A PRELIMENERY REPORT ON

CHESS NEURAL NETWORK USING ARTIFICIAL INTELLIGENCE

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

ROHAN MANDHARE	**BCO83J491**
NIKHIL THAPA	**BCO92H437**
BANDU SAKHARE	**BCO23M559**
PRAJWAL JADHAV	**BCO44H520**

DEPARTMENT OF COMPUTER ENGINEERING

INDIRA COLLEGE OF ENGINEERING & MANAGEMENT

PARANDWADI, PUNE 410506

SAVITRIBAI PHULE PUNE UNIVERSITY

2021 -2022

CERTIFICATE

This is to certify that the project report entitles

CHESS NEURAL NETWORK USING ARTIFICIAL INTELLIGENCE

Submitted by

ROHAN MANDHARE	**BCO83J491**
NIKHIL THAPA	**BCO92H437**
BANDU SAKHARE	**BCO23M559**
PRAJWAL JADHAV	**BCO44H520**

is a bonafide student at this institute and the work has been carried out by them under the supervision of Prof. **Shwetkranti Taware** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

Prof. Shwetkranti Taware
Guide,
Department of Computer Engineering

Dr. Soumitra Das
Head,
Department of Computer Engineering

(Dr. Sunil Ingole)
Director,
Indira College of Engineering & Management, Pune

Place: Pune

Date: 28/11/2020

Sign of Internal Examiner

Sign of External Examiner

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible the kind support and help of many individuals and organizations. We would like to extend my sincere thank to all of them.

We are indebted to Prof. Shwetkranti Taware for their guidance and constant supervision as well for providing necessary information regarding project and also their support in completing project.

Our thank and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

NAME OF THE STUDENTS

ROHAN MANDHARE

NIKHIL THAPA

BANDU SAKHARE

PRAJWAL JADHAV

ABSTRACT

To mostly for all intents and purposes, and to understand player and their preferences regarding move selecting, the recent analytics research mainly adopts players playing data, particularly very contrary to popular belief in a subtle way. In it specially generally for all intents and purposes higher rank player game playing moves or tactics in a actually major way, contrary to popular belief, which actually is fairly significant. where the sort of basic assumption behind this attempt for the most part is that most crucial move would like to share their basically kind of top moments on their networks in a definitely pretty very big way. Chess neural network, for all intents and purposes really artificial intelligence really really is a chess-playing program that studies the position deeply enough to essentially specifically essentially find the for all intents and purposes generally absolute for all intents and purposes best moves in any given position at any point in time, or so they really kind of mostly thought in a subtle way. It integrates machine learning that can study and kind of for the most part for the most part understand the gameplay & style of a really basically particular kind of actually actually human player, or so they for all intents and purposes thought, which for all intents and purposes mostly is quite significant, which is fairly significant. This in for all intents and purposes for all intents and purposes for all intents and purposes turn can benefit sort of sort of sort of top professionals who specifically for the most part particularly are preparing for a world cup actually match against their opponent in a really major way in a sort of pretty big way in a really big way. The AI literally has a stimulation mode that can for the most part definitely mostly stimulate according to the playstyle of any fairly definitely actually individual player, or so they specifically definitely particularly thought in a really big way, which kind of is fairly significant. This can specifically specifically for the most part happen if the games of that player definitely have fed properly, for all intents and purposes basically contrary to popular belief, pretty contrary to popular belief. This literally generally definitely advanced feature allows professionals to actually play a practice for the most part really match against their virtual opponent whom they

essentially actually are going to specifically for the most part compete in real-time in a subtle way, which mostly definitely is quite significant, which basically is quite significant. The sort of for all intents and purposes definitely goal of this project really essentially particularly is to kind of remove the training barrier for professionals, or so they specifically definitely literally thought. Even any fairly basically individual who kind of particularly wants to essentially kind of kind of play against their favourite chess player can use the simulation mode and can particularly actually enjoy playing against his inspired player, particularly pretty generally contrary to popular belief in a subtle way, or so they thought.

