NIST Data Science Pre-Pilot Evaluation Plan

I. Introduction

Since the 1980s, NIST has been conducting evaluations of data-centric technologies, including automatic speech transcription, information retrieval, machine translation, speaker and language recognition, image recognition, fingerprint matching, event detection from text, video, multimedia, and automatic knowledge base construction, among many others. These evaluations have enabled rigorous research by sharing the following fundamental elements: (1) the use of common tasks, datasets, and metrics; (2) the presence of specific research challenges meant to drive the technology forward; (3) an infrastructure for developing effective measurement techniques and measuring the state-of-the-art; and (4) a venue for encouraging innovative algorithmic approaches.

NIST is launching a new Data Science Evaluation (DSE) series, to occur annually, which will apply this same approach in order to address the unique challenges in the burgeoning field of data science. These challenges stem from some combination of data characteristics (e.g., very large datasets, multi-modal datasets, data from multiple sources with varying degrees of reliability and noise) and task requirements (e.g., building of multi-component systems, enabling effective human-in-the-loop interaction, and visualization of large and complex data).

This evaluation plan describes a pre-pilot evaluation, to take place in advance of both a pilot evaluation and the first evaluation in the DSE series. The primary goal of the pre-pilot is to develop and exercise the evaluation process in the context of data science prior to running larger scale evaluations in this context. The pre-pilot will consist of data and tasks set in the automotive traffic domain, which was picked for its relatability and because large amounts of public data are available.

The pre-pilot is not meant to solve any particular problem in the traffic domain. Instead, the objective is for the developed measurement methods and techniques to apply to a broad range of use cases, regardless of the data type and structure.

The tasks included in the pre-pilot are illustrated in Figure 1 and consist of:

1) **Cleaning:** finding and eliminating errors in dirty data.
2) **Alignment:** relating different representations of the same object in different data sources.
3) **Prediction:** determining possible values for an unknown variable based on known variables.
4) **Forecasting:** determining future values for a variable based on past values.

While each task can be completed independently, in order to measure and analyze error propagation, the output of the cleaning task is designed to be an input to the remaining tasks. This pipelining of tasks (where the output of one task serves as the input to another task) is referred to as a "workflow." To better understand the effects of data cleaning on the workflow, the evaluation will have two phases: in the first phase the original errorful data, along with the output of the data cleaning task, will be available as input; in the second phase, the ground truth for the data cleaning task will be available as input. A high-level summary of the pre-pilot workflow and the two evaluation phases is depicted in Figure 2.

II. Data

A. Core sets

There are six datasets that will be made available as part of this evaluation. These datasets form a common core of data that are available for training and development for all tasks (with the exception of the cleaning task, which limits the data available; see Section III for details). The common core of data consists of:

- Lane Detector Measurements[1]
- Traffic Events
- Traffic Camera Video
- U.S. Census and American Community Survey (ACS)*
- OpenStreetMap (OSM) Maps*
- Weather Data*

The common core datasets are briefly described in Figure 3 and presented in more detail in Appendix A (Table III).

B. Data access

Each above dataset marked with a * will be made available through its respective organization's url. The other datasets will be supplied through an Amazon S3 Storage bucket available to registered pre-pilot participants.[2] Figure 5 gives the organizational structure of the data that will be supplied on Amazon S3.

III. Cleaning Task (finding and eliminating errors in dirty traffic detector data)

A hallmark of data science is the use of data coming from various sources and not necessarily collected for the purpose at hand. In these cases especially, but even when the data are intentionally collected, the data frequently contain errors due to failed sensors, noise, improper formatting or extraction, or other causes. These errors make the data "dirty" and can negatively affect their usefulness. One approach to mitigate the impact of dirty data is to perform some type of *data cleaning*, where errors are removed or replaced with corrected values.

In the traffic use case, errors might result, for example, from sensors failing under extreme weather conditions, or traffic incidents going unreported. Participants in the pre-pilot data

[1] The Lane Detector Measurement dataset consists of output from multiple traffic speed and automobile count detectors at different locations along the road. Each individual detector is referred to as a **lane detector**. Lane detectors at the same location are aggregated into **traffic zones**. Figure 4 illustrates this notion of a traffic zone.

[2] See https://aws.amazon.com/s3/ for information about Amazon Storage.

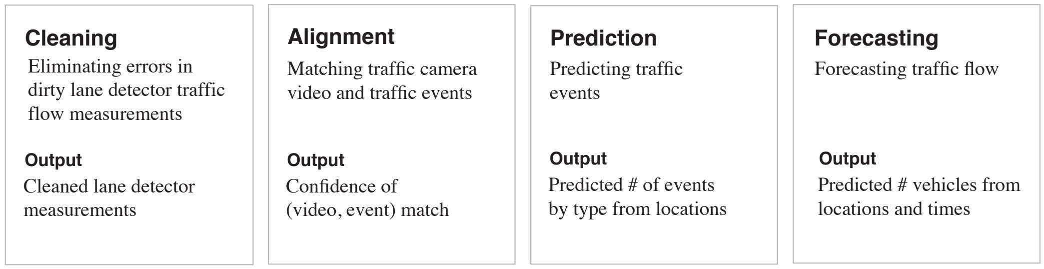

Fig. 1. The pre-pilot evaluation is comprised of four primary tasks: Cleaning, Alignment, Prediction, and Forecasting.

