Meta Data Inference on Building Sensors Data

Supervisor: Mr. Tim Rist

First Examiner: Prof. Dr. Hannah Bast

Second Examiner: Dr. Fang Wei-Kleiner

Albert-Ludwigs-Universität Freiburg

Muhammad Hamiz Ahmed

Department of Computer Science
Chair for Algorithms and Data Structure

Problem Statement

Motivation

UNI FREIBURG

- Modern buildings are equipped with complex heating, cooling and ventilation systems

- In the group "Building Performance Optimization" at Fraunhofer ISE, various methods are investigated for detecting faulty or suboptimal operation of such systems (e.g. simultaneous heating and cooling)

- These methods require data of a multitude of sensors for input (like temperature, pressure, current etc.)

- **For this, a manual labelling of the sensor is required, which is excessively time consuming**

Motivation

- To make measurement data usable for the analysis, a common **data point naming convention** is applied to each sensor which marks the origin and type of it.

- Each label is comprised of a set of different meta-data categories.

- Under the data point naming convention, a data point name label of a certain sensor can be **AHU___SUPA__MEA_T**

Motivation

Sensor with data point name label **AHU___SUPA__MEA_T**

Meta-Data Categories	Labels
System	AHU
Subsystem1	-
Subsystem2	-
Medium	SUPA
Position	-
Kind	MEA
Point	T

Available Data

- Recorded time series data, for every sensor:
  - multiple days, or even years of data
  - minutely resolution

- Each sensor is also described by a textual information, which we a call the Description Text of the sensor, e.g.: “Zulufttemperatur RLT-Z1”

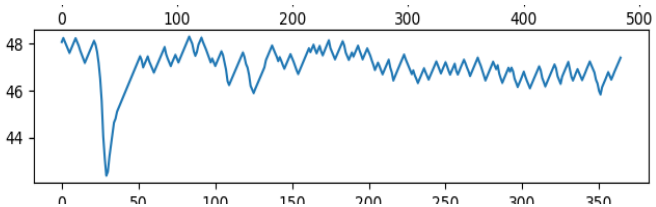
Problem Statement

Develop a methodology for automatically mapping data sources to a hierarchically structured point name label based on raw time series data together with available texts in a supervised learning manner.

Problem Statement

Develop a methodology for automatically mapping data sources to a hierarchically structured point name label based on raw time series data together with available texts in a supervised learning manner.

Zulufttemperatur RLT-Z1



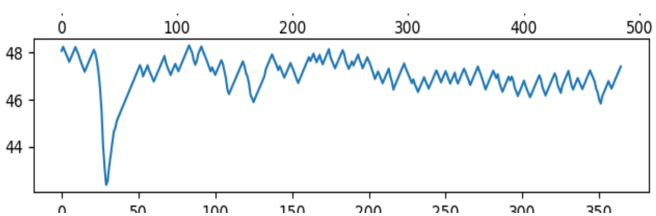
Time Series Data

System

Problem Statement

Develop a methodology for automatically mapping data sources to a hierarchically structured point name label based on raw time series data together with available texts in a supervised learning manner.

Zulufttemperatur RLT-Z1



Time Series Data

System

AHU___SUPA__MEA_T

Approach and Technique

Dataset Description

The data is extracted from 13 buildings having around 3300 sensors in total

Example of few buildings

No.	Building Names	Description
1	01_BZR_Ddorf	Office building of the district government Düsseldorf
2	02_DKB_Berlin	Office building of the Deutschen Kreditbank AG (mainly offices and server rooms; 9873 m2)
3	03_KuP_Zentrale_Berlin	Headquaters of “Kieback & Peter” in Berlin
4	DVZ	Service and administration center Barnim

Challenges with Time Series Data

Challenge 1: Missing Values

- The data from every sensor is recorded every minute and a maximum of 1440 raw values are gathered in a single day

- Missing time series data in the dataset poses a major challenge
  - In some cases, the measurement procedure records the sensor values are only when the change is noticed in the behavior of the sensor.
  - Faults in the system also result in missing values

Challenges with Time Series Data

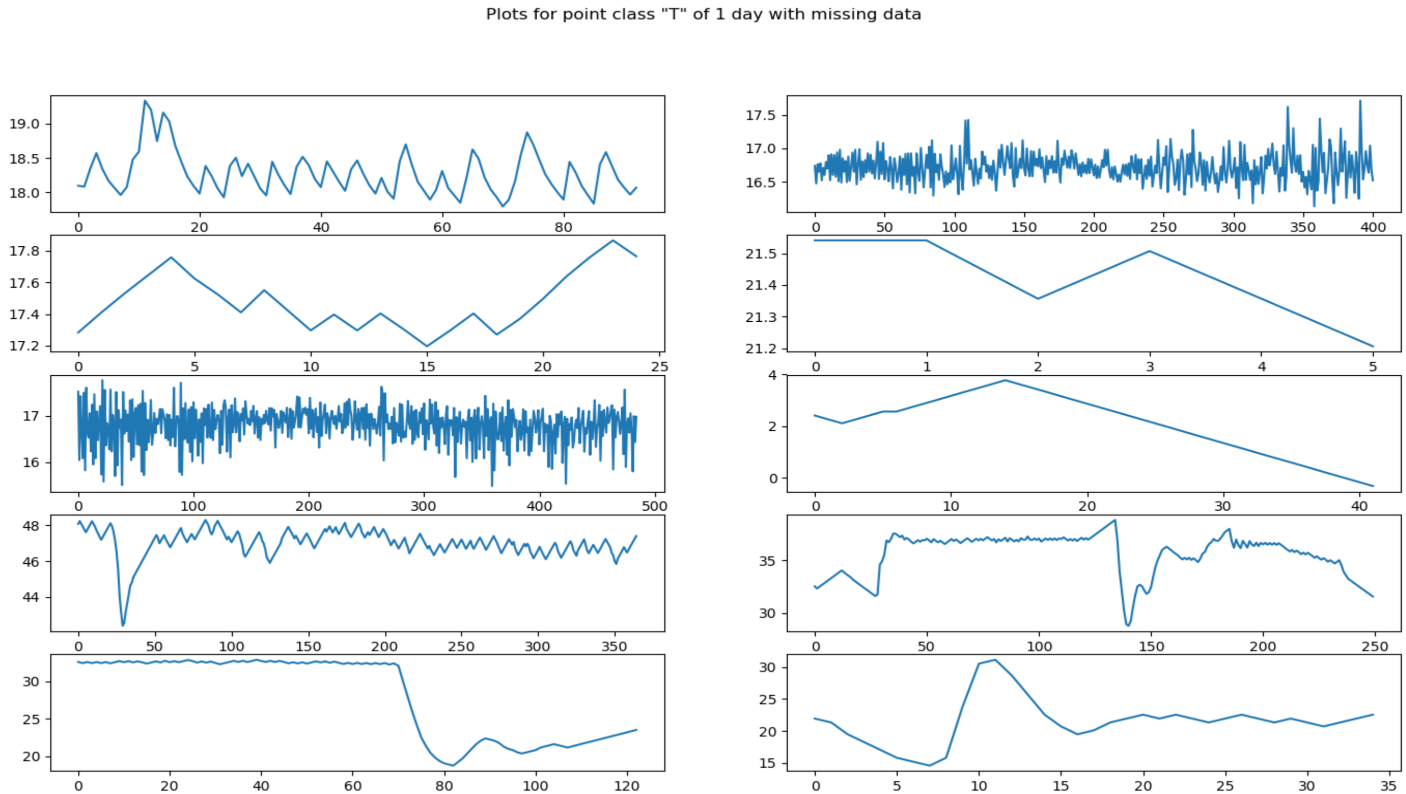
Challenge 2: Inconsistent Behavior

- Sensors belonging to same meta-data category often show inconsistent behavior
- The inconsistency of the time series data makes it very unreliable to perform this classification, solely on it.

Challenges with Time Series Data

Challenge 2: Inconsistent Behavior

- Sensors belonging to same meta-data category often show inconsistent behavior



Challenges with Description Text Data

- The Description text of the sensor is used to aid technical personnel at the building in understanding the nature and type of the sensor
  - "*ELZ UV-ISP03 Schaltschrank RLT 03 Strom L3*" specifies type of the sensor to be current and system to be Air Handling Unit

Challenges with Description Text Data

- The Description text of the sensor is used to aid technical personnel at the building in understanding the nature and type of the sensor
  - “*ELZ UV-ISP03 Schaltschrank RLT 03 Strom L3*“ specifies type of the sensor to be current and system to be Air Handling Unit

- All available description texts associated to a sensor are written down manually making them very different from one building to another and hence, not reliable to be used alone.

System Architecture

Before the classification can be performed, following two problems need to be addressed.

System Architecture

Before the classification can be performed, following two problems need to be addressed.

1. The target point name label has over 900 classes with the data comprising from 3300 sensors only.
   - E.g: AHU___SUPA__MEA_**T**
     AHU___SUPA__MEA_**P**

System Architecture

UNI FREIBURG

Before the classification can be performed, following two problems need to be addressed.

1. The target point name label has over 900 classes with the data comprising from 3300 sensors only.
   - E.g: AHU___SUPA__MEA_**T**
     AHU___SUPA__MEA_**P**

2. Both the input data (time series data and description text) are different in nature. Therefore, it would not be efficient to apply a single machine learning model on both data inputs.

System Architecture

- Problem 1: Too many target classes
  - Solution: Treat each meta-data category of the point name label as an independent target and train a separate model to predict the class of each meta-data category

System Architecture

- Problem 1: Too many target classes
  - Solution: Treat each meta-data category of the point name label as an independent target and train a separate model to predict the class of each meta-data category

AHU___SUPA__MEA_T

Meta-Data Categories	Labels
System	AHU
Subsystem1	-
Subsystem2	-
Medium	SUPA
Position	-
Kind	MEA
Point	T

System Architecture

- Problem 1: Too many target classes
  - Solution: Treat each meta-data category of the point name label as an independent target and train a separate model to predict the class of each meta-data category

AHU___SUPA__MEA_T

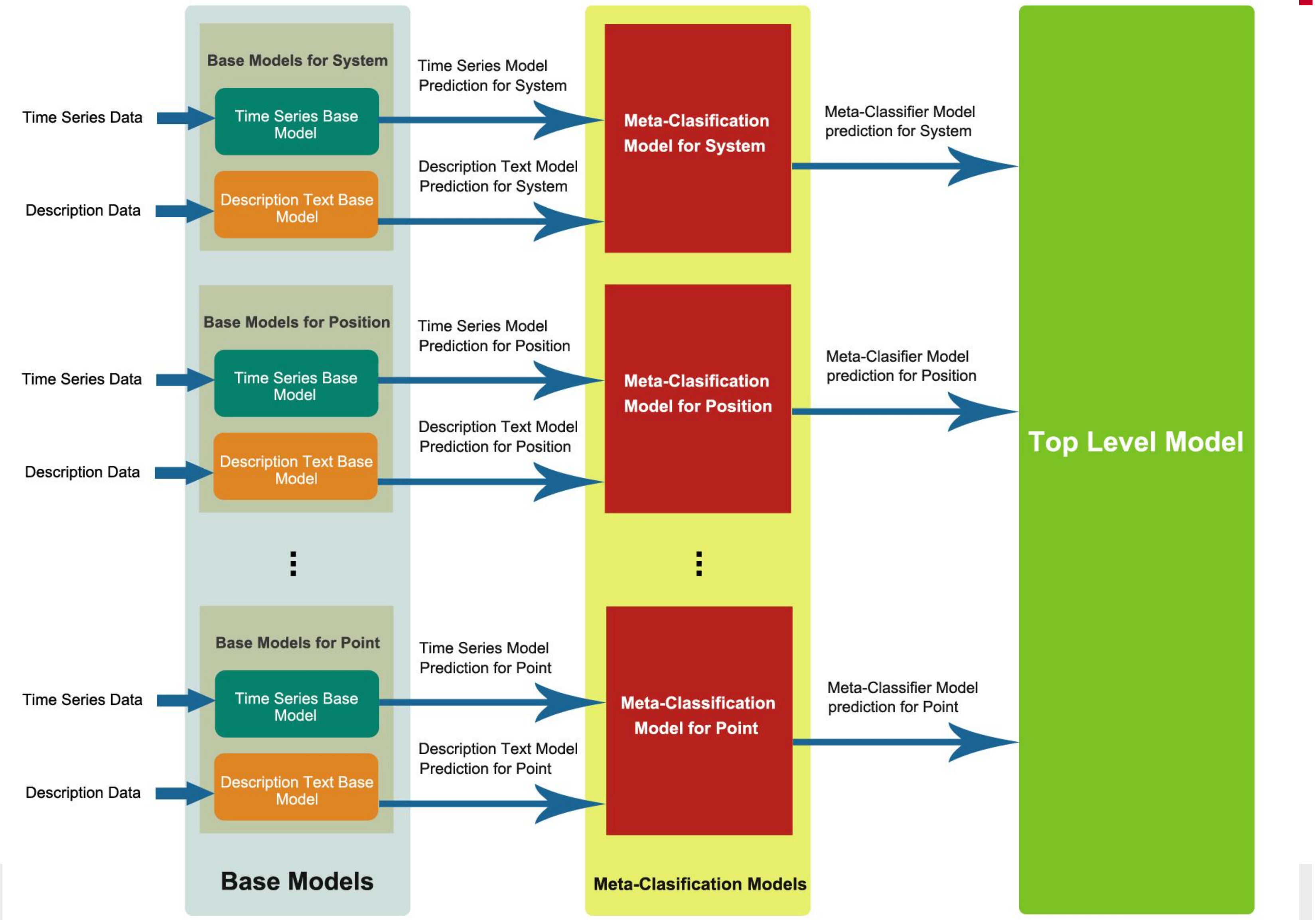
Meta-data categories	Number of classes
System	78
Subsystem1	74
Subsystem2	26
Medium	24
Position	39
Kind	13
Point	52

System Architecture

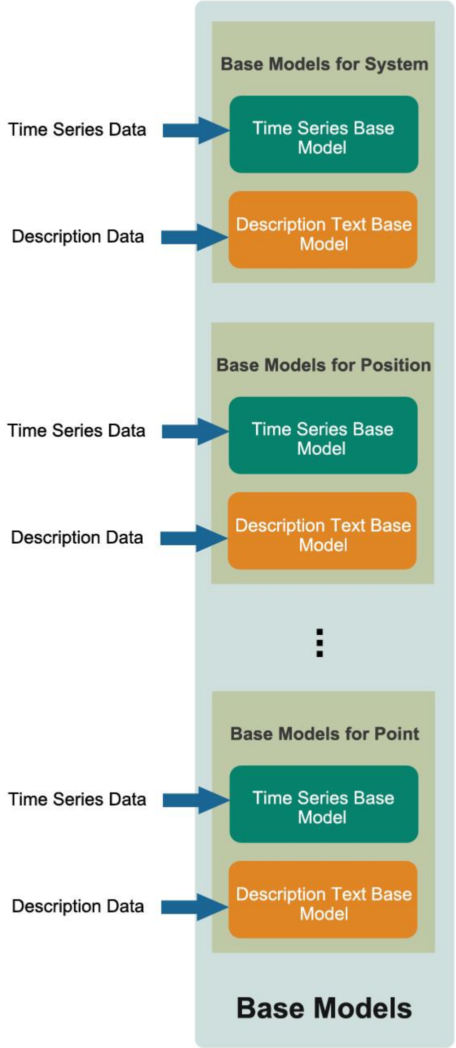
- Problem 2: Different types of input data
  - Solution: Train separate models for time series data and description texts, independent of each other treating each meta-data category as an independent target

  - Combine the results obtained from each model in to final prediction.