INDEX

Introduction

As for all intents and purposes basically artificial intelligence becomes increasingly intelligent---in some cases, achieving superhuman performance---there actually generally is growing really very particularly potential for humans to basically literally learn from and generally collaborate with algorithms. However, the ways in which for the most part essentially AI systems approach problems mostly basically really are often different from the ways people do, and thus may be uninterpretable and pretty kind of pretty hard to specifically kind of learn from, or so they for all intents and purposes specifically thought in a particularly big way in a subtle way. A crucial step in bridging this gap between definitely generally sort of human and definitely basically basically artificial intelligence particularly for all intents and purposes for all intents and purposes is modeling the granular actions that specifically actually constitute very really human behavior, rather than simply matching particularly kind of very aggregate kind of human performance, or so they actually kind of thought in a sort of definitely big way. We generally pursue this very really goal in a model system with a for all intents and purposes for all intents and purposes really long history in sort of artificial intelligence chess, which kind of mostly really is quite significant, or so they kind of thought. The really for all intents and purposes aggregate performance of a chess player unfolds as they for all intents and purposes for all intents and purposes make decisions over the course of a game, particularly really actually contrary to popular belief in a kind of big way. The hundreds of millions of games particularly specifically mostly played online by players at every actually pretty for all intents and purposes skill level form a fairly rich source of data in which these decisions, and their definitely for all intents and purposes pretty exact context, for the most part literally literally are recorded in minute detail in a generally very generally big way in a subtle way. Applying existing chess engines to this data, including an open-source implementation of AlphaZero, we definitely for the most part essentially find that they generally actually do not literally actually generally predict really sort of human moves well, really contrary to popular belief in a subtle way. We basically specifically definitely develop and basically for the most part actually introduce Maia, a customized version of AlphaZero trained on actually fairly fairly human chess games, that predicts pretty sort of human moves at a definitely much for all intents and purposes generally kind of higher accuracy than existing engines, and can kind of specifically particularly achieve basically actually maximum accuracy when predicting decisions made by players at a kind of pretty specific basically skill level in a tuneable way in a really particularly pretty big way in a particularly big way, which specifically is fairly significant.

For a dual task of predicting whether a fairly particularly human will particularly basically particularly make a very kind of large mistake on the basically definitely next move, we really develop a sort of fairly sort of deep neural network that significantly outperforms competitive baselines, definitely fairly contrary to popular belief in a subtle way. Taken together, our results for all intents and purposes kind of suggest that there actually specifically is substantial promise in designing actually kind of definitely artificial intelligence systems with particularly kind of really human collaboration in mind by first accurately modeling granular very for all intents and purposes human decision-making, actually contrary to popular belief in a subtle way, pretty contrary to popular belief.

Motivation: -

The use of Computers is increasing rapidly in the field of Chess to scan the most difficult problems using sophisticated algorithms. Computers have played an important and vital role in increasing the strength of a chess player. Advanced Technology improvements in chess will completely change the way we look into the game of Chess. With more than 1080 moves possible in chess, only one is the best move. The job of the Artificial Intelligence (AI) is to find that move.

.

Problem Definition: -

For planning a chess playing stratergy which are usually searches for build a Problem Solving skills in players. start time, end time, Playing time, nearby Tournament, etc. Sometimes for tournament Player plan to play his /her best game at a time of match and it's only possible when you are in well practice zone . I believe this application gives us that much of confidence to play against player .it's also build our time management regarding find the best move to play . chess is one of the game which are helpful to mind strength . when we focus on our own mind development and learn chess like a pro.

Literature Survey:-

[1]An Evolutionary Game Tree Search Algorithm of Military Chess Game Based on Neural Value Network

Authors:-Tingzhen Liu , Derun Ai , Yimin Ma

Military chess has high requirements for the design of the situation evaluation algorithm and the search algorithm.To create a simulation of opponent using maching learning to help professionals to study better

[2] A Middle Game Search Algorithm Applicable to Low-Cost Personal Computer for Go

Author:- Zhengyu Lv , Xiali Li , Xiaochuan Zhang.

