Reproduction of Predicting E-Learning Student's Performance

Alsulami et al. 2023

Amy Tan

An Overview:

Where we're going

1. Background
2. Methods
3. What I aimed to reproduce
4. Results
5. Discussion

Background

- Original paper (**Alsulami et al. 2023**) looked at methods to predict E-Learner student performance
    - Diversity in modality of instruction and learning can provide more accessibility, and lend itself to more equitable education opportunities.
    - Ever increasing amount of data generated about the relationship between and learning processes is generated.
    - How can we leverage that data to improve our education systems?

--> Alsulami et al. 2023: explores how to predict student performance by combining **machine learning,** where computers learn patterns from data, with **ensemble methods,** which use multiple models working together to make better predictions.

- **Machine learning (ML)**: computers learn by studying examples and figuring things out themselves ("Here's what student success looks like— figure out how to get there")
- **Ensemble methods**: several computer models work together and share their answers to make a decision (combining their strengths usually leads to a more accurate result)

Methodology

The ML Models and Ensemble Methods

ML Models

- **Decision Trees (DT)**: predict student success by splitting data into branches based on feature conditions, creating a tree-like structure that leads to a final decision.
- **Random Forests (RF)**: predict student success by combining the results of multiple decision trees, each built on a random subset of data and features, to improve accuracy and reduce errors.
- **Naive Bayes (NB)**: predict student success based on probability, assuming that features are independent and contribute equally to the decision.

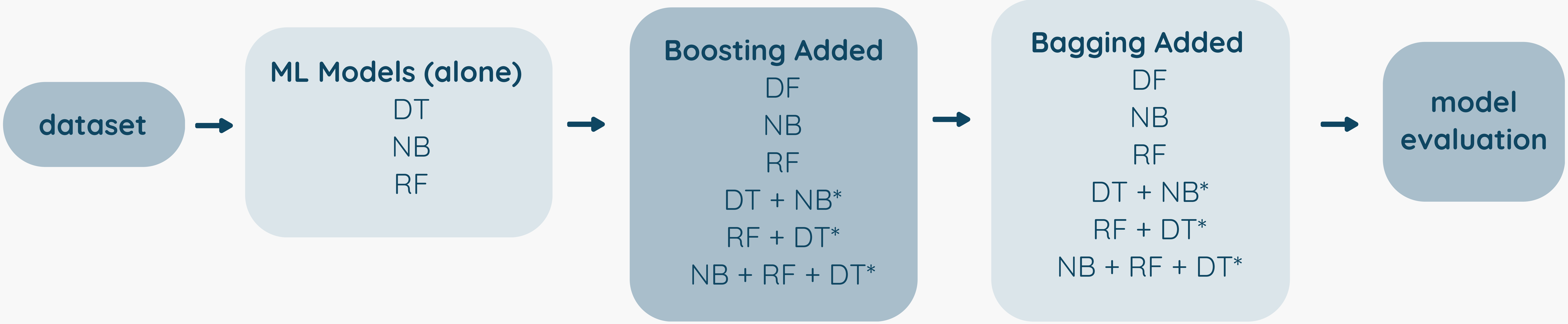
Ensemble Techniques

- **Boosting**: improves predictions by combining multiple simple models, each model focuses on correcting the errors made by the previous ones, creating a stronger overall predictor.
- **Bagging**: improves predictions by training multiple models on random subsets of the data and averaging their results to reduce variability and improve accuracy.
- **Voting**: combines predictions from multiple models, using majority rule in this instance.

Methodology

Data and What Was Done

- **Data:** obtained from the Kalboard 360 E-Learning system via the Experience API (XAPI).
    - 480 students with 17 attributes (demographic, academic, and behavioral)
- Split it into training and testing data
    - **K-fold cross-validation:** Divide data into 10 equal parts (folds). 9 of these folds are used to train the model, the remaining fold is used to test it.
    - Process is repeated for each fold, and the average accuracy from all tests is taken.
    - The entire procedure was repeated 10 times.



*voting used as well

What I wanted to reproduce:

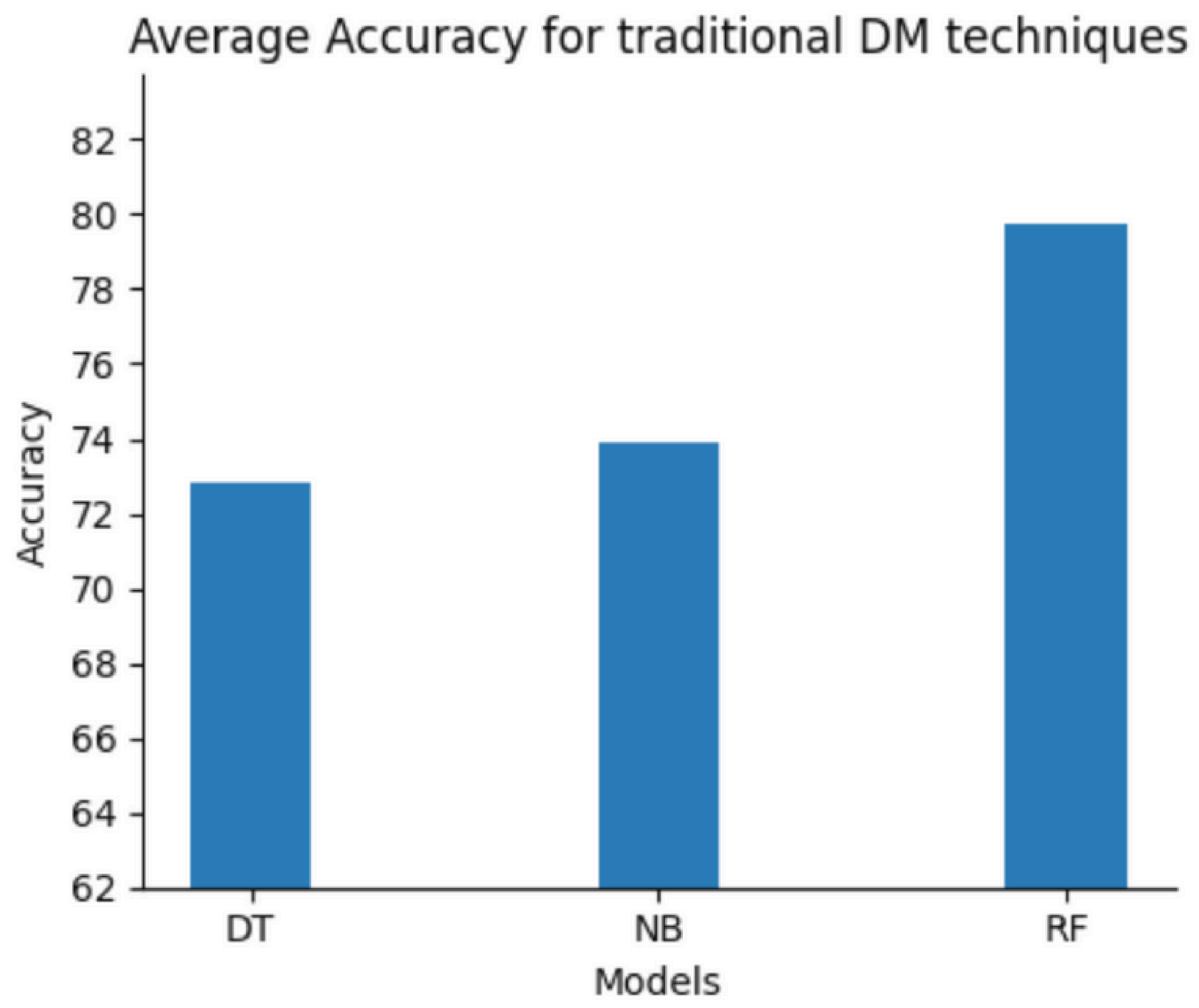
Accuracy

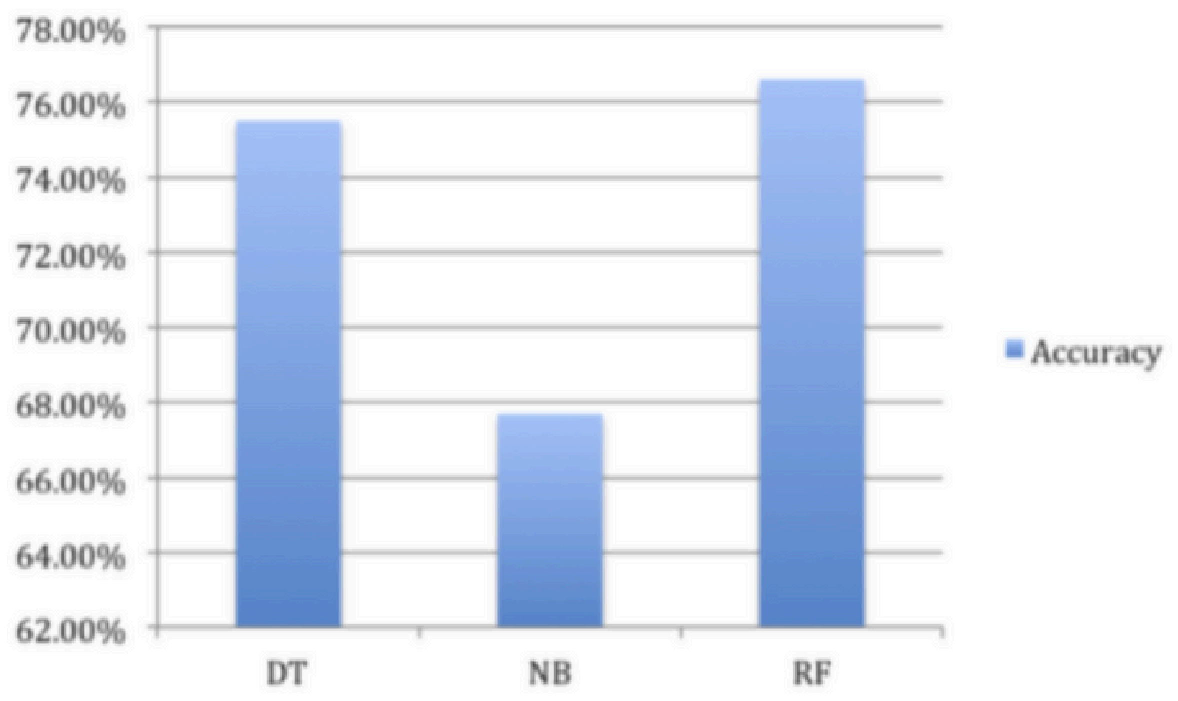
- With 15 experiments in total, how does one determine which one is the best at predicting student performance?
- **Accuracy**
  - metric used to evaluate the performance of a model
  - $\text{accuracy} = \dfrac{\text{\# correct predictions}}{\text{total \# of predictions}} \times 100$
  - In general: the higher the accuracy, the better!