System Architecture

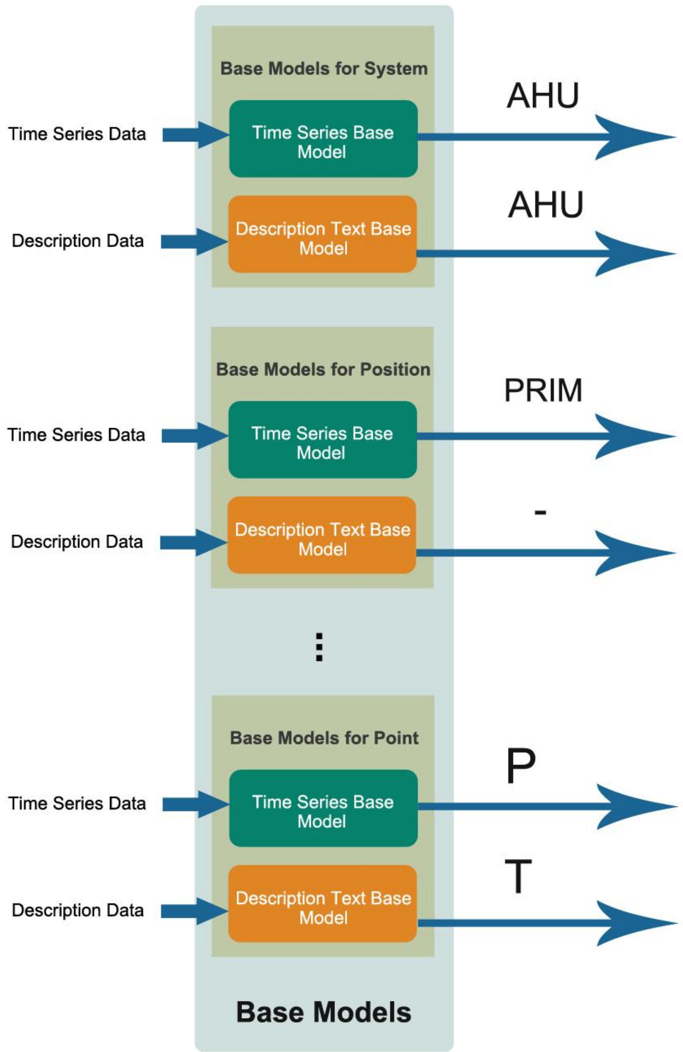


How does the system work?



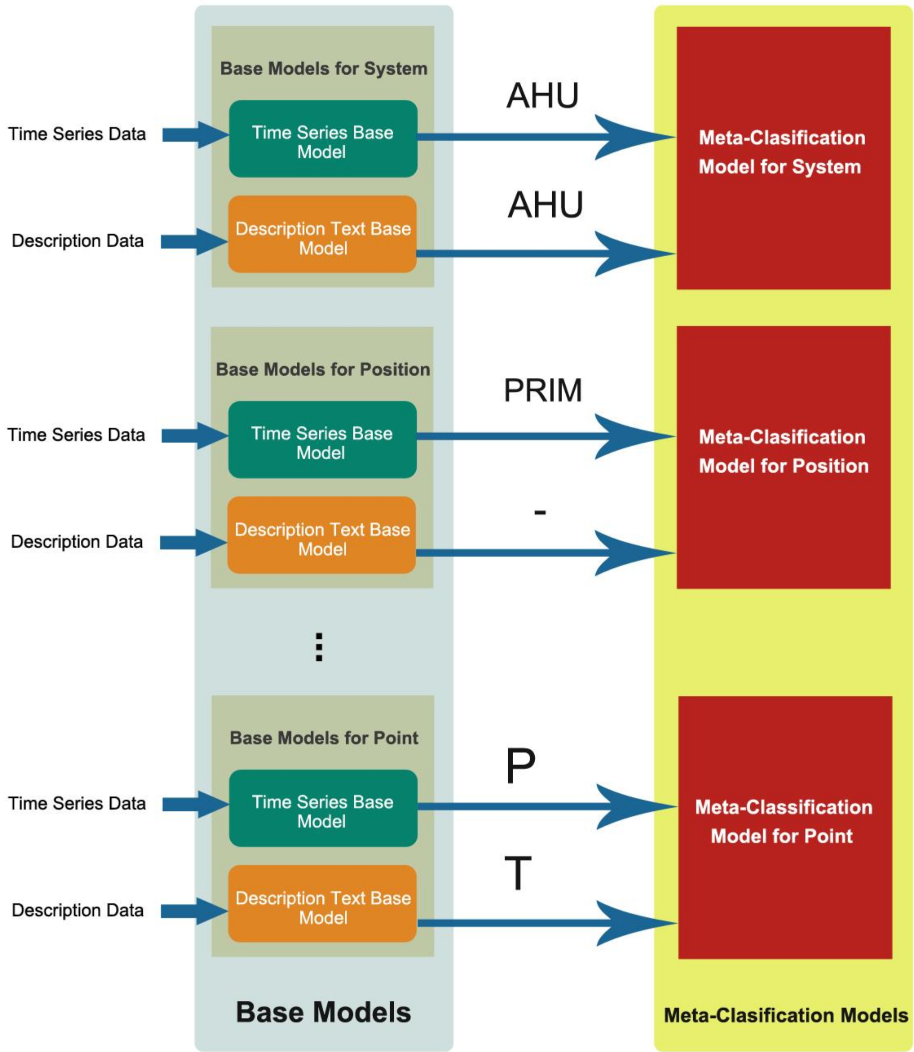
Sensor X

How does the system work?



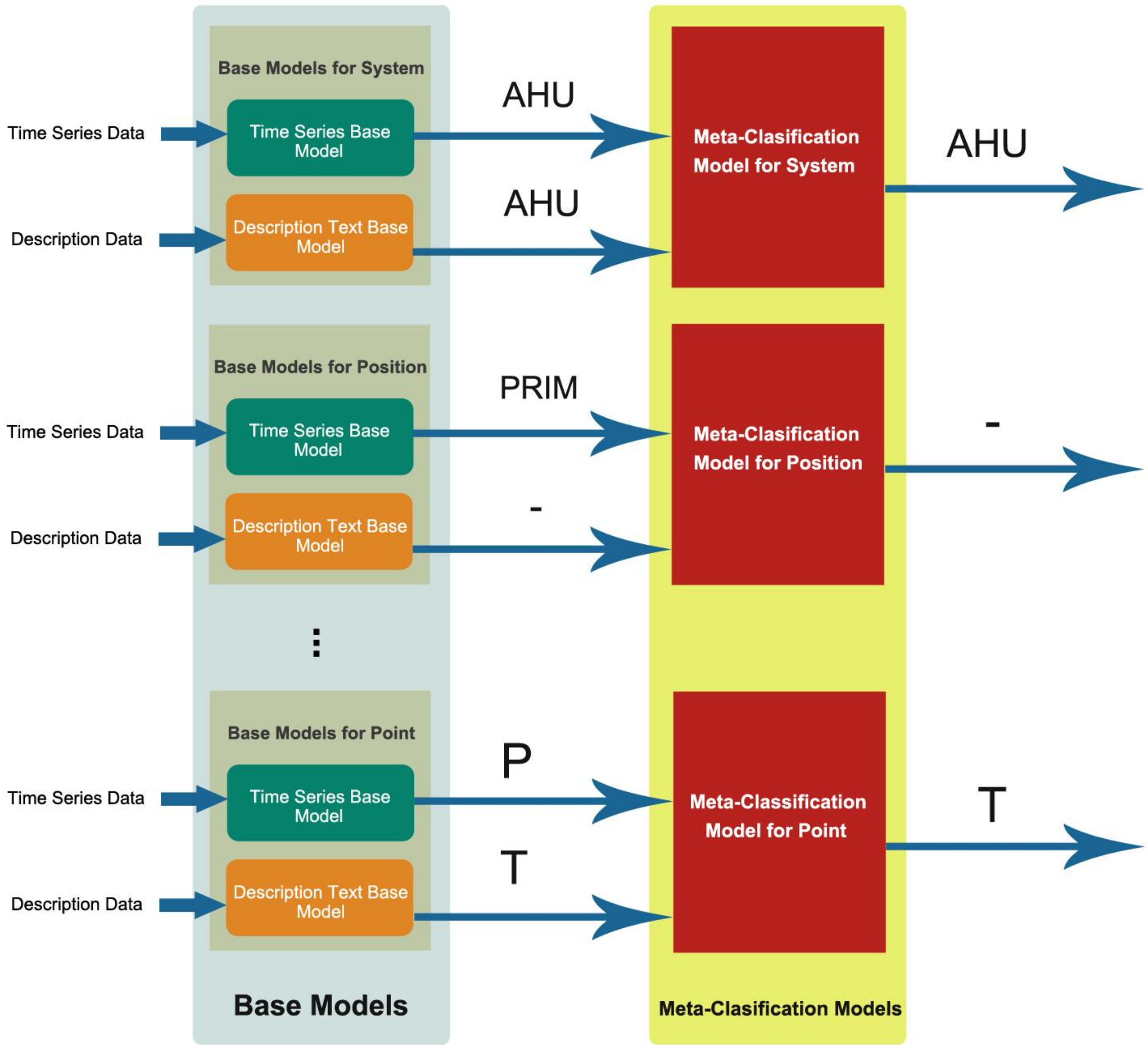
Sensor X

How does the system work?



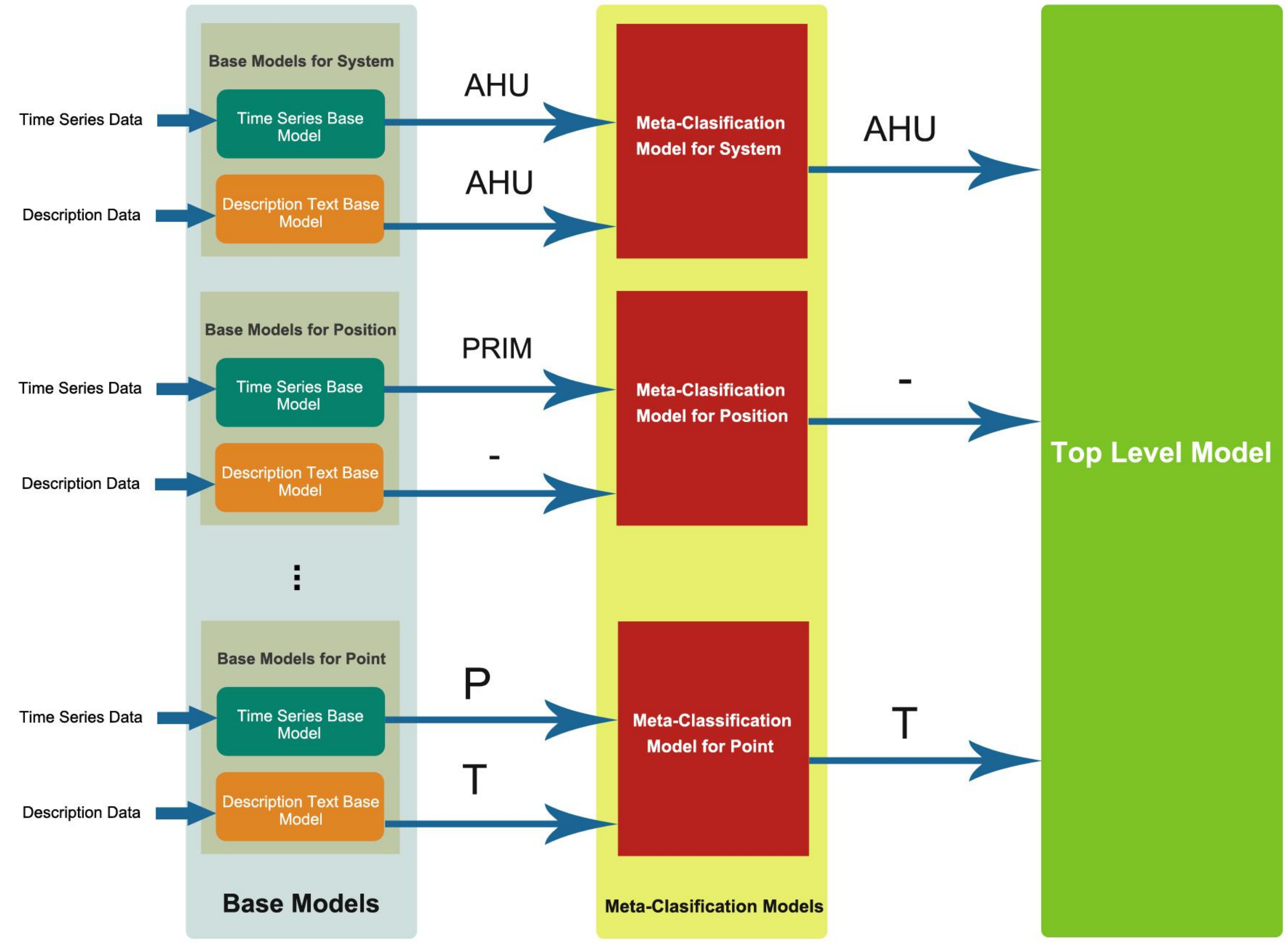
Sensor X

How does the system work?