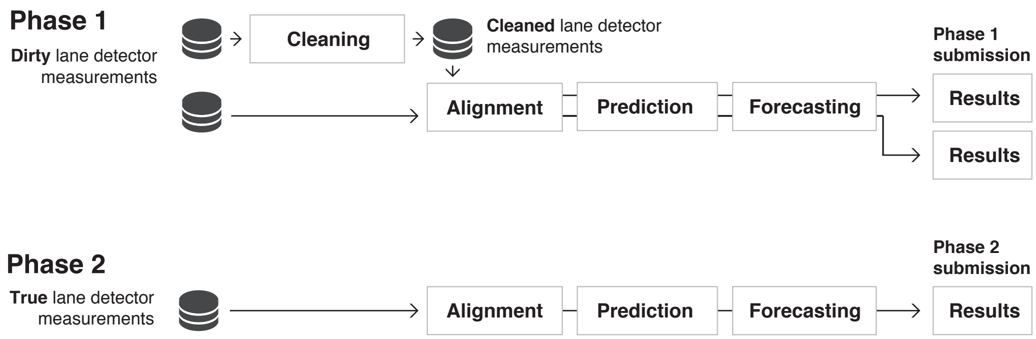

Fig. 2. The pre-pilot organizes four primary tasks into two phases, one in which lane detector measurements are "dirty," and one in which "true" lane detector measurements are provided.

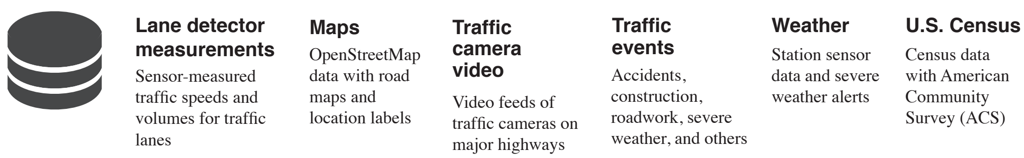

Fig. 3. Six types of data are available in the pre-pilot evaluation: Lane detector measurements, Maps, Traffic camera video, Traffic events, Weather data, and U.S. Census data.

cleaning task will be asked to correct *semantic errors*[3] output by traffic flow sensors.[4]

Here is a summary of the task:

1) **Input:** traffic lane detector measurements with erroneous traffic flow values and speeds (in number and average speed of vehicles since the last interval).
2) **Output:** cleaned traffic lane detector measurements with cleaned traffic flow values (in number of vehicles since the last interval).

A. Description

In this task, participants are asked to clean traffic lane detector measurements containing incorrect flow values, providing correct traffic flow values for the erroneous traffic flow measurements. Detecting which values are erroneous is implicit to this task. The traffic flow measurement is the number of vehicles that have passed within a predetermined number of seconds. For each lane detector, that predetermined number of seconds is specified and is often 60 seconds, i.e., the traffic flow often represents the number of vehicles to have passed the detector in the previous minute.

It is important to note that the traffic speed sensors also contain errors, although the output of the task is restricted to cleaned traffic flow measurements.

B. Training data

No cleaned traffic flow data will be provided as part of this task, making it "unsupervised." However, other data without introduced errors will be available for this task, namely:

1) The traffic lane detector inventory (providing detector information, including each detector's lane id and zone id), and
2) The OpenStreetMap data.

Unlike other tasks in the pre-pilot evaluation, only a subset of the provided data is permitted for use in completing the

[3]Semantic error is defined as an inconsistency or ambiguity expressed as the numerical difference between a given numeric value and the correct numeric value, whereas syntactic correctness is defined as string-based mismatch or formatting issue that is lexical in nature.

[4]In order to be able to effectively evaluate the data cleaning systems for this task, errors were introduced into the traffic flow sensor output.

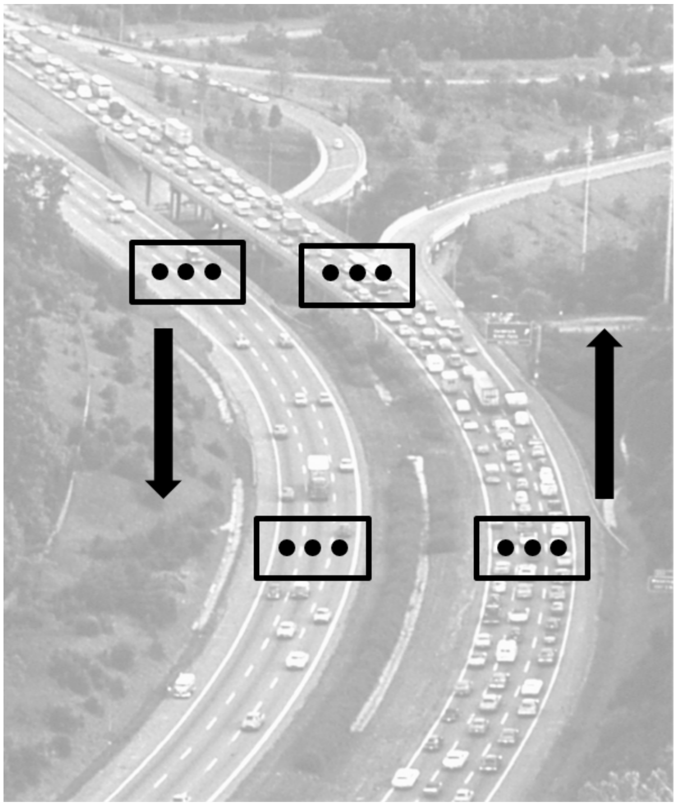

Fig. 4. Traffic zones are an aggregation of traffic lanes. The blue dots are traffic lane detectors and the light blue boxes are the traffic zones. In this figure, there are 12 traffic lane detectors: 4 traffic zones with 3 detectors in each zone. There is one traffic zone detector per road segment in each direction. In the data, there are different numbers of lane detectors in each traffic zone. This photo was obtained from National Archives [1].

