UNIVERSITÀ DI PISA

Master's degree in Artificial Intelligence and Data Engineering

Data Mining and Machine Learning project presentation

ROOM COUNTING PEOPLE

Antonino Patania

Github repository:
https://github.com/tonipata/room-people-counting.git

A.Y. 2024/2025

Contents

1. Introduction

Nowadays, work positions in offices are greatly increasing and along with these are increasing energy consumption that ensures proper quality in work, such as heating, ventilation and air conditioning systems.
One way to optimize their usage is to make them demand-driven depending on human occupancy.
Room Counting People is an application that estimates the number of people inside a room by leveraging multiple heterogeneous sensor nodes and a machine learning model.
It exploits data from 5 different types of non-intrusive sensors: CO2, temperature, illumination, sound and motion.
The machine learning models used are:

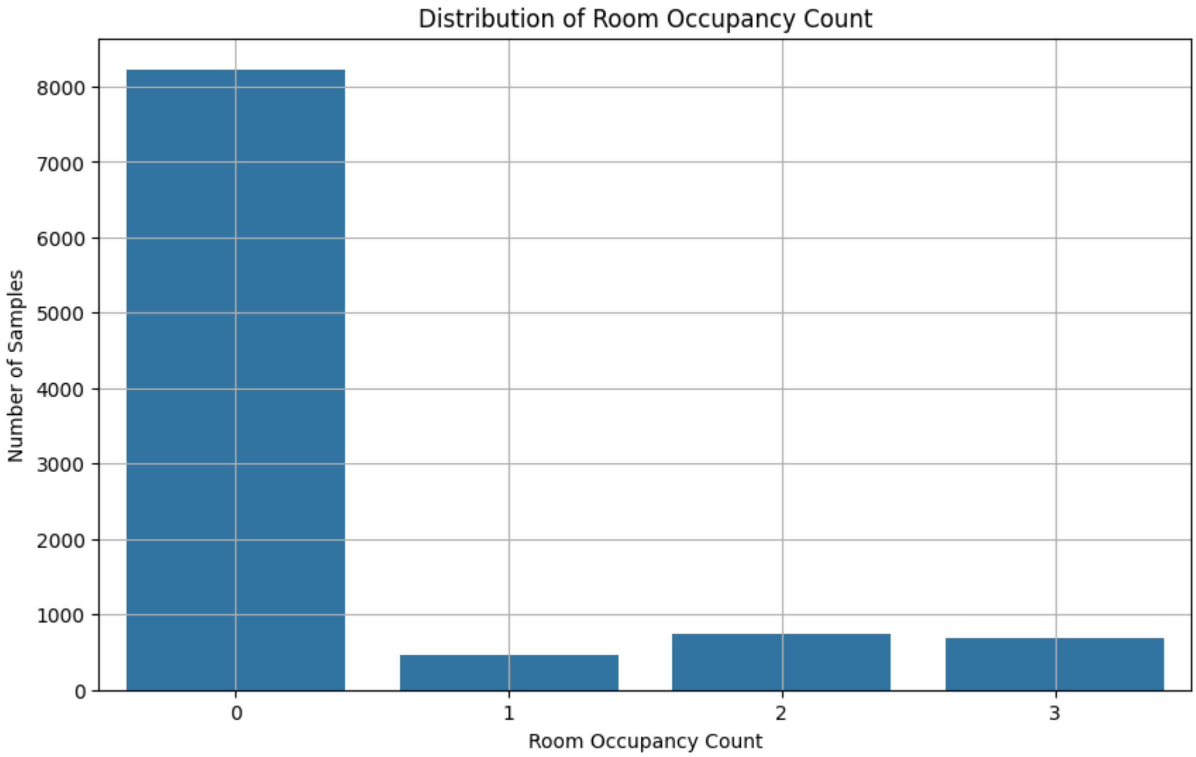
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Random Forest (RF)
- Support Vector Machines (SVM)

2. Dataset

The dataset used for this application is called “Room Occupancy Estimation” and come from https://doi.org/10.24432/C5P605
It’s composed by 18 features and 1 column for the ground truth.

