Predicting Football Match Results using Machine Learning

Ishan Jawade[1], Rushikesh Jadhav[2], Mark Joseph Vaz[3], Vaishnavi Yamgekar[4]

[1-4]*Student, Dept. of Computer Science and Engineering, MIT School of Engineering, Maharashtra, India*

Abstract- *Analyzing statistics of football teams can help clubs predict their performance over a particular time frame. In this paper we use various machine learning algorithms to predict results of Premier League season 2017-2018 for home/away win or draw and analyze the important attributes that impact the full-time result. Games routinely gather information on how the player has the play. The knowledge is fed into an algorithm which is used by humans to pull games from its predictions of what players would see. Predictions help the manager of the squad to take the next step. By spotting weaknesses at the fighting team's defensive strategy, the weakness of a specific player or selecting the statistically most possible reaction to the move from past history, coaches might get an edge over their competition. We have done a comparative study between different machine learning algorithms and used the algorithm with the highest accuracy for our project.*

Key Words: Naïve Bayes, Linear SVC, Machine Learning

1.INTRODUCTION

As a sport football is played globally and has the highest fan base which is around 3.5 billion fans across the globe. It is played in more than 180 countries in the world. Most of the fans can predict the results as who might win the match. Anyone can predict on the basis of features like home stadium, team form, squad strength, win percentage and other features. Prediction is very useful in helping club staff make the right decision regarding training and player management, it also helps the teams to prepare for their future play based on other team's performance. Premier League - the English Premier association is regarded by some to be the most fun part of football on this planet and its sort of difficult to contend against that. Some of reality's top clubs compete there and when it comes to the businesses needed, it's somewhat tough to tell they aren't in the top of the list. Manchester United are apparently thought to take this biggest family, but alongside them you've had the likes of Liverpool, Manchester City, Chelsea and some more. It's hard to quantify the human truth about predicting the outcome of football matches. Results vary according to what matches are anticipated. Predictions on various leagues and tournaments make several accuracies and humans forecast the result on a much smaller collection of leagues and tournaments than this system. This makes the system hard to equate with human reality. We have developed machine learning models in order to predict full time results of the Premier League table of the year 2017-2018. Our work predicts which team will win the match(home/away/draw).

2. PROPOSED METHOD

We have used three models in our system Linear Regression, Support Vector Machine, Logistic Regression, Random Forest and Multinomial Naive Bayes classifier. We have used LinearSVC for multinomial classification which is the problem of classifying instances into one of three or more classes. In our experiment we have classified our results into 3 classes (i.e., Win(H), Draw(D) and Loss(A)) [10].

