Supervised Learning of Units of Measure

Please Access the Interactive Poster Online

https://jacob-barhak.github.io/Poster_MSM_ML_IMAG_2019.html

Supervised Learning of Units of Measure

By: Jacob Barhak
& Joshua Schertz

Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

Abstract

U.S. law requires registration of clinical trial data in ClinicalTrials.Gov. This NIH/NLM governed registry contributed much towards providing important modeling data information by accumulating over 300,000 clinical trials. However, despite the great effort by the government to centralize the data, the entities reporting data do not follow a predetermined standard. Therefore, numerical information entered is machine readable, yet not machine comprehensible, especially due to units being entered as free text. If a machine cannot comprehend the units, it cannot comprehend the numbers. This causes human intervention in the modeling process - slowing down modeling and the uses of this important registry.

The extent of the problem requires some machine learning, as of 12 Apr 2019, all 35,926 trials with results had 24,548 different units. The authors created solution infrastructure to address this problem. The solution includes:

1) Data extraction tools for ClinicalTrials.Gov that can index data and assemble clusters of data with unsupervised learning.

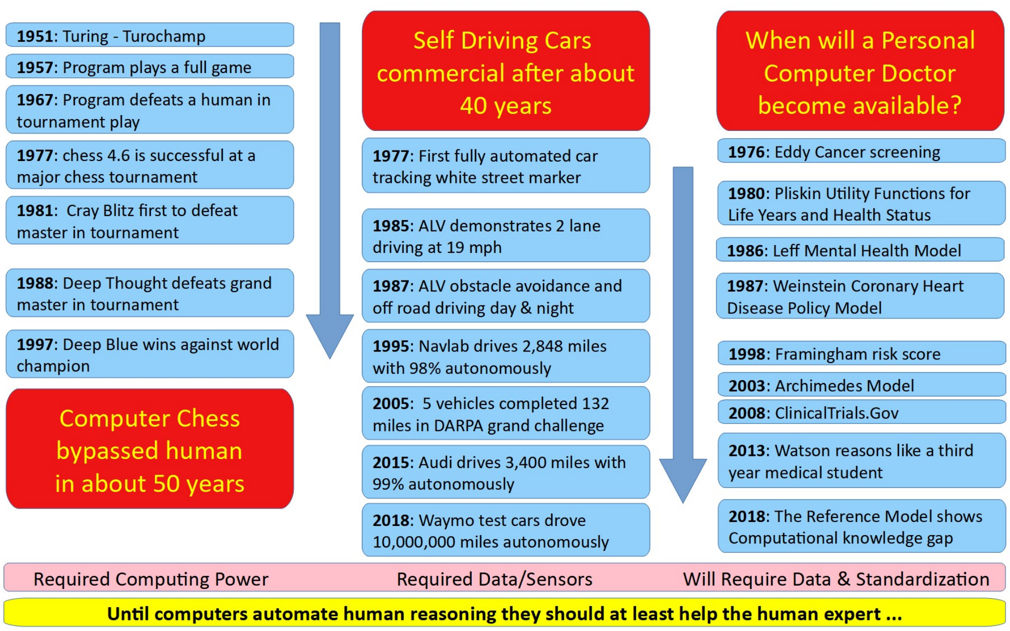
2) ClinicalUnitMaping.Com : a website for unit mapping that also demonstrates the extent of the problem.

3) A collection of existing unit standards used for medical purposes that currently holds data from CDISC, NIST / RTMMS / IEEE, Unit Ontology / Bio Portal, UCUM.

4) Supervised Machine Learning using neural networks that can predict the standardized unit given a non standard unit.

The supervised machine learning techniques are new, and their development involved many technical aspects and many attempts to solve the problem. This publication will discuss the difficulties and summarize multiple attempts, architectures, and solutions to resolve the problem.

Longer Term Motivation: Computer Automation of Human Reasoning



Supervised Learning of Units of Measure

By: [Jacob Barhak](#)
& [Joshua Schertz](#)

Venue: [Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)](#)

Proposed Solution

1. Aggregate and index all ClinicalTrials.Gov units

2. Gather auxiliary unit standards / specifications:

- [CDISC](#) - Clinical Data Interchange Standards Consortium

- [RTMMS](#) - affiliated with NIST / IEEE / ISO

- [Unit Onthology](#) from BioPortal (BIOUO)