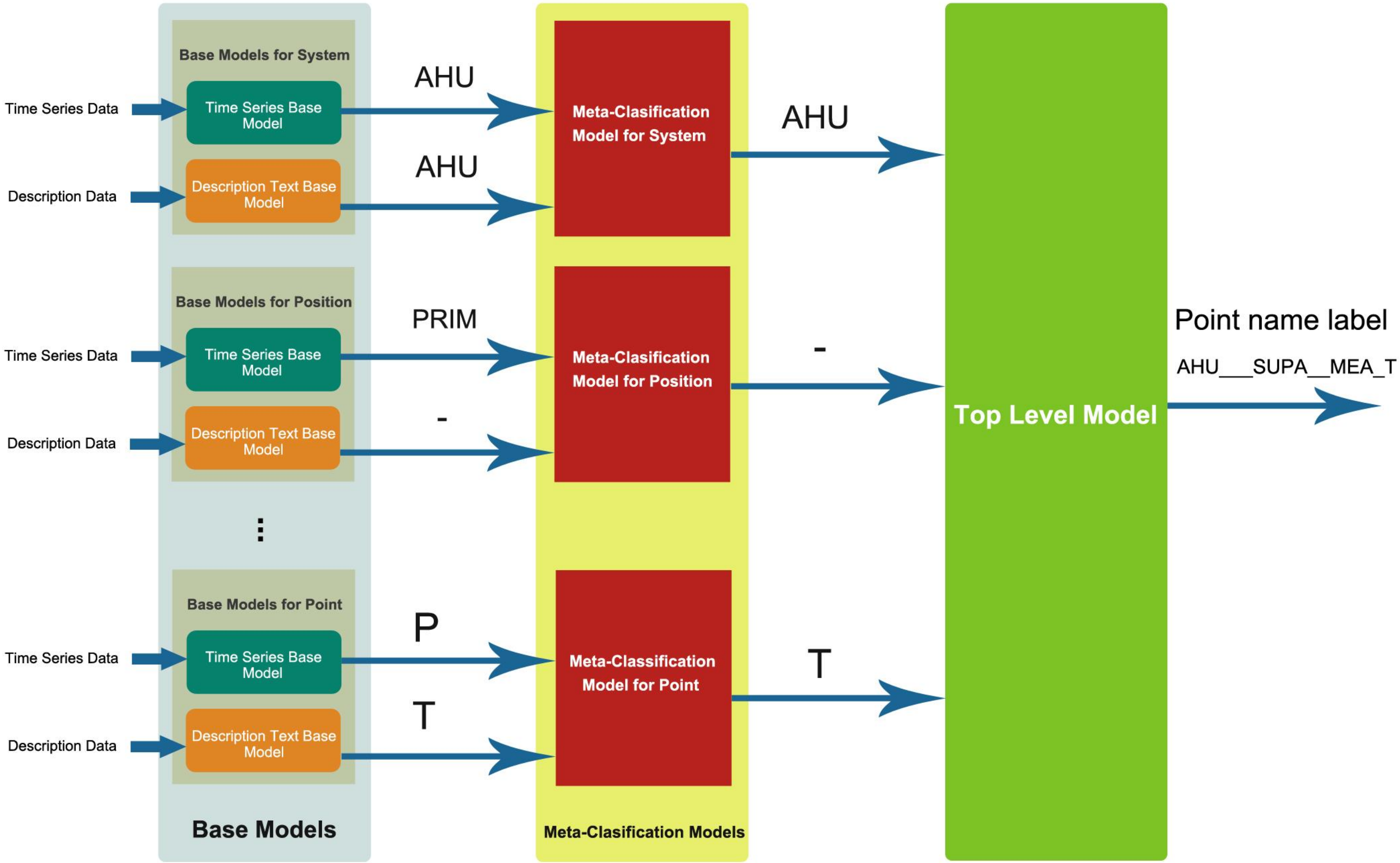
Sensor X

How does the system work?



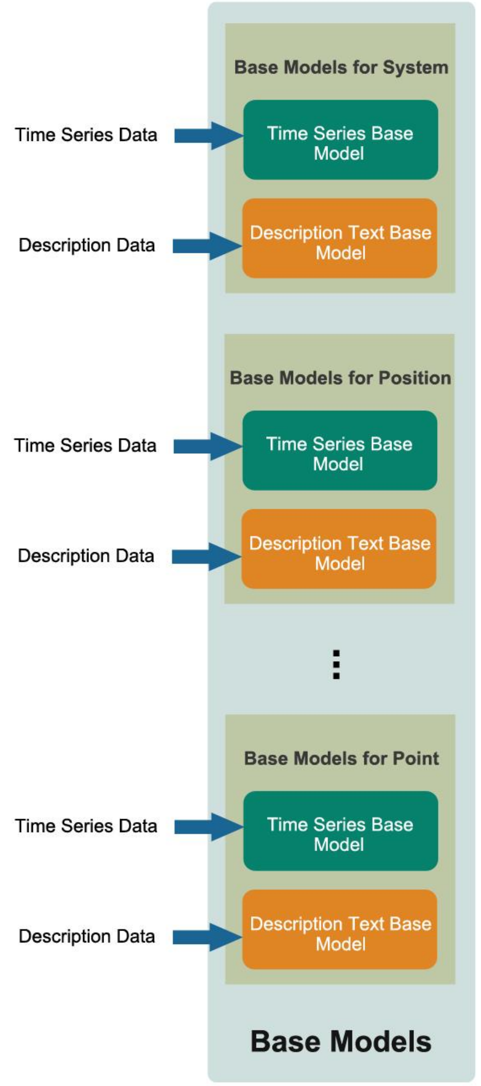
Sensor X

How does the system work?



Sensor X

Base Models

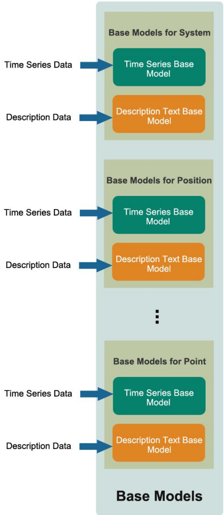

Base Models for System
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
Base Models for Position
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
Base Models for Point
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
Base Models

Base Models

Time Series Base Models

- Uses Time Series Data to predict the meta-data category class labels

- Problem: Inconsistency in Time Series Data and missing information pose a major challenge for the classification of meta-data category with raw data points.



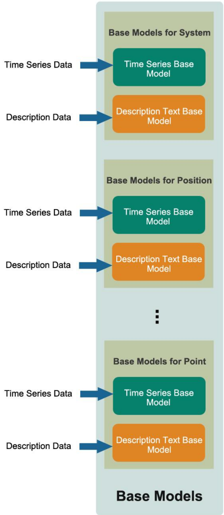
Base Models

Time Series Base Models

- Uses Time Series Data to predict the meta-data category class labels

- Problem: Inconsistency in Time Series Data and missing information pose a major challenge for the classification of meta-data category with raw data points.

- Solution: Project the data to a meaningful representation which makes it easy to classify the sensors such as extracting handcrafted features.



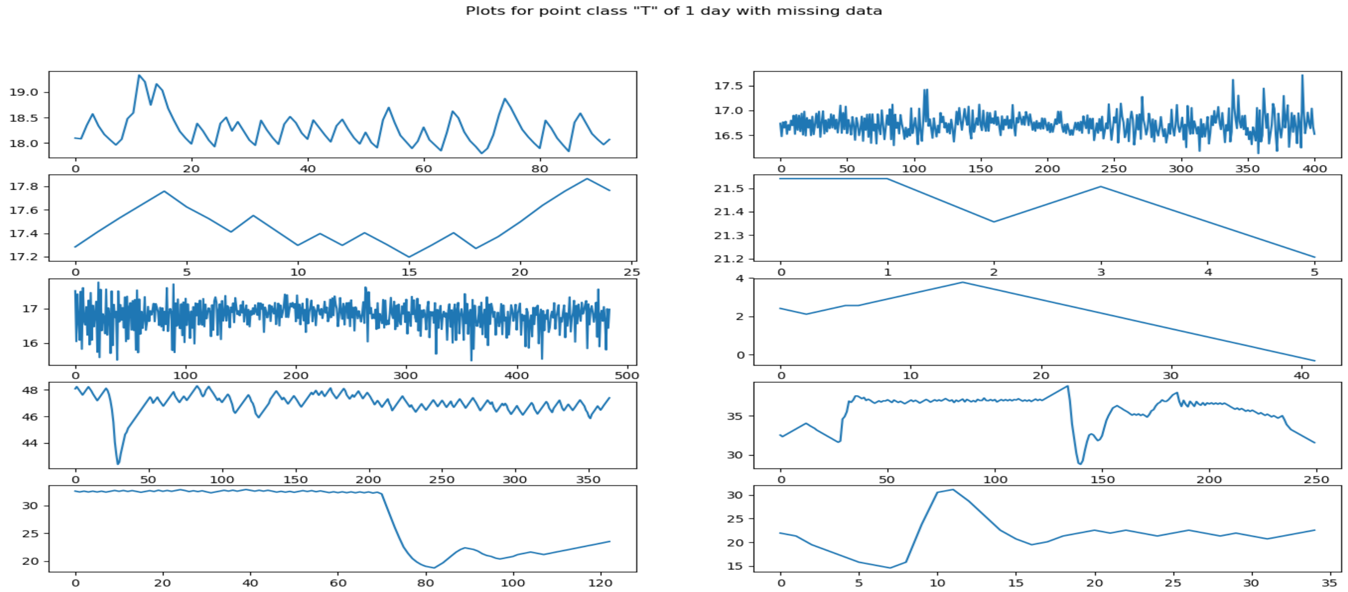
Time Series Base Model

1. Mean of a day
2. Standard Deviation of a day
3. Minimum in a Day
4. Maximum in a Day
5. Standard deviation of difference between consecutive elements in a day
6. Minimum of Difference between consecutive elements in a day
7. Maximum of Difference between consecutive elements in a day
8. Mean of hourly standard deviation
9. Standard deviation of hourly standard deviation
10. Maximum of hourly standard deviation
11. Minimum of hourly standard deviation
12. Standard deviation of absolute/real values of DFT
13. Max of absolute/real values of DFT
14. Min of absolute/real values of DFT
15. Min frequency obtained in DFT
16. Max frequency obtained in DFT
17. Median frequency obtained in DFT

