

SCIENCES

*Computer Science,*

Field Directors – Valérie Berthé and Jean-Charles Pomerol

---

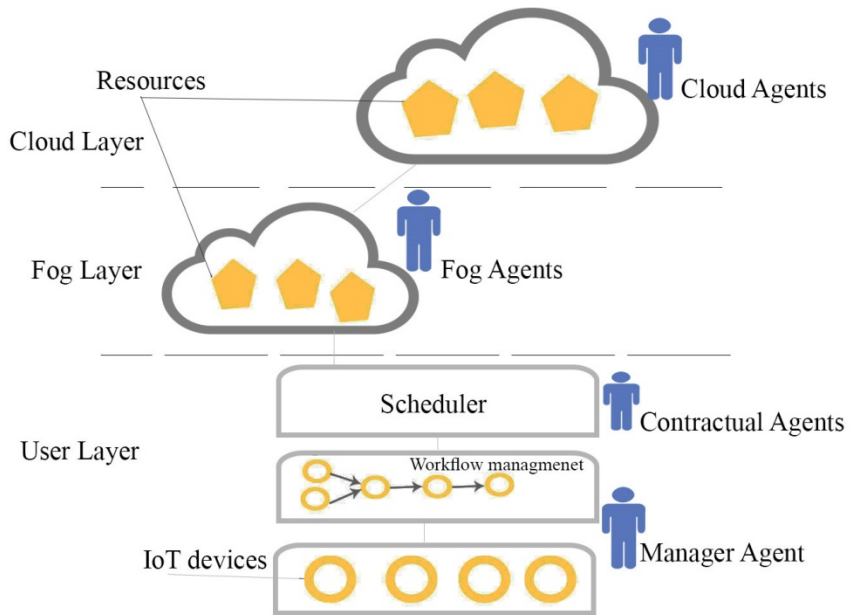
*Operational Research and Decision*, Subject Head – Patrick Siarry

# **Optimization and Machine Learning**

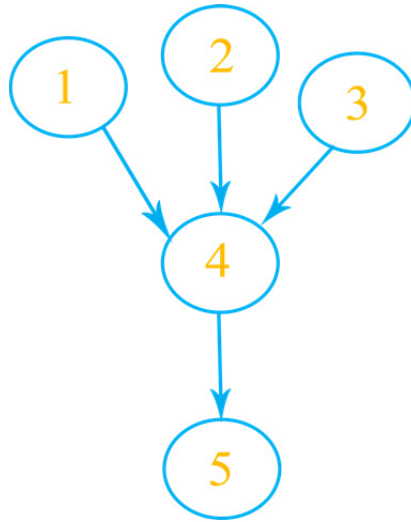
*Optimization for Machine Learning and  
Machine Learning for Optimization*

