Evaluation Plan for
Computational Cultural Understanding Program
Open Evaluation

Last update: April 16, 2024

Audrey Tong, Jonathan Fiscus, Jennifer Yu, Kay Peterson

National Institute of Standards and Technology

Contact: nist_ccu@nist.gov

Revision History

- Jan 16, 2024: Initial published version
- Apr 16, 2024: Postpone pilot evaluation from May until July. Please see the schedule for the pilot evaluation dates.

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 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 consists of 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 while TA2 is a framework for a sociocultural dialogue assistant to help monolingual operators have successful interactions in cross-cultural settings. The National Institute of Standards and Technology (NIST) is organizing a smaller scale evaluation and inviting researchers outside of the CCU program to participate in a particular technology development in CCU. This first open evaluation focuses on detection of social norms in video recordings of interactions between two or more people in Mandarin Chinese. The evaluation is being offered as a track in the NIST TREC (Text REtrieval Conference)[2]. Participants in this track are required to write a paper about their system and attend the TREC Workshop to discuss their system and results. This document covers the evaluation task, metrics, data, evaluation protocol, and schedule.

2. Evaluation Task - Sociocultural Norm Detection

Sociocultural norms are implicit rules of behavior that are generally well-understood and agreed upon within a group, community, society, and/or culture. When communicating with people from an unfamiliar culture, violating cultural norms during an interaction can derail the interaction and may cause offense that may lead to disastrous consequences. Analytics that can detect a norm in progress and inform if a violation has occurred can improve communicative understanding. The task for this evaluation is to detect whether a set of predefined norms exist in the given data and, if detected, determine if the actions in the data adhere or violate the norms in question. The norms under evaluation are listed in Table 1.

Norm ID	Norm Name	Norm ID	Norm Name
101	Doing Apology	106	Doing Thanks
102	Doing Criticism	107	Doing Taking Leave
103	Doing Greeting	108	Doing Admiration
104	Doing Request	109	Doing Finalizing Negotiation/Deal
105	Doing Persuasion	110	Doing Refusing a Request

Table 1: Norm inventory

3. Evaluation Data

The data used for development and testing consist of videos harvested from the internet from a wide variety of sources depicting conversational interactions between two or more people in Mandarin Chinese. The data were segmented into chunks to facilitate annotation, and these chunks have no

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

linguistic meaning (e.g., not a turn, not a sentence, not a complete thought, etc.). The datasets include the source videos in mp4 format, the norm definitions, and the reference annotations illustrating the given norms. These reference annotations are to be taken as the “ground-truths". Both the development and evaluation data will be released by the Linguistic Data Consortium (LDC) under a license agreement governing the use of the data. Participants will receive the following datasets (Table 2) throughout the evaluation cycle.

Dataset	**Size**
Development	~1200 recordings
Pilot evaluation	~1200 recordings
Formal evaluation	~2400 recordings

Table 2: Open CCU dataset inventory.

Upon registration and license agreement approval, the LDC will release the development dataset for model development and internal testing. Participants are free to use additional data of their choice to enhance their models.

The LDC will provide the pilot evaluation dataset to the participants a few days before the pilot evaluation period. This dataset has similar characteristics as the development dataset and will be used to get participants familiarized with the submission process as well as to serve as a checkpoint on their progress on unseen data. After the pilot evaluation, the reference annotation for this dataset will be released to participants for further system development and error analysis.

The evaluation dataset will be sent to participants a few days prior to the evaluation period. Again, this dataset has similar characteristics as the previous two datasets. See Section 10 for the schedule on when the various datasets will be distributed.

4. Metrics

Precision and recall at minimum LLR (log likelihood ratio) will be computed for each norm in the norm inventory listed in table 1. Systems are welcome to make predictions for norms outside of the norm inventory, but these norms will not be scored.