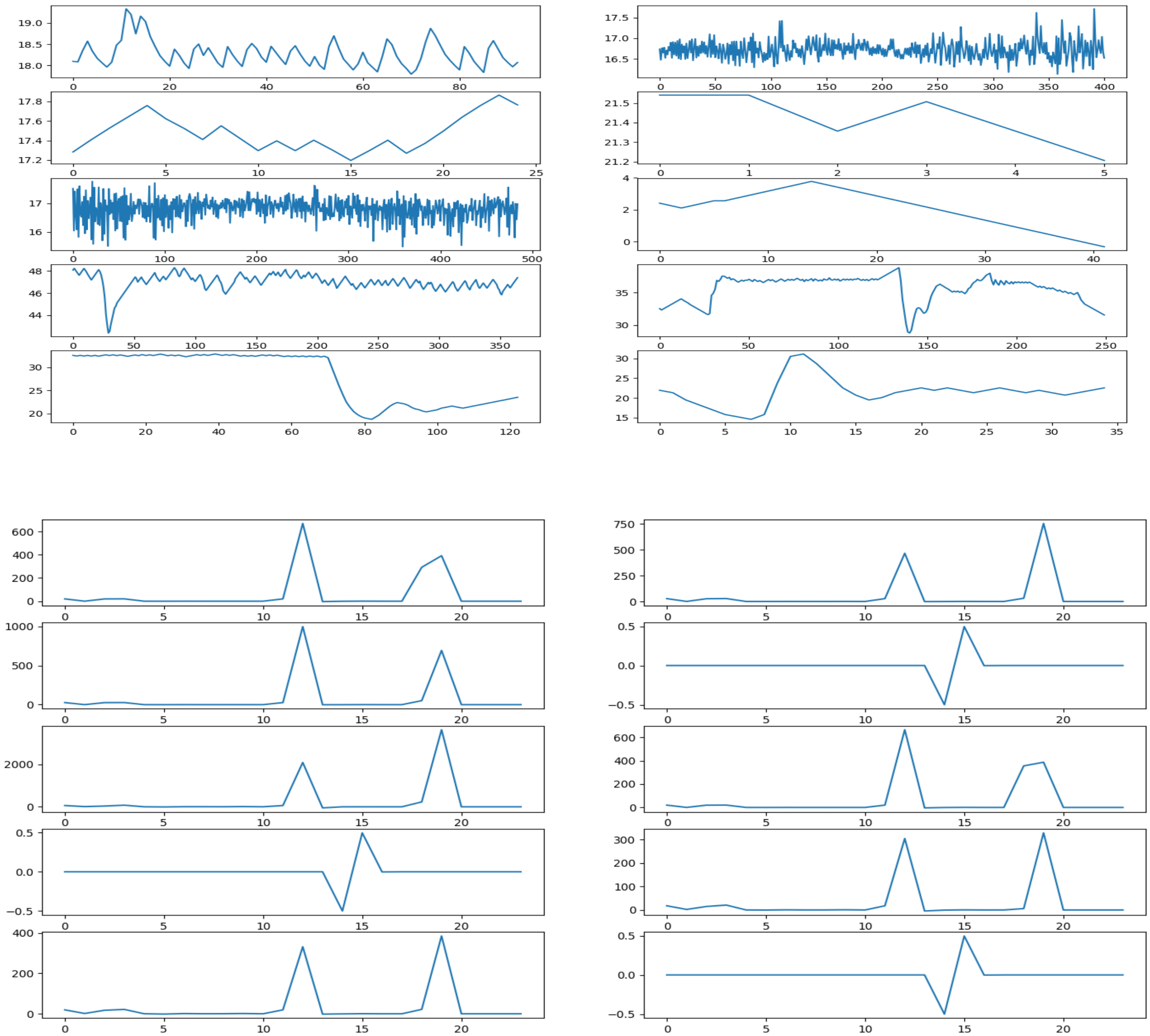
Base Models

Time Series Base Model

18. Spectral Entropy
19. Number of peaks
20. Power

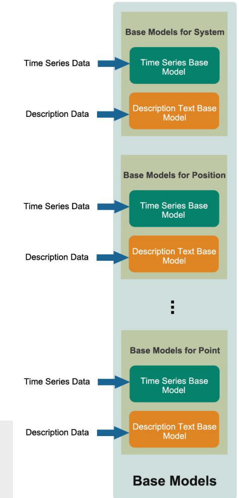

Plots for point class "T" of 1 day with missing data

Plots for point class "T" of 1 day with missing data

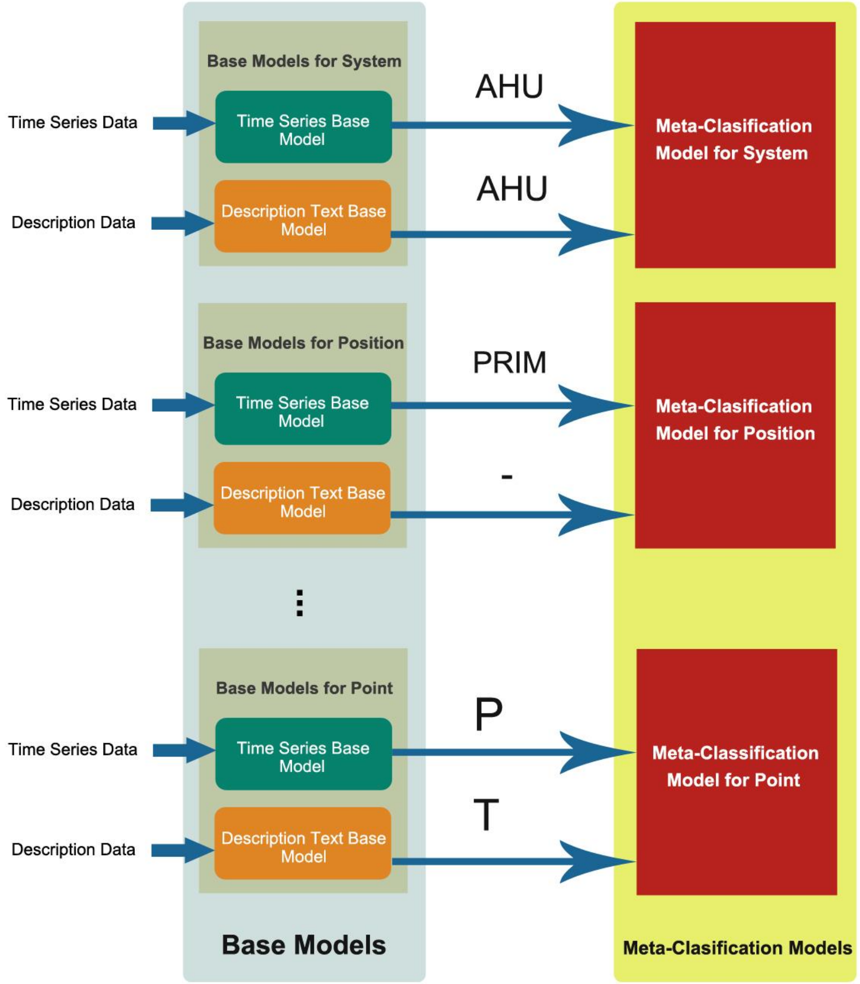


Base Models

- Description Text Base Models
  - Deep Learning Models are used with pre-trained word embedding vectors
  - Word embedding maps the words of a text data, in to a continuous low dimensional vector space such that the internal semantic and syntactic information of the words can be captured.
  - For our system, we use pre-trained vectors from Glove that have been trained on the vocabulary of Common Crawl data and use it as the seeding weights for the embedding layer of the deep learning model.

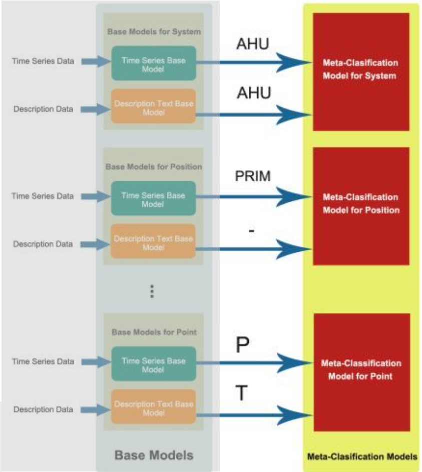


Meta-Classification Models



Meta-Classification Models

- Metaclassifiers are used on the predictions of base models
- The aim is to select the correct meta-data category class label obtained from the two base models
- We used Stacking and Voting as the classification schemes



Meta-classification Models

Stacking

- An ensemble technique which determines the reliability of the classifiers using a meta learner

- A stacked model is trained on the predictions of both base models with original meta-data class label as the target.

Time Series Base Model Prediction	Description Text Base Model Prediction	Original Label
U	U	U
I	T	T
P.EL	P	P.EL

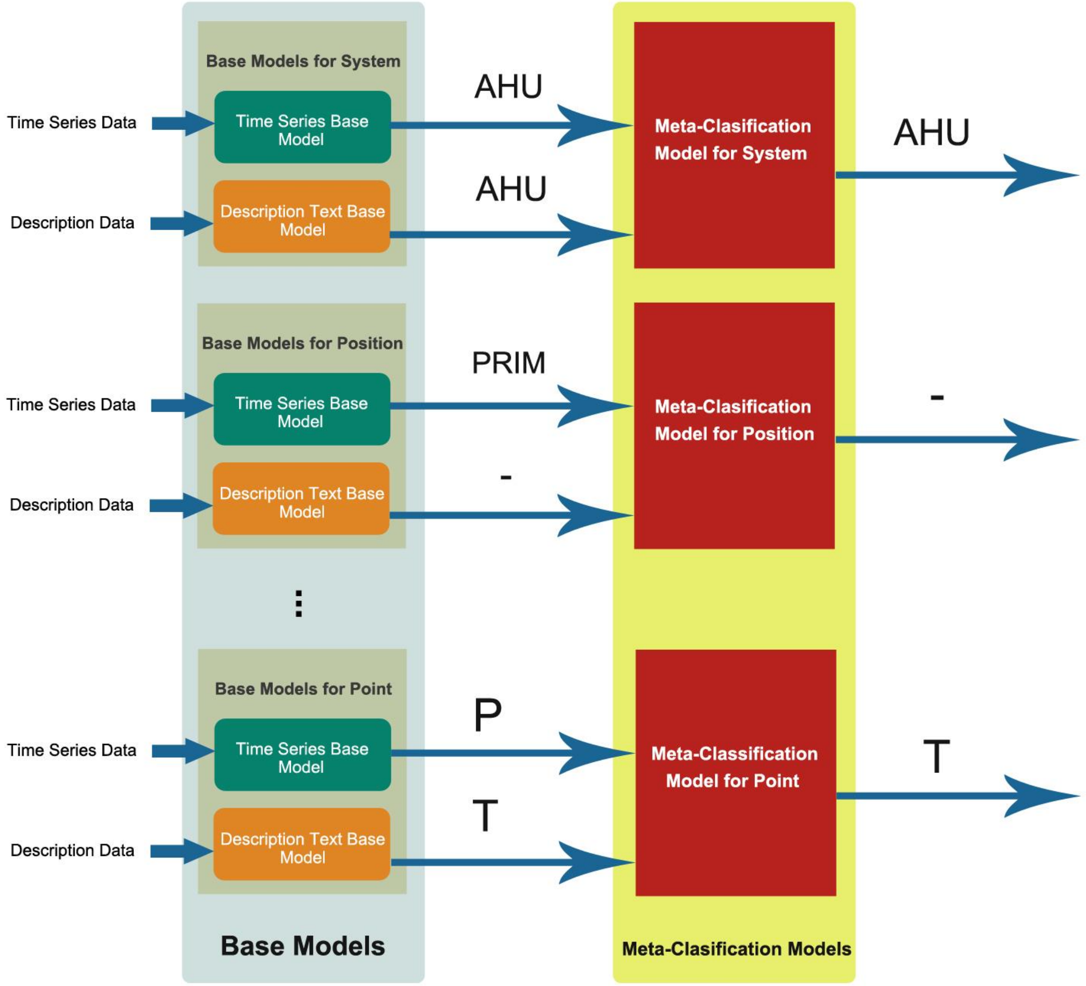
Meta-classification Models

Voting

Classifiers	Class A Pred. Probability	Class B Pred. Probability
Classifier 1	0.6	0.4
Classifier 2	0.5	0.5
Classifier 3	0.1	0.9
Mean	0.4	0.6

Soft Voting

Where are we?


Base Models for System
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
AHU
AHU
Meta-Clasification Model for System
AHU
Base Models for Position
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
PRIM
-
Meta-Clasification Model for Position
-
Base Models for Point
Time Series Data
Time Series Base Model
Description Data
Description Text Base Model
P
T
Meta-Classification Model for Point
T
Base Models
Meta-Clasification Models

Where are we?

The predictions can be used in following two ways:

- The predictions from the meta classifier model of each meta-data category can be concatenated to form a point name label

Where are we?

The predictions can be used in following two ways:

- The predictions from the meta classifier model of each meta-data category can be concatenated to form a point name label

AHU___SUPA__MEA_T

Meta-Classifier Model	Labels
System	AHU
Subsystem1	-
Subsystem2	-
Medium	SUPA
Position	-
Kind	MEA
Point	T

Where are we?

- The predictions from the meta classifier model of each meta-data category can be used to train another machine learning model, which we call the Top Level Model.

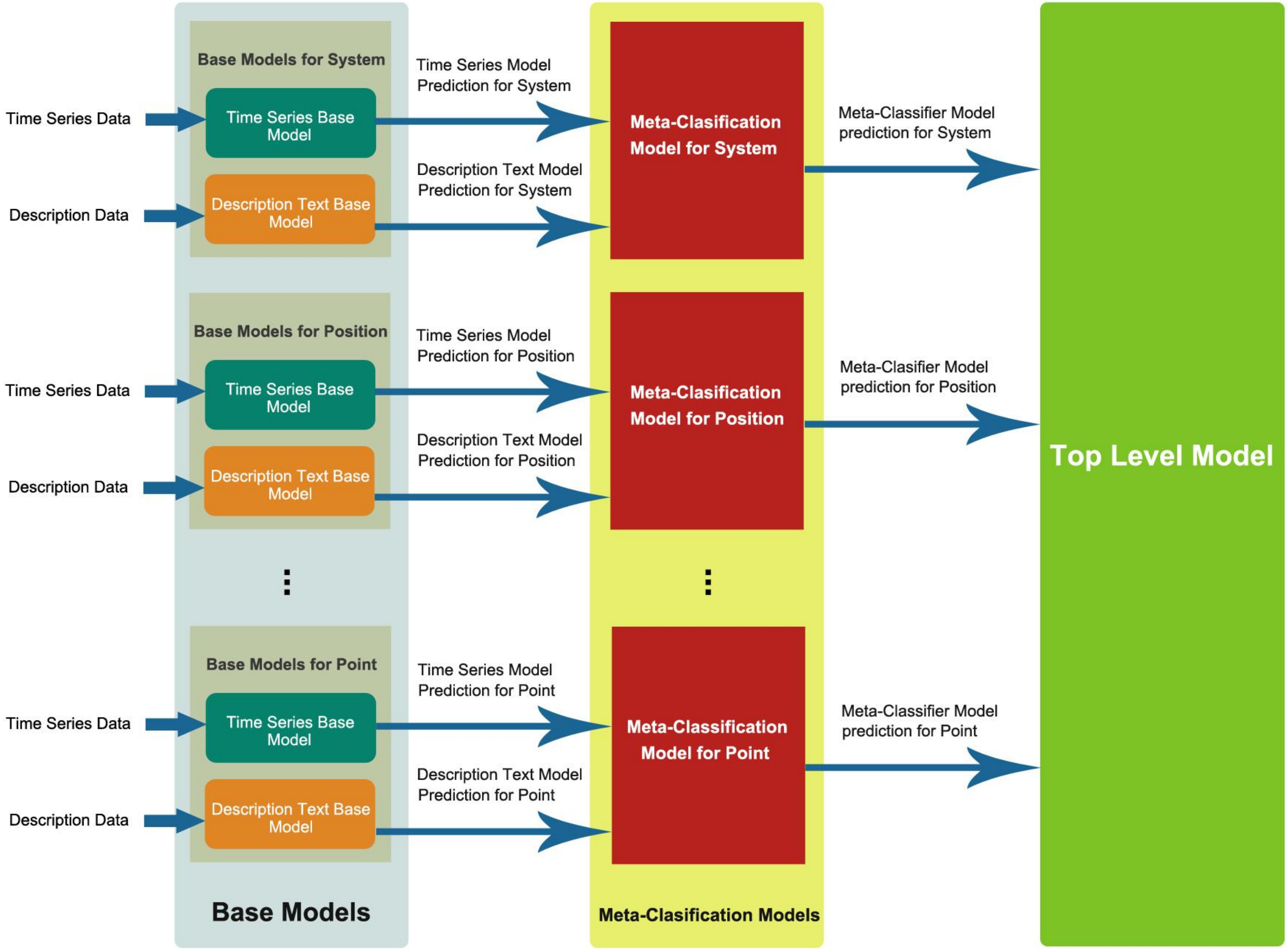
Meta-classifier for System	Meta-classifier for Subsystem1	Meta-classifier for Subsystem2	Meta-classifier for Medium	Meta-classifier for Position	Meta-classifier for Kind	Meta-classifier for Point	Original Point Name Label
-	MTR.EL	MEA	-	-	-	U	AHU_MTR.EL_MEA___U
EGEN.C	CCH	MTR.EL	MEA	-	-	U	EGEN.C_CCH_MTR.EL___MEA_U
AHU	-	MTR.EL	MEA	-	-	U	AHU__MTR.EL___MEA_T

Top Level Model

Why is training a Top Level Model helpful?