- **Decision Trees with Boosting yielded the highest accuracy (77%) out of all models run**

Results

- First examining accuracy for the ML models alone (no ensemble methods)
- My results on the left, original results on the right

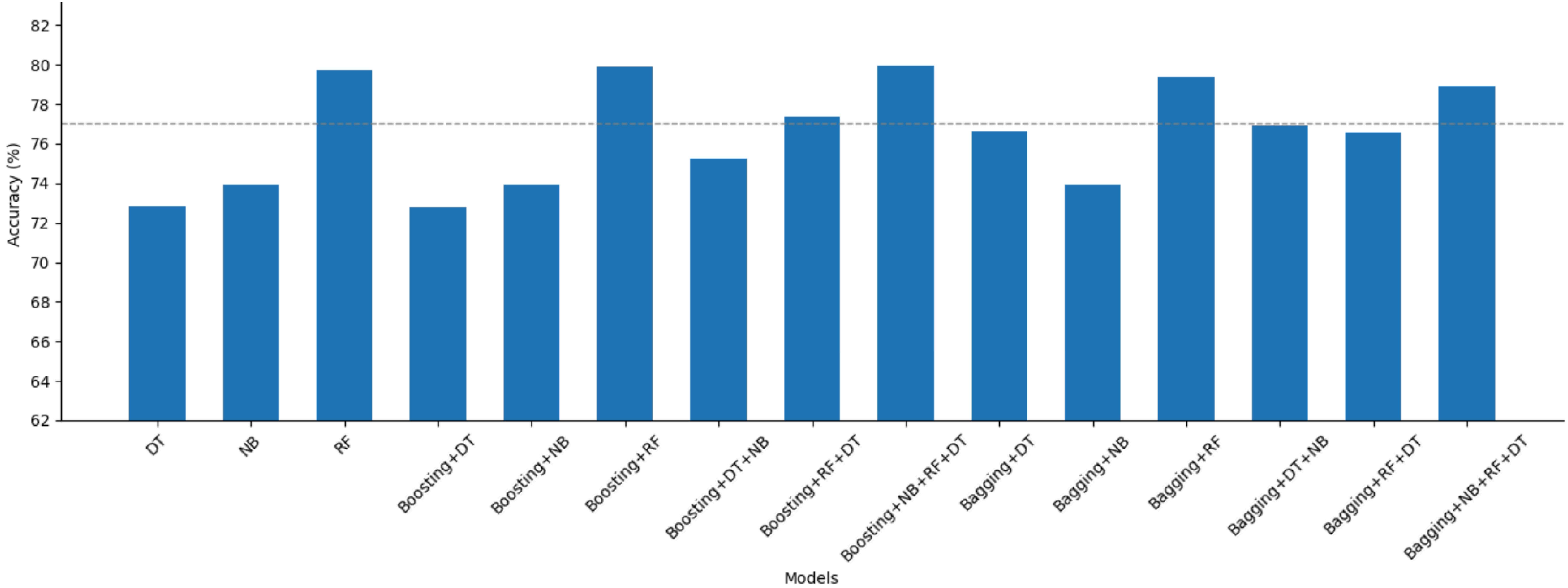




- Random forests have highest accuracy (79.81%)
- Decision trees are actually the least accurate (73.1%)

Results

- Plotting accuracies for all 15 models



- y = 77: represents the DT + boosting result from authors
- Boosting + DT + NB + RF had the highest accuracy (79.98%)

Discussion

- Failed to reproduce
    - Anticipated getting a slightly different accuracy, but I did not anticipate getting an entirely different model
- Boosting + DT + NB + RF had the highest accuracy (79.98%)
- All models had accuracy above 72% (compared to their 65%)
- Try again not using a specified random state/using a different state
- Possibly running in WEKA software, rather than Python's skitlearn package

- Using machine learning techniques in education might be worth exploring further
- If given the computational power and time, run a variety of models and comapre their performance since each dataset will be different