Precision (P) and recall (R) at a given LLR threshold are defined as follows, respectively. Refer to Section 11 to see how P and R are calculated with a simple example.

$$P_{LLR} = \frac{True\ Positive_{LLR}}{(True\ Positive_{LLR} + False\ Positive_{LLR})}$$

$$R_{LLR} = \frac{True\ Positive_{LLR}}{(True\ Positive_{LLR} + False\ Negative_{LLR})}$$

Separately, a precision-recall (P-R) trade-off curve will be computed by sweeping through all the LLR thresholds for each norm, and the area under this curve is computed as Average Precision (AP). The mean Average Precision (mAP) is also computed.

5. System Input

The input to the norm detection (ND) system will be a set of video files listed in the system input index file along with a segmentation file indicating the evaluated regions. While the full length videos are provided, only regions indicated in the segmentation file will be evaluated. The mp4 video files contain both visual and audio signals. Participants can use either or both to assist in the detection of norms.

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

Field	**Description**
file_id	(string) The ID of the input file to be processed

Table 3: Element in the system input index file.

Example of system input index file

```
system_input.index.tab

file_id
M111111SP
M222222AB
M333333AB
…
```

The segmentation file (named `segments.tab`) is an ASCII, tab-separated value file with a header and data rows with the elements listed in Table 4.

Field	**Description**
file_id	(string) The ID of the input video
segment_id	(string) The ID of the segment
start	(float) The start time of this segment (in seconds)
end	(float) The end time of this segment (in seconds)

Table 4: Elements in the segmentation file.

Example of segmentation file

```
segments.tab
file_id     segment_id         start        end
M111111SP   M111111SP_0001     24.5         39.5
M111111SP   M111111SP_0002     39.5         50.2
…
```

6. System Output

The output from the norm detection system is an ASCII, tab-separated value file with a header row and data row(s) that contains the elements listed in Table 5. For each input file, there should be one output file. Each record of the output file corresponds to a detected norm. If there is no output for a given input, 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 input video
segment_id	(string) The ID of the segment
norm	(string) A 3-digit string from Table 1 indicating the norm ID found in this segment
status	(string) An indication if the norm was adhered or violated, one of "**adhere**" or "**violate**"
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 5: Elements in a norm detection system output file.

Example Norm Detection System Output File

```
M012345QD.tab

file_id     segment_id        norm  status      llr
M111111SP   M111111SP_0001    103   adhere      0.75
M111111SP   M111111SP_0002    102   violate     0.80
M111111SP   M111111SP_0002    104   violate     0.60
M111111SP   M111111SP_0005    101   adhere      0.56
M111111SP   M111111SP_0006    106   adhere      0.65
M111111SP   M111111SP_0007    107   adhere      0.90
…
```

The example above shows the system identified norm 101 in segment M111111SP_0001, norms 102 and 104 in segment M111111SP_0002, norm 101, 106, 107 in segment M111111SP_0005, M111111SP_0006, M111111SP_0007, respectively. The gap between segment M111111SP_0002 and M111111SP_005 indicates that the system did not find any norm in segments M111111SP_0003 and M111111SP_0004.

In addition to the system output files, participants 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 cannot be processed (due to whatever reason) and one that was processed but had no output. There should be one record in the system output index file for each record in the system input index file.

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 6 and should be named as:

```
system_output.index.tab
```

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

Table 6: Elements in the system output index file.

Example System Output Index File

```
system_output.index.tab

file_id     is_processed      message                  file_path
M111111SP   true                                       ./M111111SP.tab
M222222AB   true              no output                ./M222222AB.tab
M333333AB   false             failed to process        ./M333333AB.tab
…
```

7. Evaluation Protocol

The evaluation will be conducted over a secured web server. Interested researchers must sign up to participate by creating an evaluation account at https://sat.nist.gov/openccu. After the account is created, participants can complete a data license agreement. Once the agreement is verified, participants will receive instructions on how to get the development data.

There will be two evaluation events: a pilot evaluation and a formal evaluation. The pilot evaluation intends to familiarize participants with the submission protocol and also to give participants feedback on the current performance of their system on unseen data. The fully annotated data for the pilot evaluation will be released after the pilot evaluation is completed to give participants additional data for system development and testing. The formal evaluation is the main evaluation and will use a subset of the same evaluation data used in the CCU program evaluation.