FEATURES	**UNIT**
Date	YYYY/MM/DD
Time	HH:MM:SS
S1_Temp	C
S2_Temp	C
S3_Temp	C
S4_Temp	C
S1_Light	Lux
S2_Light	Lux
S3_Light	Lux
S4_Light	Lux
S1_Sound	Volts
S2_Sound	Volts
S3_Sound	Volts
S4_Sound	Volts
S5_CO2	PPM
S5_CO2_Slope	
S6_PIR	
S7_PIR	

Ground truth has 4 categories which are: ‘0’, ‘1’, ‘2’ and ‘3’.
Pir sensor (Passive InfraRed) is used for motion detection and set the value to ‘1’ when it detects a motion, so it can only have 2 values, ‘0’ and ‘1’.
S5_CO2_Slope is a feature derived from real-time CO2 values. It’s calculated by fitting a linear regression in a window of 25 points at each instance and calculating the slope of the line.
About ground truth, we can observe the distribution by means of a histogram and note that it is unbalanced in favor of category ‘0’.



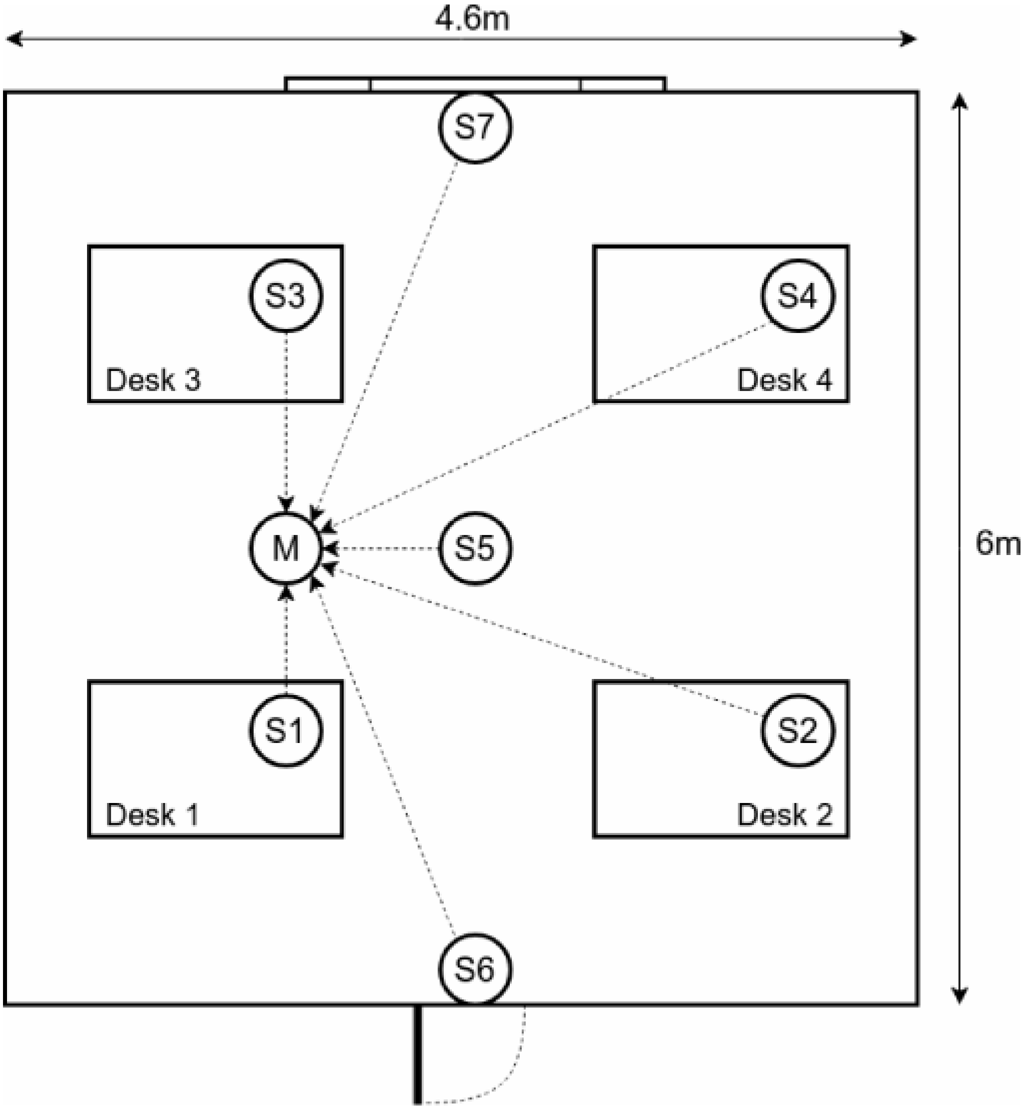
This can be explained by the fact that the system of recording sensor measurements remains active throughout the day, so during the hours when the company or office is closed, no relevant data is recorded and the number of people is obviously 0.