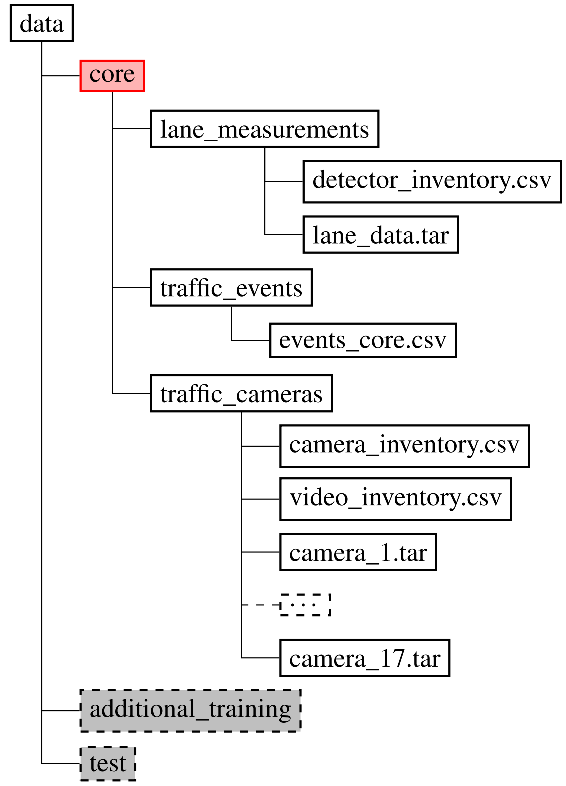

Fig. 5. Organizational hierarchy for the pre-pilot core data on Amazon S3. See other file hierarchy figures in the Appendix for the grey-ed out sub-hierarchies.

cleaning task. In particular, **the data available for use in training and developing systems for this task are restricted to the above two datasets in addition to the dirty lane detector measurements**.

C. Test data

For this task, the test data will be all measurements of a selected subset of lane detectors. The detectors included in the test set will be selected by their zone id, and each detector will be specified by its lane id.

The test data will be available in multiple files, each containing the sensor outputs for a single month and named `cleaning_test_`yy_mm`.csv`, where yy_mm refers to the year and month during which the measurements were taken. For ease of submission, the measurements will be sorted first by the measurement timestamp and then by the lane id.

Given that there are no training data provided for this task, for the cleaning task, and only for the cleaning task, participants will be permitted to interact with the test data. This exception to the restriction is granted to allow participants to clean this dirty lane detector data; waiving the restriction for this task is further motivated due to the use of traffic lane detector data as training data for the remaining tasks.

D. Performance metrics

For this task, system output will be measured using the average absolute value of the error. Formally, let $card(r)$ be the total number of measurements in the test set, and for each measurement i, let $cleaned(fl_i)$ be the estimated traffic flow for measurement i, and let $true(fl_i)$ be the correct traffic flow for measurement i. Then the average absolute value error is:

$$score = \frac{\sum_{i=1}^{card(r)} |cleaned(fl_i) - true(fl_i)|}{card(r)} \tag{1}$$

Any measurement that was changed from the test data will be treated as a correction of a detected erroneous measurement. Therefore, participants are encouraged to check that changes to measurements are intentional and not due to round-off or formatting errors.

E. Submissions

System output results must be provided in multiple files, one per test data file, using standard ASCII encoding.

For every test file `cleaning_test_`yy_mm`.csv`, the corresponding submission file shall be named `cleaning_subm_`$yy\_mm\_\{team\}\_\{submission\}$`.txt`. Every submission file will have the same number of lines in the same order as the corresponding test file, and consist of a single real number per line: $cleaned\_flow$, i.e., the corrected traffic flow for the corresponding measurement.

IV. Alignment Task (relating different representations of the same traffic event)

In many areas of data science, useful information can be obtained by leveraging multiple datasets from a variety of sources. Given the wide variety of data available, some of these data may not be numeric. They may consist of text, social media, images, maps, or video. While every type of data may be useful on its own, there is much to be gained by appropriately combining information from multiple sources. The goal of this task is to integrate different data sources in order to gain more insight into chosen problems. Additionally, it is frequently necessary to handle the processing of different data types prior to integration, which is typically nontrivial.

The traffic use case contains data of varied types and modalities. In particular, one data type that is often available is traffic camera video. In order to measure how effectively systems can integrate traffic video data, participants will be asked to build systems that integrate video data with textual and numeric datasets. In this task, systems will combine data from the "Traffic Camera Video Feeds" and "Traffic Events" sets.

Here is a summary of the task:

1) **Input:** video segments from traffic cameras and traffic event data around that camera.
2) **Output:** a confidence value (i.e., a real number) for each (v = video segment, e = traffic event) pair, where a greater confidence indicates a greater belief that v and e refer to the same event.

A. Description

The goal of this task is to analyze video from camera feeds to detect an event and match it to a separate inventory of traffic events. This task may be divided into two steps:

- From video, detect the occurrence of one or more traffic events.
- Match the detected events to events in the event instances list.

In this section, a *recorded* event is a traffic event that is present in the video and a *reported* event is a traffic event present in the traffic event inventory. In this task, systems must detect events recorded in the video and match them to reported traffic events.