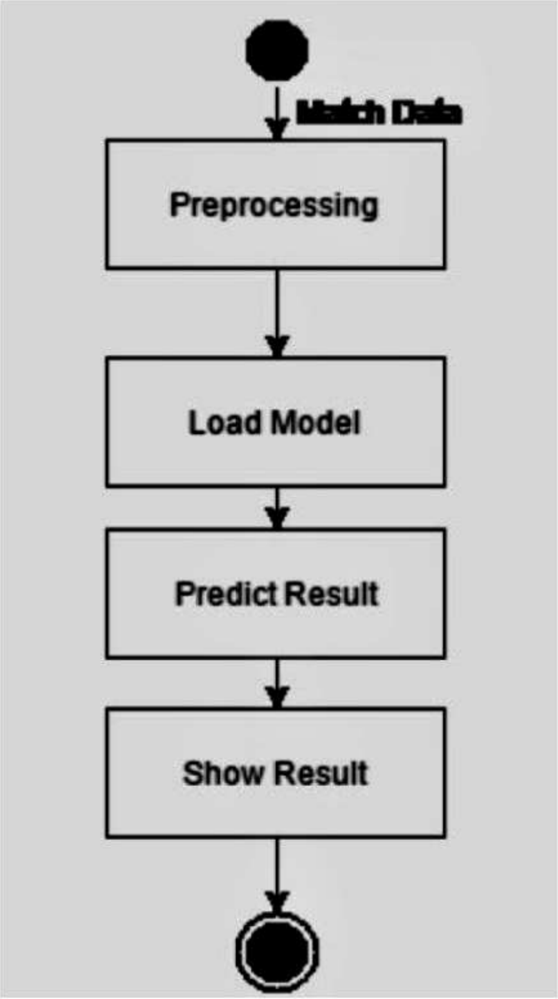
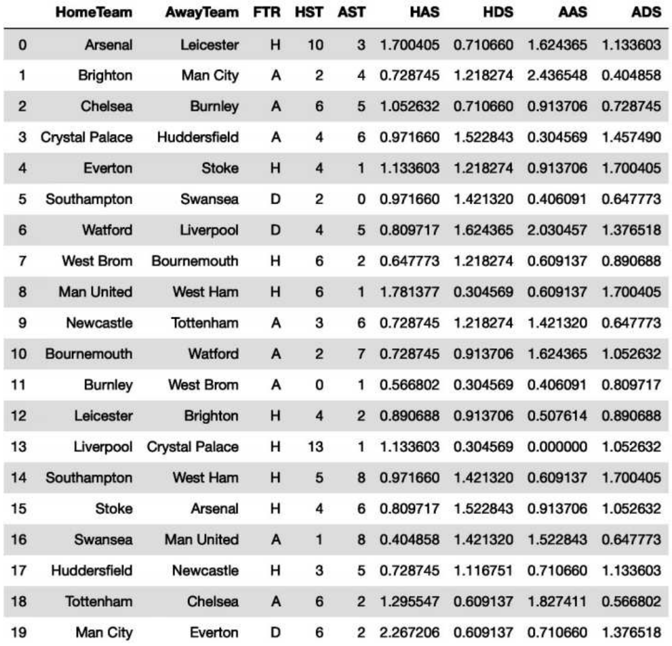


Fig -1: Proposed Testing Process Flow

We have used data obtained for 2017-2018 season of the English Premiere League on which we have used standard data pre-processing steps. We have then calculated goals scored, conceded and team form to help us calculate important attributes. We then use this data on the above mentioned machine learning models.

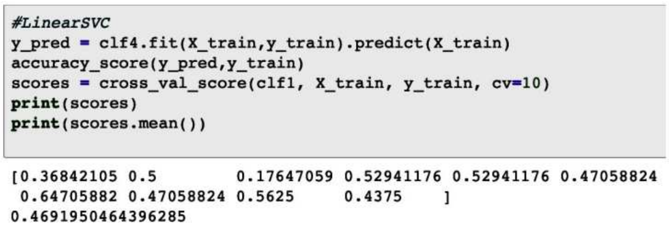


	HomeTeam	AwayTeam	FTR	HST	AST	HAS	HDS	AAS	ADS
0	Arsenal	Leicester	H	10	3	1.700405	0.710660	1.624365	1.133603
1	Brighton	Man City	A	2	4	0.728745	1.218274	2.436548	0.404858
2	Chelsea	Burnley	A	6	5	1.052632	0.710660	0.913706	0.728745
3	Crystal Palace	Huddersfield	A	4	6	0.971660	1.522843	0.304569	1.457490
4	Everton	Stoke	H	4	1	1.133603	1.218274	0.913706	1.700405
5	Southampton	Swansea	D	2	0	0.971660	1.421320	0.406091	0.647773
6	Watford	Liverpool	D	4	5	0.809717	1.624365	2.030457	1.376518
7	West Brom	Bournemouth	H	6	2	0.647773	1.218274	0.609137	0.890688
8	Man United	West Ham	H	6	1	1.781377	0.304569	0.609137	1.700405
9	Newcastle	Tottenham	A	3	6	0.728745	1.218274	1.421320	0.647773
10	Bournemouth	Watford	A	2	7	0.728745	0.913706	1.624365	1.052632
11	Burnley	West Brom	A	0	1	0.566802	0.304569	0.406091	0.809717
12	Leicester	Brighton	H	4	2	0.890688	0.913706	0.507614	0.890688
13	Liverpool	Crystal Palace	H	13	1	1.133603	0.304569	0.000000	1.052632
14	Southampton	West Ham	H	5	8	0.971660	1.421320	0.609137	1.700405
15	Stoke	Arsenal	H	4	6	0.809717	1.522843	0.913706	1.052632
16	Swansea	Man United	A	1	8	0.404858	1.421320	1.522843	0.647773
17	Huddersfield	Newcastle	H	3	5	0.728745	1.116751	0.710660	1.133603
18	Tottenham	Chelsea	A	6	2	1.295547	0.609137	1.827411	0.566802
19	Man City	Everton	D	6	2	2.267206	0.609137	0.710660	1.376518