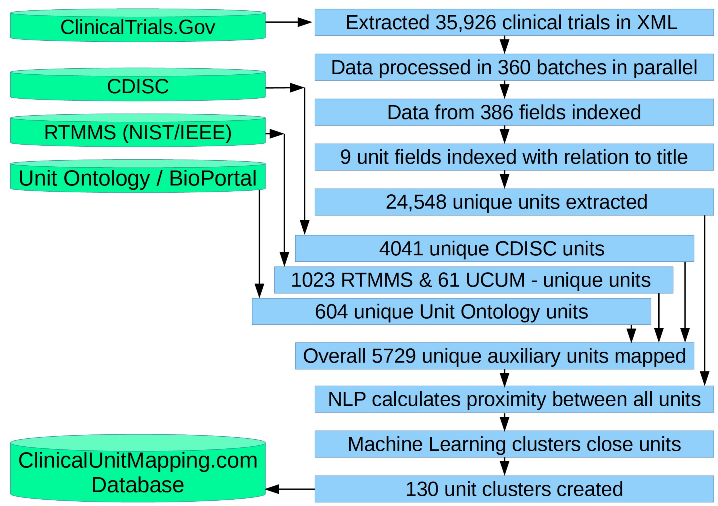
- [UCUM](#) - The Unified Code for Units of Measure (RTMMS / CDISC)

3. Use python tools to:

- Find unit proximity using unsupervised Machine Learning and Natural Language Processing (NLP)

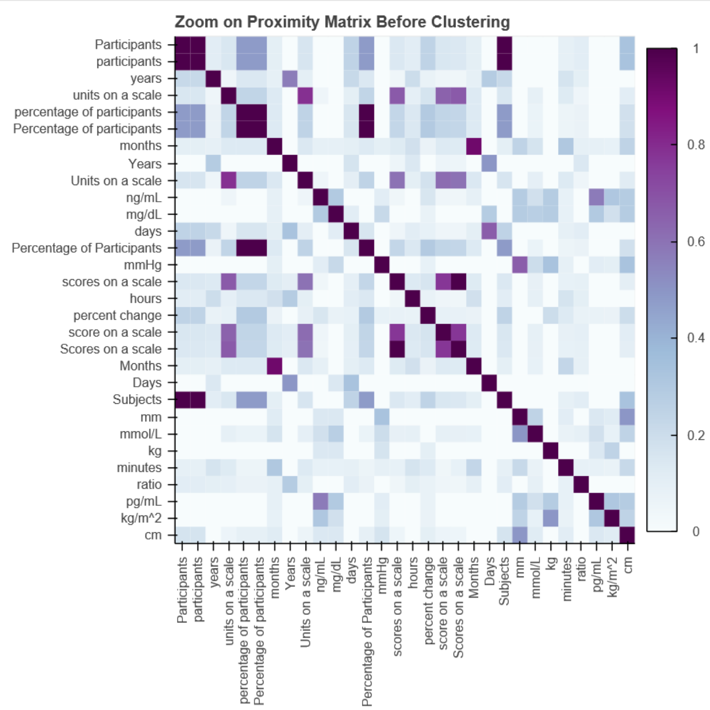
- Create a web site for crowd mapping of the unit corpus

- Create supervised learning technique to comprehend units



Unit Proximity with NLP + Clustering

Zoomed Unit Proximity | Before Clustering | After Clustering



Collaborative Unit Mapping Web Site

The web site is accessible using [ClinicalUnitMapping.com](#)

- A reduced database was used for demonstration purposes

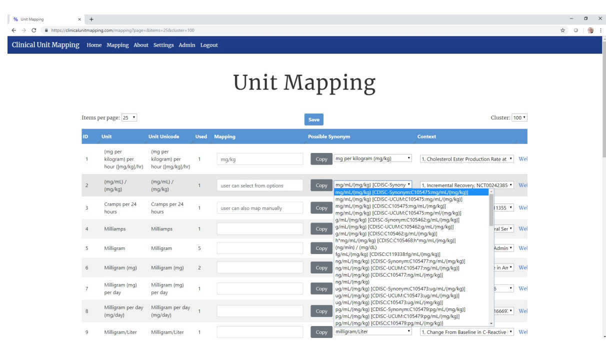
- An administration system allows multiple user management

- Similar units clustered together and user can switch clusters

- Unit context and statistics displayed

- User can map units using user or machine suggested units

- Highlighted auxiliary units: RTMMS / CDISC / UCUM / BIOUO



Supervised Machine Learning for Mapping units

Difficulties:

- There are too many target units to use ordinary classification
- Many units map to the same result so the translation is many-to-one rather than one-to-one
- Data distribution is unbalanced with many examples for some mappings
- Context of units has a large vocabulary
- Training data is limited - although growing in time