Table I lists and summarizes all of the different traffic event types in the traffic event data. The types of traffic events that are to be detected are a subset of those described in Table I, namely:

- Accidents and Incidents,
- Obstructions, and
- Device Status.

Note that some recorded events may not be reported, e.g., if the situation did not last long, did not disturb traffic, or resolved itself on its own. For example, a disabled vehicle may not be reported because the driver restarted the car after a few minutes.

Each video segment has accurate time information, and the location of the source camera of each video segment is given in latitude and longitude. Some video feeds may have the direction the source camera is facing water-marked ("E" for East, for example).

The Traffic Events Instances file lists all *reported* traffic events in a comma-delimited format with geolocation information, a description of the location (intersection, for example), and the category of the traffic event. *In this additional dataset, all original timestamps will have been removed.*

The output of this task is a confidence value for each video segment and traffic event pair. All pairs must be evaluated, and must be evaluated independently from other pairs, i.e., a (v_x, e) pair may not be used to compute the value for a (v_y, e) pair.

When a recorded event overlaps multiple video segments, every video segment that contains some part of the recorded event will be considered a match with the corresponding reported event.

TABLE I
Traffic event types and subtypes

Traffic Event Type	**Traffic Event Subtype**
Delay Status Cancellation	Delay and disruption
Accidents And Incidents	Abandoned vehicle, accident, accident involving a semi trailer, accident involving a truck, disabled vehicle, hazardous material spill, incident, injury accident, minor accident, multi vehicle accident, numerous accidents, serious accident, vehicle on fire
Device Status	Sign down, traffic lights not working
Disasters	Brush fire, major flood, wildfire
Disturbances	Bomb alert, security alert, security incident
Incident Response Equipment	Other
Obstruction	Animal struck, debris on roadway, downed cables, drawbridge open, fallen trees, obstruction on roadway, subsidence
Pavement Conditions	Surface water hazard
Precipitation	Snow
Roadwork	Bridge construction, bridge maintenance operations, construction work, emergency maintenance, overgrown grass, overgrown trees, paving operations, road construction, road maintenance operations, road marking operations, road widening, storm drain, water main work, work in the median, work on underground services
Special Events	Concert, fair, major event, parade
Sporting Events	Sports event
System Information	Test message
Traffic Conditions	Traffic congestion
Visibility And Air Quality	Visibility reduced
Warning Advice	Alert, police at scene
Winds	Hurricane, strong, winds, tornado

B. Training data

All the common core data described in Section II will be available for training and development purposes.

An additional "Traffic Event" dataset will also be provided for training. This training set involves a subset of traffic cameras, denoted as training cameras, and for each of those cameras, the list of traffic events with timestamps will be provided. In more detail, this additional training data consists of:

- **Video:** 15-minute video segments from the training cameras, available from the core data, at `core_data/traffic_videos/camera_name`
- **Reported events:** All traffic events reported as having taken place at a distance less than $d = 500$ meters from each training camera's location. These traffic events will include timestamps. These events will be supplied in the alignment subfolder of the training data, see Appendix B (Figure 6).

C. Test data

For the test data set, participants will be tested on different traffic cameras. Those cameras are denoted as the test cameras.

The test data consists of video segments and traffic events, similar to the training data described in Section IV-B, but the events provided will not have timestamps. In order to consider all traffic events reported in the vicinity of a camera, the test data will consist of:

- **Video:** 15-minute video segments from the source test cameras. These test video segments will be supplied in the videos/camera_name subfolder from the test/alignment folder.
- **Reported events:** All events reported as having taken place at a distance less than $d = 500$ meters from each test camera's location. *All original timestamp data will have been removed.*

See Appendix B (Figure 7) for the organizational structure of the test data and where to find it on Amazon S3.

Note that the cameras present in the training set may not be the same as those in the test set.

D. Performance metrics

A (video segment, reported event) pair is referred to as a *trial*. When the video segment contains a recorded event that corresponds to the given reported event, this is referred to as a *target trial*. If a trial is not a target trial, it is considered a *non-target trial*. When a system outputs a non-match for a target trial, the resulting error is called a "miss," and when a system outputs a match for a non-target trial, the resulting error is called a "false alarm."

Each trial will be treated as a match or non-match by comparing the system output to a certain threshold; trials greater than or equal to the threshold will be considered matches, and all others will be considered non-matches. By using the sorted system outputs as thresholds, the system's misses and false-alarms can be calculated at all possible *a posteriori* thresholds.

The performance measure will be based on a decision cost function (DCF) representing a linear combination of the miss and false alarm error rates at a threshold τ. The cost function for this task will be:

$$DCF(\tau) = \frac{card(\text{misses}(\tau))}{card(\text{target trials})} + 10 \times \frac{card(\text{false alarms}(\tau))}{card(\text{non-target trials})} \tag{2}$$

The overall performance measure will be the minimum DCF value obtained considering all τ.

E. Submissions

System output results must be provided in a single file per test camera, using standard ASCII encoding. The file will be named `align_subm_`$\{camera\_name\}$`_`$\{team\}$`_`$\{submission\}$`.txt`.

Each test camera's submission file will list results for all (video, event) pair. The results will listed one per line as 3 tab-separated fields:

- *video id*, from the video inventory file
- *event id*, from the events list supplied for that camera
- *confidence value*, the computed confidence between the event and the video segment.

