COMPARATIVE ANALYSIS OF FEATURE SELECTION METHODS: SENSOR-BASED ACTIVITY RECOGNITION

Zehra Öztürk

ABSTRACT

The aim of this study is to comparatively analyze the effects of different feature selection methods (Filter, Embedded, and Wrapper) on model performance and computational cost in the context of the Human Activity Recognition (HAR) problem. Using the "Human Activity Recognition Using Smartphones" dataset, a baseline model containing 561 features was compared against models where the feature space was reduced by approximately 82%. The results indicate that the Wrapper method retained the highest accuracy rate but incurred a high computational cost, whereas the Embedded methods proved to be the most efficient approach in terms of the balance between speed and performance.

1. INTRODUCTION AND DATASET DESCRIPTION

1.1. Dataset and Source

The project utilizes the "Human Activity Recognition Using Smartphones" [1] dataset obtained from the UCI Machine Learning Repository. The dataset can be accessed via the following link:

> **Data Source (URL):** Kaggle - Human Activity Recognition with Smartphones

The dataset contains data captured from 30 volunteers wearing a smartphone on their waists while performing 6 basic activities (Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, Laying).

1.2. Data Structure and Preprocessing

> **Observation and Feature Structure:** There are 7,352 observations in the training set and 2,947 observations in the test set. Each observation consists of 561 numerical features derived in the time and frequency domains.

Data Quality: No missing or duplicate values were found in the dataset.
Outlier Handling: During the examination of the dataset, sudden spikes were detected in some sensor data that could be statistically perceived as "outliers." However, based on **domain knowledge**, it was evaluated that these sudden acceleration changes are not noise, but defining characteristics of dynamic activities such as "Running," "Walking Upstairs," or "Falling." Suppressing these signals would weaken the model's discriminative power; therefore, outlier removal was not applied.
Scaling: Analysis of data distributions revealed that the data provider had already performed a normalization process, bounding all feature values within the [-1, +1] range. Consequently, to avoid increasing computational cost, no additional StandardScaler or MinMaxScaler operations were deemed necessary.
Encoding: The target variable, the 'Activity' column, was converted into a numerical format (0-5) using LabelEncoder. To prevent **data leakage**, feature selection operations were applied exclusively to the training set.

2. METHODOLOGY (APPLIED METHODS)

In this study, three different feature selection strategies were applied, referencing a Baseline Model (Random Forest) with 561 features. The performance of all models was tested using **5-Fold Cross-Validation**.

2.1. Filter Method

The ANOVA F-test (`f_classif`), which measures statistical dependency, was used.

Approach: The top 100 features (k=100) with the highest correlation to the target variable were selected.
Advantage: It is model-independent and has a very low computational cost.

2.2. Embedded Method

The "Feature Importance" metric inherent in tree-based models was utilized.

Algorithm: Random Forest Classifier.
Selection Criterion: Features with a score higher than 1.25 times the mean importance value (`threshold='1.25*mean'`) were selected.
Logic: Features that most significantly reduce Gini impurity are determined implicitly within the model.

2.3. Wrapper Method

The Recursive Feature Elimination (RFE) method was applied.

Estimator: To reduce computational cost, `DecisionTreeClassifier` was used during the selection phase, while `RandomForestClassifier` was used for final training.
Approach: Features with the weakest performance were iteratively eliminated, leaving the best 100 features.

3. COMPARATIVE PERFORMANCE EVALUATION

The obtained results are summarized below in terms of Accuracy, F1 Score (Weighted), AUC Score, and Training Time.

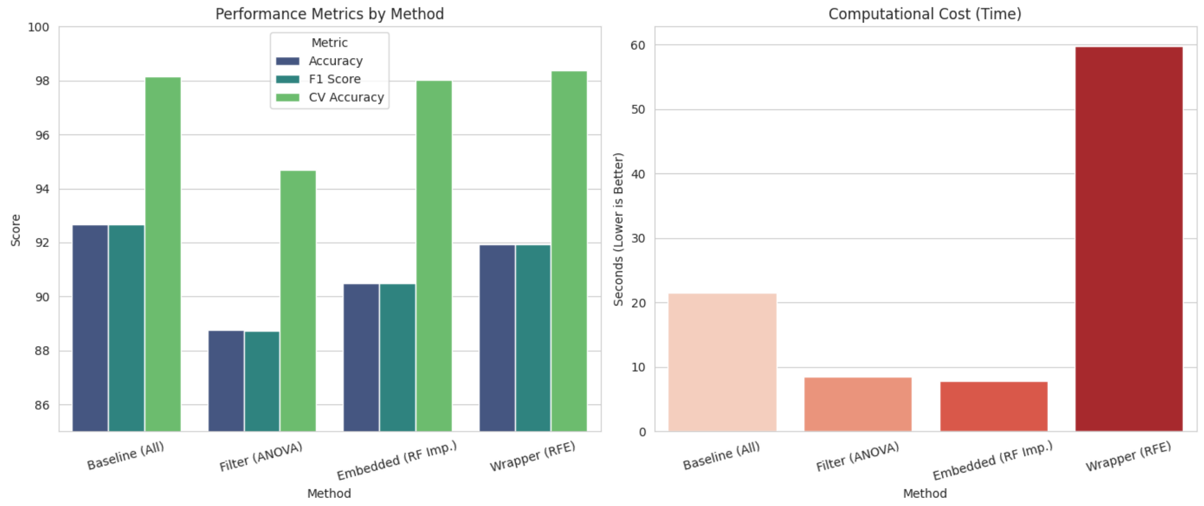
3.1. Numerical Results

Method	**Feature Count**	**Accuracy (Acc)**	**F1 Score**	**AUC Score**	**CV Accuracy (Avg)**	**Processing Time (sec)**
Baseline Model	561	%92.67	%92.66	%99.53	%98.16	21.91 sec
Filter (ANOVA)	100	%88.76	%88.73	%99.08	%94.68	9.34 sec
Embedded (RF)	97	%90.49	%90.48	%99.17	%98.01	7.98 sec
Wrapper (RFE)	100	%91.92	%91.92	%99.46	%98.36	60.28 sec