Multiple Solutions Attempted

Solution	Main Layers	Encoding	Comments	References
Simple Classification	Dense	One Hot / Feature	Simple solution, yet this problem has many classes and therefore not practical	1, 2
Feature Classification	LSTM / CNN	One Hot	Can be simple and fast yet requires mapping and sensitive and complex features require dealing with sequences	3
Sequence to Sequence Preset Length	LSTM / CNN	One Hot / Embedding	Relatively simple flexible and reliable, training reasonable, and inference is reasonably fast	4, 5, 6
Sequence to Sequence Encoder/Decoder	LSTM	One Hot / Embedding	Works well for short sequences, non trivial implementation. However slow inference since GPU is not used in decoding	5, 7, 8, 9, 10
Learning to Rank - Pairwise	CNN + Dense Twin	Embedding	High complexity O(N^2) difficult inference due to pairwise nature	11, 12, 13, 14, 15

Chosen Solution Implemented Sequence to Sequence Networks so the Modeler Can:

- Control:
  - switch neural network architecture
  - decide on execution mode: New Netowrk, retrain old, or just test or plot using previously trained network
- Preprocessing:
  - add noise to input units simulating typing errors, and control noise type
  - decide if to duplicate dataset without one or more inputs to test missing unit context
- Neural Network Inputs:
  - decide if to use unit input as one hot
  - decide if to use unit input as integer for embedding
  - decide if to use unit context as input
  - decide if to add input attention to unit input when using both one hot and integer
- Training:
  - decide network layer sizes and network depth
  - decide if to use LSTM or CNN for context
  - decide if to use LSTM or CNN for Units - only LSTM in Encoder/Decoder architecture
  - set dropout rate for LSTM
  - determine input data clusters used when training - automate multiple networks for multiple clusters
- Debug:
  - request accumulated training history even when retrained
  - look inside network layers of interest during training - this is beyond tensorboard support - implemented with PyViz
- Post-Processing:
  - choose inference from between best Validation model. last trained model, or both
  - decide on verbosity of output - e.g. number of closest units to output
- More Details:
  - Mock training data mapped 24,548 units to 6,891 mock interpretations derived from clustering.
  - In addition, post processing matched char sequence output to allowed unit interpretations within the same cluster.
  - Closest units with a certain distance from prediction were explored for accuracy.
  - Multiple distance metrics were used to deduce closest unit to predicted string.

Supervised Learning of Units of Measure

By: Jacob Barhak
& Joshua Schertz

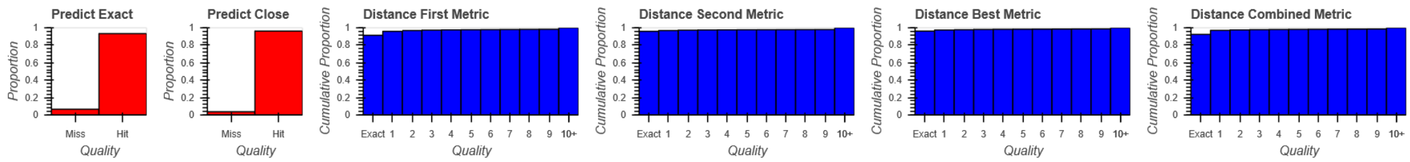
Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

Neural Network Training and Validation Results For Multiple Architectures

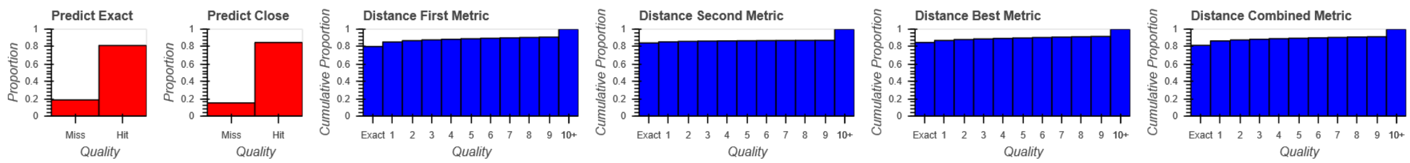
Encoder Decoder | LSTM Unit LSTM Context | LSTM Unit CNN Context | CNN Unit CNN Context