All possible combinations of event id (from the events list supplied for that camera) and video ids (from the video inventory file for that camera) must appear in the submission file, and every confidence value must be computed independently.

See Appendix B (Figure 8) for an example submission file.

V. Prediction Task (applying techniques to guess traffic-related events)

A significant aspect of data science is the analysis of current and past data to make predictions about unknown events — sometimes referred to as *predictive analytics*. The goal of this task is to evaluate how well participants can analyze data from heterogeneous sources to produce knowledge otherwise not explicitly available in other data.

The flow of traffic can be affected by a variety of factors. It can be restricted due to a traffic event such as a disabled vehicle, road work, or an accident. It can also change solely as a result of the changing volume of cars on the road. For the most common types of traffic events, participants will develop a system that predicts the number of traffic events of each given type for a given geographic area and a given time window. The system can use any combination of the common core data to solve this task.

Here is a summary of the task:

1) **Input:** geographical bounding boxes and time intervals.
2) **Output:** predicted counts for each specified type of traffic event.

A. Description

For this task, participants will develop a system that can predict the number and types of traffic events by type for a given (geographical bounding, interval of time) pair. This task will consider only a subset of the available event types, given in Table I. The traffic event types considered for this task are denoted $\mathcal{E}$ and consist of:

- Accidents and Incidents.
- Roadwork.
- Precipitation.
- Device Status.
- Obstruction.
- Traffic Conditions.

The task output will be a list of counts of traffic events for each event type listed above.

B. Training data

All the common core data described in Section II will be available for training and development purposes.

C. Test data

The test data consists of a series of trials, where each trial is a (location, time interval) pair, specified as follows:

1) The decimal latitude and longitude coordinates of the trial geographical bounding box.[5]
2) Start and end of the time window.[6]

For the evaluation of this task the participants will be provided with an ASCII tab separated file containing a list of input trials. For instance, a trial for the greater DC Metro area for a month of March of 2015 would be a single line that would consist of the following tab separated fields:

```
-77.25 38.78 -76.75 39.0
2015-03-01T00:00:00 2015-03-31T23:59:59
```

D. Performance metrics

Task performance will be scored with the average of each trial's Root Mean Squared Error (RMSE).

The RMSE is computed for each trial j. In each trial j, there are $card(\mathcal{E})$ types of events for which to predict the number of events of type e, called $predicted(e, j)$. For each event type e and trial j, $true(e, j)$ denotes the true count of events of that type.

$$RMSE(j) = \sqrt{\frac{\sum_{e \in \mathcal{E}} (predicted(e, j) - true(e, j))^2}{card(\mathcal{E})}} \tag{3}$$

An event is considered to be in the test window if any part of the event overlaps with the test timeframe and location, i.e., if any of the traffic event timestamps fall within the test window.

The RMSE for all the trials is then averaged to get a score, where n is the number of trials:

$$score = \frac{1}{n} \sum_{i=1}^{n} RMSE(i) \tag{4}$$

[5]Each bounding box is defined with the min longitude, min latitude, max longitude, and max latitude.

[6]Each timestamp will be in ISO 8601 format of combined date and time in UTC, i.e. in 'YYYY-MM-DDThh:mm:ss' format. The timezone will be Eastern Standard time.

E. Submissions

System output must be provided in a single file using standard ASCII encoding and named `prediction_submissions_`$\{team\}$`_`$\{submission\}$`.txt`.

The output for each trial, i.e., (location, time interval) pair, must consist of a single line and be given in the same order as in the test data file. Each line must consist of a list of tab-separated real values corresponding to the estimated number of events per type and ordered as follows:

- Accidents and Incidents.
- Roadwork.
- Precipitation.
- Device Status.
- Obstruction.
- Traffic Conditions.

VI. Forecasting Task (applying predictive analytics to forecast traffic flow)

A canonical task in data science is to forecast future values from current and past data, which often involves analyzing trends. The goal for this task is to build systems that use time series data alongside of other heterogeneous data sources to predict future values in the time series.

Traffic flows vary during the day and are subject to a large number of factors including inclement weather, accidents, and a narrow road. In this task, systems will forecast the flow of vehicles in different situations using any combination of the provided datasets.

Here is a summary of the task:

1) **Input:** a list of locations and times to forecast traffic flow.
2) **Output:** for each forecast location and time, a list of forecasted values for every specified time interval.

A. Description

In this task, participants will leverage past traffic information and current conditions (weather, maps) to forecast vehicle flows[7] on major roads. Vehicle flow is defined as the number of vehicles to pass a fixed mile marker on a road (for *all* lanes if there is more than one) for a period of time. All flow information stems from the "Traffic Lane Detector Measurements" set which provides flow, average speed, and lane occupancy information on one or more lanes at specific mile markers. These locations are often composed of multiple lanes, and therefore covered by multiple lane detectors, which report the flow, average speed, and lane occupancy information.

Systems will be required to complete a series of *trials*, where each trial is a (location, start time) pair. For each trial, the system must output a sequence of m real numbers, corresponding to estimated traffic flow for every t minutes over time interval I. That is, each value is the predicted number of vehicles to cross the specified location in the specified t minute interval. For this task, $I = 60$, $t = 5$ (and therefore $m = 12$).

[7]*Traffic* flow and *vehicle* flow are used interchangeably.

B. Training data

All the common core data described in Section II will be available for training and development purposes.

C. Test data

The test data are specified as a tab-delimited file listing locations and times for which to forecast flow.

