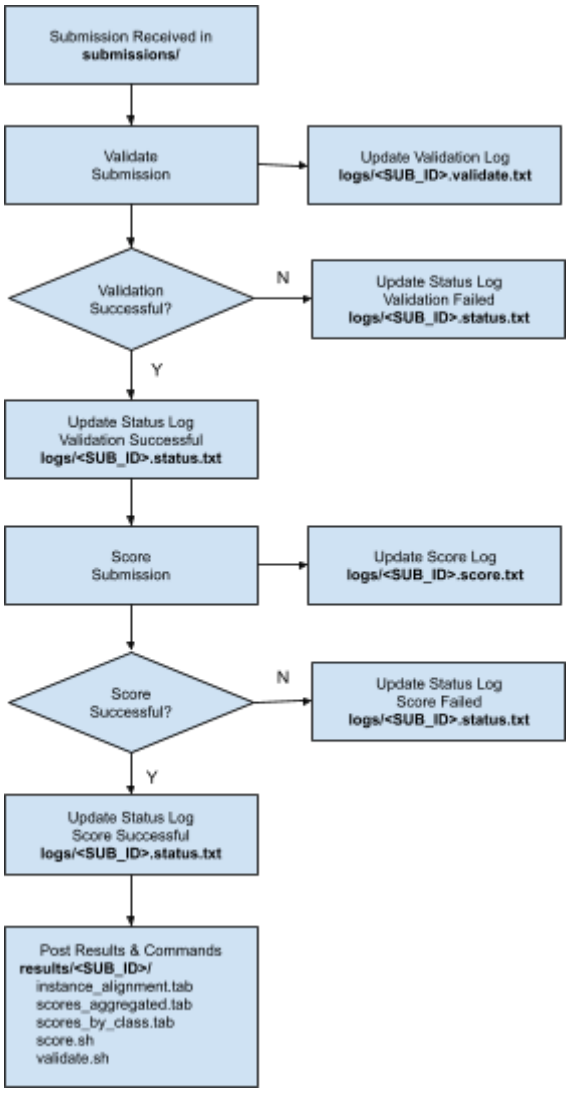


Figure 3: A flow chart showing the action sequence of the scoring pipeline.

```
submissions/
        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236.tgz
logs/
        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236.status.txt
        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236.validate.txt
        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236.score.txt
results/
        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236/
                validate.sh
                score.sh
                instance_alignment.tab
                scores_by_class.tab
                scores_aggregated.tab
```

# Appendix G: Norm Detection Scoring for CCU LC1 Evaluation

**Overview**

The CCU evaluation tool supports many different scoring protocols for all CCU TA1 tasks. What follows is a description of the Norm Detection (ND) evaluation protocols although the same metrics/protocols will be applied to Emotion Detection (ED) systems. NIST will score each ND system for the follow to scenarios:

- Scenario 1: Finding Norms
    - Under this scenario, norms are correctly detected if the norm type is correct and the reference/system instances overlap (i.e., the overlap is non-zero). The norm status is ignored presently. The reference segment annotations will be merged when adjacent segments are of the same norm and the gap is <= 30 seconds or 300 characters.
- Scenario 2: Finding Norms – Optimum System Merge
    - In addition to Scenario 1, NIST will sweep through a set of system merging parameters and report the scores optimizing each system's performance for the 'Scaled F1 at Minimum LLR' described below using a 15 second/150 character overlap forgiveness collar.

The ND performance assessment retains the view that ND is a streaming detection task requiring systems to report the following for each instance: the norm category (e.g., 101, 102, ..), the temporal/text span, and an LLR score. The reported performance assessments will make use of the LLR scores in different ways as follows:

- Mean Average Precision (mAP) – mAP requires a sweep through sorted detections. For mAP, the LLRs are used as a sorting function.
- Minimum LLR – All provided detections are used in the metric calculations; thus, the LLR scores are functionally ignored. This method uses a single LLR threshold; therefore, the triple of Precision, Recall, and F1 are the performance measures reported. Two forms of the triple will be reported: traditional values using instance counts for the computations and a new, temporal overlap-scaled Precision, Recall, and F1 described below.

The traditional Precision, Recall, F1 scores will determine how well a norm is detected regardless of how much the reference overlaps the system output using the Minimum LLR threshold. In this case, a true positive (TP) is counted anytime there is a temporal overlap for both the reference and the system output. A false positive (FP) is indicated whenever the system output has a time associated with it that is not present in the reference (or there is another system segment with higher LLR value that overlaps with the reference segment). The total number of norms is computed whenever a time is indicated in the reference (Ref). An example is shown in Table 15.

| File | ref_beg | ref_end | sys_beg | sys_end | Ref | TP | FP |
|---|---|---|---|---|---|---|---|
| F1 | 30 | 100 | | | 1 | 0 | 0 |
| F2 | 70 | 120 | | | 1 | 0 | 0 |
| F3 | | | 20 | 200 | 0 | | 1 |
| F4 | 25 | 55 | 20 | 60 | 1 | 1 | 0 |
| F5 | 20 | 60 | 25 | 55 | 1 | 1 | 0 |
| F6 | 25 | 55 | 5 | 75 | 1 | 1 | 0 |
| F7 | 5 | 75 | 25 | 55 | 1 | 1 | 0 |
| F8 | 20 | 55 | 25 | 75 | 1 | 1 | 0 |
| F9 | 5 | 55 | 25 | 65 | 1 | 1 | 0 |
| F10 | 25 | 75 | 20 | 55 | 1 | 1 | 0 |
| F11 | 25 | 65 | 5 | 35 | 1 | 1 | 0 |
| Total | | | | | 10 | 8 | 1 |

Table 15: Example reference and system segments to illustrate the scoring process.