Sequence to Sequence Encoder Decoder

Training Unit & Context

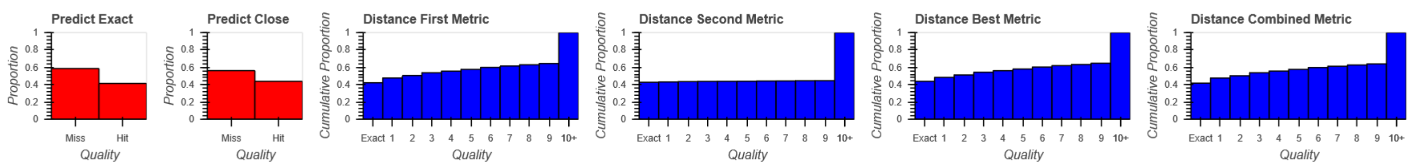


Training Unit Only

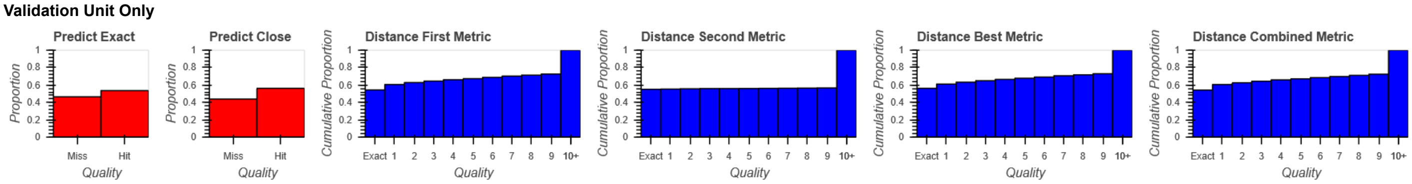


Training Context Only

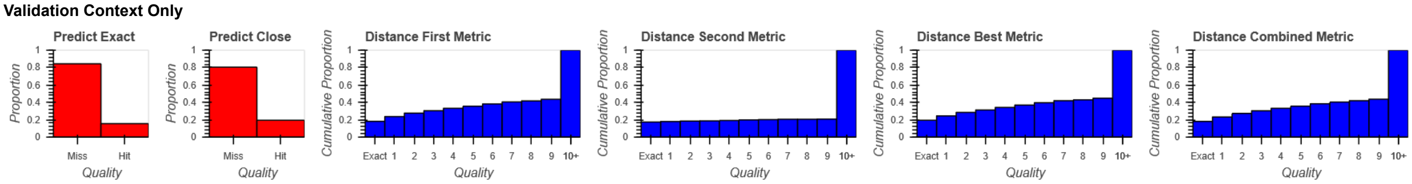
Validation Unit & Context



Validation Unit Only



Validation Context Only



Rows = input scenarios: unit / context / both. Columns = comparison metrics: Red= first prediction attempt accuracy. Blue = prediction quality with multiple close units.

Supervised Learning of Units of Measure

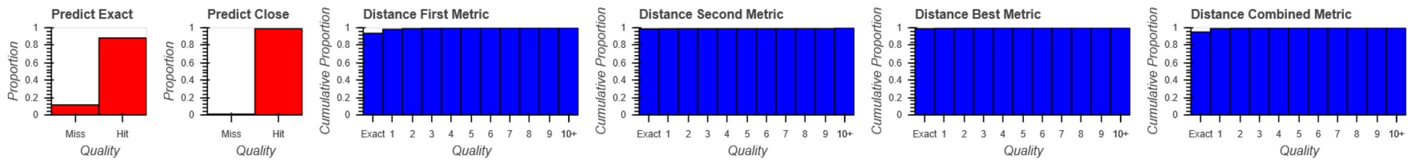
By: Jacob Barhak
& Joshua Schertz

Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

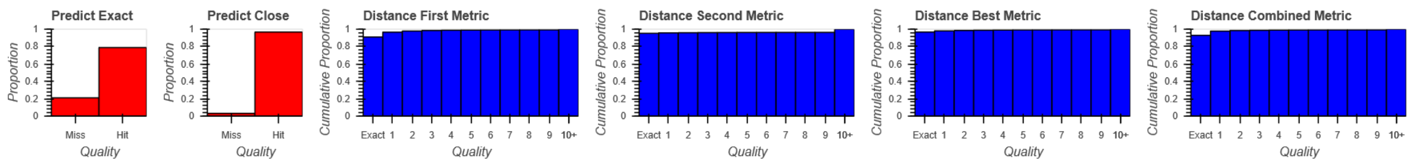
Neural Network Training and Validation Results For Multiple Architectures