8. Submission File

The evaluation follows a “take-home” protocol where the data provider (LDC) will send the evaluation data to the participants who, in turn, will send their system output to the evaluator (NIST) for scoring. Please refer to the schedule in Section 10 on when these events will take place including when the data will be released, when the system output will be due, and when the results will be reported.

Participants are to package their system output files (following the format described in Section 6) and system output index file into a compressed, tar submission file and upload it via their evaluation account.

```
% mkdir my_submission
% cp system_output.index.tab my_submission/
% cp M012345QD.tab my_submission/
% tar zcvf my_submission.tgz my_submission/
```

Participants can submit up to 5 submissions. If a submission did not pass validation or could not be scored for any reason, it will not be counted toward the limit. Participants can see the status of their submission and the results in their evaluation account.

9. Participant Paper & Workshop Attendance

To fully satisfy the participation requirements, participants must submit a paper discussing their system as well as attend the TREC workshop at their own cost to present their results. Failure to fully complete the evaluation may result in not being invited to participate in future evaluations. A template for the paper will be available shortly. Registration for the TREC workshop will be announced at a later time.

10. Schedule

Milestone	**Date**
Registration opens; Development data available for release	February 13, 2024
Pilot evaluation data release	April 30, 2024
Pilot evaluation period	July 9-16, 2024
Pilot evaluation full results release	at submission time
Pilot evaluation annotation release	July 23, 2024
Registration ends	June 25, 2024
Evaluation data release	August 27, 2024
Evaluation period	September 3-10, 2024
Evaluation partial results release	at submission time
Evaluation full results release	September 17, 2024

Evaluation annotation release	September 24, 2024
Participant's paper due (not peer review)	October 15, 2024
TREC registration	TBD
TREC workshop	November 18-22, 2024

11. Appendix

This section illustrates how the metrics will be computed using a toy example where a file has only 3 segments:

Reference:

```
user_id     file_id     segment_id        norm  status
123         M111111SP   M111111SP_0001    103   adhere
123         M111111SP   M111111SP_0002    104   violate
123         M111111SP   M111111SP_0002    105   violate
123         M111111SP   M111111SP_0003    none  EMPTY_NA
```

System Output:

```
file_id     segment_id        norm  status      llr
M111111SP   M111111SP_0001    103   adhere      0.75
M111111SP   M111111SP_0002    104   violate     0.80
M111111SP   M111111SP_0003    104   adhere      0.60
M111111SP   M111111SP_0003    106   adhere      0.60
```

This scoring assumes a minimum threshold where all llr values are used. The scorer scores each norm in the reference. For '103' it removes segments not identified as '103' from both the reference and system output.

Reference:

```
user_id     file_id     segment_id        norm  status
123         M111111SP   M111111SP_0001    103   adhere
```

System Output:

```
file_id     segment_id        norm  status      llr
M111111SP   M111111SP_0001    103   adhere      0.75
```

P = #correct / # systems = 1/1 = 1
R = #correct / # reference = 1/1 = 1

For '`104`':

Reference:

```
user_id     file_id     segment_id        norm  status
123         M111111SP   M111111SP_0002    104   violate
```

System Output:

```
file_id     segment_id        norm  status      llr
M111111SP   M111111SP_0002    104   violate     0.80
M111111SP   M111111SP_0003    104   adhere      0.60
```

P = 1/2 = 0.5
R = 1/1 = 1

For ‘`105`’
Reference:

```
user_id     file_id     segment_id        norm  status
123         M111111SP   M111111SP_0002    105   violate
```

System Output:

```
file_id     segment_id        norm  status      llr
```

P = 0 (hard coded as 0 because division by 0 is undefined)
R = 0/1 = 0

Since ‘`106`’ is not in the reference, ‘`106`’ output by the system is not scored. The reference also indicates there is no norm in `M111111SP_0003` so this segment is also ignored from scoring regardless of what the system output.