Fig -2: Dataset with new attributes

3. EXPERIMENT

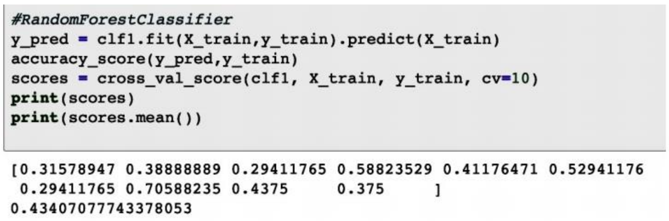
We then test it against a matchday where 10 games are played on the weekend by the 20 teams in the Premier League. We then predict the accuracy for each of the algorithms. We then run our machine learning algorithms on them and calculate the accuracy. Test result for the algorithms can be seen in Fig2, Fig3 and Fig4.



```
#LinearSVC
y_pred = clf4.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf1, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.36842105 0.5        0.17647059 0.52941176 0.52941176 0.47058824
 0.64705882 0.47058824 0.5625     0.4375    ]
0.46919504643962850
```

Fig -3: Linear SVC Results

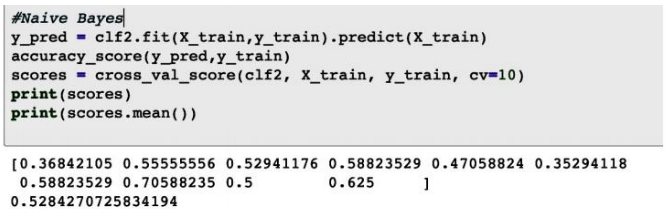


```
#RandomForestClassifier
y_pred = clf1.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf1, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.31578947 0.38888889 0.29411765 0.58823529 0.41176471 0.52941176
 0.29411765 0.70588235 0.4375     0.375     ]
0.43407077743378053
```

Fig-4: Random Forest Results

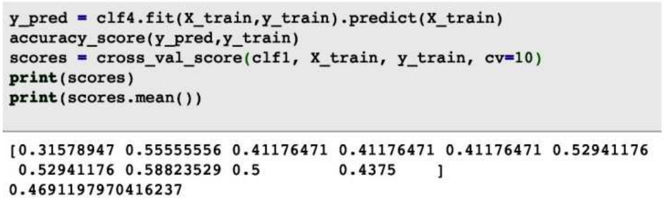
The accuracy is lower so we add more important attributes which are influential to the result of a game. We add recent performances of the teams to improve the accuracy. In football the particular form of a team is very important factor which can be very effective especially while predicting the outcome of the game. We calculate the form of the team based on their previous six results.



```
#Naive Bayes
y_pred = clf2.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf2, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.36842105 0.55555556 0.52941176 0.58823529 0.47058824 0.35294118
 0.58823529 0.70588235 0.5        0.625     ]
0.5284270725834194
```

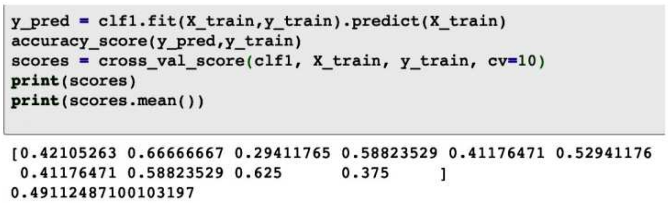
Fig-5: Naive Bayes Results



```
y_pred = clf4.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf1, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.31578947 0.55555556 0.41176471 0.41176471 0.41176471 0.52941176
 0.52941176 0.58823529 0.5        0.4375    ]
0.4691197970416237
```