2.1 Setup of environment

The room where the wireless sensors network (WSN) were deployed is 6 m x 4.6 m with 4 office desks.

The network is a Zigbee based star network[1] with 7 slave nodes feeding data to the master node.

In each sensor node, a microcontroller board sampled data from the sensors and transmitted it periodically every 30 s.



[1] The Zigbee protocol is a low-power communication system designed for IoT, sensor and automation applications. In the star topology, all nodes in the network (slaves) communicate with a central node (master, coordinator Zigbee).

Instead PIR and sound sensors need a constant polling or else the events of interest are lost. Since the output pin of the PIR sensor remains high for about 3 s in repeat trigger mode. If even a single motion event was captured in the frame of 30 s, a ‘1’ was sent to the master.
For the sound sensors, the algorithm churned out the maximum peak to peak voltage that was achieved in the time frame of 30 s.

3. Preprocessing

The dataset undergoes several preprocessing steps before getting to use the machine learning models:

- Cleaning
- Normalization;
- Discretization
- PCA

3.1 Cleaning

Rows that have missing sensor values are discarded for 2 mainly reasons:

- Zigbee protocol is reliable, so there is a low probability that a sensor data is not valued;
- Sensor data arrive at the system every 30 seconds and it's statistically difficult for the number of people to change in such a small time window;

In addition, it was decided to remove the feature 'S5_CO2_Slope' because computing a regression for each input row would involve a significant computational cost and could introduce latency.

3.2 Normalization

Because the SVM, LDA and QDA models and PCA technique can be affected by different magnitude values, it was decided to use normalization.

The normalization used is the 'min-max normalization':

$$v' = \frac{v - min_A}{max_A - min_A}$$

This type of normalization is chosen because a typical range of values is expected.

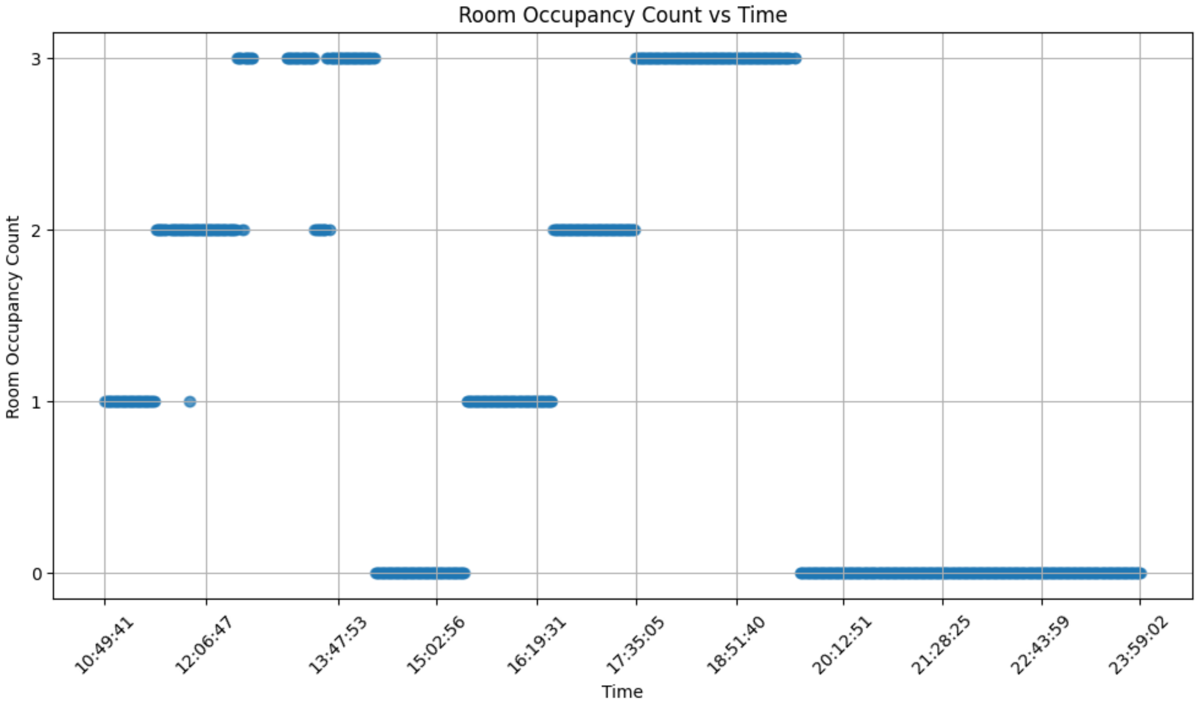
All sensor values, excluding those of PIR sensors, will have values in the range [0,1].

