Phase 1 Evaluation Plan for Computational Cultural Understanding Program

Last update: February 21, 2023

Jonathan Fiscus, Audrey Tong, Jennifer Yu, Kay Peterson

National Institute of Standards and Technology

Contact: nist_ccu@nist.gov

Revision History

- July 15, 2022: Initial version; released to CCU T&E
- July 28, 2022: Released to CCU T&E
    - Triple annotations are merged before scoring rather than averaging over the separate references.
    - The instance scoring protocol in Appendix A now uses a fast match algorithm rather than a bipartite graph minimization algorithm.
    - We added index files for system input and output.
    - We will treat non-annotated segments as no-score regions.
    - The Valence and Arousal systems will produce numeric scores between 1 and 1000 rather than discrete labels.
- August 10, 2022: Released to performers
    - Fixed typos.
- August 12, 2022: Released to CCU T&E
    - Added content for TA2 evaluation.
- August 24, 2022: Released to performers
    - We now refer to undisclosed norms as “hidden” rather than “latent.”
    - We allow a more flexible mapping of system-produced norms to hidden norms.
    - Fixed typos.
- October 3, 2022:
    - We now refer to “norm” as “norm category” to ensure consistent terminology.
    - Included a notice about segmentation of text data.
    - Clarified how emotion annotations will be collapsed (Appendix D).
    - Clarified how gaps between reference segments will be handled (Appendix D).
    - Clarified that unannotated regions will be treated as no-score regions (Appendix D).
    - Included a table to summarize all the scoring constants that will be used (Appendix E).
- October 26, 2022:
    - Updated submission protocol for TA1 evaluation (each team gets its own Google Team Drive).
    - Added scoring pipeline data flow to show how submissions will pass through the NIST scoring pipeline (Appendix F).
- November 3, 2022:
    - Clarified how to make a submission file (section 3.1).
- December 1, 2022: Addressed WERB reviewer’s comments.
- February 21, 2023: Minor formatting edits for public release compliance.

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 new 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.

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 and detecting hidden norms.[3] 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 not disclosed during the development period but are discovered by the system during development.

During the evaluation period, systems will process the evaluation collection, detecting both known norms and automatically discovering and detecting putative 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-discovered norms and the disclosed hidden norms. Known and hidden norms will be evaluated using the same methodology, but their scores will be reported separately.

2.1.1. Task Definition

Given a document set, an ND system automatically detects all instances of known norms and discovers and detects all instances of hidden norms within the documents. The systems are allowed to use both the development data and evaluation documents to discover hidden norms.

2.1.2. Evaluation Metrics

The primary metric for ND will be Mean Average Precision (mAP) averaged over norms using the single Intersection over Union (IoU) threshold value of 0.2. The performance for known norms and hidden norms will be reported separately. The performance for each norm will also be reported. Thus, the reported metrics are:

[1] https://www.darpa.mil/news-events/2021-05-03a
[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.
[3] Throughout this document, norm is synonymous to norm category or norm type, and norm instance is an exemplar of a norm category.

- $ND_{K_k}\ AP$ -> $AveragePrecision$ of Known Norm K_k
- $ND_{L_l} AP$ -> $AveragePrecision$ of Hidden Norm L_l
- $ND_K\ mAP$ -> $AveragePrecision$ averaged over the Known norms ($Mean(ND_{K_k}\ AP)$)
- $ND_L\ mAP$ -> $AveragePrecision$ averaged over the Hidden norms ($Mean(ND_{L_l}\ AP)$)

The performance will be computed using the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C. The distance function $d()$ for the ND task is given in Table 1. Consult Appendix A for further details.

Media type	**Stream coordinates**	**Minimum overlap for correct detection**
text	character	$IoU_{character}(s_x, r_y) > 0.2$
audio or video	time (in seconds)	$IoU_{time}(s_x, r_y) > 0.2$

Table 1: Parameters of the distance function for norm instance alignment.

NIST will release the scoring tool to performers. The tool will be able to compute performance at additional user-specified IoU thresholds.

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. The text data will come pre-segmented as part of the LDC automated text processing pipeline and may not correspond to the segments used in annotations. The audio and video data will come unsegmented. Please refer to the README in the LDC data package for information.

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 2 and is named `system_input.index.tab`.

Field	**Description**
file_id	(string) The ID of the input document to be processed
type	(string) The modality of the input document, one of "**text**", "**audio**", or "**video**"
file_path	(string) The file path of the document in the data package
length	(numeric) The length of the document. If the document is text, the length is the character count including spaces. If the document is audio or video, the length is the duration (in seconds).

Table 2: Elements in the system input index file.