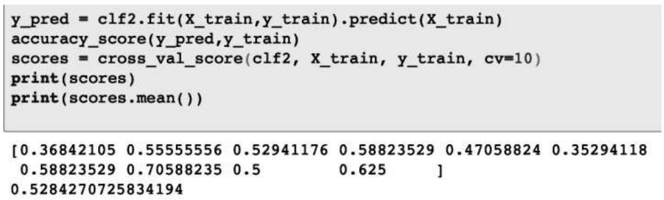
Fig-6: Linear SVC Results



```
y_pred = clf1.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf1, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.42105263 0.66666667 0.29411765 0.58823529 0.41176471 0.52941176
 0.41176471 0.58823529 0.625      0.375     ]
0.49112487100103197
```

Fig-7: Random Forest Results

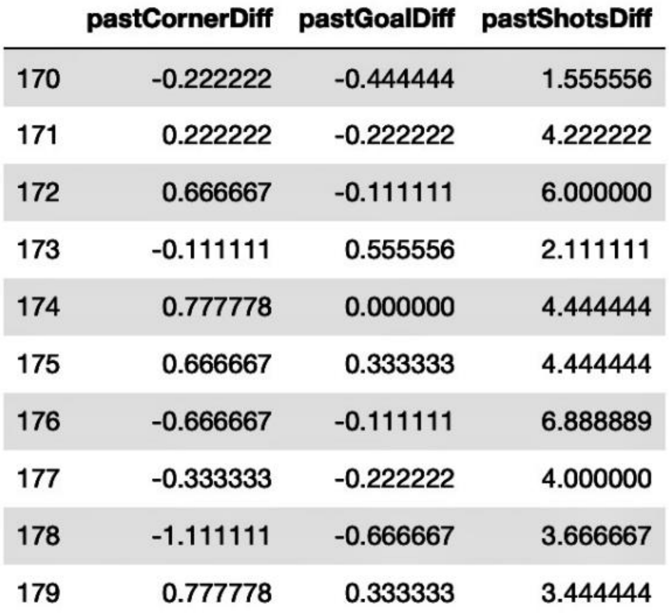


```
y_pred = clf2.fit(X_train,y_train).predict(X_train)
accuracy_score(y_pred,y_train)
scores = cross_val_score(clf2, X_train, y_train, cv=10)
print(scores)
print(scores.mean())
```

```
[0.36842105 0.55555556 0.52941176 0.58823529 0.47058824 0.35294118
 0.58823529 0.70588235 0.5        0.625     ]
0.5284270725834194
```

Fig-8: Naive Bayes Results

We see an improvement in the results after adding recent performance. To further improve our model, we use stadium advantage. In football stadiums of the teams are very difficult for opponent teams to play at and majority of the times the home team wins therefore adding this attribute will improve our results. We use various attributes to calculate how much home advantage a team has at their stadium. We use attributes such as past home shots, past home corner, past away shots, past away corners, past home goals, past away goals, result, past corner difference, past goal difference and past shots difference. The resultant data can be see of the most important attribute in Fig9.



	pastCornerDiff	pastGoalDiff	pastShotsDiff
170	-0.222222	-0.444444	1.555556
171	0.222222	-0.222222	4.222222
172	0.666667	-0.111111	6.000000
173	-0.111111	0.555556	2.111111
174	0.777778	0.000000	4.444444
175	0.666667	0.333333	4.444444
176	-0.666667	-0.111111	6.888889
177	-0.333333	-0.222222	4.000000
178	-1.111111	-0.666667	3.666667
179	0.777778	0.333333	3.444444

Fig-9: Attributes for recent performance

For this experiment we only use Logistic Regression and k-Nearest Neighbors. Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable whereas k-NN is a type of instance based learning [13] where the function is only approximated locally and all computation is deferred until function evaluation [14]. We visualize the accuracy graph for better understanding of the results. The graphs can be seen in Fig10 and Fig11.