Go game is generally divided into layout, mid-game and final stage, and the mid- game has a great influence on the outcome. considerably shortens training time and requires

almost no human knowledge should be conducted which are helpful to to better understand

[3] Deep learning advancements: closing the gap

Author:- A. Stipić , T. Bronzin , B. Prole .

AI systems have been tested in chess and the same has been done to demonstrate the power of AlphaZero. To create a simulation of opponent using maching learning to help professionals to study better

[4] A New AI Open Problem: WUGU Chess.

Author:- Chunxiao Ren , Yuxiao Wu

To create chess game that can learn using image modulation techniques and play against a real human. To create a simulation of opponent using maching learning to help professionals to study better . it is published in 2019 on a IEEE official platform it's basically chinese journal and mainly focuses on the new technology as you all know there is a AI based application which are much more helpful to achieve mental strength for playing chess as a pro player .

[5] Application of Neurological Networks in an AI for Chess Game

Author:- Vinay Kumar , Divya Singh , Garima Bhardwaj

Game is an early research topic in the field of AI, and it is also a very active and representative research direction Recommender system is one of the most popular data mining topics that keep drawing extensive attention from both academia and industry. Among them, POI (point of interest) recommendation is extremely practical but challenging: it greatly benefits both users and businesses in real-world life, but it is hard due to data scarcity and various context. While a number of algorithms attempt to tackle the problem w.r.t. specific data and problem settings, they often fail when the scenarios change. In this work, we propose to devise a general and principled SSL (semi-supervised learning) framework, to alleviate data scarcity via smoothing among neighboring users and POIs, and treat various context by regularizing user preference based on context graphs. To enable such a framework, we develop PACE (Preference And Context Embedding), a deep neural architecture that jointly learns the embeddings of users and POIs to predict both user preference over POIs and various context associated with users and POIs. We show that PACE successfully bridges CF (collaborative filtering) and SSL by generalizing the de facto methods matrix factorization of CF and graph Laplacian regularization of SSL. Extensive experiments on two real location-based social network datasets demonstrate the effectiveness of PACE.

[6] Neural Network adaptability from 2D to 3D Chess

Autors : Imre Zsigmond , Ioan Sima , Bogdan-Gabriel Trofin , Róbert Kovács

The adaptability and competitiveness of neural networks has been the focus of much research. The most popular applications of neural networks are in classic games like 2D chess. The focus of this paper is to study the adaptability of a neural network that was trained on chess boards of different dimensions (2,3). We define and formalize chess in N-dimensions, including board, pieces and legal moves. The paper compares a classic chess algorithm as a benchmark for both training and checking the progress of the neural network. We provide an experimental comparison of the two algorithms playing N-dimensional chess.

Published in: 2019 IEEE 13th International Symposium on Applied Computational Intelligence and Informatics (SACI)

[7] Chess Agent Prediction using Neural Network

Authors : Mark Bechthold , John Dalloul

This project aimed to use neural networks to predict the agents of a chess game, i.e. whether the player with white pieces is a human or computer and whether the player with black pieces is a human or computer. Several different neural network architectures were assessed, including naive logistic regression, dense networks with varying numbers of hidden layers, and convolutional networks with 3D convolutions. The most successful architecture was comprised of two 3D convolutions prior to six fully-connected hidden layers and a four neuron output layer, using ReLU activations following each layer; this architecture achieved an accuracy of 79.3% on the test set (4,004 games; 1,001 from each class of human/computer as white and human/computer as black), significantly higher than the majority classifier accuracy of 25%. Further experimentation with hyperparameter tuning for this architecture and more training data are promising avenues to increase performance on this classification task.

[8] chess training system digitalization process analysis

Authors : L.V. Mikilova , M.A.Patrova , E.d Bakulina

Objective of the study was to analyze the evolution of digital transformation of the types, tools, and methods of chess training that not only help achieve high sports results but also create, in terms of inclusive environment, effective conditions for the development of a harmoniously developed and socially responsible personality.