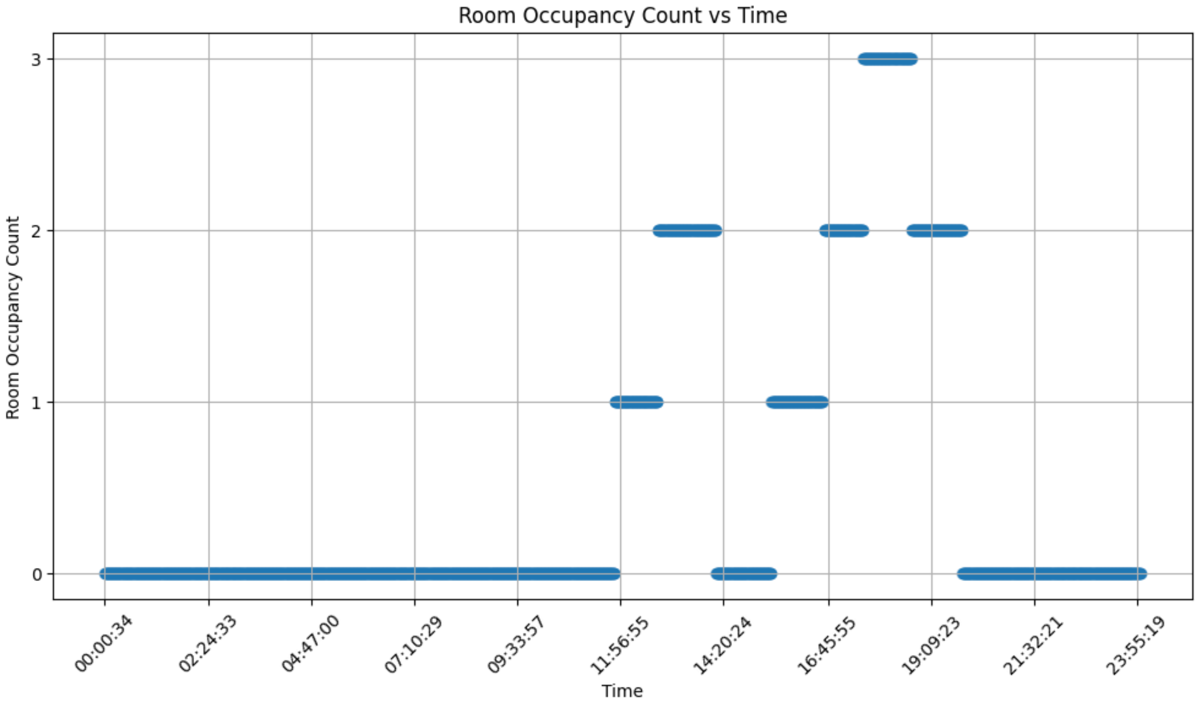
3.3 Discretization

The system remains active 24 hours a day, but its use is aimed to companies, particularly employee offices.

For this reason, it's interesting to discretize the values of the time (column 'Time') into 2 categories, 'WORK' and 'SLEEP'.

Time slots ranging from 9 a.m. to 7 p.m. fall into the first category, while the rest end up in the second.