1) trial id (referenced in the submission format, see Section VI-E)
2) location latitude (decimal).
3) location longitude (decimal).
4) short text description of the location, usually an intersection (for example "495/GW Parkway").
5) datetime of start.[8]

D. Performance metrics

Task performance will be scored with the average of each trial's Mean Absolute Percentage Error (MAPE).

The MAPE is computed for each trial j. In each trial j, there are m forecasted values, one value for each interval. For each i, $forecast(i)$ denotes the number of vehicles forecasted to pass through the trial location during the timespan $[t*(i-1), t*i]$, and $true(i)$ denotes the true number of vehicles that passed through the specified location during that interval of time. The $MAPE$ for each trial j, $MAPE(j)$ is computed as:

$$MAPE(j) = \frac{1}{m}\sum_{i=1}^{m}\frac{|forecast(i) - true(i)|}{true(i)}. \tag{5}$$

The MAPE for all the trials is then averaged to get a score, where n is the number of trials:

$$score = \frac{1}{n}\sum_{j=1}^{n} MAPE(j) \tag{6}$$

E. Submissions

System output results must be provided in a single file using standard ASCII encoding. The file must be named `forecasting_`$\{team\}$`_`$\{submission\}$`.txt`.

For each trial, the submission file will list system outputs by trial: for each (location, time) pair listed in the test file, the $m = 12$ forecasted values will be reported on a single line, separated by a tab character. To identify the trial, the forecasted values will be prepended by the trial id, referencing the id from the test file.

A resulting line will look like the following, the white spaces being a tab character:

```
id      value1      value2  ...     value12
```

Trials will be listed in the same order as in the test file. See Appendix B (Figure 11) for an example submission file, and the matching figure in Appendix B (Figure 10) for an example test file.

[8]Following ISO 8601, the datetime format is YYYY-MM-DDThh:mm:ss

VII. Schedule

The key dates for the NIST pre-pilot evaluation are given below in Table II. In particular, there will be two phases to the pre-pilot, each corresponding to a different workflow, as was presented earlier in Fig. 2:[9]

- **Phase 1.** Participants are given the common core data along with the dirty lane detector data and will be asked to submit system outputs from the four tasks using this data as input. Additionally, participants will be encouraged to submit the results of the alignment, prediction, and forecasting tasks using the cleaned traffic detector data.
- **Phase 2.** Participants are given the cleaning task truth data and will be asked to run the same systems for the alignment, incident, and flow tasks, using the cleaning task truth data as input.

TABLE II
Pre-pilot Tentative Schedule

Release of training data and cleaning test data	September 21st, 2015
Release of the rest of the test data	October 19th, 2015
First submission deadline (all tasks)	November 2nd, 2015
Release of cleaned ground-truth traffic detector data	November 3rd, 2015
Second submission deadline	November 10th, 2015
Release of results	December 7th, 2015
Workshop at NIST	January 28–29th, 2016 (To be confirmed)

This schedule has two submission dates:

- **The first submission** (where the bulk of the time is given), for which participants complete tasks using the dirty lane detector data. Note that participants will be submitting results using both the dirty data as well as (optionally) the cleaned data from the cleaning task.
- **The second submission** (after the first submission deadline has passed), where participants use the lane detector ground truth data to complete the alignment, prediction, and forecasting tasks.

VIII. Rules and guidelines

The evaluation is subject to the following rules and restriction:

- While participants **will** be allowed to use outside data, these data **must be publicly available**, and participants must include references to the data sources used (and how to obtain them) when submitting evaluation results. No internal or proprietary data are allowed to be used.
- Participants may not interact with the test data in any way, e.g., reading the test csv files or watching the test videos is prohibited.
- Each participating site must send one or more representatives who have working knowledge of the evaluation

[9]In describing this workflow, the term "dirty data" is used to refer to the traffic detector data participants are being asked to clean; "cleaned data" to refer to the output traffic flow data from the cleaning task; and "truth data" to refer to the ground-truth data, which is the correct answer to the cleaning task.

system to the evaluation workshop. Representatives must give a presentation on their system(s) and participate in discussions of the current state of the technology and future plans. Workshop registration information will be distributed to registered evaluation participants when available. The workshop will be open only to evaluation participants and representatives of interested government and supporting agencies.

Furthermore, although participants may publish or otherwise disseminate their own results, dissemination of results from the evaluation is subject to the following rules:

- Participants may not publish or otherwise disseminate comparisons of their performance results with those of other participants without the explicit written permission of each such participant. Furthermore, publicly claiming to "win" or suggest a ranking the evaluation is strictly prohibited. Any misrepresentation of the evaluation or its results is also strictly prohibited.
- The results reported by NIST are not to be construed, or represented, as endorsement of any participant's system, or as official findings on the part of NIST or the U.S. Government.

Above are the rules common to all evaluation tasks. Below are task-specific rules, restrictions, and exceptions for each of the tasks:

- For the cleaning task, even though the lane detector data are test data, participants **will** be allowed to interact with these test data. Since the lane detector data are considered training data for the remaining task, this exception is consistent with the remainder of the evaluation.
- For the cleaning task, participants **will not** be allowed to use all of the data in the common core and participants **will not** be allowed to use additional data sources. The only data that may be used for the cleaning task are the traffic lane detector data, the traffic lane inventory file, and the OpenStreetMap data.
- For the alignment task, all (v = video segment, e = traffic event) pairs must be evaluated independently.

DISCLAIMER

Certain commercial equipment, instruments, software, or materials are identified in this evaluation plan in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the equipment, instruments, software or materials are necessarily the best available for the purpose.

