Case Study #2

By Bashir Gulistani

Objectives

Scalability & Efficiency

Analyze model performance with varied training data

Focus on resource optimization for practical deployment.

Robustness Testing

Evaluate model resilience with noise and missing data.

Ensure adaptability in diverse real-world scenarios.

Algorithm Comparison

Assess accuracy and efficiency of algorithms like LinearSVC, Logistic Regression, KNN

Select the most effective algorithm for precise location predictions

Location Accuracy Optimization

Develop a robust WLAN fingerprint-based model for precise indoor location predictions

Enhance accuracy using innovative feature engineering techniques.

Feature Engineering Impact

Experiment with approaches like signal strength normalization on WLAN fingerprints

Evaluate the impact on classification accuracy, refining feature selection for better predictions.

Data

Features

Focused Selection: Utilized Intensity values for WAPs (features 0-520) to capture relevant data.

Label

Enhanced Labeling: Combined Floor and Building ID to create a streamlined and informative new feature.

Model Optimization

Simplicity for Efficiency: Emphasized ease of application by excluding unnecessary features, ensuring a lean and effective model implementation.

Things I have Done

- Understanding Dataset
- Data Preprocessing
  - Combining Building and Floor
    - Creating 13 different categories, e.g. 00,..., 24 (Building, Floor)
    - Building Separation: Presenting Building ID before Floor ensures clear separation between buildings. This helps the model understand the distinction between different buildings, especially when each building has a different number of floors.
    - Consistency: Building ID provides a consistent reference point for different floors, making it easier for the model to learn patterns across buildings.
- Model Engineering
  - Use Models that are compatible for 20,000 data points and 520 features
  - Combining Train and Validation set, creating 80% train, 10% validation, 10% test randomly
  - Doing Experiments
  - Cross-Validation

Experiments

- **Exp 1: training data without any feature engineering**
  - Logistic Regression: 91.9%
  - Random Forest: 99% (100 n-estimator)
  - SVC: 96%
  - KNN: 94% (11 k_neighbor)
  - Gaussian Naive Bayes: 47%
  - Linear SVC: 85.7%
- **Exp 2: Normalize data (min/max) (only changes)**
  - Logistic Regression: 90.5%
  - SVC: 95
  - Linear SVC: 89%

...

- **Exp 3: Changing 100 to -105**
  - Logistic Regression: 97%
  - Random Forest: 99% (100 n-estimator)
  - SVC: 99% (11 k_neighbor)
  - KNN: 98.6%
  - Linear SVC: 98.4%
- **Exp 4: Modifying the range of numbers 0-(-104) and 100 to 0 to 105 (0 = 100, 1= weak, 105=strong) (OVERFIT)**
  - Logistic Regression: 98%
  - Random Forest: 99% (100 n-estimator)
  - SVC: 99%
  - KNN: 99% (11 k_neighbor)
  - Linear SVC: 98%

...

- **Exp 5: Standardization**
  - Logistic Regression: 93.5%
  - Random Forest: 99% (100 n-estimator)
  - SVC: 94%
  - KNN: 94.5% (1 k_neighbor->Grid Search and Cross Validation)
  - Linear SVC: 89.8%
- **Exp 6: Standardization and changing 100 to -105**
  - Logistic Regression: 98.8%, SVC: 98.1%, KNN: 97.2%, LinearSVC: 97.8%
  - Cross-Validation
    - Logistic Regression: [0.98752969 0.98812352 0.99168646 0.98456057 0.98990499 0.98812352
    - 0.99109264 0.99287411 0.9869281 0.99108734]
  - Given very high accuracy, and concern about potential overfitting, sticking with just standardization is a safer approach
  - Achieving ~93% accuracy with just standardization might be more interpretable and robust in the long run

Random Forest & Linear SVC

- **Computationally Expensive**
- **Performance Gap**
- **Random Forest:**
  - **Model Complexity**
  - **Hyperparameter Tuning**
    - **Time-Consuming**
  - **Prone to Overfitting on this dataset**
    - Too Few-> High Bias
    - Too Many -> High Variance
  - **Data Changes Over Time**
- **Linear SVC:**
  - **Prone to Underfitting**
  - **Linear Boundaries: not entirely linear in our case**

Chosen Models

- **Logistic Regression**
  - Interpretability
  - Speed
  - Robustness
- **KNN**
  - Flexibility
  - Intuitive
  - Adapt to various patterns
- **SVC**
  - Versatility
  - Maximum Margin
  - Handle complex data distributions without being sensitive to noise

KNN: n-neighbors?