Example of system input index file

```
system_input.index.tab

file_id     type        file_path                           length
M012345QD   text        ./data/text/M012345QD.ltf.xml       250
M987654BY   audio       ./data/audio/M987654BY.flac.ldcc    306.7
M111111SP   video       ./data/video/M111111SP.mp4.ldcc     500.1
```

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 3.

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) The ID of the norm. 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.
start	(numeric) The start location in the document where the norm instance is found. If the document is text, the value is character offset. If the document is audio or video, the value is time (in seconds) offset.
end	(numeric) The end location in the document where the norm instance is found. If the document is text, the value is character offset. If the document is audio or video, the value is time (in seconds) offset.
status	(string) The indication if the norm was adhered or violated, one of "**adhere**" or "**violate**". Presently, this is not evaluated.
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.

Table 3: 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   001    12     40     adhere      0.75
M012345QD   001    50     75     adhere      0.60
M012345QD   A1     88     99     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 4 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 within the submission file.

Table 4: 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                                     ./out/M012345QD.tab
M987654BY   true              no output              ./out/M987654BY.tab
M111111SP   false             failed to download     ./out/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. 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 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 scores for norms that the systems have identified as known norms. The mapping file simply allows the scorer to link the system-produced norms to the hidden norms. System-produced norms not listed in the mapping file will not be scored, and therefore, the system will not be penalized or given credit.[4]

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 5 per T&E-provided hidden norms. The mapping file should be named as:

```
nd.map.tab
```

There should be one mapping file for each norm submission file. Please refer to the Submission Protocol section 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 norm (from LDC)
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: CCU_<phase>_<technical_area>_<task>_<team>_<dataset>_<timestamp> <phase> := P1 | P2 <technical_area> := TA1 | TA2 <task> := ND | NDMAP | VD | AD | ED | CD <team> := COL| LCC | MON | PAR | … <dataset> := <LDC catalog ID>-<version number> (e.g., LDC2022R17-V1) <timestamp> := YYYYMMDD_HHMMSS

Table 5: The required elements for a performer-provided norm mapping file.

Example Norm Discovery Mapping File

nd.map.tab

```
sys_norm    ref_norm    sub_id
A1          005         CCU_P1_TA1_ND_LCC_mini-eval1_20220719_110203
A3          042         CCU_P1_TA1_ND_LCC_mini-eval1_20220719_110203
```

[4] Performers are welcome to pool their results as a way to validate the norms that their systems discovered but not annotated by the data provider.

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. The N-way segment annotations will be converted to a single reference annotation using the "Judgment Averaging" procedure defined in Appendix D.

The valence value is not a function of any emotion. This means that valence can be 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. The performance will be assessed by comparing the system output to the average judgment reference. The reported metric is:

- Average Valence CCC_{AvgRef}

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 6.

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. If the document is audio or video, the value is time (in seconds) offset.
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. If the document is audio or video, the value is time (in seconds) offset.
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 6: 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     10    350
M012345QD   10    35    500
M012345QD   35    59    900
M012345QD   59    78    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. The N-way segment annotations will be converted to a single reference annotation using the "Judgment Averaging" procedure defined in Appendix D.

The arousal value is not a function of any emotion. This means that arousal can be 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. The performance will be assessed by comparing the system output to the average judgment reference. The reported metric is:

- Average Arousal CCC_{AvgRef}

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 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 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. If the document is audio or video, the value is time (in seconds) offset.
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. If the document is audio or video, the value is time (in seconds) offset.
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 7: 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	10	350
M012345QD	10	35	500
M012345QD	35	59	900
M012345QD	59	78	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.[5] For this evaluation, the emotion category label will be independently annotated three times at the segment level where the segmentation boundaries are at the convenience of the annotation process. The N-way segment annotations will be converted to a single emotion instance reference for scoring.

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 in the document. Systems must limit emotion decisions to individual documents.

2.4.2. Evaluation Methodology and Metrics

The primary metric for ED will be Mean Average Precision (mAP) for the emotion instance reference using the single Intersection over Union (IoU) threshold value of 0.2. The unit of detection is an 'instance' of an emotion which is a span exhibiting the emotion. The evaluation tool will convert the segment-based annotations into 'instance' annotations by reducing the N-way annotations using the "Judgment Collapsing by Majority Voting" steps in Appendix D and then merging the spans of adjacent segments with the same emotion and applying "Instance Merging" as defined in Appendix D. NIST will report AP for each emotion and Mean Average Precision over the eight emotions. The reported metrics are:

- $ED_{E_j}AP$ -> $AveragePrecision$ of emotion E_j
- $ED\ mAP$ -> $AveragePrecision$ averaged over the eight emotions ($Mean(ED_{E_j}AP)$)

The performance will be computed with the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C. The distance function $d()$ for the ED task is as given in Table 8. Consult Appendix A for further details.