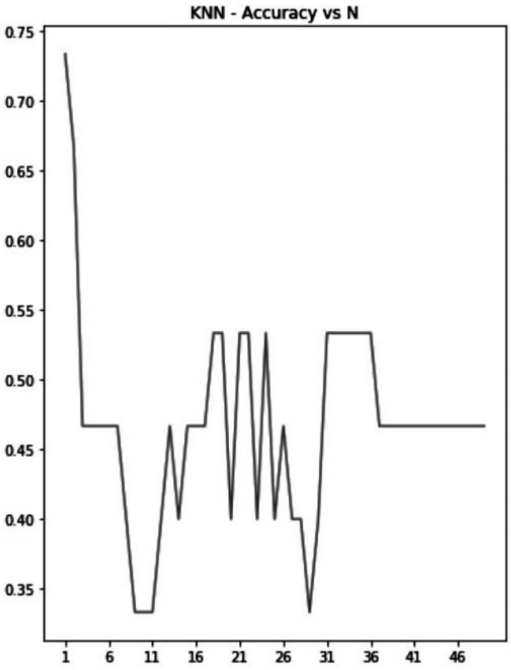


Fig-10: K-Nearest Neighbor

The accuracy for the two models can be seen in Fig12

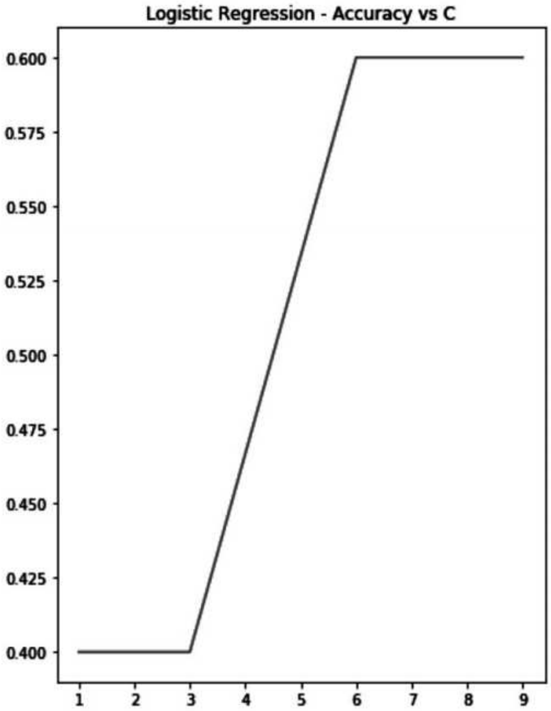
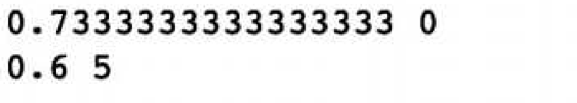


Fig-11: Logistic Regression

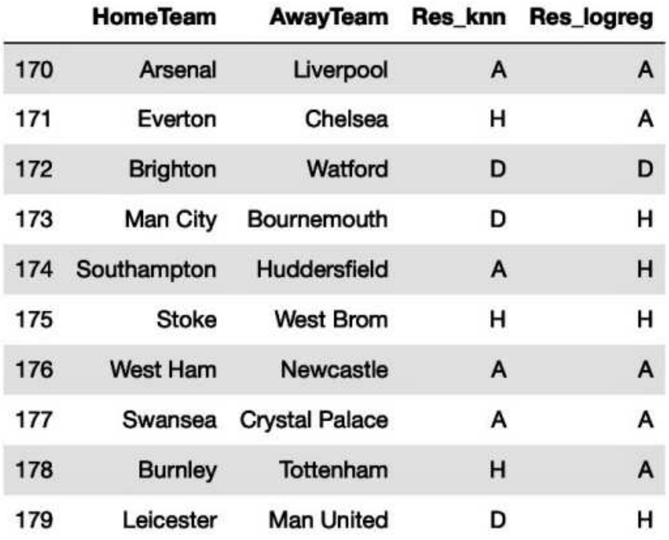


```
0.7333333333333333 0
0.6 5
```

Fig-12: 1.KNN 2.Logistic Regression

4. RESULT ANALYSIS

From our experiments we found out that Linear SVC, Random Forest Classifier and Naive Bayes doesn't give us good results. We then use KNN and Logistic Regression to predict the results and compare them with each other. We have used various models throughout the experiment to find the algorithm which gives us the best accuracy and we can conclude that K-Nearest Neighbors is the best for predicting the outcomes.