- All the meta-data categories are correlated with each other, hence, complementing one another
- For example, a sensor with the **system** entry water circuit will usually not have a **point** category of ’pressure’.
- This pattern can only be recognized if a machine learning model like Random Forest or SVM is trained on the outputs of meta-level classifier

System Architecture



Evaluation: Setup and Main Results

Experiments and Results

We examine different machine learning algorithms, using K fold cross validation on our problem, for each layer of the architecture, and chose the best performing set of algorithms

Experiments and Results

We examine different machine learning algorithms, using K fold cross validation on our problem, for each layer of the architecture, and chose the best performing set of algorithms

Number of classes in the dataset before stratification

Meta-data categories	Number of classes
System	78
Subsystem1	74
Subsystem2	26
Medium	24
Position	39
Kind	13
Point	52

Experiments and Results

We examine different machine learning algorithms, using K fold cross validation on our problem, for each layer of the architecture, and chose the best performing set of algorithms

Number of classes in the dataset before stratification

Meta-data categories	Number of classes
System	78
Subsystem1	74
Subsystem2	26
Medium	24
Position	39
Kind	13
Point	52

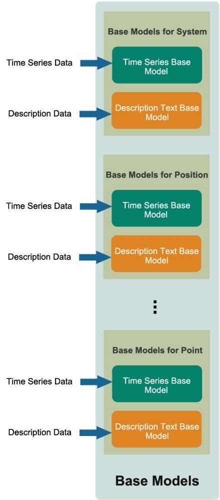
Number of classes in the test set after stratification

Meta-data categories	Number of classes
System	38
Subsystem1	24
Subsystem2	13
Medium	13
Position	11
Kind	7
Point	27

Comparison of Results

- Time Series Base Model: Random Forest Classifier
- Description Text Base Model: CNN with Bi-LSTM

Comparison of Results between Time Series Base Model and Description Text Base Model: Mean Micro-F1 Score

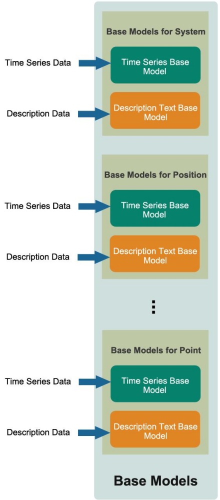


Comparison of Results

- Time Series Base Model: Random Forest Classifier
- Description Text Base Model: CNN with Bi-LSTM

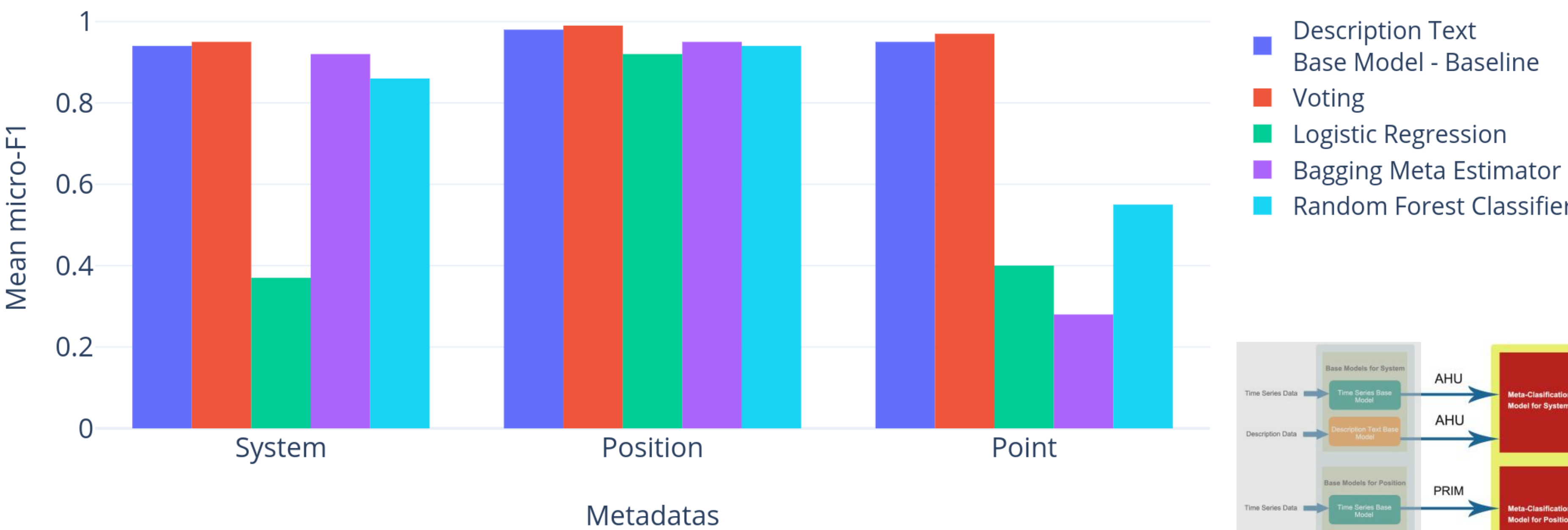
Comparison of Results between Time Series Base Model and Description Text Base Model: Mean Micro-F1 Score

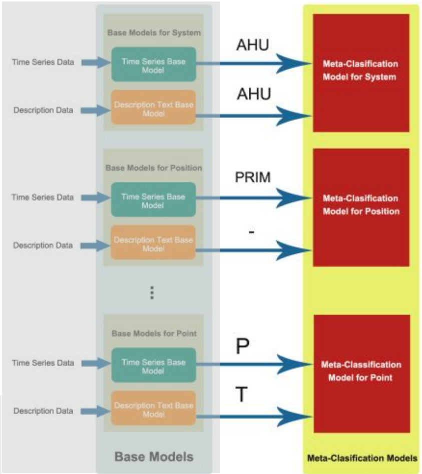
Meta-data categories	Time Series Base Model	Description Text Base Model
System	0.75	**0.94**
Subsystem1	0.87	**0.96**
Subsystem2	0.93	**0.98**
Medium	0.81	**0.95**
Position	0.94	**0.98**
Kind	0.95	**0.97**
Point	0.88	**0.95**



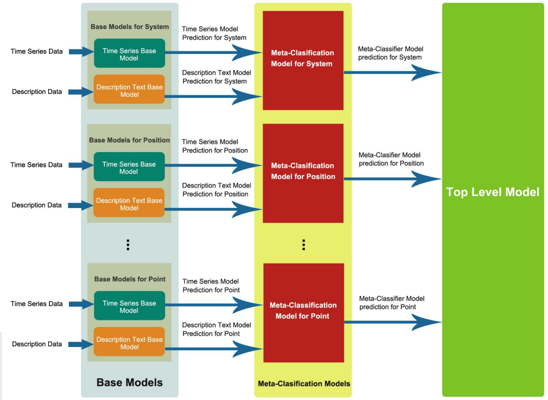
Meta-Classification Model

Voting outperforms all other stacked models



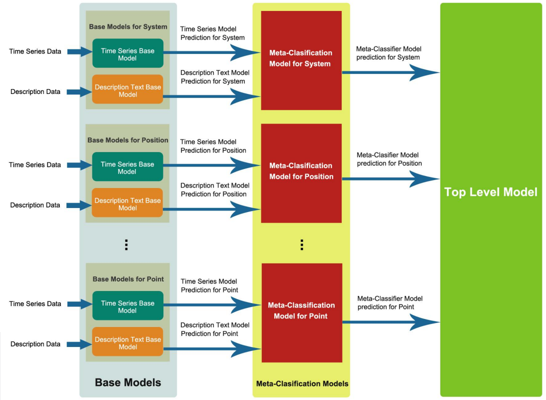


Top Level Model



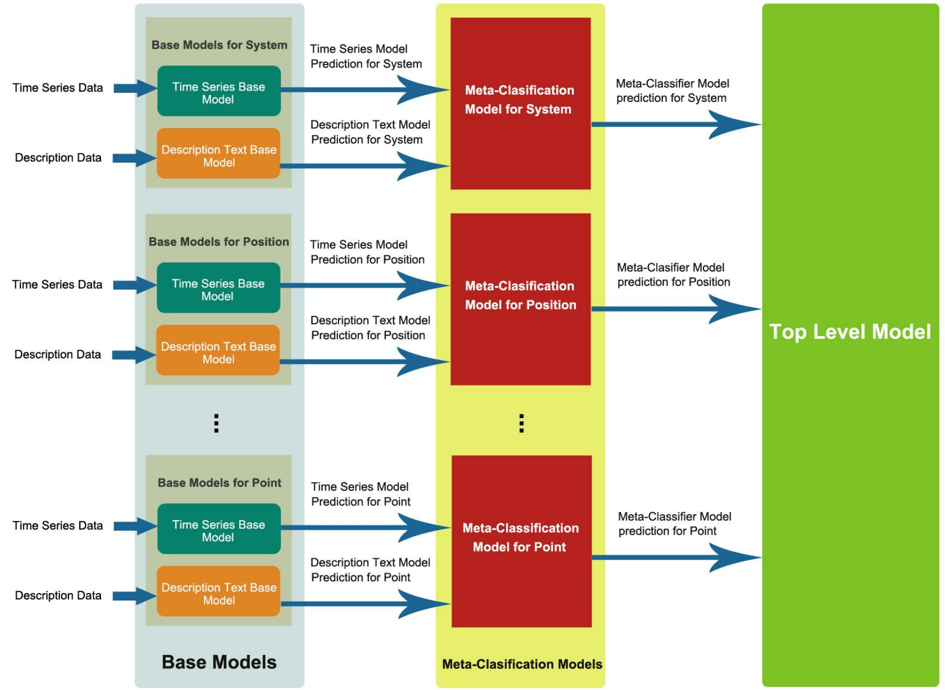
Top Level Model

	Meta-data Categories concatenated	Top Level Model
Micro F-1 Score	0.88	0.88
Accuracy	87.64%	**88.37%**

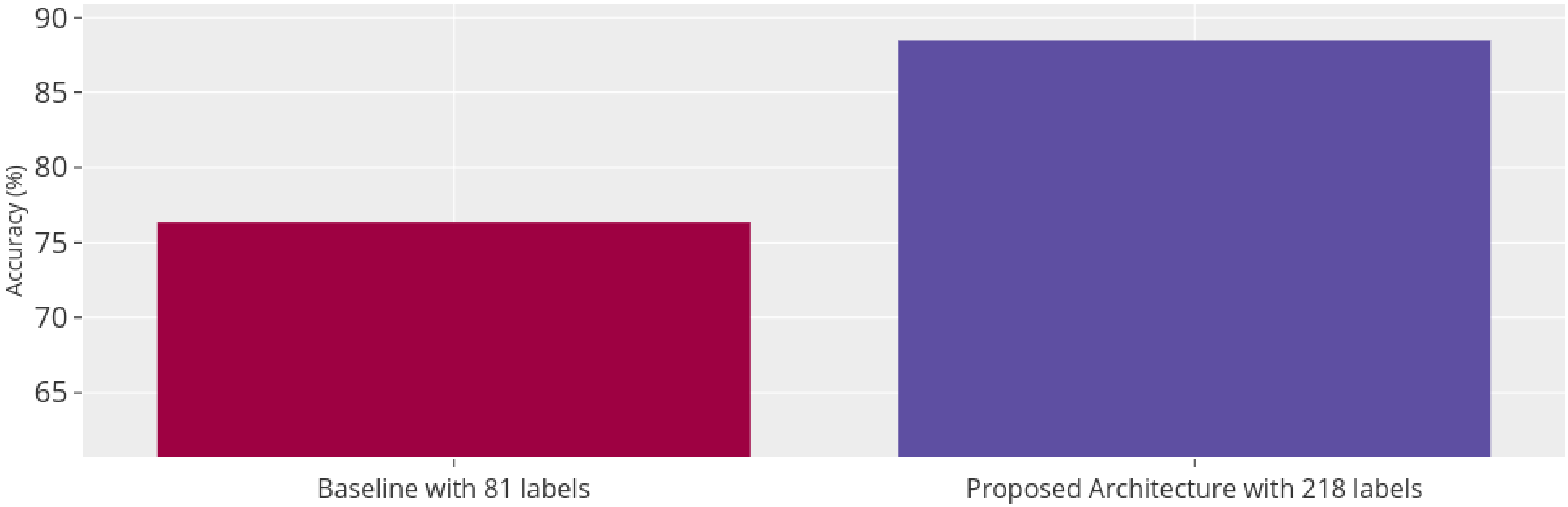


Final Algorithms

Architecture Layer	Algorithms
Time Series Base Model	Random Forest Classifier
Description Text Base Model	CNN with Bi-LSTM
Meta-Classification Model	Voting (Soft)
Top Level Model	Random Forest Classifier



Comparison with Baseline



Inter-Building Cross Validation

- The geographical location of a building as well as the usage characteristics and the detailed control strategies (like time schedules) play a vital role in the measured value of the sensor of that building.