[5] 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.

.

Media type	Stream coordinates	Minimum overlap for correct detection
text	character	$IoU_{character}(s_x, r_y) > 0.2$
audio or video	time (in seconds)	$IoU_{time}(s_x, r_y) > 0.2$

Table 8: 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 9.

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. If the document is audio or video, the value is time (in seconds) offset.
end	(numeric) The end location in the document where the emotion instance is found. If the document is text, the value is character offset. If the document is audio or video, the value is time (in seconds) offset.
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.

Table 9: 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	40	0.75
M012345QD	trust	50	75	0.60
M012345QD	fear	88	99	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. [This will mirror LDC's annotation definition.]

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

The primary metric for CD will be Average Precision (AP) by data type (text, audio, or video). 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) (ΔCP_{time}) for audio and video. Source documents will be singly annotated. The reported measures will be by signal type (text, audio, or video):[6]

- $CD_{text}AP$ -> $AveragePrecision$ for text documents
- $CD_{audio}AP$ -> $AveragePrecision$ for audio documents
- $CD_{video}AP$ -> $AveragePrecision$ for video documents

The performance will be computed with the "Streaming Instance Detection" protocol found in Appendix A and with the metrics defined in Appendix C. The distance function $d()$ for the CD task is given in Table 10. Consult Appendix A for further details.

[6] 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 IoU used for ND and ED encompasses reference instance spans in the calculation so that the IoU thresholds are more or less similar across data types.

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 10: 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 11.

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
timestamp	(numeric) The point location in the document where the change is detected. If the document is text, the value is character offset. If the document is audio or video, the value is time (in seconds) offset.
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.

Table 11: The required elements in a CD system output file.

Example Change Detection System Output File

M012345QD.tab

```
file_id     timestamp   llr
M012345QD   12.2        0.75
M012345QD   42.6        0.60
M012345QD   56.8        0.60
```

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 alone (Operation condition 1), 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 four test conditions (see Table 12) 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 1		Limited language ability	TA2	
Operation 2		No language ability	TA2 with MT	
Baseline		No language ability	MT only	

Table 12: TA2 test conditions

2.6.1. Task Definition

Per the BAA[7], 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 TA2 evaluation will employ a variation of the PARADISE framework.[4] The overall metric will be a linear combination of task success and dialogue costs. For mini-eval1, only task success will be measured while dialogue costs are being deferred to future evaluations.

Task success will be approximated by the responses from a subset of questions obtained from the post-scenario questionnaire that both speakers will complete after each interaction. The full questionnaire is given in [5]. In particular, questions 1-5 and 8-11 from the questionnaire will be used to approximate task success. The responses to these questions will be compared against the ideal responses to a perfect scenario where each of these questions would get a “Completely Agree” rating. Cohen’s Kappa will be used to compare the responses given by the speakers against the ideal responses of a perfect scenario.

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

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 TA2 delivery will include a mechanism (e.g., a shell script) to extract TA1-style outputs from the log for all the TA1 evaluation tasks (see section 2.1 - 2.5). The analysis of the 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 5 for the components for <SUB_ID>.

```
% 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>
```

```
% mkdir CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236
% cp system_output.index.tab CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236
% cp M012345QD.tab CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236
% tar zcvf CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236.tgz \
                                        CCU_P1_TA1_ND_NIST_LDC2022R17-V1_20220531_050236
```

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. 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 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] Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, and Alicia Abella. 1997. PARADISE: A Framework for Evaluating Spoken Dialogue Agents. In 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, pages 271–280, Madrid, Spain. Association for Computational Linguistics.

[5] 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.

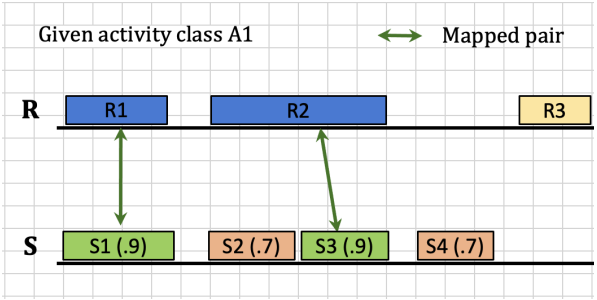

Figure 1: Depiction of finding correct system detections using IoU and detection score. In the system output (S), the first number indicates instance id and the second indicates detection score. For example, S1 (.9) represents the instance S1 with corresponding detection score 0.9. Green indicates TP 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 δ_{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 δ_{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 δ_{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 δ_{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_τ, MD_τ, FA_τ) 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'.[8] 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 use the average of 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