For this example, Recall R = TP/Ref = 8/10 = 80%, Precision P = TP/(TP+FP) = 8/9 = 89%, and F1 = 84%.

The overlap-scaled Precision, Recall, and F1 scores take into account how much overlap there is between the reference and the system. TP is now scaled by what portion of the reference is found in the system output allowing for a collar of 15 seconds on either side (that is, subtracting up to 15 seconds from the sys_beg and adding up to 15 seconds to the sys_end[8]. For FP, we need to ensure that the collar does not affect the score by adding time that would cause a false positive, so we choose to add only as much collar as is necessary for the FP calculations. For example, in F5 the system output already totally overlaps the reference, so no additional collar is needed to ensure total TP while the collar would detract from the FP score so no collar is needed.

Table 16 shows how the times have been altered to scale TP and FP. In this table, the sys_beg column shows two numbers separated by a dash. The first number is the original time (from the table above) and the second number is the adjusted time. The same applies for the sys_end column.

[8] For text documents, the collar is 150 characters.

| File | ref_beg | ref_end | sys_beg | sys_end | Ref | Scaled TP | Scaled FP |
|---|---|---|---|---|---|---|---|
| F1 | 30 | 100 | | | 1 | 0 | 0 |
| F2 | 70 | 120 | | | 1 | 0 | 0 |
| F3 | | | 20 | 200 | 0 | | 1 |
| F4 | 25 | 55 | 20 — 20 | 60 — 60 | 1 | 1 | 0.33 |
| F5 | 20 | 60 | 25 — 20 | 55 — 60 | 1 | 1 | 0 |
| F6 | 25 | 55 | 5 — 5 | 75 — 75 | 1 | 1 | 1.33 |
| F7 | 5 | 75 | 25 — 10 | 55 — 70 | 1 | 0.86 | 0 |
| F8 | 20 | 55 | 25 — 20 | 75 — 75 | 1 | 1 | 0.57 |
| F9 | 5 | 55 | 25 — 10 | 65 — 65 | 1 | 0.90 | 0.20 |
| F10 | 25 | 75 | 20 — 20 | 55 — 70 | 1 | 0.90 | 0.10 |
| F11 | 25 | 65 | 5 — 5 | 35 — 50 | 1 | 0.63 | 0.50 |
| Total | | | | | 10 | 7.28 | 4.04 |

Table 16: Adjusted system segment times to illustrate scaled TP and FP.

The TP is now scaled by the portion of overlap between the system and reference, so for F7, scaled TP becomes 60/70. For F11, the overlap is between 25 and 50 so TP = 25/40.

FP is scaled by calculating all regions in the system output that are not in the reference. Looking at F4, there is an extra 5 seconds at both beginning and end of the system output, and the interval is 30; so FP is scaled as FP=10/30. For F11, only one side contributes to FP, so the new FP is 20/40.

This time, scaled R, P, and F1 are 73%, 64% and 68%, respectively. However, the examples show a great deal of overlap between the reference and the system output. In practice, the scores will be much lower after scaling.

Table 16 is also shown in the figure at the end of this document. The shifted system timeline (in dashes) shows the inserted collar of up to 15 seconds.

**Scoring code**

Version 1.2.1 of the Computational Cultural Understanding (CCU) Evaluation Validation and Scoring Toolkit available https://github.com/usnistgov/ccu_validation_scoring . Please git pull for the latest version. The example above is a test case in the release and can be used after the library dependencies are installed. To run the test, execute the commands below. Note the options at the end of the command set the specific parameters for this evaluation.

```
% cd ccu_validation_scoring
% mkdir -p outdir
% python -m CCU_validation_scoring.cli score-nd \
       -ref test/reference/WeightedF1 \
       -s test/pass_submissions/pass_submissions_WeightedF1/ND/system1 \
       -i test/reference/WeightedF1/index_files/WeightedF1.scoring_input.index.tab \
       -o outdir \
       -n test/known_norms_WeightedF1.txt \
       -aR 30 -xR 300 -vR class -t intersection:gt:0 -aC 15 -xC 150 -d
```

The following commands extracts the various scores and reviews scoring output:

# Get the Average Precision for Norm 101

```
% grep '101.*all' outdir/scores_by_class.tab  | grep average_precision
```

# Get the traditional F1 at the Minimum LLR for Norm 101

```
% grep '101.*all' outdir/scores_by_class.tab  | grep f1_at_MinLLR | grep -v scaled
```