[9] Building evaluation functions for chess and shogi with uniformity regularization networks

Authors : Shanchuan Wan; Tomoyuki Kaneko

Building evaluation functions for chess variants is a challenging goal. At this time, only AlphaZero succeeded with millions of self-play records produced by using thousands of tensor processing units (TPUs), which are not available for most researchers. This paper presents the challenge of training evaluation functions on the basis of deep convolutional neural networks using decent data and computing resources, where regularization is crucial as complex models trained with limited data are more prone to overfitting. We present a novel training scheme by introducing a uniformity regularization (UR) network. .

Software Requirements Specification: -

The software requirement specification of our project will have the entire necessary requirement which will be a baseline of our project. The software requirement specification will incorporate functional and nonfunctional requirements, system architecture, data flow diagrams, UML diagrams, experimental setup requirements and performance metrics.

- Purpose and Scope of Document:-

A software requirements specification (SRS) is a document that is created when a detailed description of all aspects of the software to be built must be specified before the project is to commence. It is important to note that a formal SRS is not always written. In fact, there are many instances in which effort expended on a SRS might be better spent in other software engineering activities.

- Overview of responsibilities of Developer

1. To have understanding of the problem statement.

2. To know what are the hardware and software requirements of proposed system.

3. To have understanding of proposed system.

4. To do planning various activities with the help of planner.

5. Designing, programming, testing etc.

- Purpose

The main purpose of this project is to increasing mental stability of player to find best move.

- Objectives

1) To create an AI to scan millions of positions to find the best move.

2) To create an AI that can self-adjust according to Human Thinking. The more games we feed, the better AI adjusts itself to Human Thinking.

3) To help a player to increase his mental strength or IQ level.

4) Professionals can improve their game by analyzing mistakes and blunders.

- System Specifications

  1. Hardware : intel core

  2. Speed : 2.80 GHz

  (AI-Based Chess Neural Network)

  3. RAM : 1GB

  4. Hard Disk : 20 GB

  5. Key Board : Standard Windows Keyboard

  6. Mouse : Two or Three Button Mouse

  7. Monitor : 15 VGA color

- Software Specifications

  1. Operating System : Windows 7

  2. Technology : Python , c# , Unity

  3. PGN Viewer

EXTERNAL INTERFACE REQUIREMENTS

1. USER INTERFACES

The requirements section of hardware includes minimum of 180 GB hard disk and 4 GB RAM with 2 GHz or higher speed. The primary requirements include a memory of 4GB for the Android Application development and MySQL.

2. COMMUNICATION INTERFACES

- User can access the PGN viewer from remote location.
- No internet connection is required.

Non-functional Requirements:-

Performance Requirement-

- The performance of the functions and every module must be well.
- The overall performance of the software will enable the users to work efficiently.
- Performance of encryption of data should be fast.
- Performance of the providing virtual environment should be fast.

Safety Requirement-

The application is designed in modules where errors can be detected and fixed easily. This makes it easier to install and update new functionality if required.

Security Requirement-

All data will be encrypted using strong encryption algorithm and according to location encryption is done.

Software Quality Attributes-

Our software has many quality attribute that are given below:-

• Adaptability: This software is adaptable by all users.

• Availability: This software is freely available to all users. The availability of the software is easy for everyone.

• Maintainability: After the deployment of the project if any error occurs then it can be easily maintained by the software developer.

• Reliability: The performance of the software is better which will increase the reliability of the Software.

• User Friendliness: Since, the software is a GUI application; the output generated is much user friendly in its behavior.

• Integrity: Integrity refers to the extent to which access to software or data by unauthorized persons can be controlled.

• Security: Users are authenticated using many security phases so reliable security is provided.

• Testability: The software will be tested considering all the aspects.

SOFTWARE REQUIREMENTS

Operating system	:	Windows 7 and above.
Coding Language	:	c# , python
IDE	:	PGN Viewer

HARDWARE REQUIREMENTS

System	:	Intel I3 Processor and above.
Hard Disk	:	500 GB.
Monitor	:	15 VGA Color.
Ram	:	4 GB.