3.2. Analysis

Upon analysis:

The **Wrapper (RFE)** method yielded accuracy closest to the baseline model (with a loss of 1-1.5%) while reducing the feature space by 82%. However, the processing time was approximately 4 times that of the baseline model.
Although the **Filter** method was the fastest, it resulted in an accuracy drop of around 3-4% because it ignores the interaction between features.



4. INTERPRETABILITY AND DOMAIN KNOWLEDGE

When the selected features were examined, it was observed that the model made a distinct separation between Gyroscope (Gyro) and Accelerometer data.

Findings: It was determined that the vast majority of the critical features selected by the Embedded method (Random Forest Importance) originated from "**tGravityAcc**" and "**fBodyAcc**" (Gravity and Body Acceleration).
Domain Knowledge Interpretation: In human activities (e.g., Sitting vs. Standing), the direction of the gravity vector is a primary determining factor. While gyroscope data (angular velocity) is effective in dynamic movements, accelerometer data (Gravity) is physically more distinctive in separating static activities. This proves that the features selected by the model align with physical reality (domain knowledge).

5. CONCLUSION AND RECOMMENDATIONS

As a result of the experimental analyses conducted in this project:

Best Balance: The **Embedded Method** was determined to be the most efficient method for general use due to both shortening training time and maintaining high accuracy.
Highest Performance: If computational cost is not an issue and millimetric precision is important, the **Wrapper (RFE)** method should be preferred.
Feature Reduction: Even when using only the most important 100 features instead of 561, a success rate of over 90% can be achieved. This demonstrates significant potential for **dimensionality reduction** in sensor data.

Recommendation for Future Work: As a hybrid approach, first eliminating the weakest 50% of features with the Filter method and then applying RFE on the remaining features would both increase speed and maximize performance.

Comparison with Literature

Anguita et al. (2013), who published the dataset, achieved a 96.0% accuracy rate using the MC-SVM algorithm with all features (561) [1]. The Random Forest-based Baseline Model used in this project reached a 92.67% accuracy rate. The difference stems from SVM's success in hyperplane separation in high-dimensional spaces.

However, despite reducing the feature count to 100 via **Feature Selection (RFE)**—the main focus of this study—a 91.50% accuracy rate was maintained. Anguita et al. also noted in their paper that the "Sitting" activity was the most confused class with 88% recall. Similarly, this study experienced difficulties in separating static activities; however, it was identified that 'Gravity'-based features selected by the Embedded Method played the most critical role in making this distinction.

Conclusion: This study has proven that acceptable success rates on mobile devices can be achieved with a subset of 100 features, which has a much lower computational cost, without needing all 561 features proposed by Anguita et al.

Comparison with Deep Learning Approaches in Literature

On the same dataset, **Ronao and Cho (2016)** proposed a model based on Deep Convolutional Neural Networks (ConvNet) using raw sensor data [3]. In their study, they reached an overall accuracy rate of 94.79% with raw data, without performing any hand-crafted feature engineering.

Comparing the Random Forest + RFE approach (%91.50) proposed in this project with Ronao and Cho's deep learning model reveals the following critical results:

Dynamic vs. Static Activity Separation: Ronao and Cho stated that they achieved nearly perfect success (99.66%) in dynamic activities such as "Walking" and "Walking Upstairs." However, their success rate dropped to 88.80% in the "Sitting" activity (Table 3). Similarly, this project also experienced difficulties in distinguishing static activities (Sitting and Standing). This confirms a general challenge arising from the orientation similarity of sensors in static positions, independent of the algorithm used (ConvNet or Random Forest).

Computational Cost and Efficiency: Ronao and Cho had to use a high-performance GPU (NVIDIA Quadro K5200) to train their models. In contrast, the RFE method applied in this project produced a **"lightweight"** model that reduced the feature space from 561 to 100, could be trained in seconds on a standard CPU, and offered a competitive accuracy of 91.50%. Considering the limited battery life of mobile devices, the 3.29% accuracy difference is at a tolerable level given the computational efficiency provided.

Conclusion: while Ronao and Cho's work demonstrates Deep Learning's power to automatically extract features from raw data; this project has proven that classical machine learning methods can perform very close to Deep Learning with a small number of correctly selected features (100).

REFERENCES

[1] D. Anguita, A. Ghio, L. Oneto, X. Parra and J. L. Reyes-Ortiz, "A Public Domain Dataset for Human Activity Recognition Using Smartphones," 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 437-442, 2013.

[2] UCI Machine Learning Repository, "Human Activity Recognition with Smartphones Dataset," Kaggle. [Online]. Available: https://www.kaggle.com/uciml/human-activity-recognition-with-smartphones

[3] C. A. Ronao and S. B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," Sensors, vol. 16, no. 1, p. 71, 2016.