Sequence to Sequence LSTM for Unit LSTM for Context

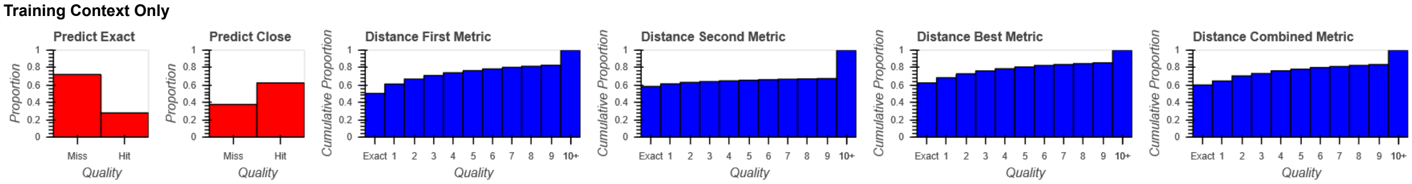
Training Unit & Context



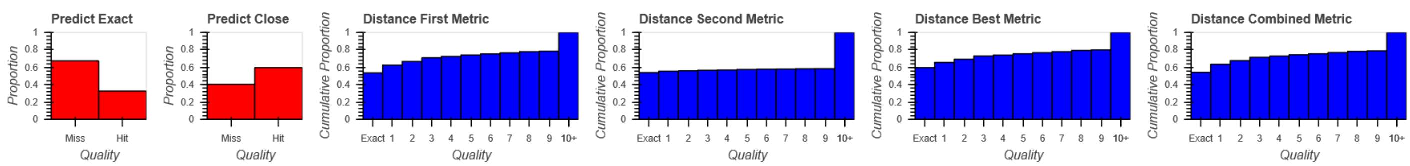
Training Unit Only



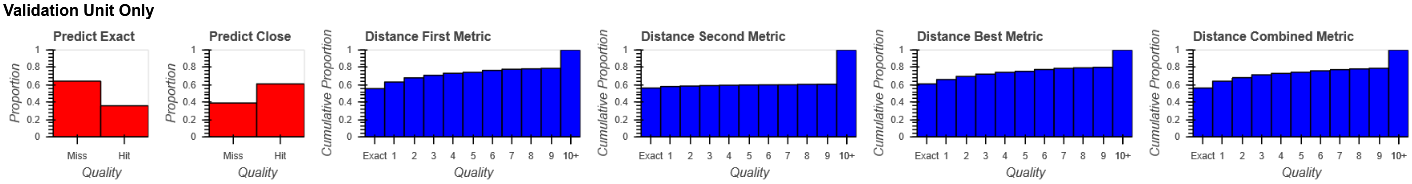
Training Context Only



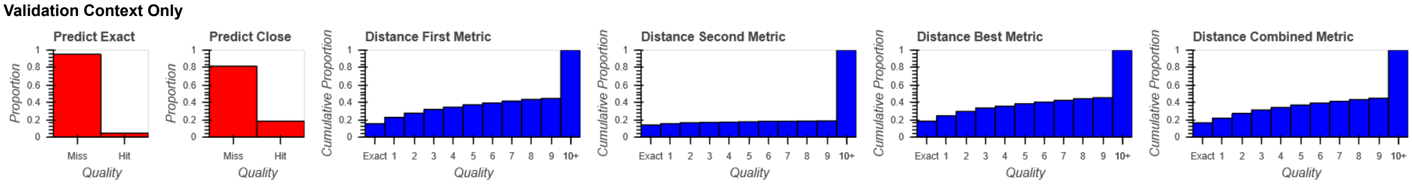
Validation Unit & Context



Validation Unit Only



Validation Context Only



Rows = input scenarios: unit / context / both. Columns = comparison metrics: Red= first prediction attempt accuracy. Blue = prediction quality with multiple close units.

Supervised Learning of Units of Measure

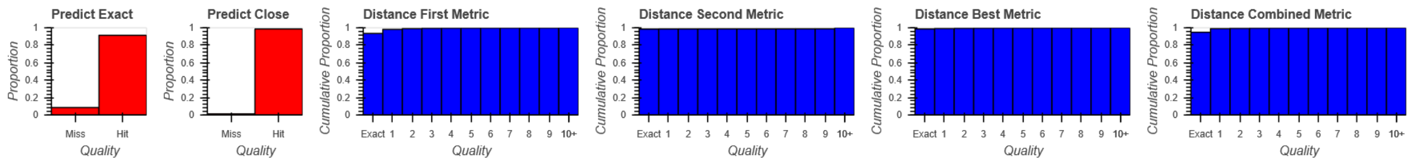
By: Jacob Barhak
& Joshua Schertz

Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

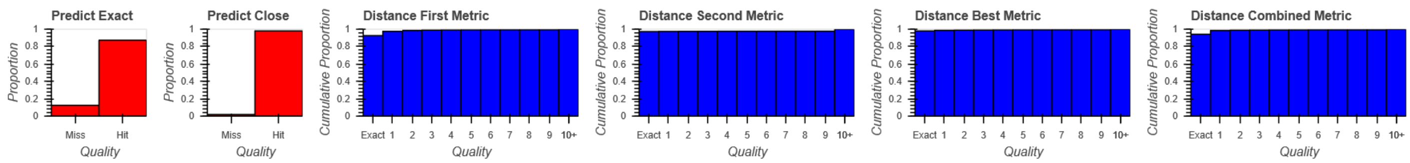
Neural Network Training and Validation Results For Multiple Architectures