*Coordinated by*  
**Rachid Chelouah**  
**Patrick Siarry**

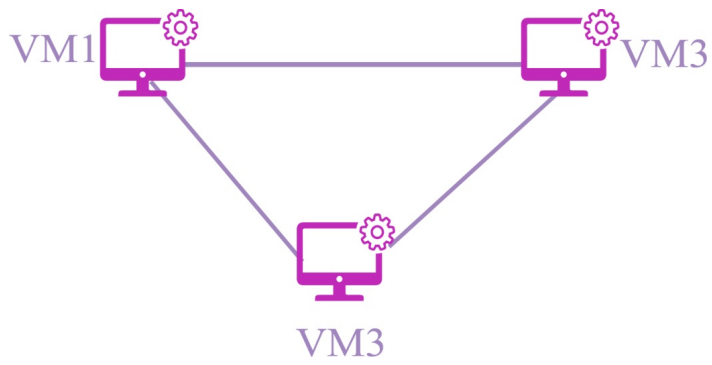
Color section



**Figure 2.1.** *Solution architecture*

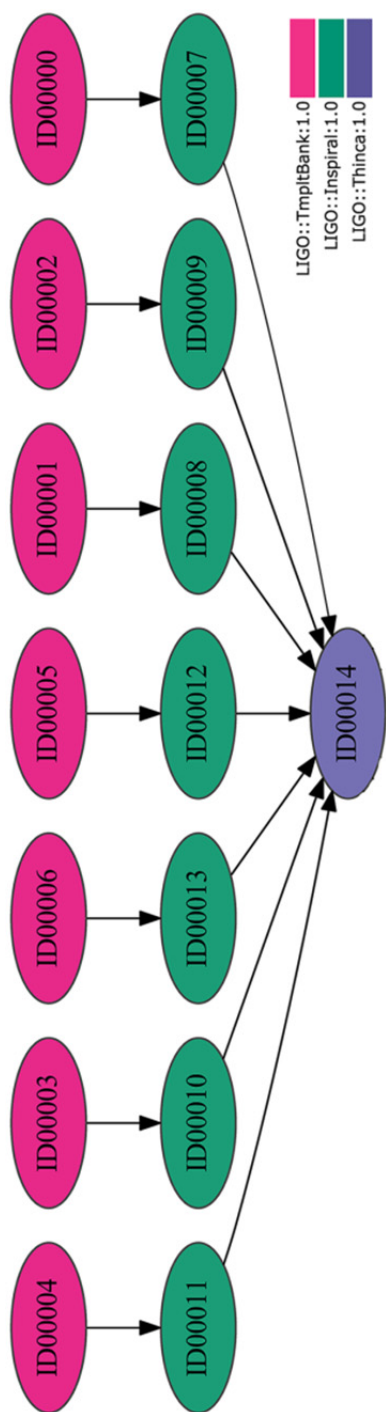


**Figure 2.4.** *Workflow example*

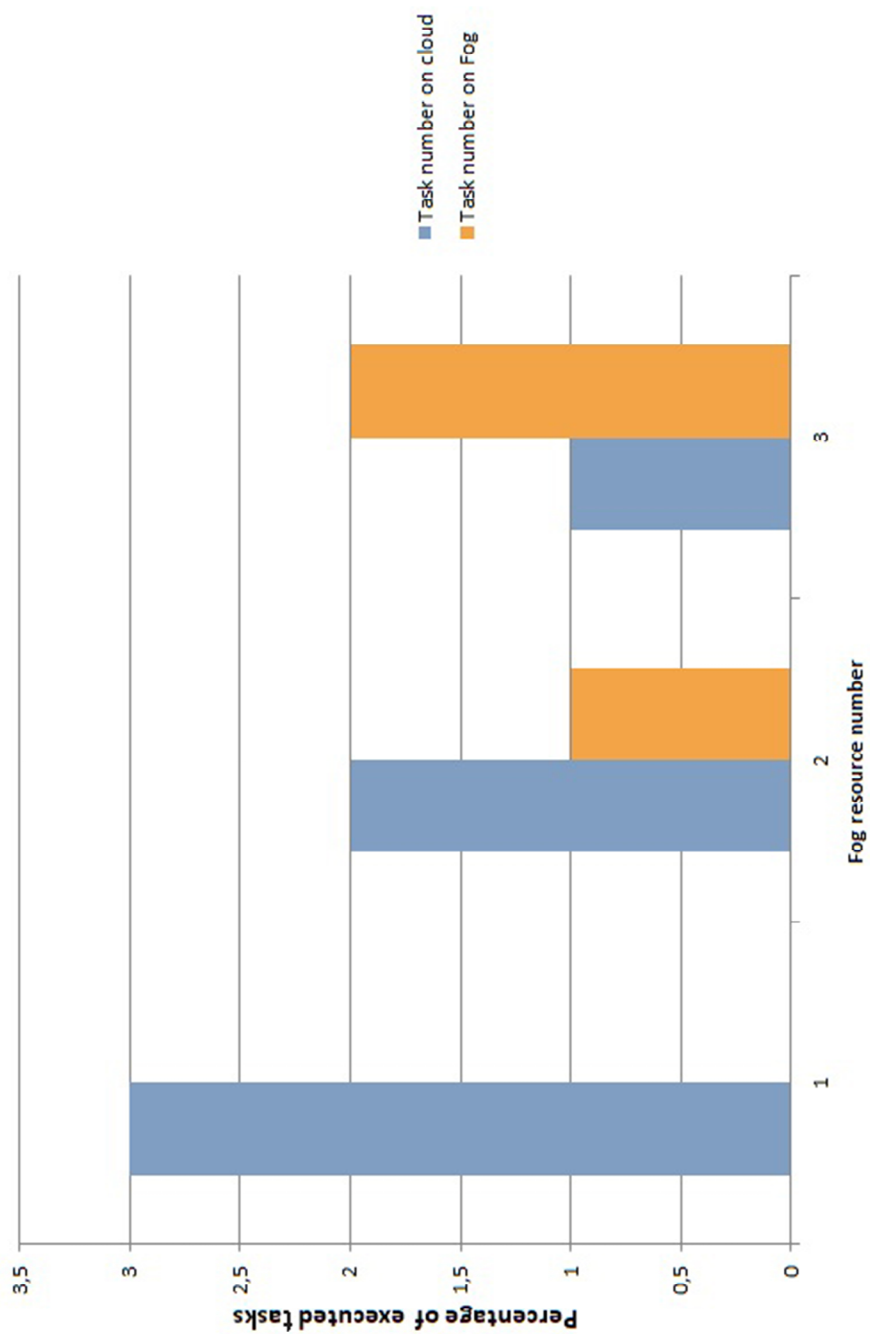


**Figure 2.5.** *Example of resources*

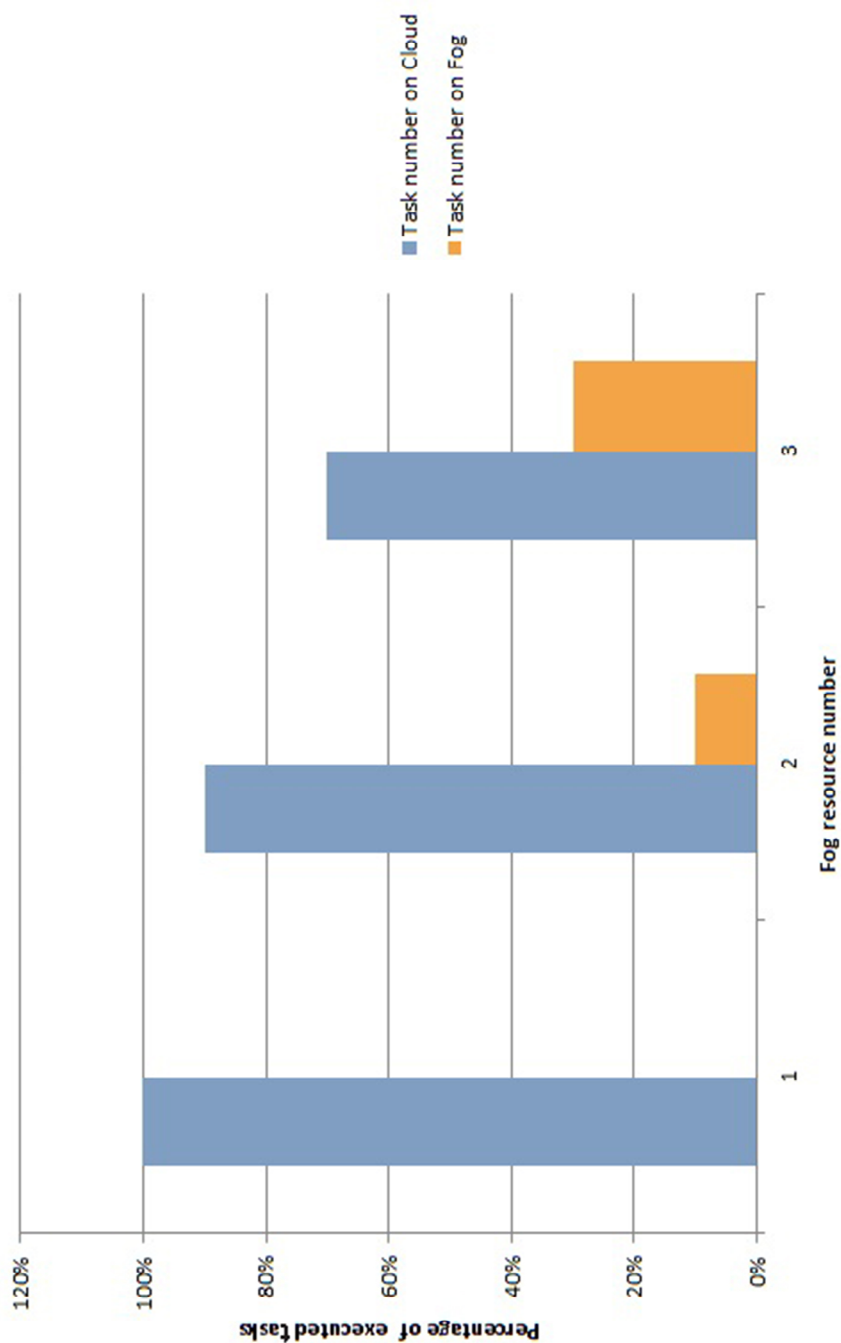




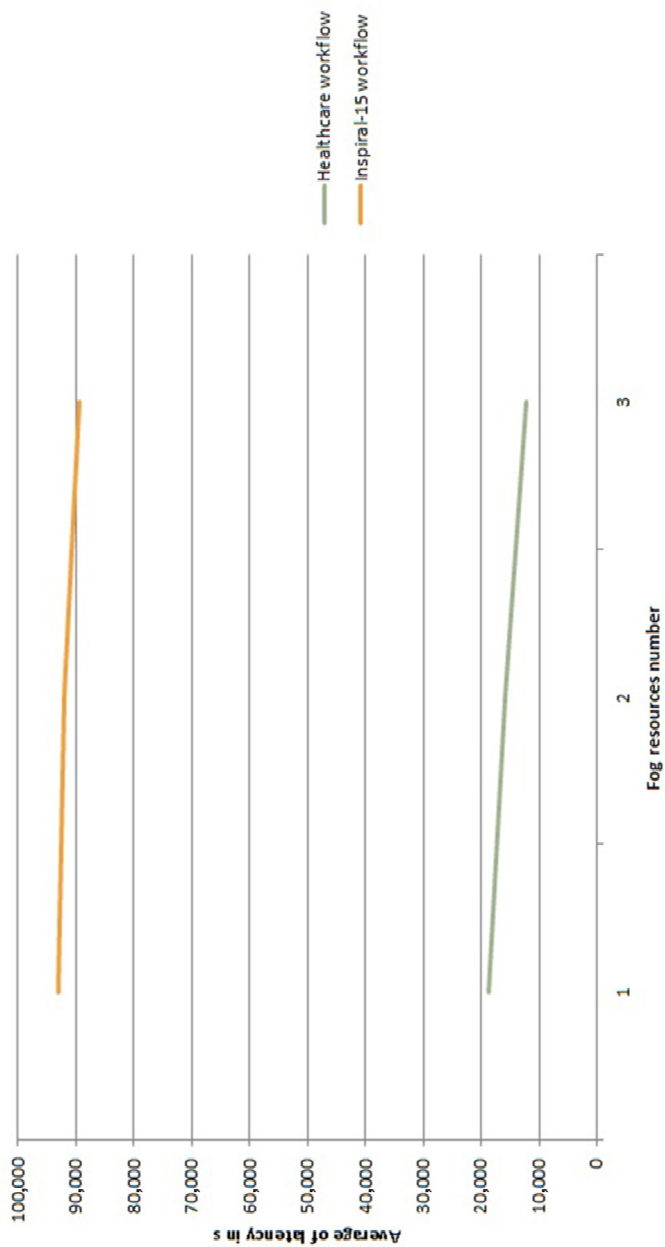
**Figure 2.9.** *Inspiral-15 workflow benchmark*



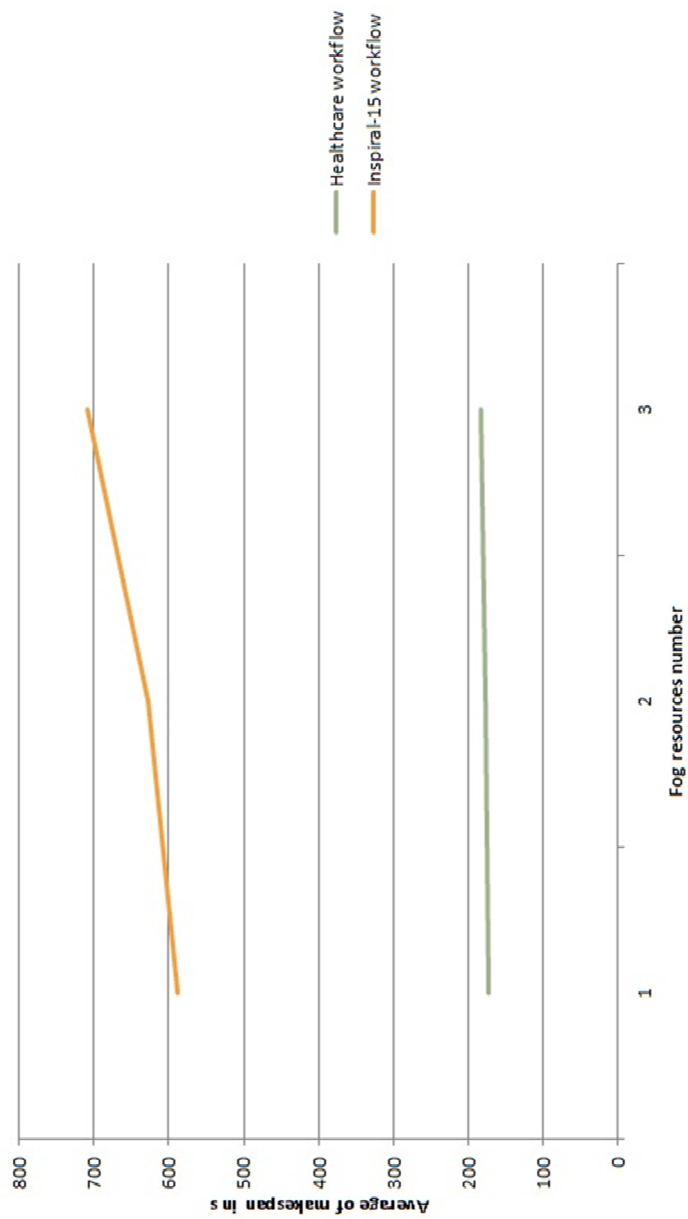
**Figure 2.10.** *Percentage of IoT workflow executed tasks on Fog computing and Cloud computing*



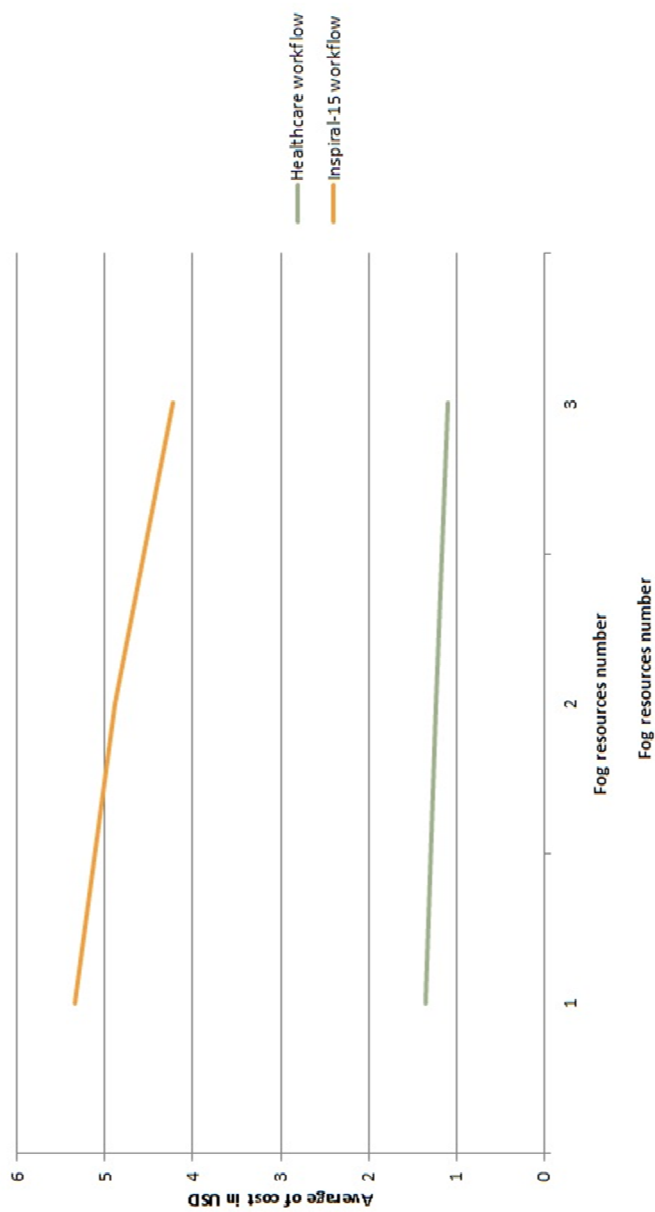
**Figure 2.11.** Percentage of Inspiral-15 workflow executed tasks on Fog computing and Cloud computing