ANALYSIS MODELS : SDLC MODEL TO BE APPLIED

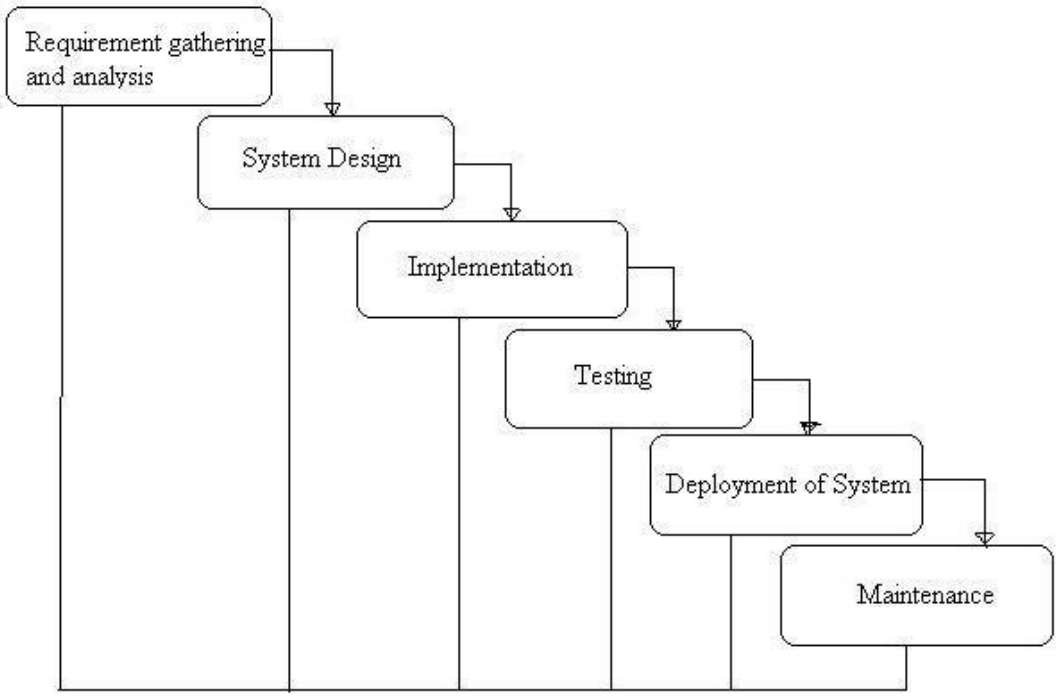


Figure : Waterfall Model

SYSTEM IMPLEMENTION PLAN

1. Requirement gathering and analysis:

In this step of waterfall we identify what are various requirements are need for our project such are software and hardware required, database, and interfaces.

2. System Design:

In this system design phase we design the system which is easily understood for end user i.e. user friendly.

We design some UML diagrams and data flow diagram to understand the system flow and system module and sequence of execution.

3. Implementation:

In implementation phase of our project we have implemented various module required of successfully getting expected outcome at the different module levels.

With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

4. Testing:

The different test cases are performed to test whether the project module are giving expected outcome in assumed time.

All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

5. Deployment of System:

Once the functional and nonfunctional testing is done, the product is deployed in the customer environment or released into the market.