Sequence to Sequence LSTM for Unit CNN for Context

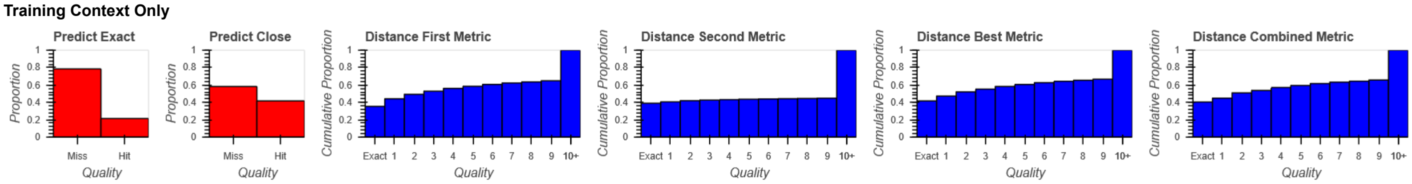
Training Unit & Context



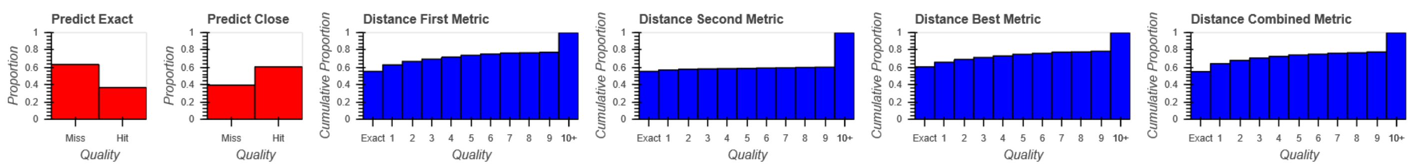
Training Unit Only



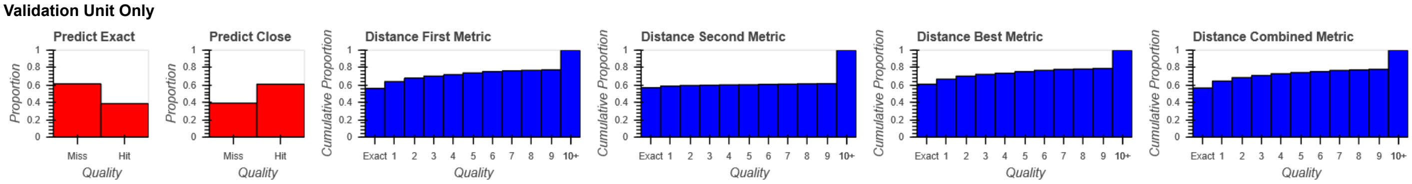
Training Context Only



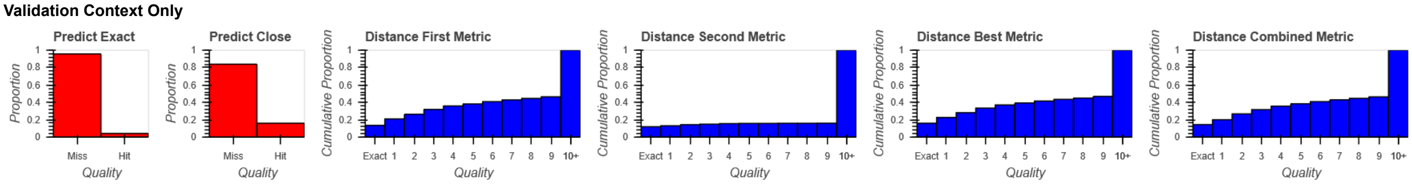
Validation Unit & Context



Validation Unit Only



Validation Context Only



Rows = input scenarios: unit / context / both. Columns = comparison metrics: Red= first prediction attempt accuracy. Blue = prediction quality with multiple close units.

Supervised Learning of Units of Measure

By: Jacob Barhak
& Joshua Schertz

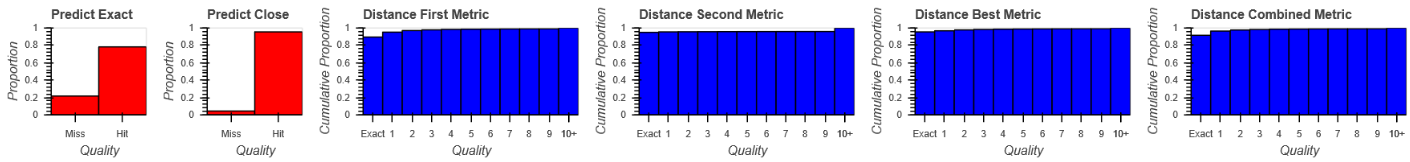
Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