**Figure 2.12.** Average of latency when executing the healthcare workflow and Inspiral-15 workflow



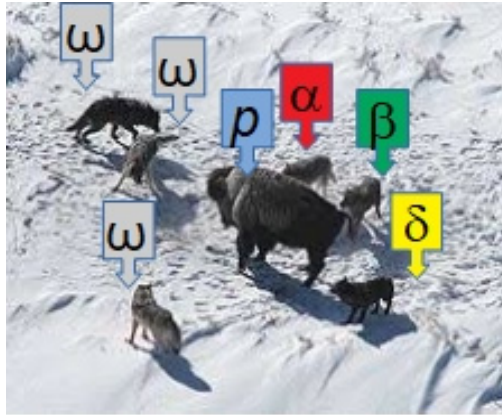
**Figure 2.13.** Average of makespan when executing the healthcare workflow and Inspiral-15 workflow



**Figure 2.14.** Average of cost when executing the healthcare workflow and Inspiral-15 workflow

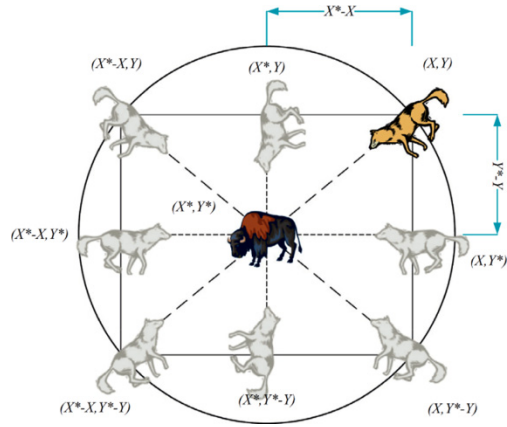


**Figure 3.2.** Search, pursue, surround-harass and attack the prey  
(source: <https://phys.org> (ecologists larger group))

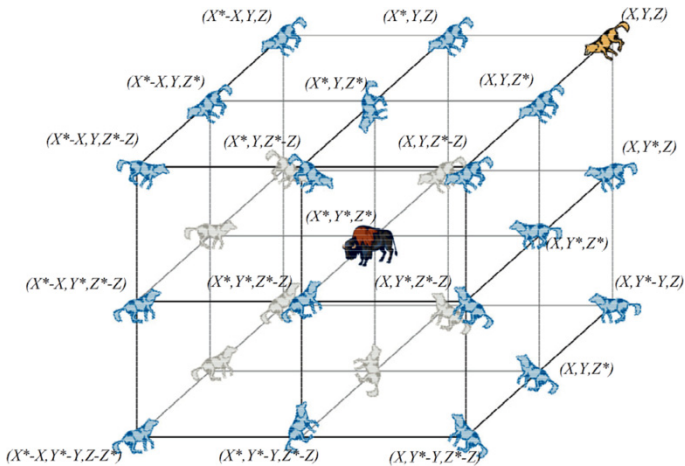


**Figure 3.4.** Alpha, Beta, Delta and Omega Wolf positions around prey  
(source: <https://phys.org> (ecologists larger group))

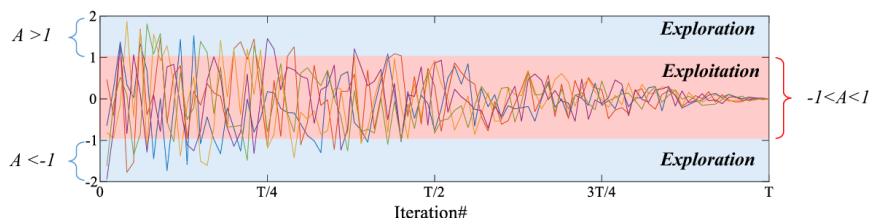




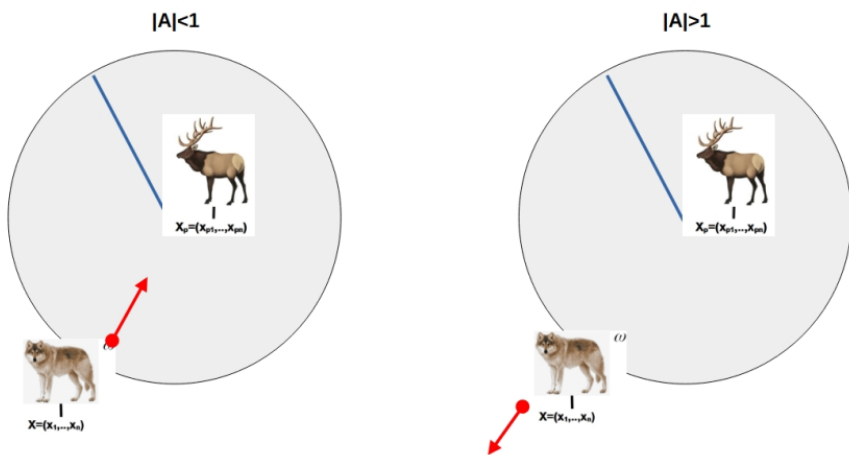
**Figure 3.5.** Position of the wolf agent and its prey in a square with  $N = 2$  (Mirjalili et al. 2014)



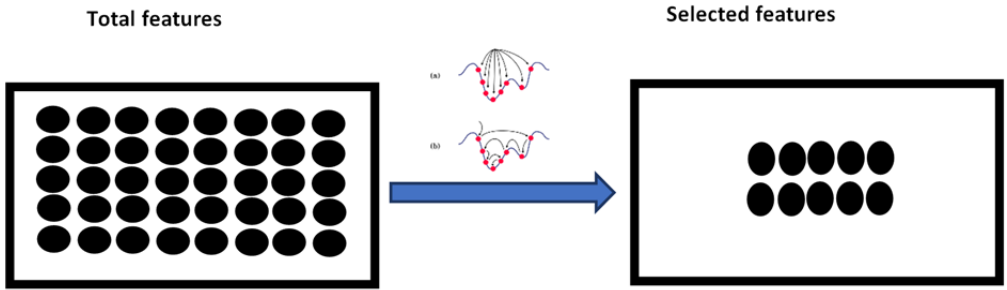
**Figure 3.6.** Position of the wolf agent and its prey in a cube with  $N = 3$  (Mirjalili et al. 2014)



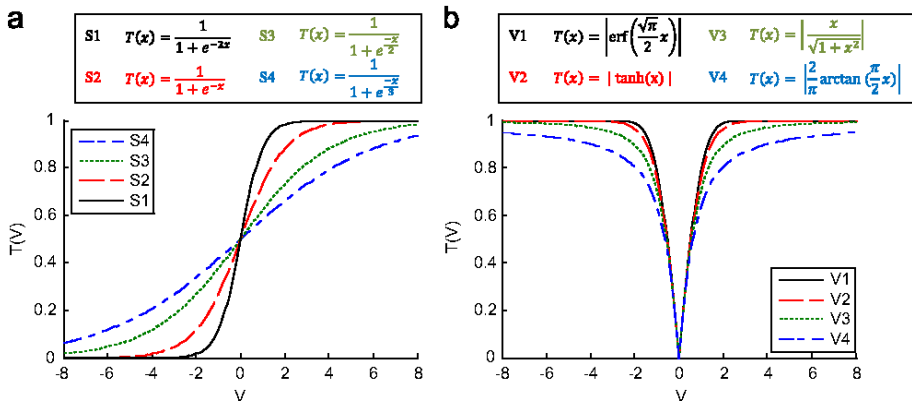
**Figure 3.7.** Steering of exploration and exploitation phases with vector  $A$  during iterations (Faris et al. 2017)



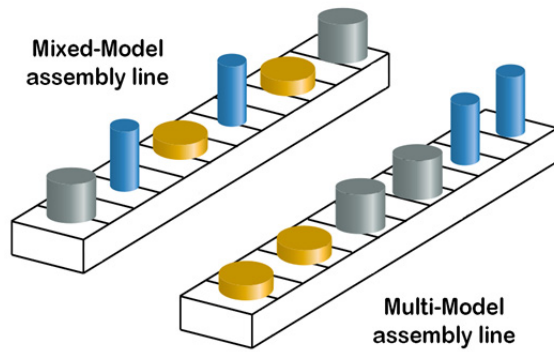
**Figure 3.8.** Search exploration phase  $|A| \geq 1$  and attack exploitation phase  $1 > |A| \geq 0$



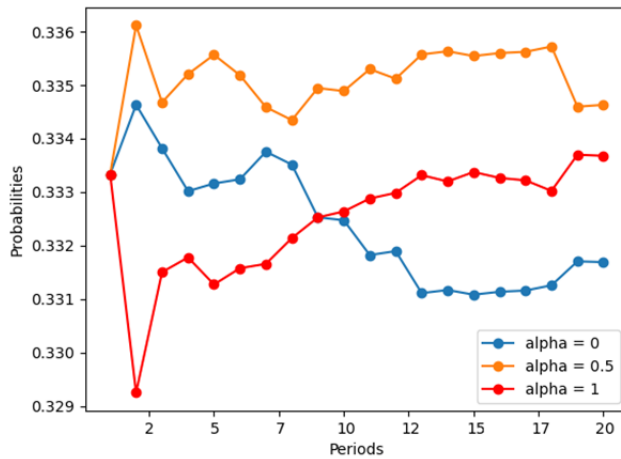
**Figure 3.9.** Feature selection process



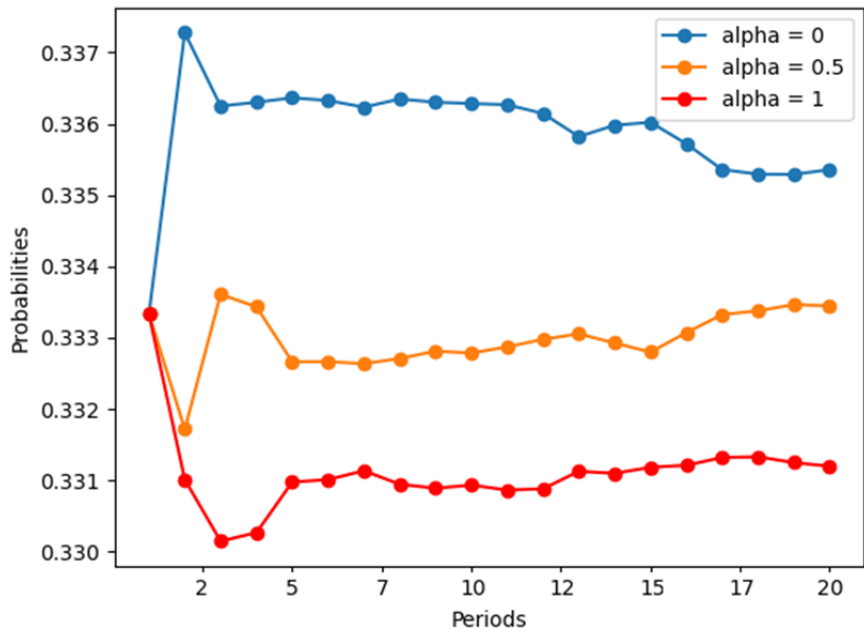
**Figure 3.13.** S and V transfer function (Mirjalili and Lewis 2013)