	HomeTeam	AwayTeam	Res_knn	Res_logreg
170	Arsenal	Liverpool	A	A
171	Everton	Chelsea	H	A
172	Brighton	Watford	D	D
173	Man City	Bournemouth	D	H
174	Southampton	Huddersfield	A	H
175	Stoke	West Brom	H	H
176	West Ham	Newcastle	A	A
177	Swansea	Crystal Palace	A	A
178	Burnley	Tottenham	H	A
179	Leicester	Man United	D	H

Fig-13: Accuracy1.KNN 2Logistic Regression

5. CONCLUSION

In this research paper, we have built multiple machine learning models to predict 2017-18 English Premier League match results. We can conclude that some attributes are more important than others, but prediction cannot be done using only these attributes. Usage of significant attributes increases the accuracy for the result prediction. We have also proved that algorithms like Support Vector Machine, Random Forest and Multinomial Naive Bayes classifier aren't effective for football prediction. K-Nearest Neighbors gives us the best accuracy compared to all the different algorithms used throughout the research. We also concluded that addition of important attributes such as recent performances and home advantage improves the model substantially.

ACKNOWLEDGEMENT

We wish to express our gratitude towards our mentor, Prof. Seema Wazarkar and Prof. Sonali Deshpande who at every stage in this project, contributed their valuable input. We also thank our H.O.D Dr. Rajneesh Kaur Sachdeo Bedi, for providing the necessary facilities for the completion of this project work in our college.

REFERENCES

[1] Bailey, M.J. (2005). Predicting Sporting Outcomes: A Statistical Approach. Swinburne University of Technology: Faculty of Life and Social Sciences.

[2] aio,G., & Blangiardo,M.(2010). Bayesian Hierarchical Model for The Prediction of Football Results. Journal of Applied Statistics, 253-264

[3] Hosmer, D.W. Lemeshow, S. & Sturdivant, R.X. Applied Logistic Regression 3rd ed. Hoboken, New Jersey: John Wiley & Sons, Inc

[4] Igiri, C.P., & Nwachukwu, E.O.(2014). An Improved Prediction System for Football a Match Result. IOSR Journal of Engineering Volume 04 Issue 12, pp12-20

[5] Min, B., et al. (2008). A Compound Framework for Sports Result Prediction: A Football Case Study. Journal of Knowledge-Based System Volume 21 Issue 7, 551-562. The Netherlands: Elsevier Science Publishers

[6] Peng,etal.(2002).An Introduction to Logistic Regression Analysis and Reporting. Indiana University-Bloomington: EBSCO Publishing.

[7] Reddy, V., & Movva, Sai V. K. (2014). The Soccer Oracle: Predicting Soccer Game Outcomes Using SAS® Enterprise Miner TM. SAS® GLOBAL FORUM. Washington, D.C.

[8] Shin, J., & Gasparyan, R. (2014). A Novel Way to Soccer Match Prediction. Stanford University: Department of Computer Science.

[9] Snyder, Jeffrey A.L. (2013). What Actually Wins Soccer Matches: Prediction of the 2011-2012 Premier League for Fun and Profit. Thesis, University of Washington, WA: Department of Computer Science.

[10] https://en.wikipedia.org/wiki/Multiclass classification

[11] https://en.wikipedia.org/wiki/Random forest

[12] https://en.wikipedia.org/wiki/Naive Bayes classifier

[13] https://en.wikipedia.org/wiki/Logistic

[14] https://en.wikipedia.org/wiki/K-nearest neighbor's algorithm

BIOGRAPHIES

Ishan Jawade is a Computer Enthusiast, currently pursuing B.Tech in Computer Science and Engineering at MIT School Of Engineering MIT ADT University, Pune. He worked on projects in various domains like Web Development and Machine Learning.

Rushikesh Jadhav is a computer science and engineering student from MIT ADT University. He is deeply passionate and interested in Armed Forces. He wants to make remarkable contributions to our Armed Forces with is knowledge in the fields of computer science and artificial intelligence.