Neural Network Training and Validation Results For Multiple Architectures

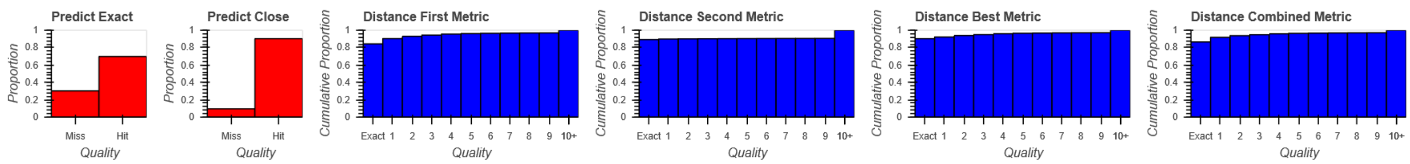
Encoder Decoder | LSTM Unit LSTM Context | LSTM Unit CNN Context | CNN Unit CNN Context

Sequence to Sequence CNN for Unit CNN for Context

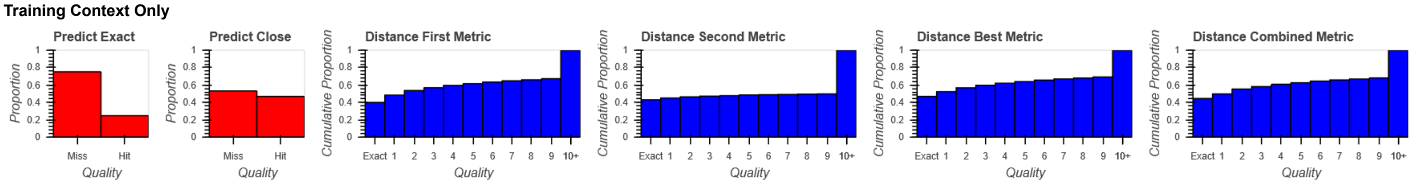
Training Unit & Context



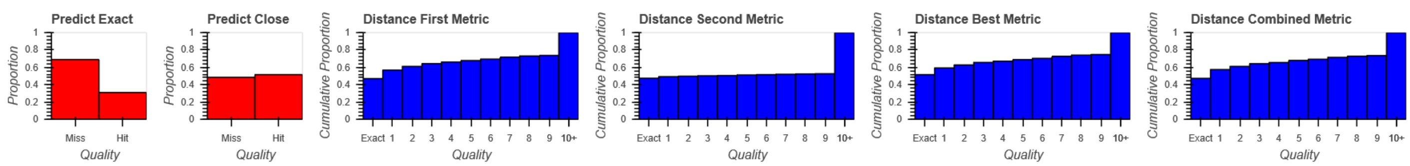
Training Unit Only



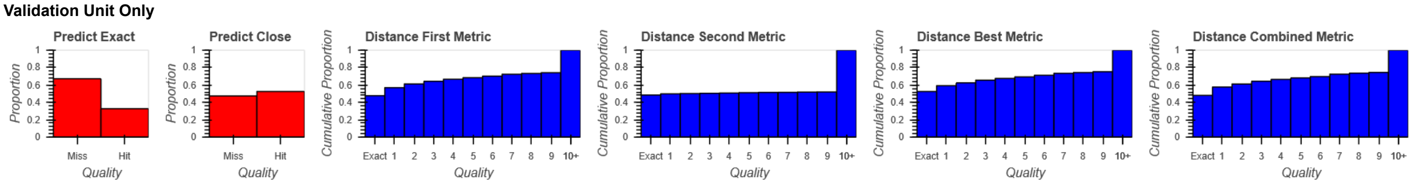
Training Context Only



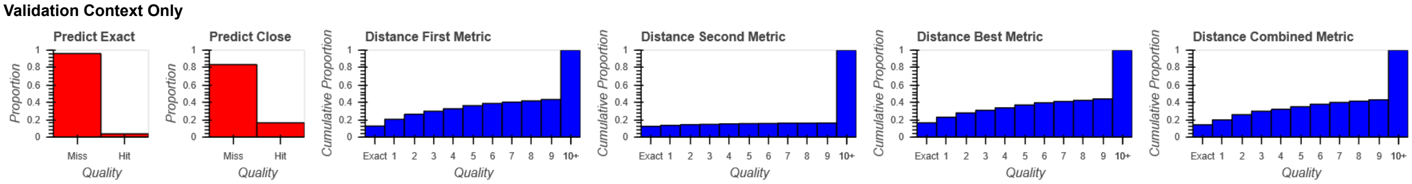
Validation Unit & Context



Validation Unit Only



Validation Context Only



Rows = input scenarios: unit / context / both. Columns = comparison metrics: Red= first prediction attempt accuracy. Blue = prediction quality with multiple close units.