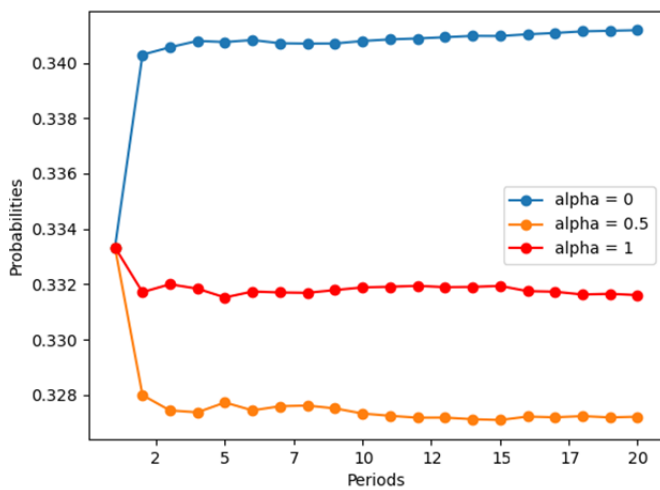
**Figure 4.2.** *Mixed- and multi-model assembly lines*



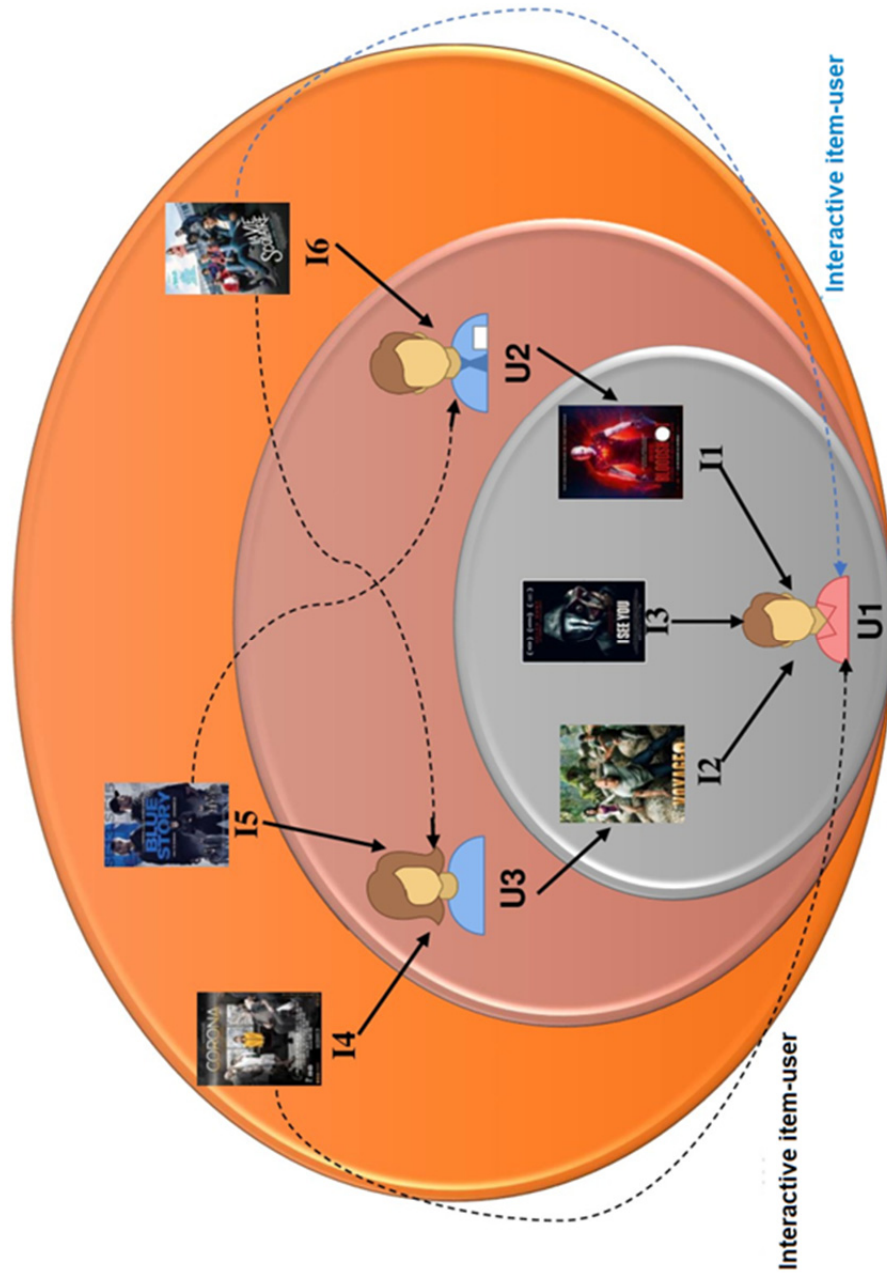
**Figure 4.5.** *Variation of probabilities during solving problem 01*



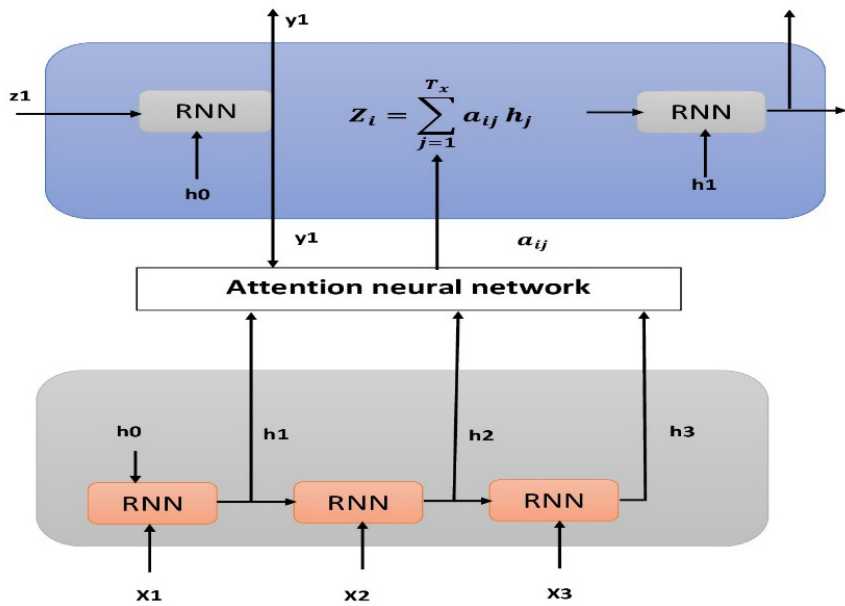
**Figure 4.6.** Variation of probabilities during solving problem 02



**Figure 4.7.** Variation of probabilities during solving problem 03

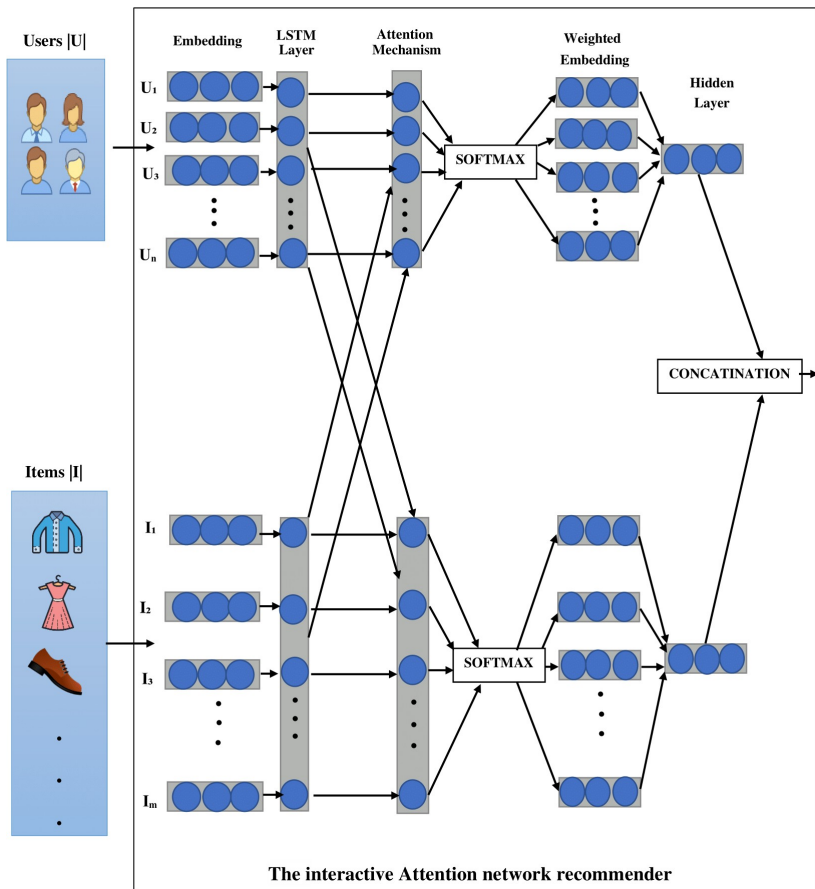


**Figure 5.1.** *The Interactive Personalized Recommender framework models the mutual influence between the different entities*