- **Using Grid Search and Cross Validation if providing range from 1**
  - K=1
    - Provides the highest accuracy possible (94.5%) based on our current dataset
    - Can detect underlying patterns in the current dataset
- **Choosing K=3**
  - Nature of Data: WiFi fingerprints can be noisy
  - Dynamic Environment
  - Spatial Granularity
  - Temporal Stability
  - Consistency at the cost of accuracy

Challenges

- Finding a way to combine Floor and Building ID
  - Only considering building ID
    - Not Inclusive
  - Creating 13 different categories
- Overfitting
  - Using Cross Validation
  - Choosing Better Suited Model Based on Dataset
- Very High Accuracy Rate
  - Exploring various feature engineering techniques and combinations
  - Striking a balance between bias and variance

Findings

- Model Selection and Performance
  - Going for simple models first
- Random forest can provide very high accuracy. However,
  - it is computationally expensive, e.g. it can take more than 10 minutes to do grid search/random search based on different paramaters
- Test/Validation set should be at least 10% each to accurately perform
- Data Normalization/Standardization should be done to increase the performance and reduce the noise
- Unneccessary features should be excluded before model engineering
- Similar Results in test set compared to validation set (less than 1% increase or decrease)
  - Good Generalization
  - Positive Sign that the model is doing well in unseen data

Logistic Regression

- Accuracy: Achieved an overall accuracy of 93.8%. (validation: 93.5%)
- Precision: Ranged from 88% (for class 02) to 97% (for class 21), indicating that the model is reliable with its positive predictions across most classes.
- Recall: Ranged from 87% (for class 02) to 97% (for class 23), showing that the model effectively identified the actual positives for each class.
- F1-Score: Ranged from 87% (for class 02) to 97% (for class 20), reflecting a well-balanced performance between precision and recall for most classes.
- Class-Specific Performance:
  - Classes 20 and 23 continue to excel in both precision and recall, showcasing the model's effectiveness for these categories.
  - Class 02 has shown slightly lower recall and precision compared to other classes, suggesting potential areas for improvement.
- Macro and Weighted Averages:
  - Macro average for precision, recall, and F1-score: 93%
  - Weighted average for precision, recall, and F1-score: 94%

KNN

- Accuracy: Achieved a high accuracy of 93.44% (validation: 93.2%)
- Precision: The precision spanned from 87% (for class 00) to a perfect 100% (for class 24), indicating the model's reliability in its positive predictions across the majority of the classes.
- Recall: The recall values ranged between 81% (for class 02) and 99% (for class 24), highlighting the model's effectiveness in capturing the actual positives.
- F1-Score: With scores spanning from 86% (class 02) to a perfect 100% (class 24), the model consistently balanced precision and recall across most classes.
- Class-Specific Performance:
  - Classes 24 and 23 displayed exemplary performance in both precision and recall. Particularly, class 24 achieved a flawless precision score.
  - Class 02 showed a slightly reduced recall, suggesting there's potential for refining the model's sensitivity for this class.
  - Class 00 had the lowest precision, pointing towards possible improvements in reducing false positives for this category.

SVC

- Accuracy: The model achieved an impressive accuracy of 94.58% (validation: 94.1%)
- Precision: Precision values varied between 81% (for class 02) and a perfect 100% (for class 24). This demonstrates the model's high reliability in its positive predictions across the majority of classes.
- Recall: Recall metrics ranged from 90% (for class 02) to a flawless 100% (for class 23), emphasizing the model's proficiency in identifying actual positives across classes.
- F1-Score: F1-scores spanned from 86% (class 02) to a perfect 98% (class 23 and class 24), underlining a consistent balance between precision and recall for the various classes.
- Class-Specific Performance:
  - Classes 23 and 24 showcased outstanding precision and recall, with class 24 achieving perfect precision.
  - Class 02, however, lagged in terms of precision, suggesting that there's room for improvements in reducing false positives for this category.

More Data=Higher Accuracy?

- **Increasing Training Data Boosts Performance:**
  - Accuracy consistently improves as training data increases.
- **Performance Plateau:**
  - Performance plateau observed beyond 80% training data.
- **Near Maximum Performance:**
  - 100% Training data achieves 94% accuracy
- **Initial Data Boost:**
  - model needs a certain baseline amount of data to effectively learn the WiFi fingerprints
- All models improve with more training data.
- SVC is the top performer across all data sizes, peaking at 94.7% with 80% data.
- KNN surpasses Logistic Regression by 60% data.
- Models show diminishing accuracy gains after 80% data utilization.

Model Performance