By: Jacob Barhak
& Joshua Schertz

Venue: Integrating Machine Learning with Multiscale Modeling for Biomedical, Biological, and Behavioral Systems (2019 ML-MSM)

Summary

We created tools necessary to merge units of measure standards.

With such tools is will be possible for machines to:

- Recognize medical units, even if misspelled

- Comprehend medical units

- Comprehend numbers associated with units

Such AI will eventually replace tedious human tasks.

Future Work

- Perform human mapping of units using ClinicalUnitMapping.Com

- Apply the supervised machine learning tools to the mapped units

- Add the supervised learning API to ClinicalUnitMapping.Com

- Contribute to (UMLS), (CDISC), (SISO)

Acknowledgments:

- Many thanks to the PyViz team: Philipp Rudiger, James Bednar, Jean-Luc Stevens.
- Thanks to John Rice for the fruitful discussions regarding standardization
- Thanks to government persons who helped and specifically to:
  - Nick Ide from NIH/NLM ClinicalTrials.Gov team on advice to process the site
  - Erin E Muhlbradt from NIH/NCI for advice on CDISC unit data
  - John Garguilo from NIST for information on RTMMS
  - Grace Peng for maitaining the IMAG community and connecting to the NLM team
- Thanks to Paul Schluter for information about RTMMS and the IEEE unit standard
- Thanks for Tipton Cole, Rocky Reston, Andrew Simms for useful directions
- Thanks to Becky Ruppel, Yuval Merchav Uri Goren, Ari Bornstein, Ryan Baxley, Bhargav Srinivasa Desikan, Blaize Berry for NLP advise

Reproducibility:

This presentation is accessible here. The code that generated the presentation can be accessed here. This presentation is generated using Python 2.7.16, panel-0.5.1, holoviews 1.12.3, bokeh-1.1.0. Code and data for this work are archived in the file: AnalyzeCT_2019_05_13.zip. Web site database was created using the database PartUnitsDB_2019_05_13.db Supplemental code archived in the files: AnalyzeCT_Images_2019_10_10.zip, AnalyzeCT_Code_2019_05_15.zip. Clinical Trials data archived in StudiesWithResults_Downloaded_2019_04_12.zip. Bio Ontology Units downloaded on 2019_04_09, CDISC data downloaded on 2019_03_30 , RTMMS units downloaded on 2019_03_24 . Mock database used in training was ModifiedUnitsDB_Remodified.db . Tensorflow 2.0.0 was used for Neural Network execution in Python 3.7.4 environment . This tensorflow version is unstable, so results presented may not be reproducible. PYTHONHASHSEED was set to 0. Execution transcripts were saved in the files: AnalyzeCT_TF2_LargeMod_Mixed_LSTM_Unit_LSTM_Context_NewMetric_2019_10_14.zip , AnalyzeCT_TF2_LargeMod_Seq2Seq_NewMetric_2019_10_15.zip , AnalyzeCT_TF2_LargeMod_Mixed_LSTM_Unit_CNN_Context_NewMetric_2019_10_15.zip , AnalyzeCT_TF2_LargeMod_Mixed_CNN_Unit_CNN_Context_NewMetric_2019_10_15.zip .

Publications:

- J. Barhak, The Reference Model Models ClinicalTrials.Gov. SummerSim 2017 July 9-12, Bellevue, WA. Paper
- J. Barhak, The Reference Model: A Decade of Healthcare Predictive Analytics with Python, PyTexas 2017, Nov 18-19, 2017, Galvanize, Austin TX. Video
- J. Barhak, C. Myers, L. Watanabe, L. Smith, M. J. Swat , Healthcare Data and Models Need Standards. Simulation Interchangeability Standards Organization (SISO) 2018 Fall Innovation Workshop. 9-14 Sep 2018 Orlando, Florida Presentation
- J. Barhak, Python Based Standardization Tools for ClinicalTrials.Gov. Combine 2018 . Boston University Poster
- J. Barhak, J. Schertz, Clinical Unit Mapping for Standardization of ClinicalTrials.Gov . MSM/IMAG meeting. IMAG Multiscale Modeling (MSM) Consortium Meeting March 6-7, 2019 @ NIH, Bethesda, MD . Poster
- J. Barhak, Clinical Data Modeling with Python, AnacondaCon , Austin, Texas, April 3-5, 2019. Video , Presentation
- J. Barhak, J. Schertz, Standardizing Clinical Data with Python . PyCon Israel 3-5 June 2019, Video Presentation
- J. Barhak, J. Schertz, Clinical Unit Mapping with Multiple Standards . 2019 CDISC U.S. Interchange, Poster