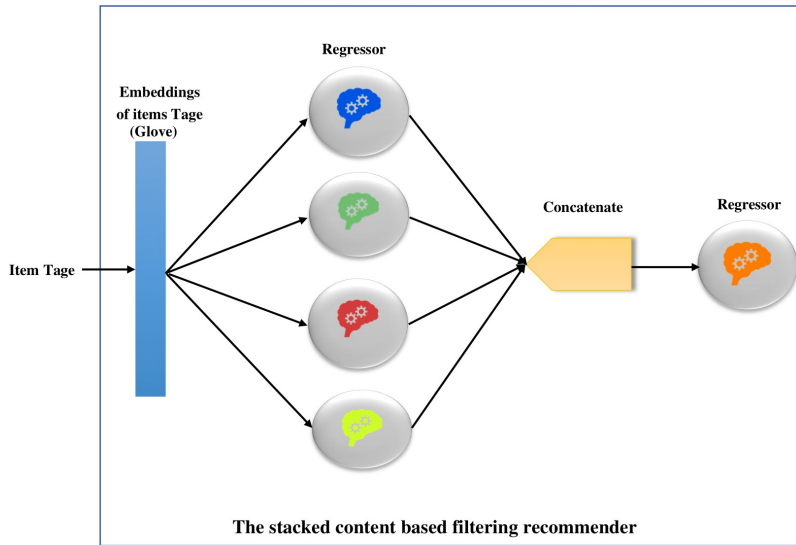


**Figure 5.2.** *The Interactive Personalized Recommender framework models the mutual influence between the different entities*

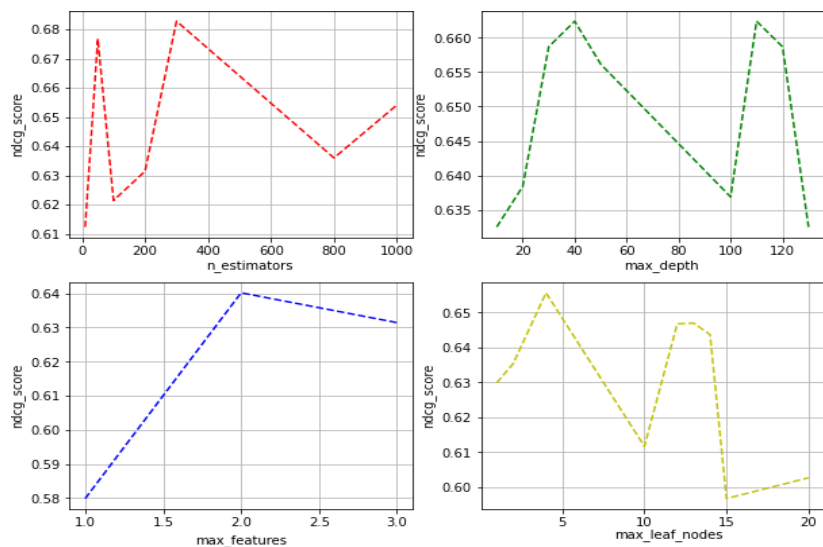




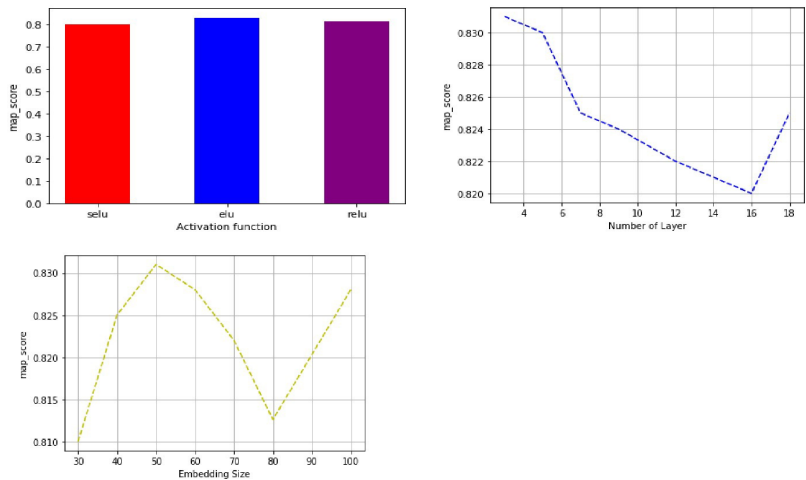
**Figure 5.3.** *The interactive attention network recommender*



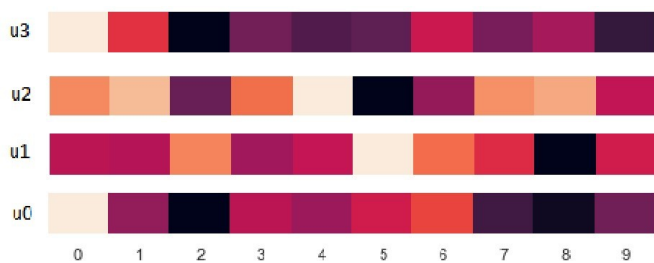
**Figure 5.4.** *The stacked content-based filtering recommender*



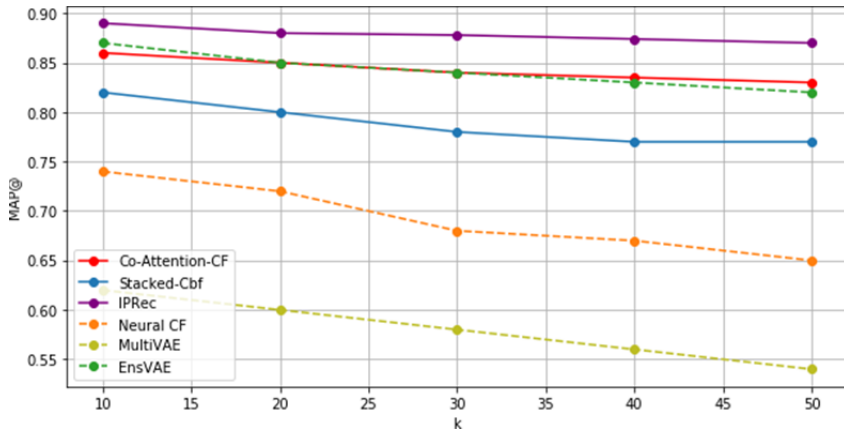
**Figure 5.5.** *Hyperparameter searching for the stack-based random forest*



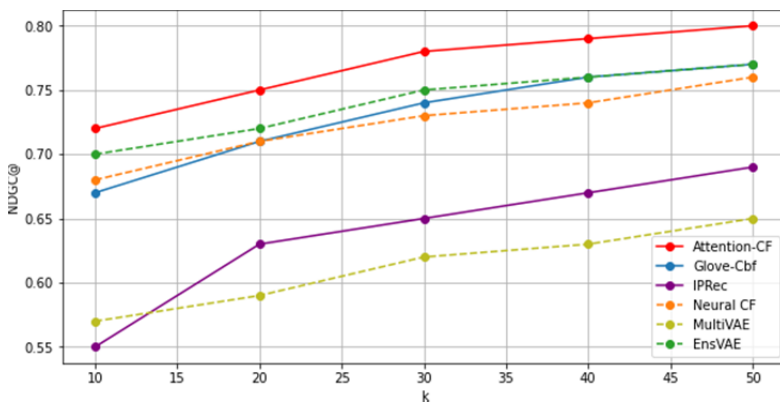
**Figure 5.6.** Hyperparameters analysis for interactive attention network recommender



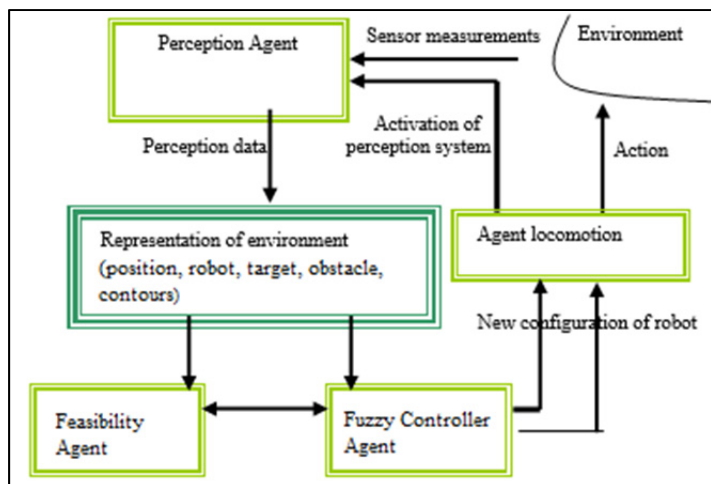
**Figure 5.7.** Visualization of the interactive co-attention weights for 10 items and four users, dark colors refer to low attention weights



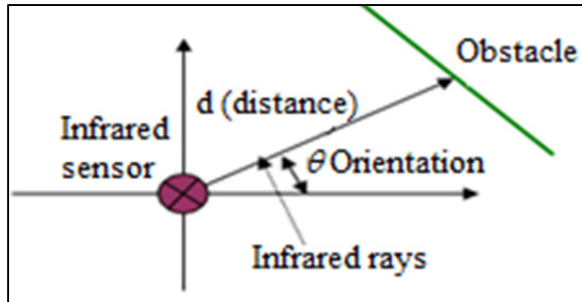
**Figure 5.8.** Results of the comparison on MovieLens dataset. Evaluation of the performance of top-K recommended lists, in terms of MAP. The ranking position K ranges from 1 to 50



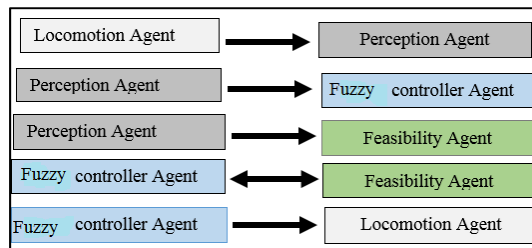
**Figure 5.9.** Results of the comparison on MovieLens dataset. Evaluation of the performance of top-K recommended lists, in terms of NDCG. The ranking position K ranges from 1 to 50



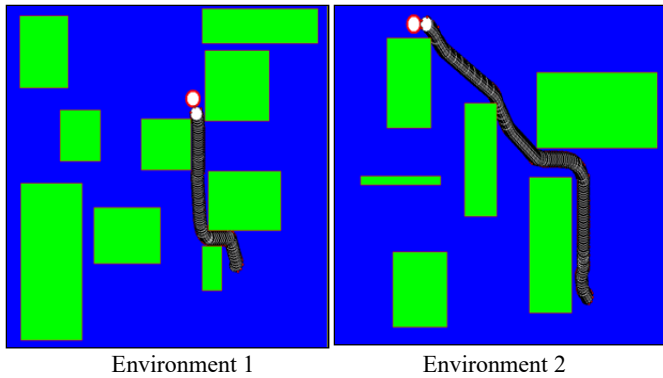
**Figure 7.3.** *Mobile robot control architecture based agent*