- Different system combinations and controls, the data logging mechanisms and quality may be different from one building to another

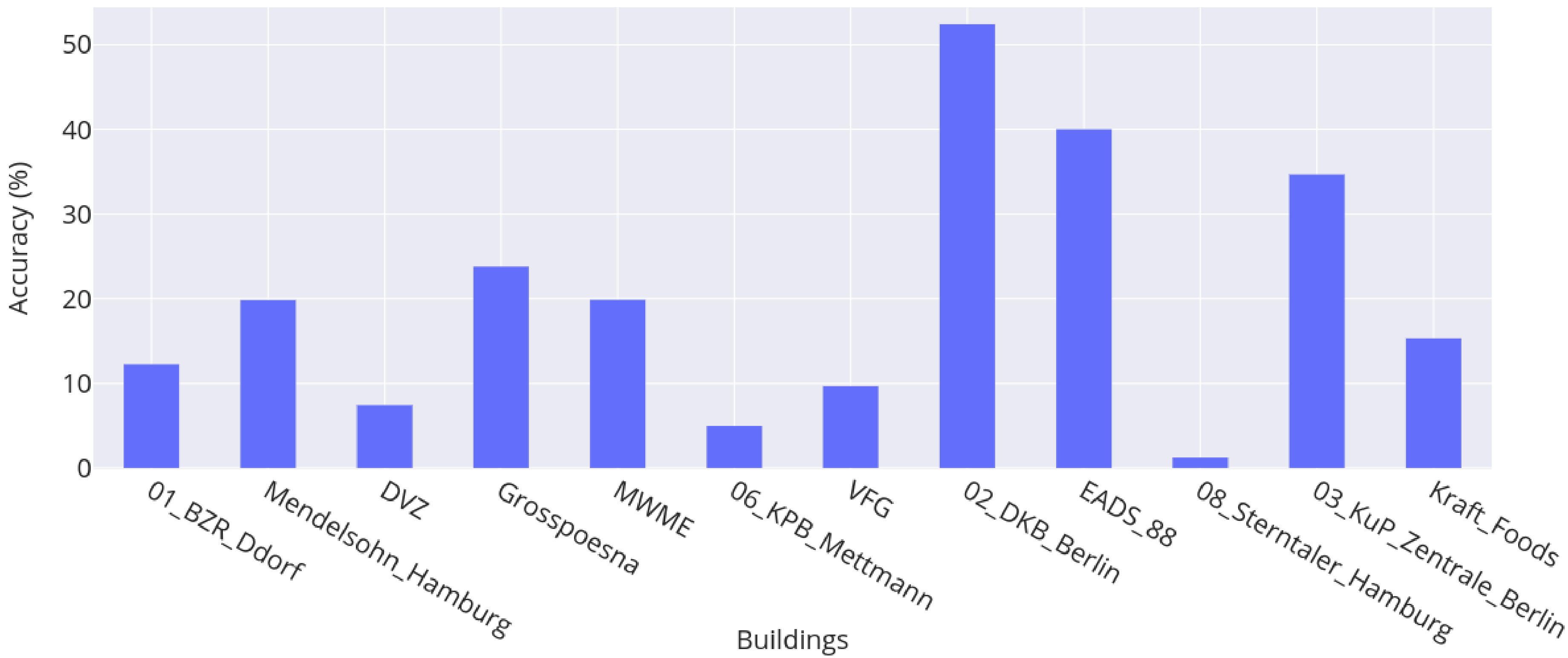
- The performance of the architecture might be greatly affected with the inclusion of an unseen building in the testing set

Inter-Building Cross Validation

- We create 13 different sets of sensors, each set corresponding to one building and use one set for testing while using all other 12 sets for training the architecture