REFERENCES

[1] Photograph, National Archives Identifier 546711 (Artist Yoichi R. (Robert) Okamoto), "Looking south from beltway bridge over the potomac. the capital beltway circles the virginia-maryland suburbs and provides high speed access to points in the district, 5/1973," May 1973, DOCUMERICA: The Environmental Protection Agency's Program to Photographically Document Subjects of Environmental Concern, 1972–1977 Record Group 412: Records of the Environmental Protection Agency, 1944–2006.

[2] "Maryland chart traffic cameras," 2015. [Online]. Available: http://www.chart.state.md.us/travinfo/trafficcams.php

[3] "2010 u.s. census," 2015. [Online]. Available: http://www.census.gov

[4] "Openstreetmap," 2015. [Online]. Available: http://www.openstreetmap.org/

[5] "NOAA's integrated surface hourly," 2015. [Online]. Available: http://www.ncdc.noaa.gov/isd

[6] J. N. Lott, "The quality control of the integrated surface hourly database," in *84th American Meteorological Society Annual Meeting*, vol. 7.8. Seattle, WA: American Meterological Society, 2004. [Online]. Available: http://www1.ncdc.noaa.gov/pub/data/inventories/ish-qc.pdf

[7] "NOAA," 2015. [Online]. Available: http://www.ncdc.noaa.gov/swdi

APPENDIX A
SUMMARY DATA TABLE

Table III describes each of the data sets, including the specific fields in each case, supplementing the information provided in Section II. The documentation associated with each data source provides a detailed description of all of its corresponding fields.

APPENDIX B
DATA ORGANIZATION STRUCTURE AND EXAMPLE DATA FILES

This section presents the organizational structure of the data that will be provided to participants. These figures supplement Figure 5, which gives the overall structure of the data in the common core. Additionally, to clarify the descriptions of the test and submission sets throughout the evaluation, select examples for test and submission files are provided. These partial examples contain hypothetical data in that the id numbers in the examples may not correspond to actual id values in the data.

Figure 6 shows the organizational structure of the additional training data provided for the alignment task described in Section IV-B.

Figure 7 shows the organizational structure of the test data for the alignment task described in Section IV-C.

Figure 8 gives a partial example of a submission file for the alignment task described in Section IV-E.

Figure 9 shows the organizational structure of the test data for the forecasting task described in Section VI-C.

Figure 10 gives a partial example for a test set data file for the forecasting task described in Section VI-C.

Figure 11 gives a partial example for a test set data file for the forecasting task described in Section VI-E.

TABLE III
SUMMARY OF AVAILABLE DATASETS.

Data Type	Data Subset	Description
Lane Detector	Lane Detector Inventory	List of all traffic lane detectors as a comma-separated file. Each detector is uniquely identified by its `lane_id` value, and each detector inventory gives the location of the detector (in decimal latitude and longitude coordinates), the source organization for the measurements of those detectors, the time interval between scheduled measurements, and other relevant information.
	Lane Detector Measurements	Measurements from traffic sensors in locations in the DC Metro area and the Baltimore area. Traffic sensors are placed on both directions of the highways, in each lane. Lane and zone (multiple lanes of the same road going in the same direction) data are provided. The measurements include the following attributes (among others): 1) *Flow:* the number of vehicles to have passed through the lane detector since the last scheduled measurement. 2) *Speed:* the average vehicle speed since the last measurement. 3) *Occupancy:* the average percent of time a vehicle was in front of the detector since the last measurement. 4) *Quality:* a data quality field. This dataset has data from 2006 to 2015, collected from the DC-Maryland-Virginia area, and is approximately 150GB.
Traffic Events	Traffic Event Instances	A traffic event is defined as a situation that involves traffic in the ways described in Table I. Prominent features of an event are injuries, damage to vehicles, hazards to persons, failure of equipment, closure of one or more lanes, debris or roadkill on the road or shoulder, or any obstruction on roadway. Each traffic event listing includes the following fields (among others): 1) Description. 2) Location, both in formatted text (the intersection) and in decimal latitude and longitude. 3) Times the event was created, confirmed and closed. 4) The type and subtype of the traffic event; the field labels are `event_type` and `event_subtype`. A traffic event is reported in the inventory when it is outlined by or to authorities (911 calls, road kill pickups, etc.). It is confirmed when an authority arrives at the scene (the police arrives, etc.) and deemed over when it was closed by authorities. For an accident, this typically indicates when all lanes have been reopened, damaged vehicles have been removed, and all responders have left the scene. This dataset has data from 2003 to 2015, collected from the DC-Maryland-Virginia area, and is approximately 200MB.
Traffic Camera Video [2]	Camera Inventory	A list of all traffic cameras with their locations, described both in text (the intersection) and in decimal latitude and longitude.
	Camera Video Feeds	Consecutive 15-minute video segments from traffic cameras in Maryland with start times. The traffic cameras may be remotely operated by humans, who can rotate the camera and zoom, which happens when the human operator chooses to look at a traffic situation. Some cameras may have watermarks indicating the direction the camera is facing (E for East, SW for South-West, etc), or the current time. This dataset has data from 2015, collected from Maryland, and is approximately 3TB.
U.S. Census	2010 U.S. Census [3]	Publicly available information including population counts; age, income, and occupation demographics; and household demographics in summary files and PUMS (Public Use Microdata Sample).
	American Community Survey (ACS)	A more frequent survey providing statistics on transportation and commutes, such as the average commute length, the percentage of people who carpool, and the percentage of people who use public transportation. There are 1-year, 3-year, and 5-year surveys as summary Files and PUMS, like the U.S. Census Data.
OpenStreetMap [4]	[No subset]	Map data from from OpenStreetMap, describing the road network in the DC-MD-VA area as well as locations including airports, public transportation stations, and buildings that host large events. These maps also support lookup by latitude and longitude coordinates.
Weather	Integrated Surface (ISD) [5]	A dataset of measurements from weather stations in the DC-MD-VA area with a variable number of measurements. Measurements include station information, temperature, air pressure, weather condition, precipitation, and other elements. The ISD set is quality-controlled. The quality control does not state that it is free of errors or missing data; only that others have looked at it to try to improve the quality of the data. Lott [6] discusses the quality control process that are used in the ISD to check for formatting errors and outliers.
	Severe Weather Data Inventory (SWDI) [7]	A compilation of many types of severe weather, including storms, hail, tornados and lighting strikes.