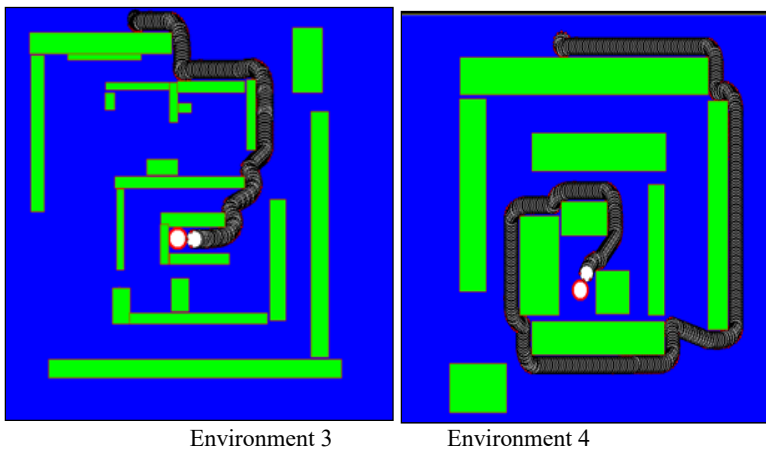
**Figure 7.4.** Measurements provided by infrared sensors (information of perception agent and provided to all agents)



**Figure 7.9.** Agent interaction and communication

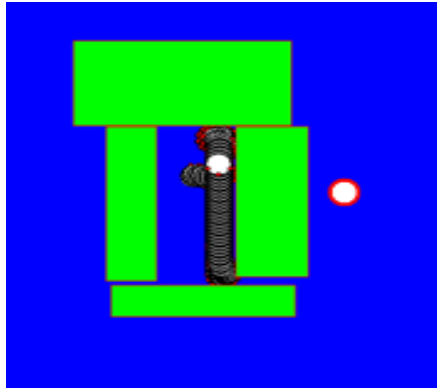


**Figure 7.26.** *The robot, in environments 1 and 2, reaches the target (all obstacles are avoided)*

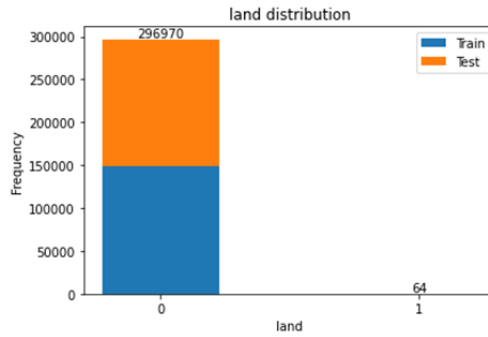


**Figure 7.27.** *The robot reaches the target in complex environments (environments 3 and 4) (all obstacles are avoided)*

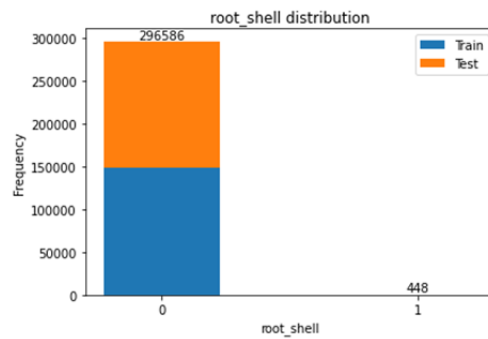




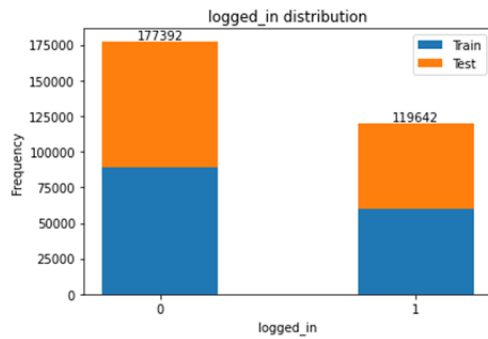
**Figure 7.28.** *The robot in a deadlock situation*



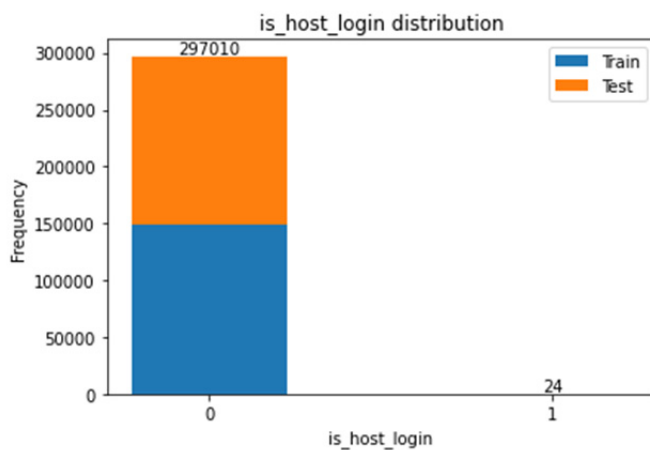
**Figure 8.1.** Land category distribution



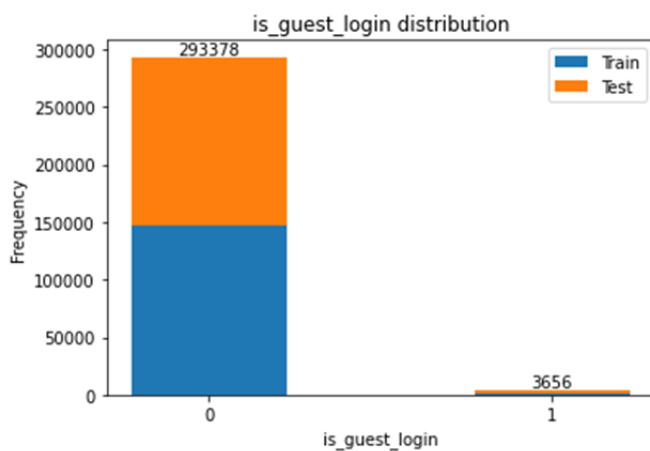
**Figure 8.2.** Root\_shell category distribution



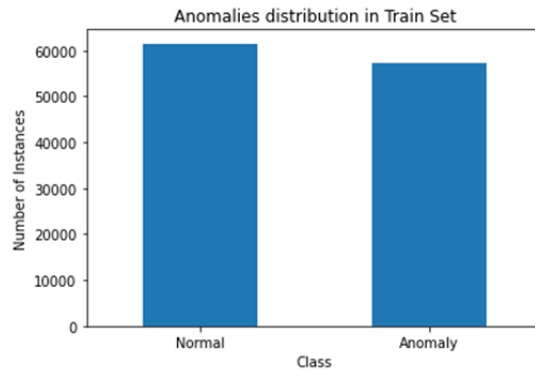
**Figure 8.3.** Logged\_in category distribution



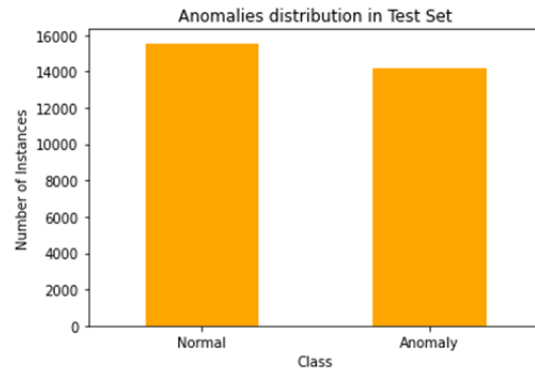
**Figure 8.4.** *Is\_host\_login* category distribution



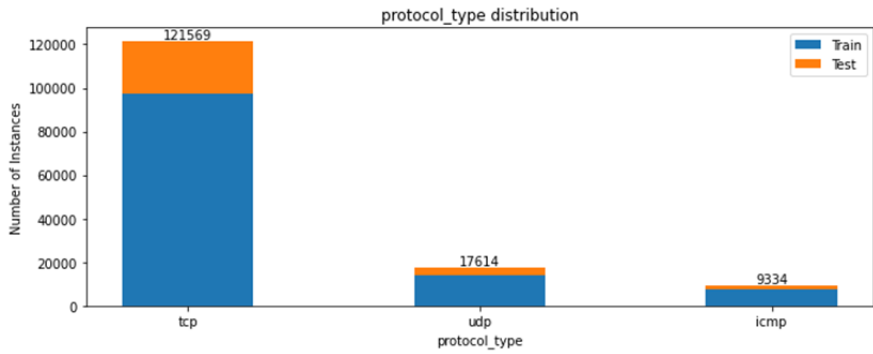
**Figure 8.5.** *Is\_guest\_login* category distribution



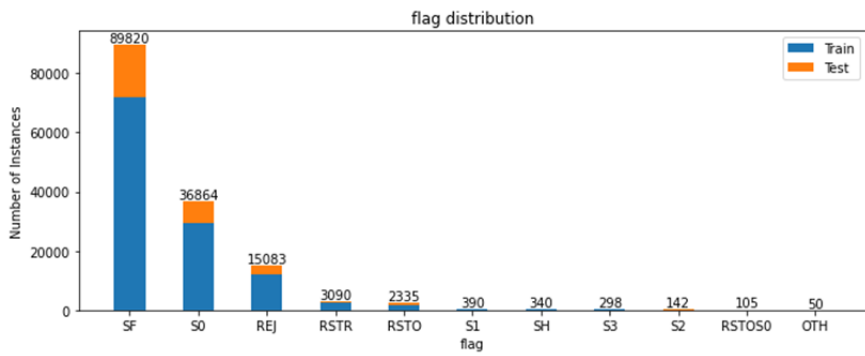
**Figure 8.6.** *Distribution of the target column values in the train set*



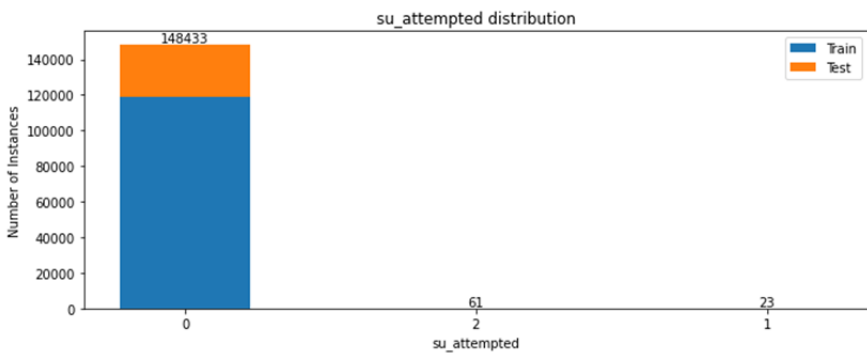
**Figure 8.7.** *Distribution of the target column values in the test set*