#Get the traditional Scaled F1 at the Minimum LLR for Norm 101

```
% grep '101.*all' outdir/scores_by_class.tab  | grep f1_at_MinLLR | grep scaled
```

# To see the reference annotations as read

```
cat outdir/inputs.ref.read.tab
```

# To see the reference annotations as scored (after transformations)

```
cat outdir/inputs.ref.scored.tab
```

# To see the reference annotations as read

```
cat outdir/inputs.sys.read.tab
```

# To see the reference annotations as scored (after transformations)

```
cat outdir/inputs.sys.scored.tab
```

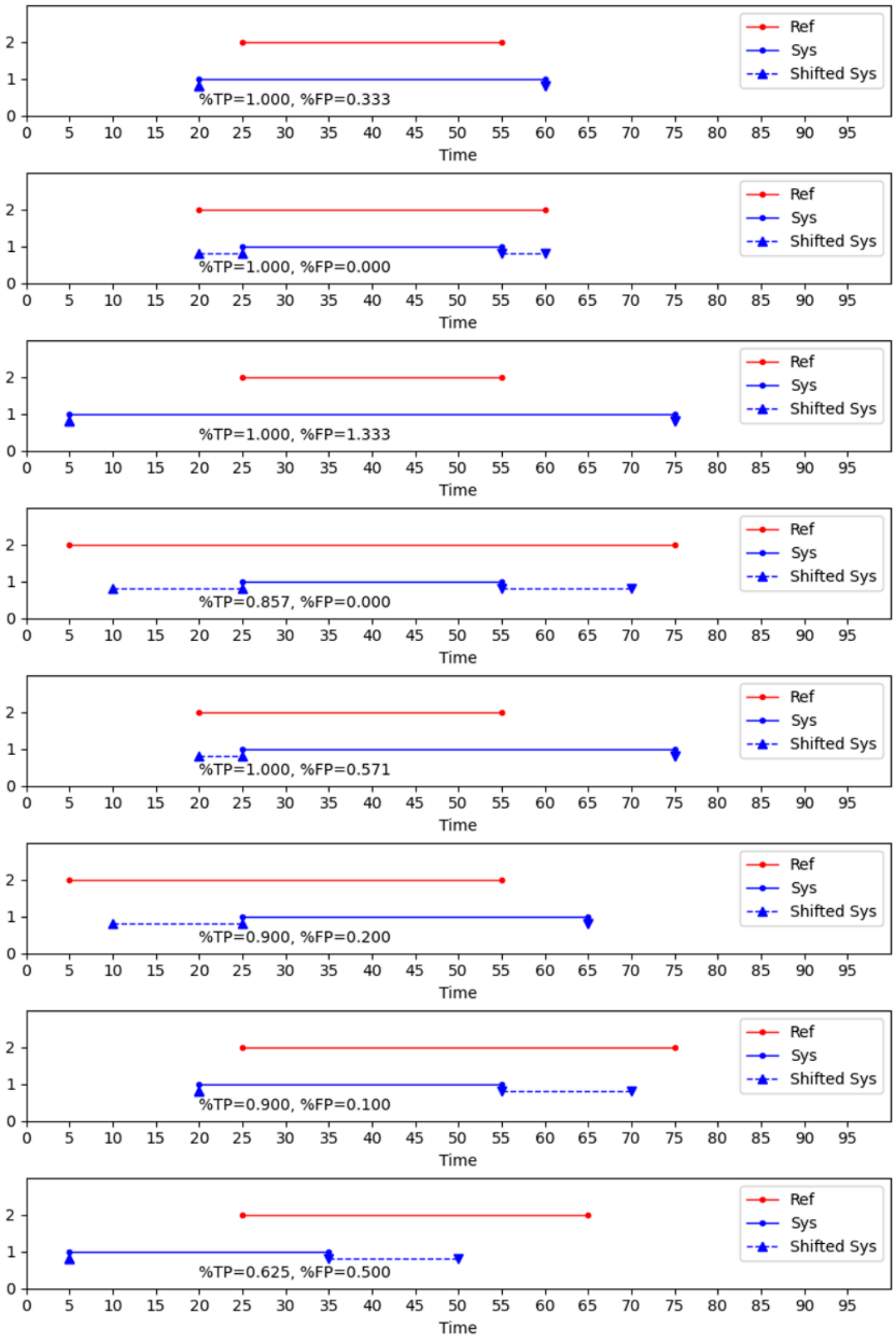
Ref
Sys
Shifted Sys
%TP=1.000, %FP=0.333
%TP=1.000, %FP=0.000
%TP=1.000, %FP=1.333
%TP=0.857, %FP=0.000
%TP=1.000, %FP=0.571
%TP=0.900, %FP=0.200
%TP=0.900, %FP=0.100
%TP=0.625, %FP=0.500
Time

Figure 4: Graphical representation of segments listed in Table 16.