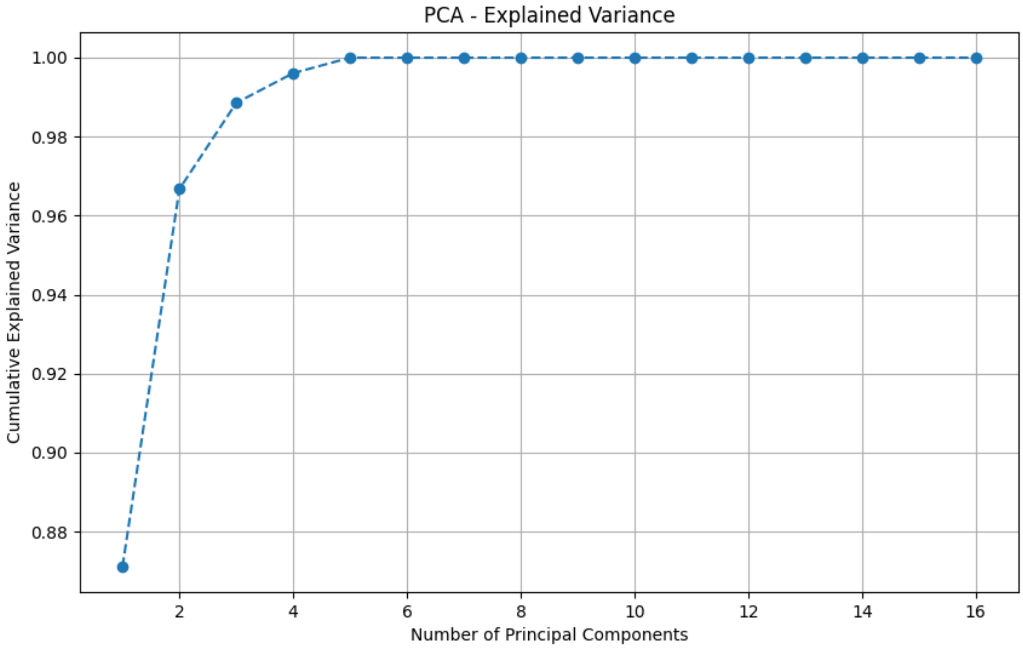


3.4 PCA

Having multiple measurements available for the same type of sensor and the system being real time type, it's important to apply PCA.

In this way, the number of variables is reduced and even a heavy model such as RF or SVM improves run time.

The dataset contains 16 features and the graph showing the comulative explained variance for choosing the number of components is calculated.



The graph shows that almost all of the comulative variance is explained by 4 components.

4. Classification models

As said before, the dataset is unbalanced, so accuracy cannot be relied on as a comparison parameter.
Stratified K-Fold Cross-Validation with 10 layers is used to select the best model maintaining the proportion of each class in each subdivision.
The evaluation metrics used are: precision, recall and f1.
In particular, these metrics are calculated for each class in each fold and then you do the arithmetic average (macro averaging).
After all 10 iterations have been completed, the metric calculated for each fold are averaged to obtain an overall estimate of model performance across all data.

4.1 Linear Discriminant Analysis (LDA)

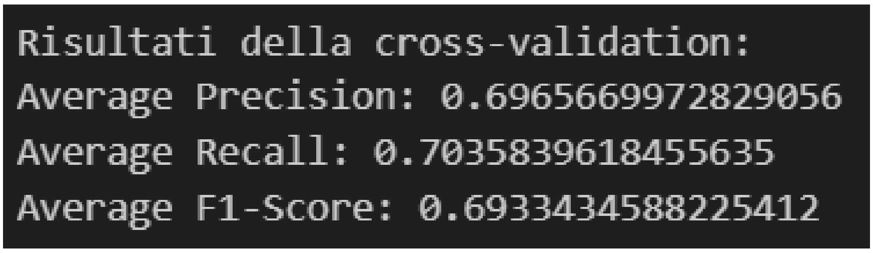
LDA tries to find a linear combination of the independent features that maximizes the separation between classes.
It assumes that the data follow a Gaussian distribution within each class and have identical covariance among all classes.