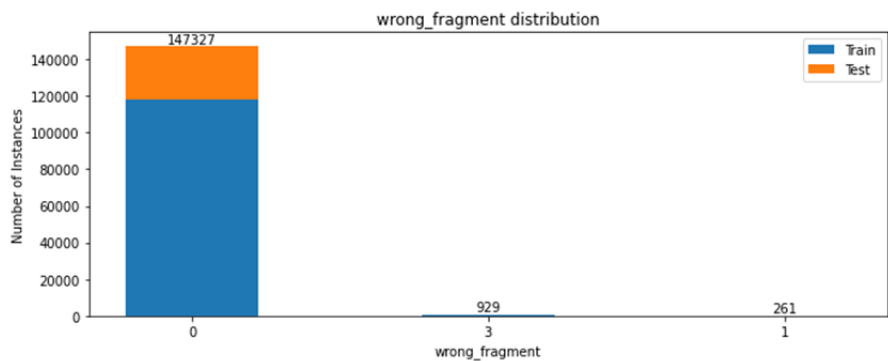
**Figure 8.8.** Protocol type value distribution



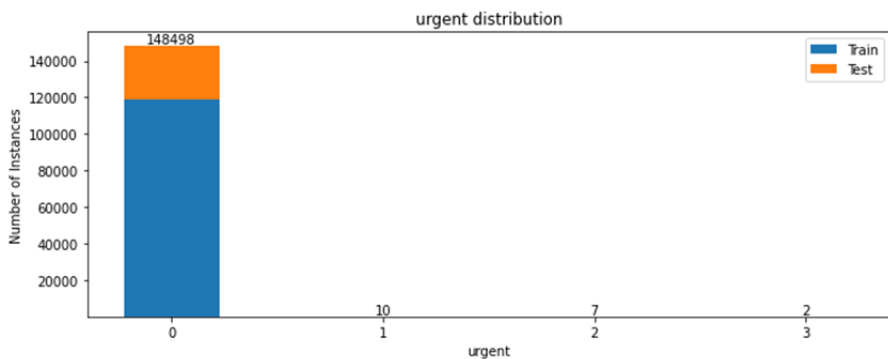
**Figure 8.9.** Flag value distribution



**Figure 8.10.** Su attempted value distribution



**Figure 8.11.** *Wrong fragment value distribution*



**Figure 8.12.** *Urgent value distribution*

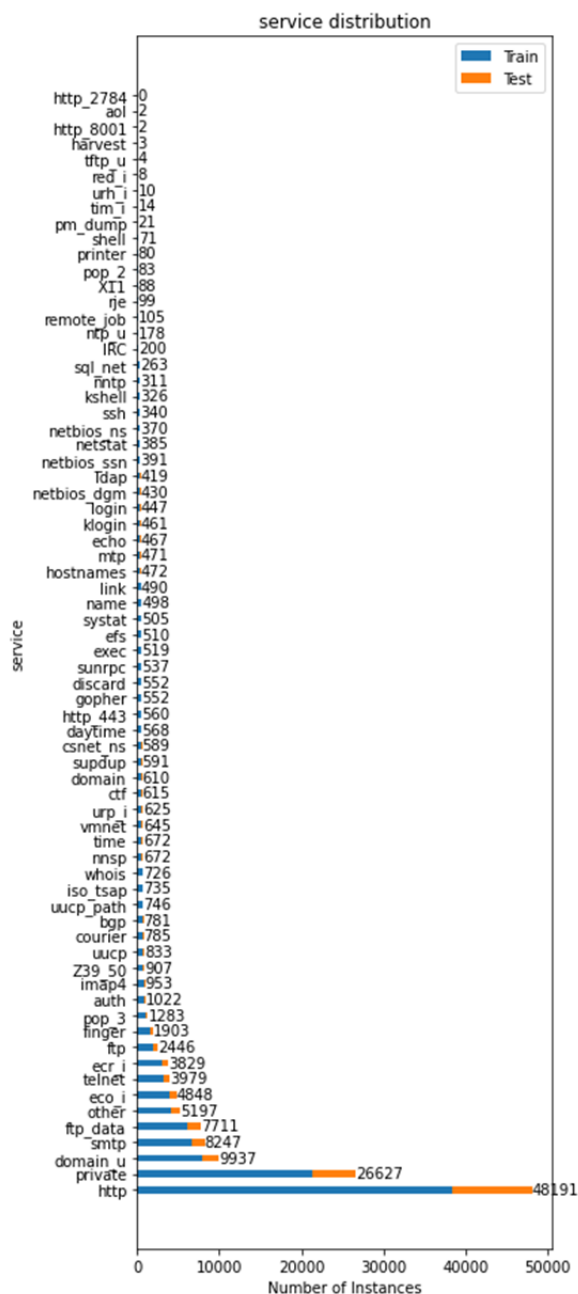
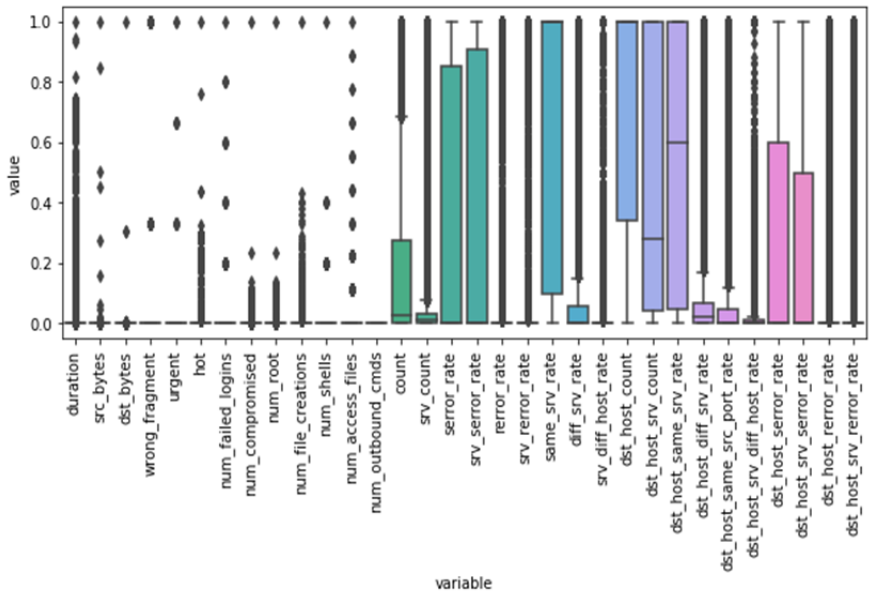
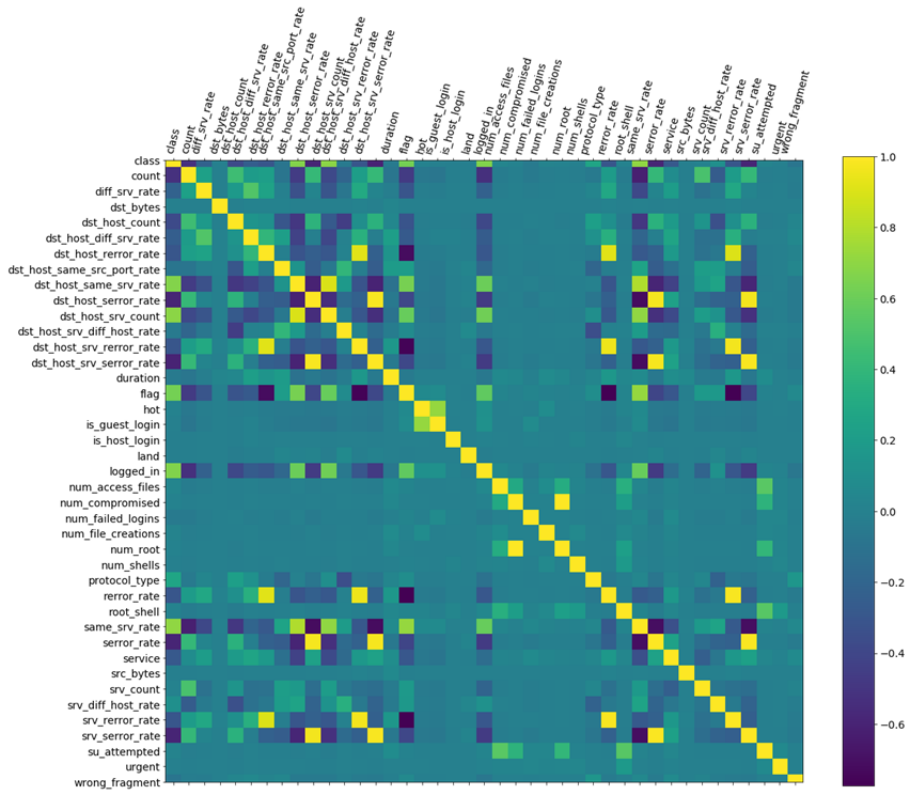


Figure 8.13. Service value distribution

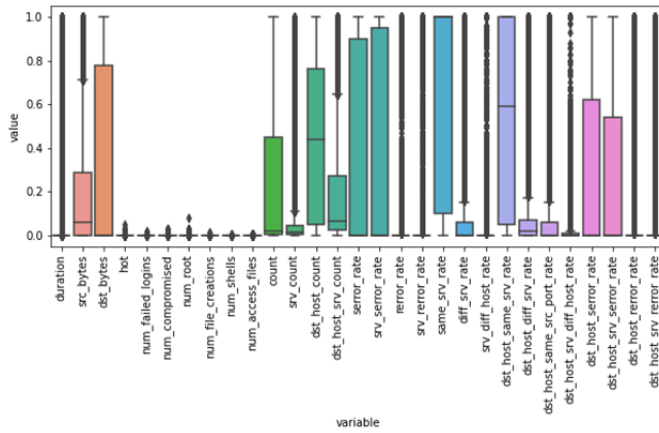


**Figure 8.14.** *Distribution of discrete and continuous features inside the normalized dataset*

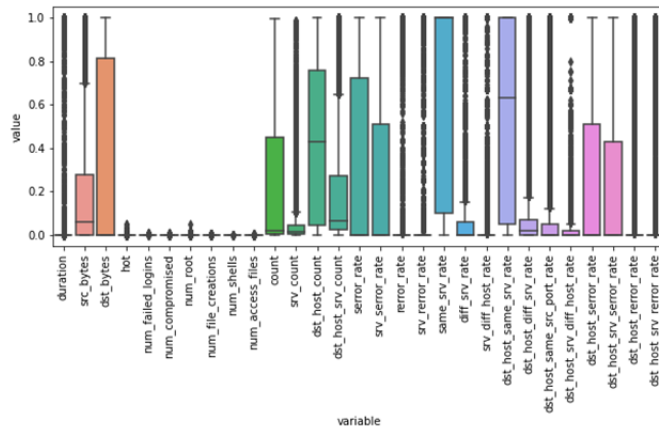




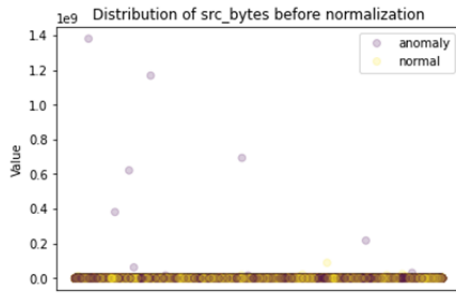
**Figure 8.15.** Correlation matrix of all the features of the dataset. A lighter color means higher correlation