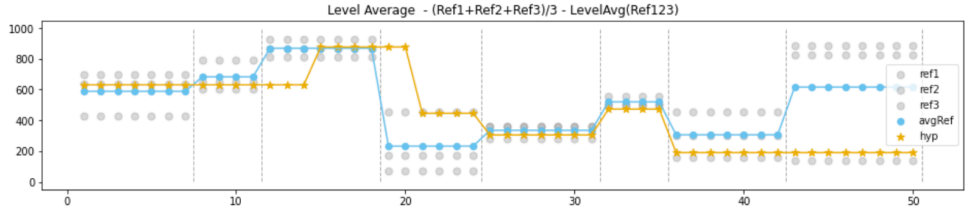

Figure 2: An illustrative example showing 3-way valence reference annotation, the average reference, and a hypothesized system output rendered on a single timeline.

[8] 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. (ΔCP)

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 δ_{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 δ_{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 δ_{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 δ_{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 δ_{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 δ_{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 δ_{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 1 second for audio/video documents or less than 10 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. The emotion is present if:

- For 3-way annotation: At least two of the three annotators agree
- For 2-way annotation (if a 3rd is missing): If two annotators agree
- For 1-way annotation (if a 2nd and 3rd are missing): Translate as a no-score region

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

<u>Example Norm Discovery Reference Preprocessing</u>

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
Seg6 33s-43s joy
Seg5 23s-33s joy
Seg7 43s-53s noann
Seg8 63s-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 easy reference, Table 13 lists all the constants mentioned throughout this document.

Type	Task	Text	Audio/Video
Agreement Threshold	Norm/ Emotion	IoU(ref,sys) <= 0.2	IoU(ref,sys) <= 0.2
	Changepoint	Distance(ref,sys) <= 100 characters	Distance(ref,sys) <= 10 seconds
Reference Norm and Emotion Instance Merging (no merging will be done for system instances)	Norm/ Emotion (same category)	< 10 characters	< 1 second
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. • 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. • For 1-way annotation (if a 2nd and 3rd are missing): The value is a no-score region.	
	Valence/ Arousal	• For 3-way annotation: Average the three • For 2-way annotation (if a 3rd is missing): Average the two • 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

Table 13: Summary of constants used in the scoring process.

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.

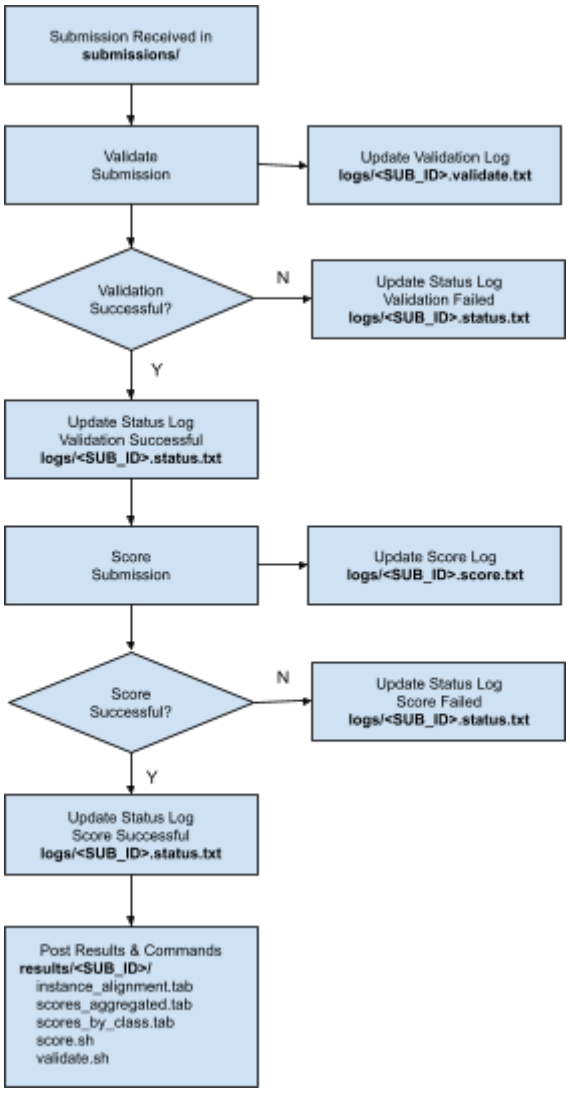

Figure 3: A flow chart showing the action sequence of the scoring pipeline.

```
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
```