$$P(X|y = k) \sim N(\mu_k, \Sigma)$$

μ_k: average of class k
Σ: covariance matrix

The probability of class membership is calculated using Bayes:

$$P(y = k|X) = \frac{P(X|y = k)P(y = k)}{\sum_i P(X|y = i)P(y = i)}$$



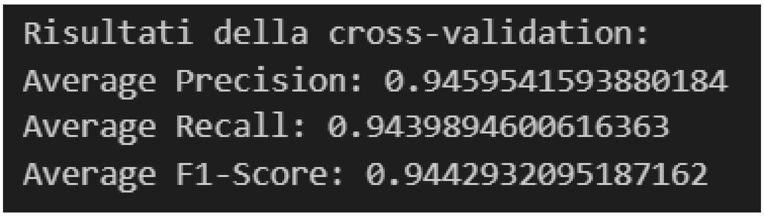
```
Risultati della cross-validation:
Average Precision: 0.6965669972829056
Average Recall: 0.7035839618455635
Average F1-Score: 0.6933434588225412
```

4.2 Quadratic Discriminant Analysis (QDA)

QDA is a variant of LDA that allows more flexibility in classification due to one key difference: QDA does not assume classes have the same covariance matrix.

$$P(X|y = k) \sim N(\mu_k, \Sigma_k)$$

Σ_k: covariance matrix of class k

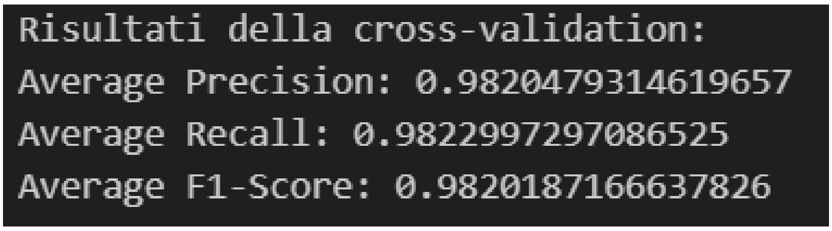


```
Risultati della cross-validation:
Average Precision: 0.9459541593880184
Average Recall: 0.9439894600616363
Average F1-Score: 0.9442932095187162
```

4.3 Random Forest (RF)

The random forest was used with the following parameters:

- n_estimators=50
- max_depth=20
- min_samples_split=2
- min_samples_leaf=1
- criterion=gini

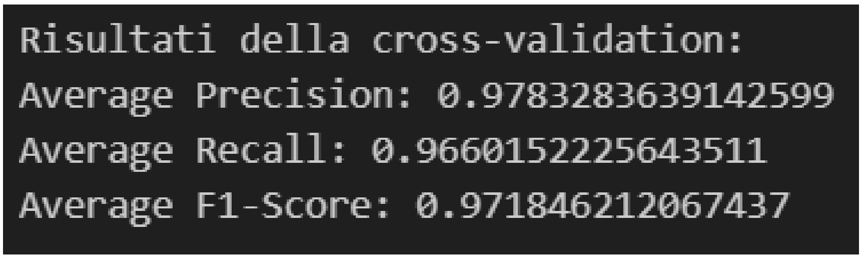


```
Risultati della cross-validation:
Average Precision: 0.9820479314619657
Average Recall: 0.9822997297086525
Average F1-Score: 0.9820187166637826
```

4.4 Support Vector Machines (SVM)

SVM deals with separating points in a dataset into 2 or more classes by finding a line or plane in multi-dimensional spaces that divides the

classes as best as possible.
In a 2D want to find the line that is as far as possible from the nearest points of both classes.
If data are not linearly separable, SVM uses a technique called kernel. In particular, it transforms data into a space with multiple dimensions (e.g. from 2D to 3D). In this new space, data become separable with a hyperplane.



```
Risultati della cross-validation:
Average Precision: 0.9783283639142599
Average Recall: 0.9660152225643511
Average F1-Score: 0.971846212067437
```

5. Conclusion

According to the evaluation metrics, the choice of model falls on the Random Forest, a very robust classifier since it consists of a set of decision trees.

Although the co2 regression feature was discarded, the model performed very well achieving an F1-score close to the excellent.

Future developments could include the addition of new types of sensors, such as those for moisture detection to further improve the robustness and quality of the environment.

The positive results obtained demonstrate the potential of this approach in the area of energy management and improving occupant comfort.

6. Bibliography

Singh, A. & Chaudhari, S. (2018). Room Occupancy Estimation [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C5P605.

Paper: By A. Singh, Vivek Jain, S. Chaudhari, F. Kraemer, S. Werner, V. Garg. 2018
https://www.semanticscholar.org/paper/e631ea26f0fd88541f42b4e049d63d6b52d6d3ac