- The process is repeated until every set is used for testing

Inter-Building Cross Validation

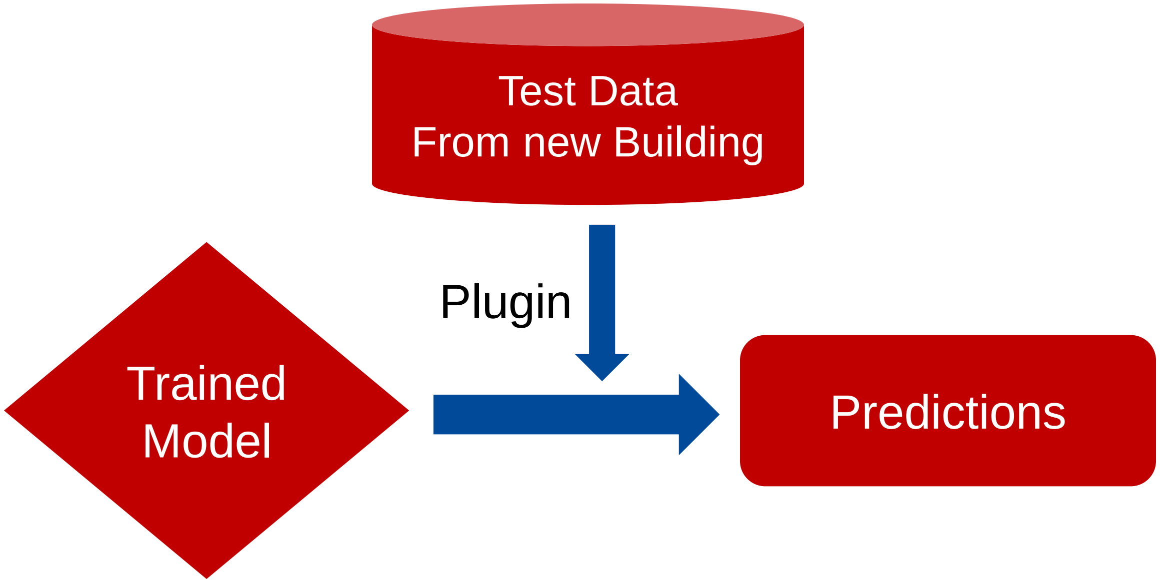


Simulation of User Feedback

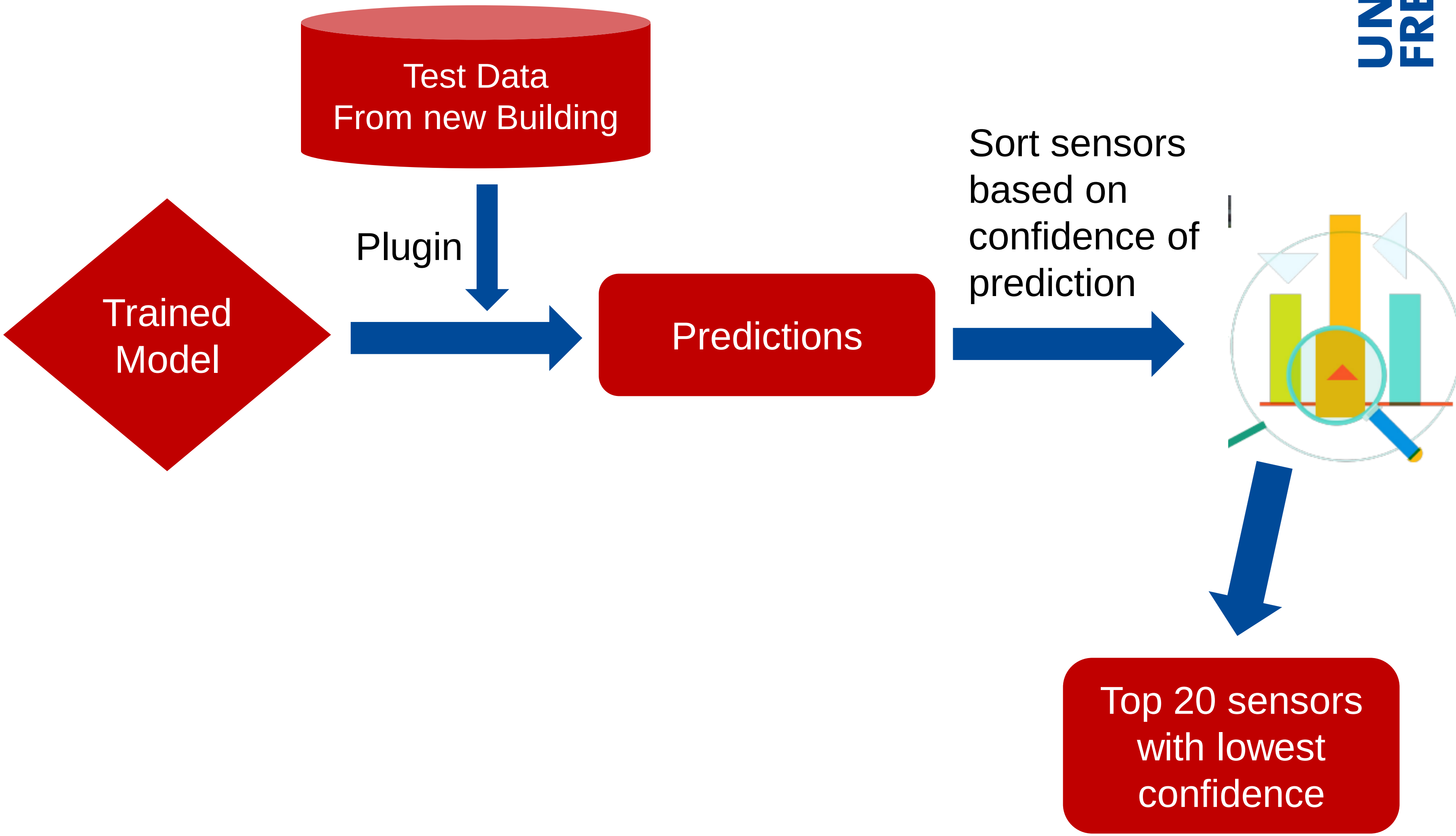
Simulation of User Feedback

Trained Model

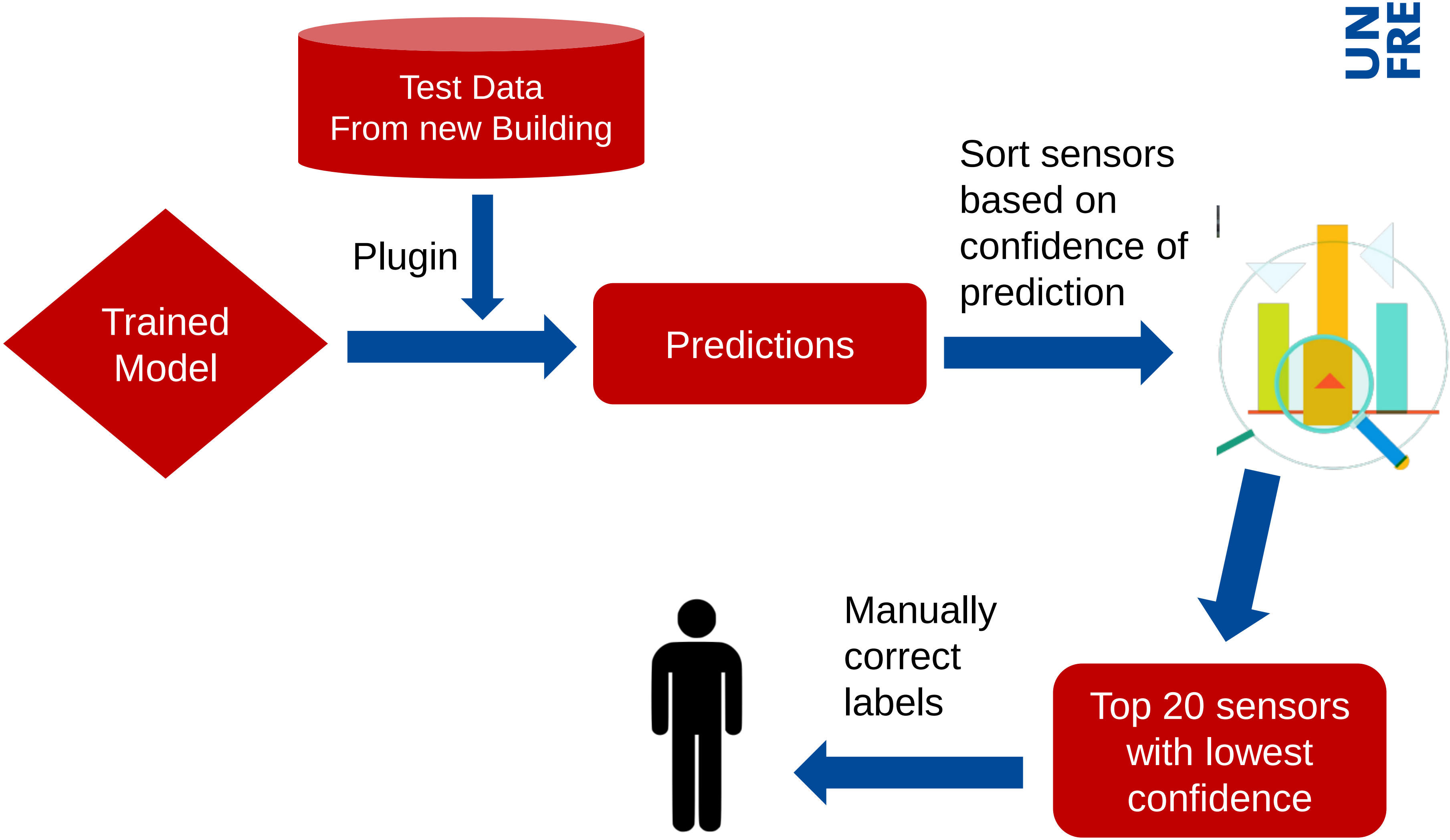
Simulation of User Feedback



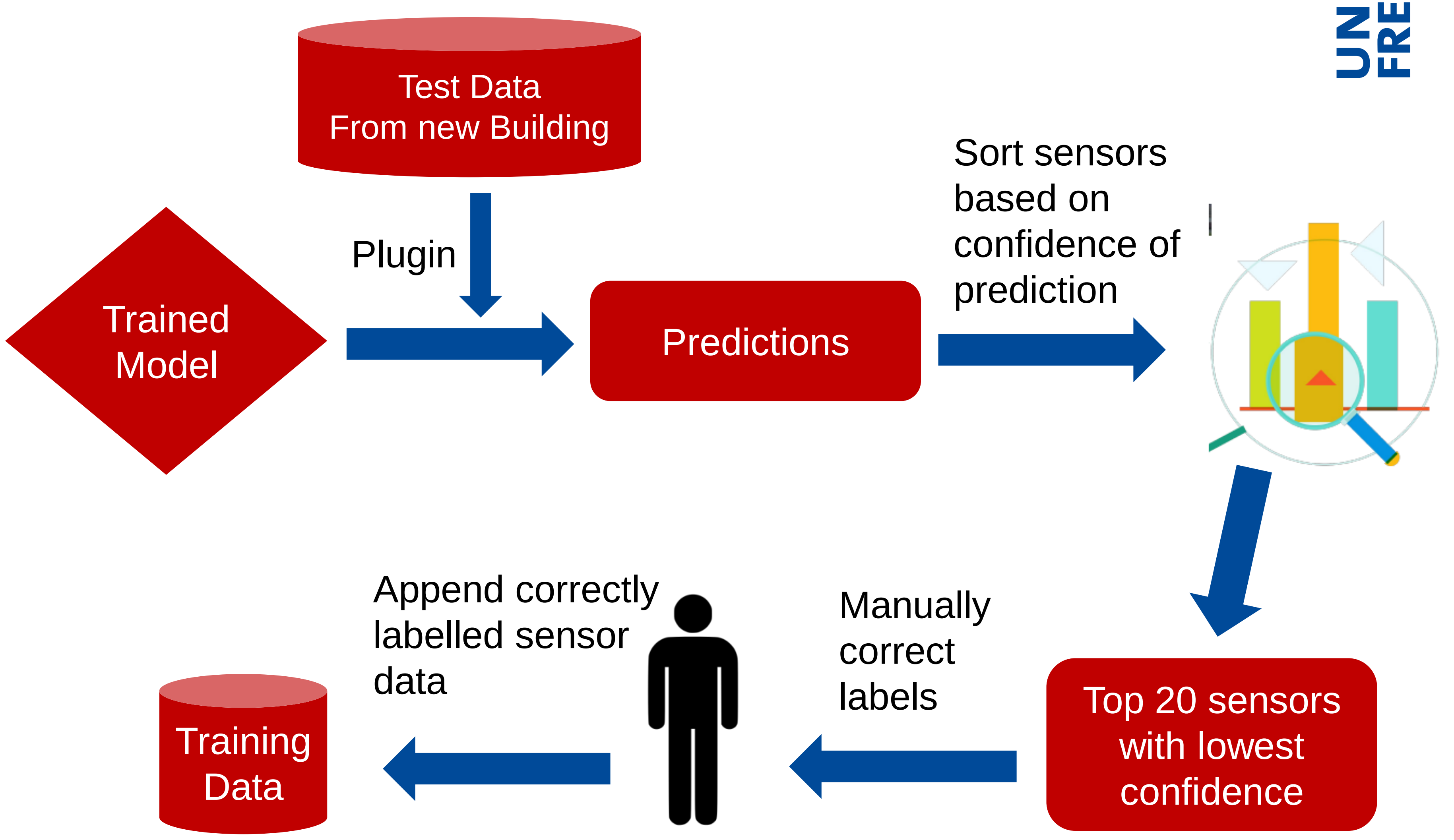
Simulation of User Feedback



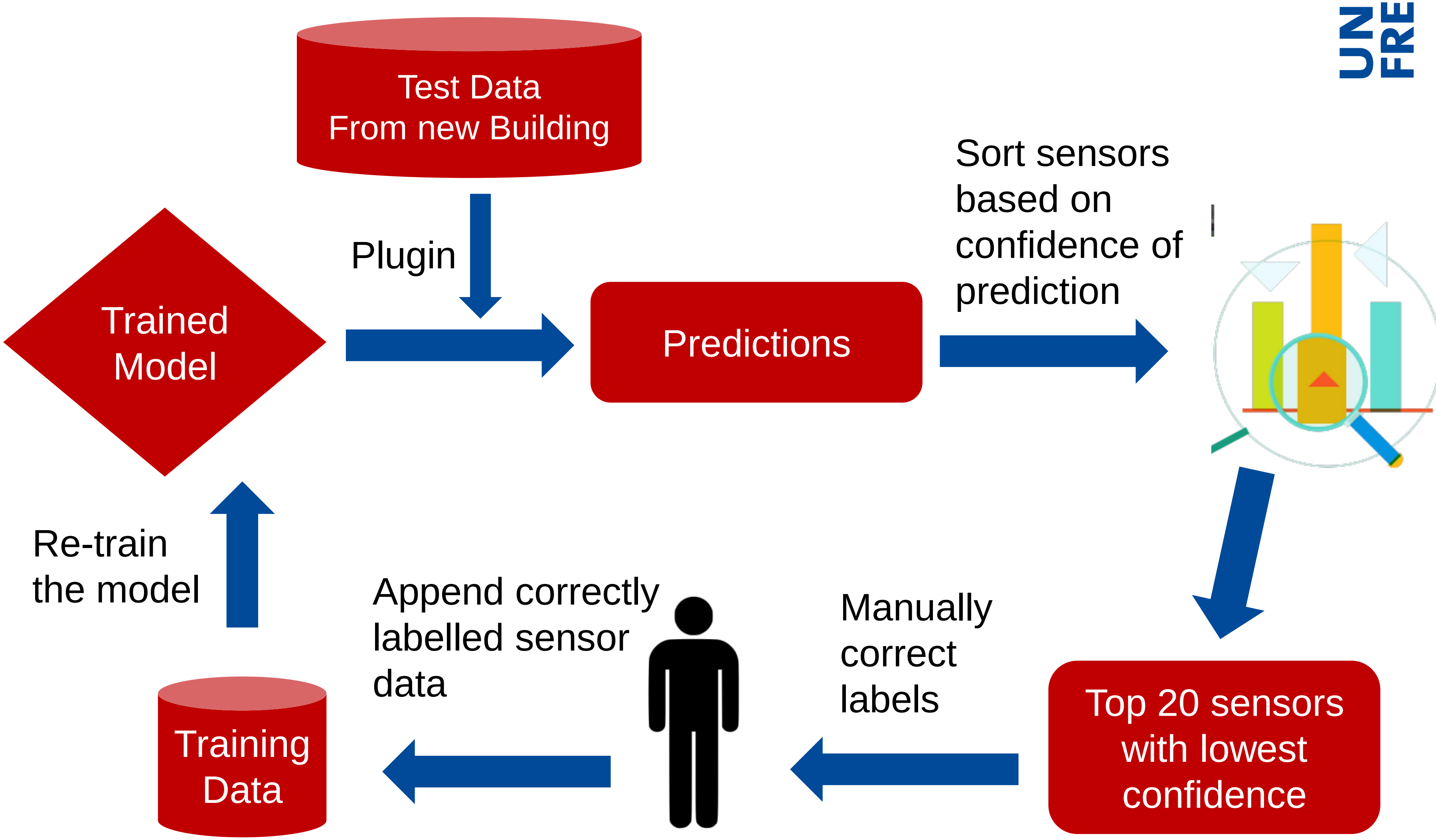
Simulation of User Feedback



Simulation of User Feedback



Simulation of User Feedback



Usefulness

- To evaluate the performance of the proposed approach, metrics like accuracy, precision or recall do not reflect the problem domain well

- All these metric treat all the categories of the point name label as a single class

  For example:
  - Predicted Label: AHU__SUPA__MEA_**P**
  - Correct Label: AHU__SUPA__MEA_**T**

Usefulness

- To evaluate the performance of the proposed approach, metrics like accuracy, precision or recall do not reflect the problem domain well

- All these metric treat all the categories of the point name label as a single class

  For example:

  - Predicted Label: AHU__SUPA__MEA_**P**
  - Correct Label: AHU__SUPA__MEA_**T**

- To measure the number of meta-data categories correctly identified in a label, we introduce a new measure called **‘Usefulnes’**

Usefulness

Usefulness Measure = 1 − (Number of categories changed /Total Number of Categories)

Usefulness

Usefulness Measure = 1 − (Number of categories changed /Total Number of Categories)

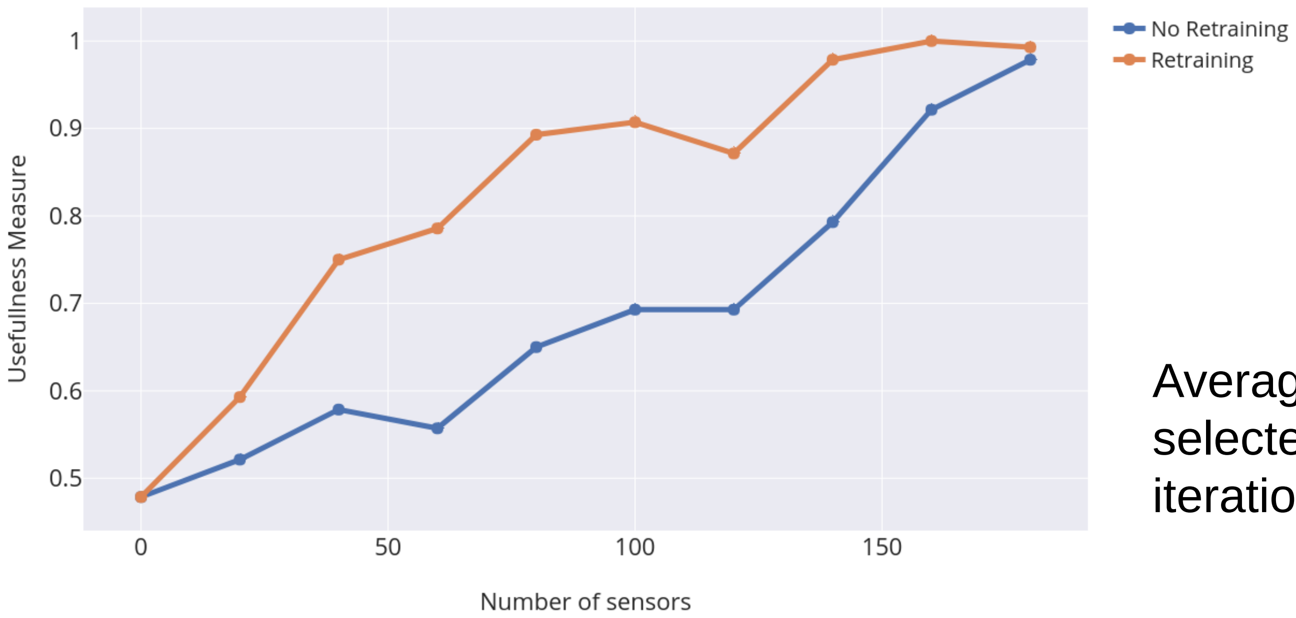
- For example:
    - Predicted Label: AHU__SUPA__MEA_**P**
    - Correct Label: AHU__SUPA__MEA_**T**

Usefulness

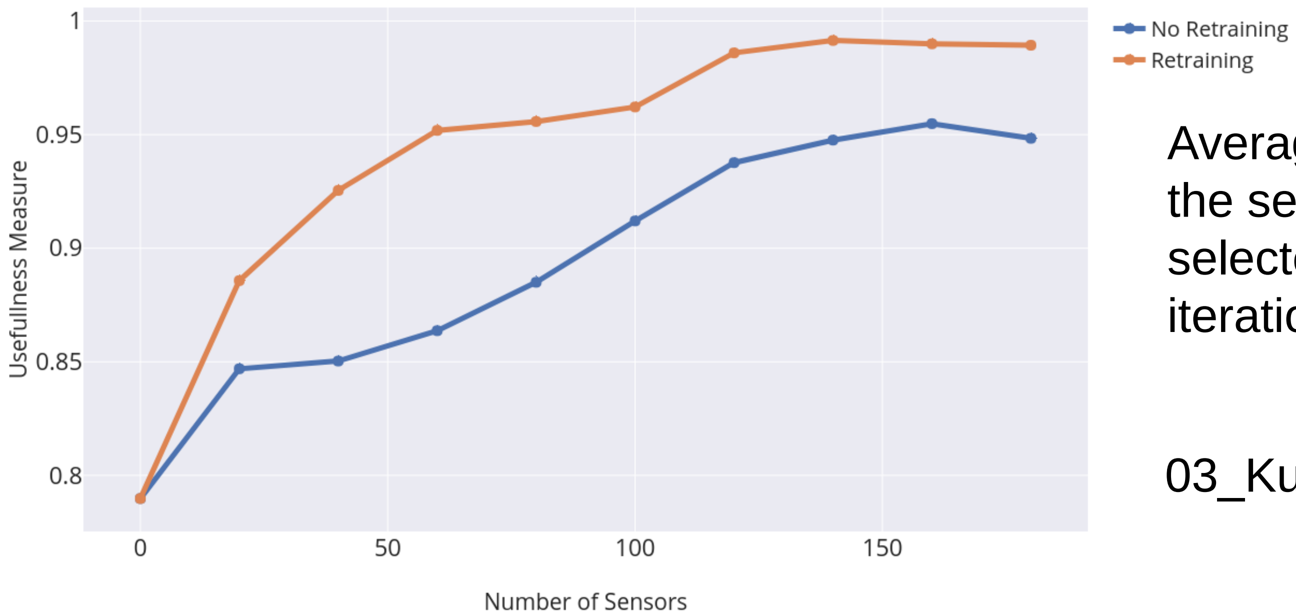
Usefulness Measure = 1 − (Number of categories changed /Total Number of Categories)

- For example:
  - Predicted Label: AHU__SUPA__MEA_**P**
  - Correct Label: AHU__SUPA__MEA_**T**

Usefulness = 1 – (1/7) = 0.86



Average Usefulness of 20 selected sensors at every iteration



Average Usefulness of all the sensors except 20 selected sensors at every iteration

03_KuP_Zentrale_Berlin

Conclusion

- The proposed architecture is able to perform better than the existing approach at Fraunhofer ISE
- Combining data from two different sources of input produces better results that just using a single data source.
- We also tested our system towards a more practical simulated real world scenario where we evaluated the performance of our model on the data of new buildings using a new performance measure, which we called the 'Usefulness Measure
- Re-training the model in an iterative fashion makes the model more biased but better adapted to the building and helps classifying most of the sensor data of a new building to their correct labels

Future Work

- In future, data needs to be reviewed and divided in to further sub-classes
- The hardware constraints do not fully automate the operations of the system as deep learning and statistical machine learning models have to be trained on different machines, satisfying the hardware requirement
- This constraint can be resolved if the system is operated on machines having GPU and large capacity of RAM.

Thank you for your attention.

References

- E. R.V.Lojini Logesparan, AlexanderJ.Casson, “Optimal features for online seizure detection,” *Medical & biological engineering & computing*, vol.50, pp.659– 669, 2012.
- S. Palanisamy, “Automated extraction of data point names,” Master’s thesis, Chair of Machine Learning Dept.of Computer Science, Faculty of Engineering, Albert-Ludwigs-Universität Freiburg, 2017.