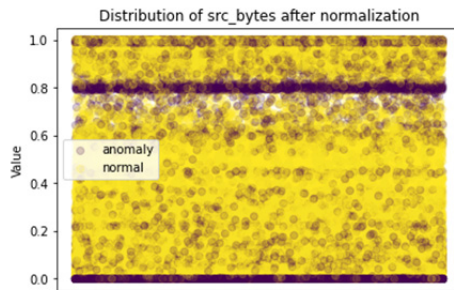
**Figure 8.16.** Distribution of the values of each numerical feature in the normalized train set



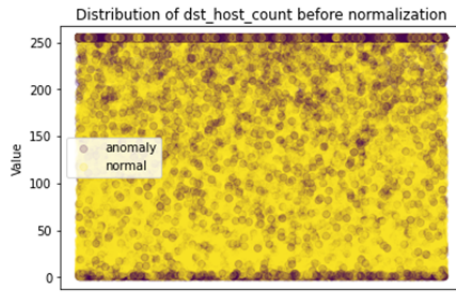
**Figure 8.17.** Distribution of the values of each numerical feature in the normalized test set



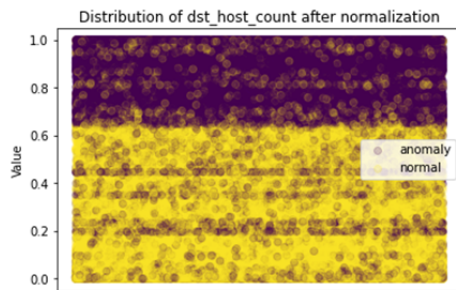
**Figure 8.18.** *Distribution of src\_bytes before normalization*



**Figure 8.19.** *Distribution of src\_bytes after normalization*

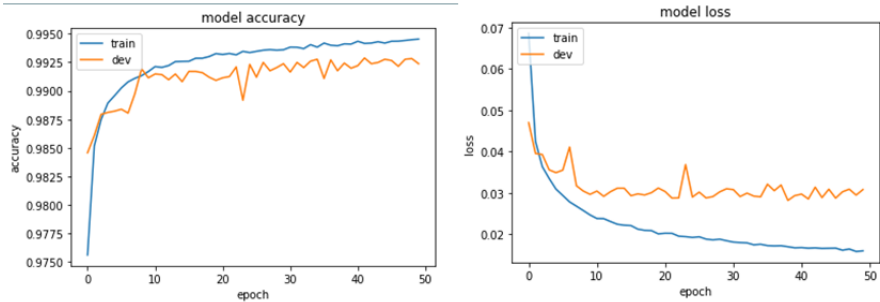


**Figure 8.20.** *Distribution of dst\_host\_count before normalization*

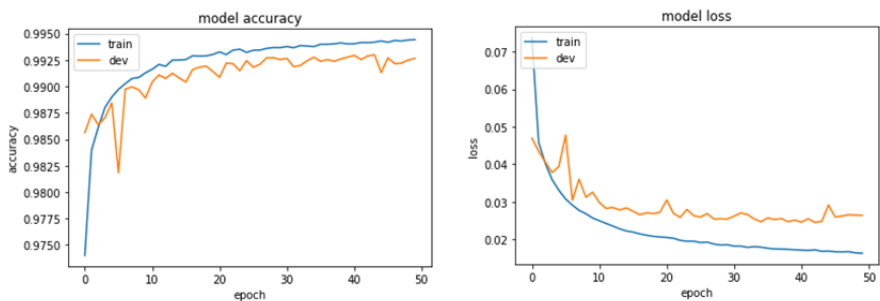


**Figure 8.21.** *Distribution of dst\_host\_count after normalization*

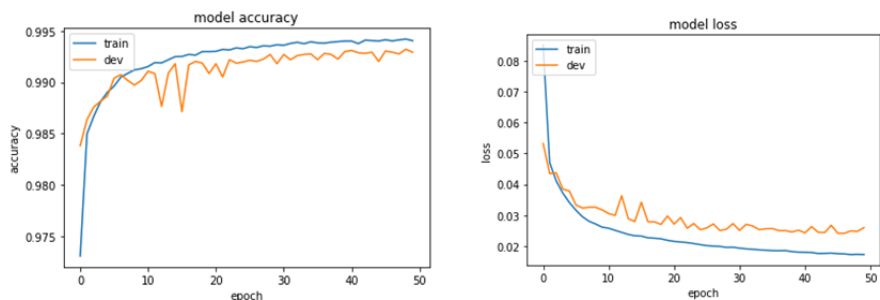




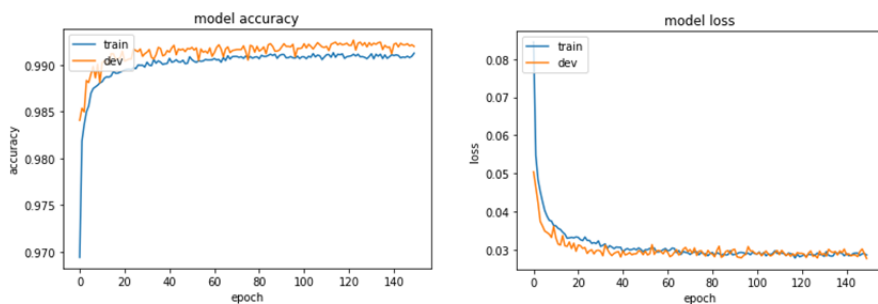
**Figure 8.24.** Learning curves (accuracy and loss) of the first model (all features, no dropout, relu activation function for hidden layers)



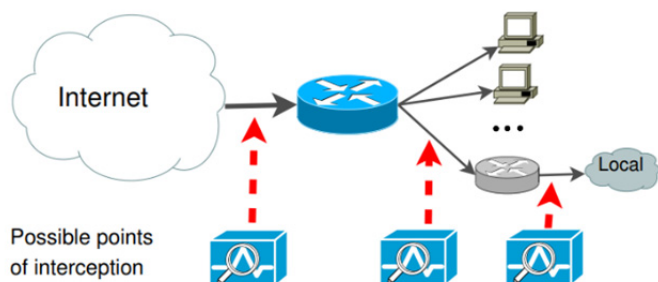
**Figure 8.25.** Learning curves (accuracy and loss) of the second model (all features, no dropout, tanh activation function for hidden layers)



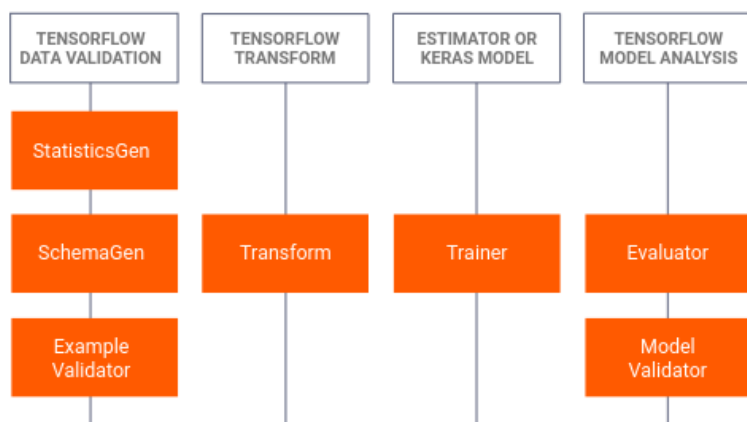
**Figure 8.26.** Learning curves (accuracy and loss) of the third model (all features, no dropout, sigmoid activation function for hidden layers)



**Figure 8.27.** Learning curves (accuracy and loss) of the low dropout model

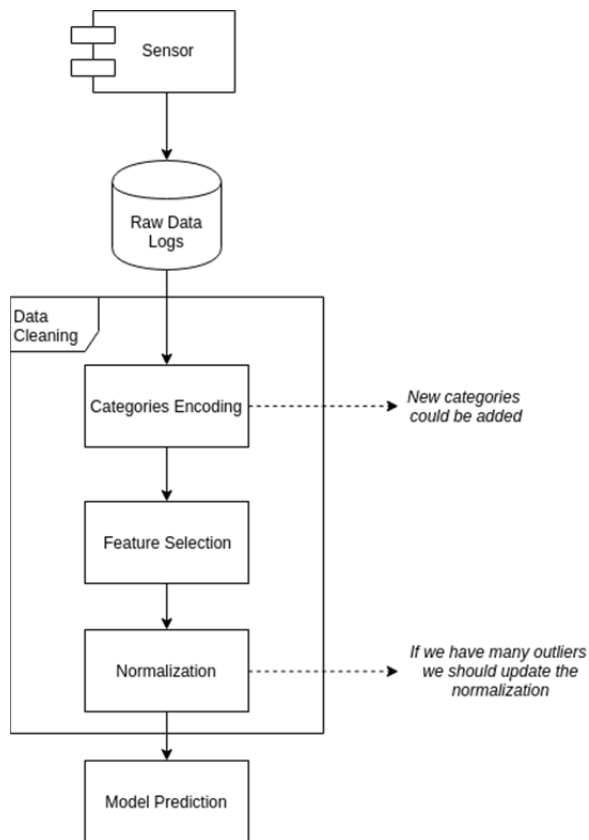


**Figure 8.28.** Possible points of interception for an NIDS (source: Kim et al. (2012))



**Figure 8.29.** Possible steps of the IDS once deployed in a real system (source: <https://www.tensorflow.org/tfx/guide/serving>)





**Figure 8.30.** Components of the TFX framework as described in the official guide