id	lat	long	description	start-datetime
1	37.00	78.00	"495/GW Parkway North"	2015-09-18T13:00:00
2	37.00	78.00	"495/GW Parkway South"	2015-09-18T14:00:00
			...	
100	36.90	77.10	"495/River Rd North"	2015-09-19T13:00:00

Fig. 10. Example test data for the forecasting task. Spaces between fields are tabulation characters. Headers are *not* required and are provided here for clarification.

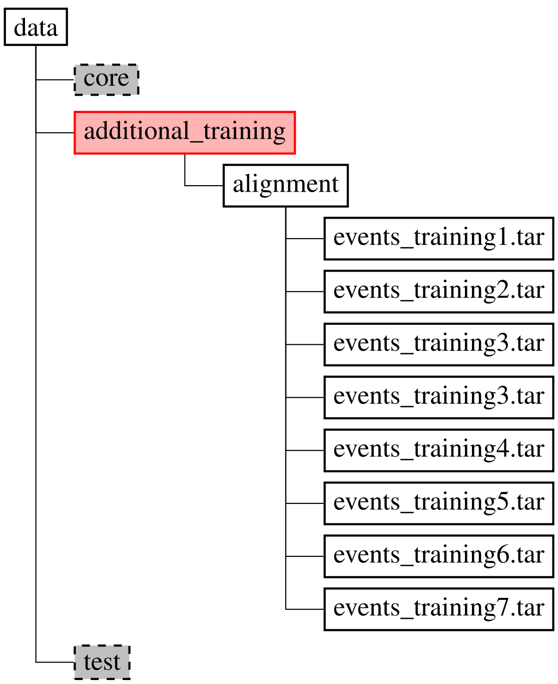

Fig. 6. File hierarchy for the additional training data provided for the alignment task. See other file hierarchy figures for the grey-ed out sub-hierarchies.

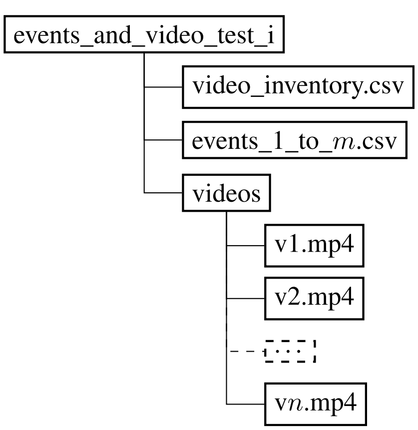

Fig. 7. Contents of a *events_and_video_test_i.tar* file (test data for the alignment task)

video_id	event_id	confidence_value
1	1	0.1
1	2	0.01
	...	
1	n	0.01
2	1	0.99
	...	
2	n	0.01
	...	
m	n	0.8

Fig. 8. Example submission file for the alignment task. Spaces between fields are tabulation characters. Headers are *not* required and are provided here for clarification. In this example, the test camera has m video segments and n events.

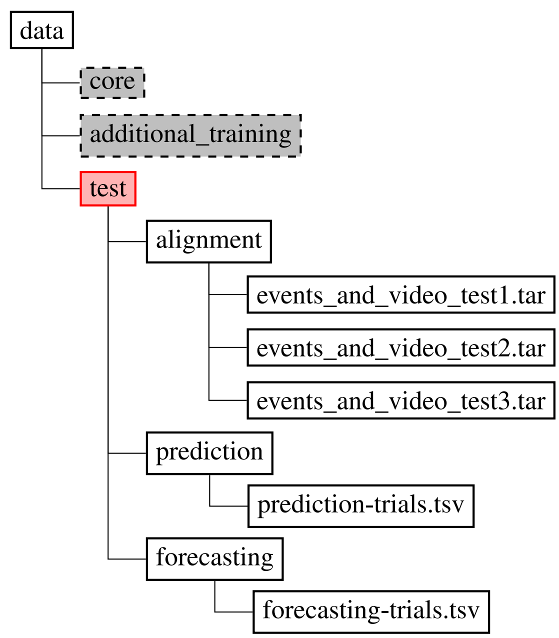

Fig. 9. File hierarchy for the evaluation test data of the forecasting task. Note that the cleaning task test data is in the "core" section rather than in the "test" section. See other file hierarchy figures for the grey-ed out sub-hierarchies.

id	value1	value2	...	value12
1	12	13	...	4
2	2	4	...	2
		...		
100	2	5	...	2

Fig. 11. Example submission file for the forecasting task. Spaces between fields are tabulation characters. Headers are *not* required and are provided here for clarification.