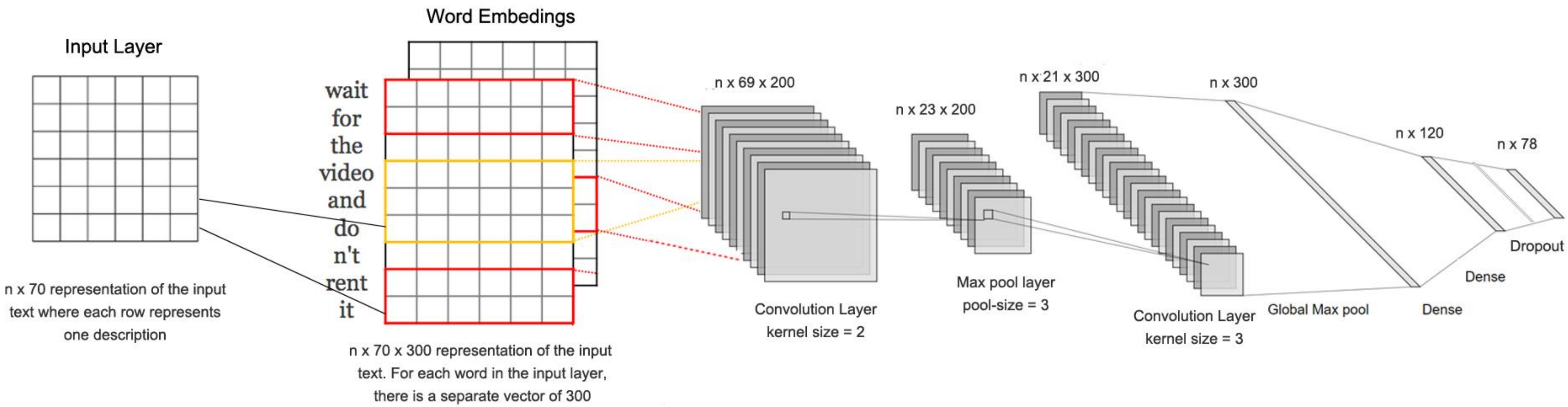
Backup Slides

Base Models

- Description Text Base Model: CNN with Bi-LSTM produced best results
  1. Convolutional Neural Networks
  2. Convolutional Neural Networks with Bidirectional LSTM
- The input to both the architectures of the Description Text Base Model are the word vectors obtained through pre-trained Word Embeddings from Embedding Layer

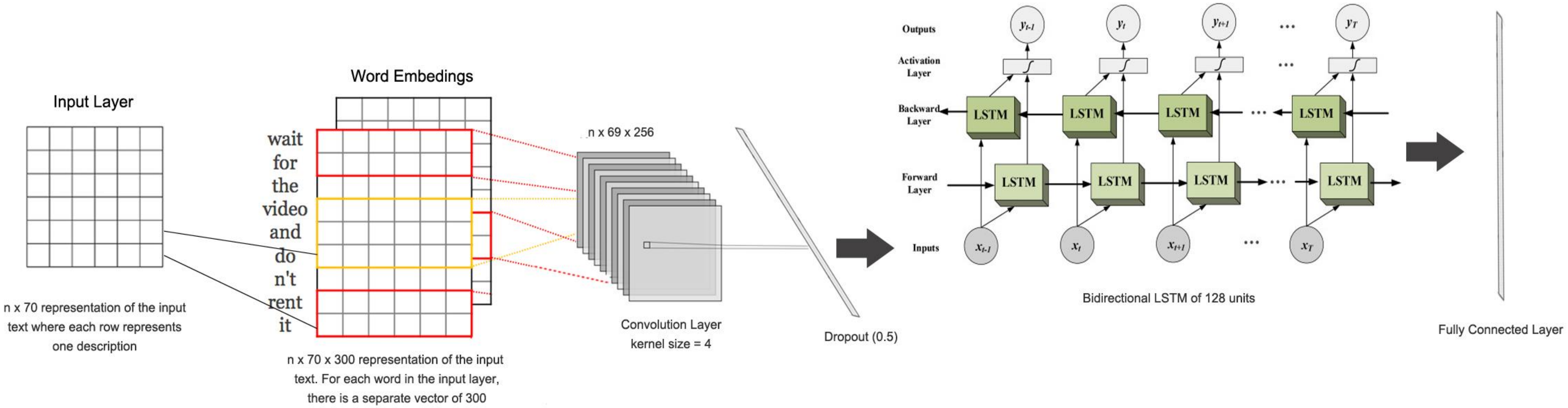
Description Text Base Model

- Architecture 1: CNN



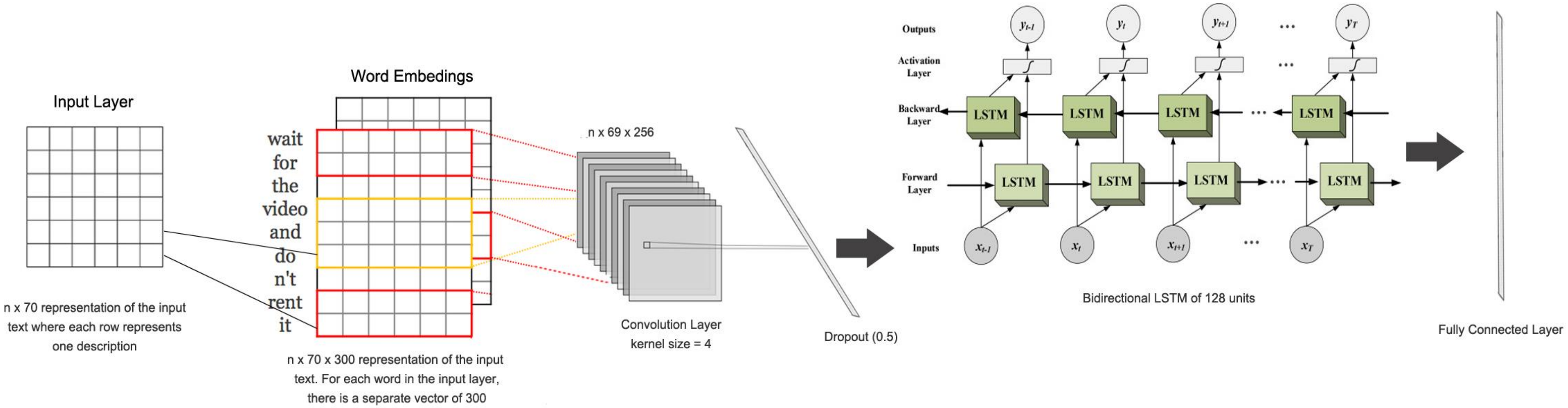
Description Text Base Model

- Architecture: CNN with Bi-LSTM



Description Text Base Model

- Architecture 2: CNN with Bi-LSTM



Description Text Base Model

- Results

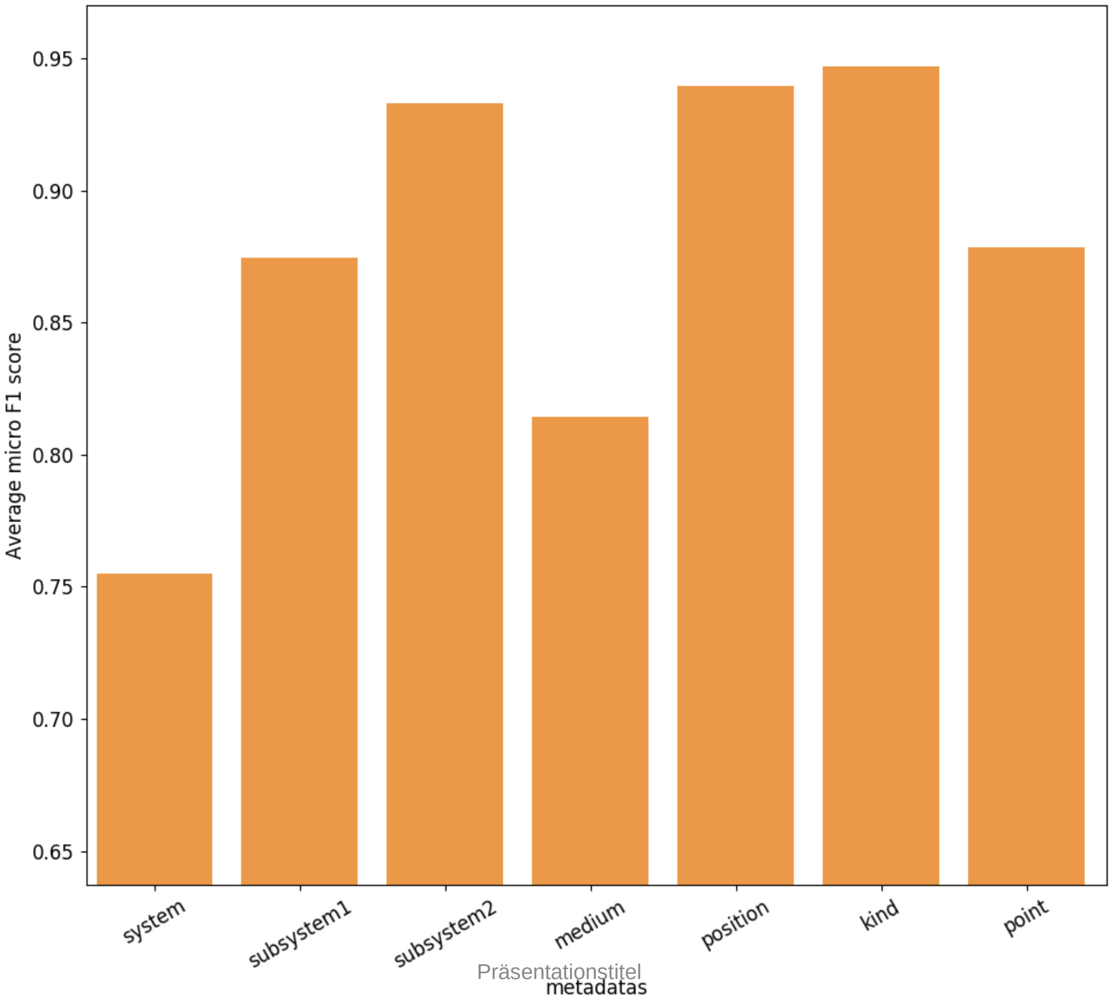
Description Text Base Model

- Results

Meta-data categories	Architecture 1 - CNN	Architecture 2 – CNN - BiLSTM
System	0.92	**0.94**
Subsystem1	0.96	0.96
Subsystem2	0.97	**0.98**
Medium	0.94	**0.95**
Position	0.98	0.98
Kind	0.97	0.97
Point	0.93	**0.95**

Base Models

Time Series Base Model: Random Forest Classifier



Base Models

- Description Text Base Model: CNN with Bi-LSTM produced best results

- The input to the architectures of the Description Text Base Model are the word vectors obtained through pre-trained Word Embeddings from Embedding Layer

Meta-Classification Model

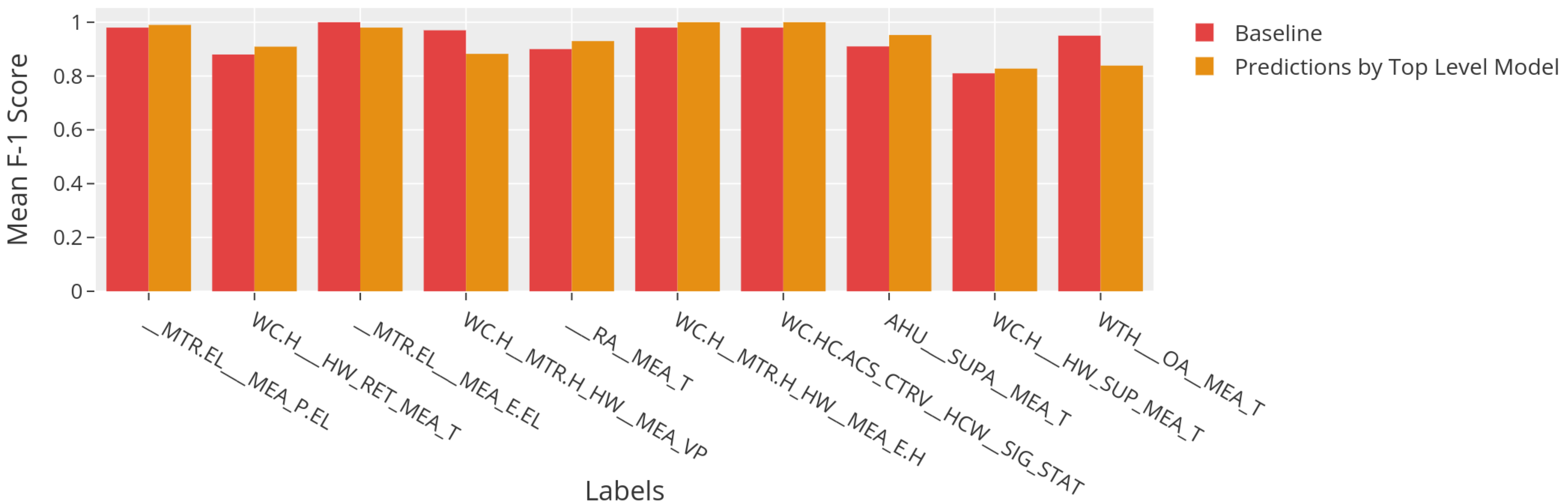
- We use Stacking and Voting as meta-classification scheme
- For Stacking, we used Logistic Regression, Bagging meta-estimator with Decision Trees as base classifier and Random Forest Classifier
- For Voting, we used soft voting method
- Description Text Base Model Results are taken as baseline

Components of Point Name Label

No.	Category	Remark
1	System	Main system to which sensor belongs to
2	Subsystem1	If appropriate: subsystem of system
3	Subsystem2	If appropriate: subsystem of subsystem1
4	Medium	Medium in which the sensor is placed
5	Position	Position of the sensor
6	Kind	Kind of the data point
7	Point	The physical quantity which is measured by the sensor

Comparison with Baseline

- Top 10 labels: F1-score comparison



Word Embedding