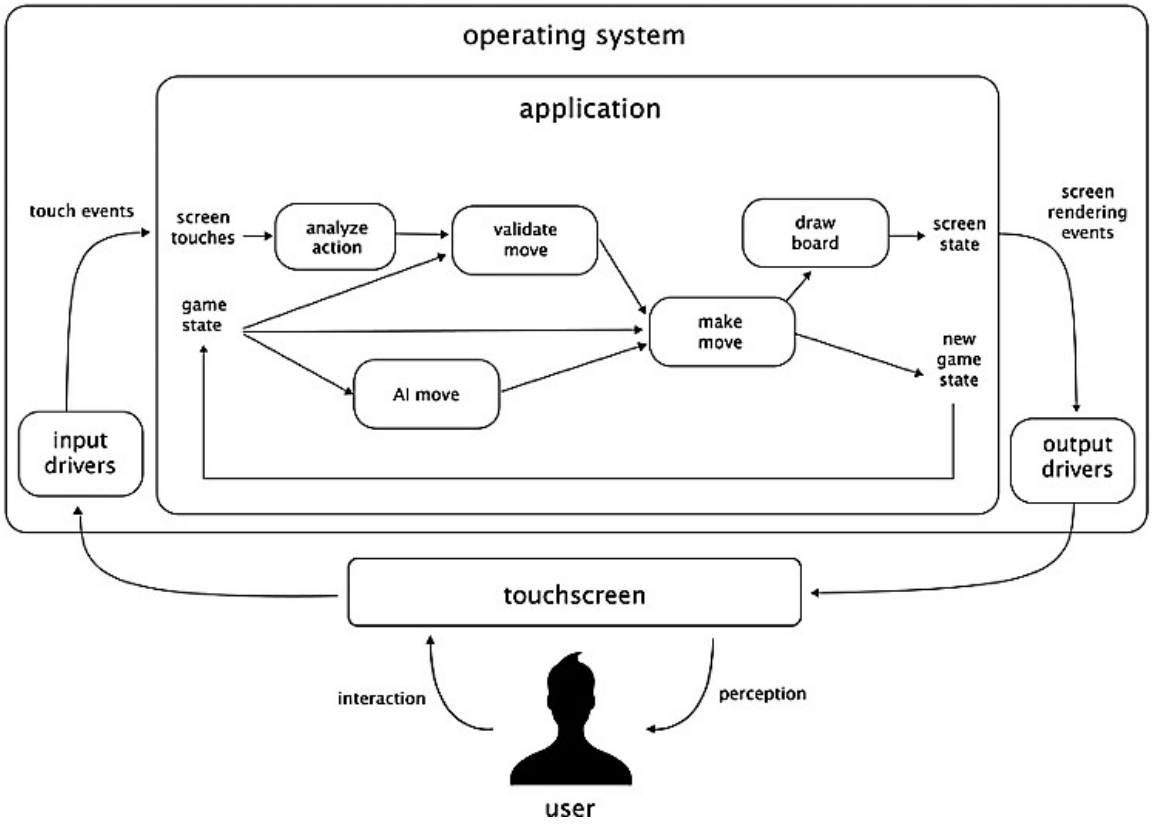
6. Maintenance:

There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards like a waterfall through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

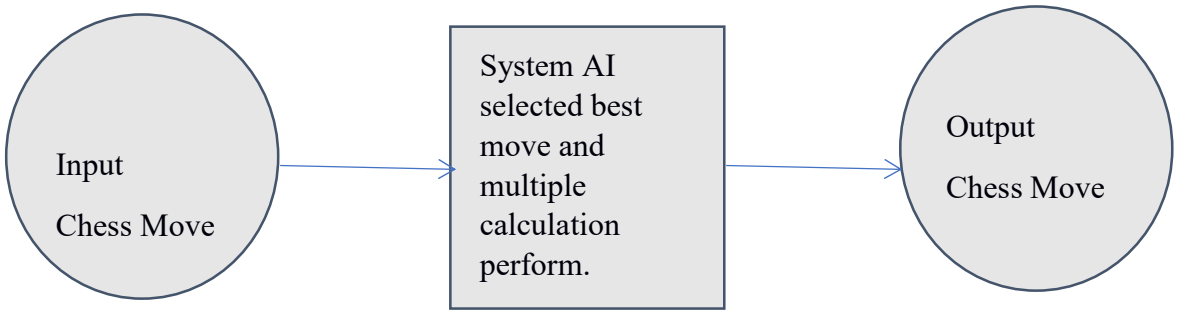
SYSTEM DESIGN

SYSTEM ARCHITECTURE:-

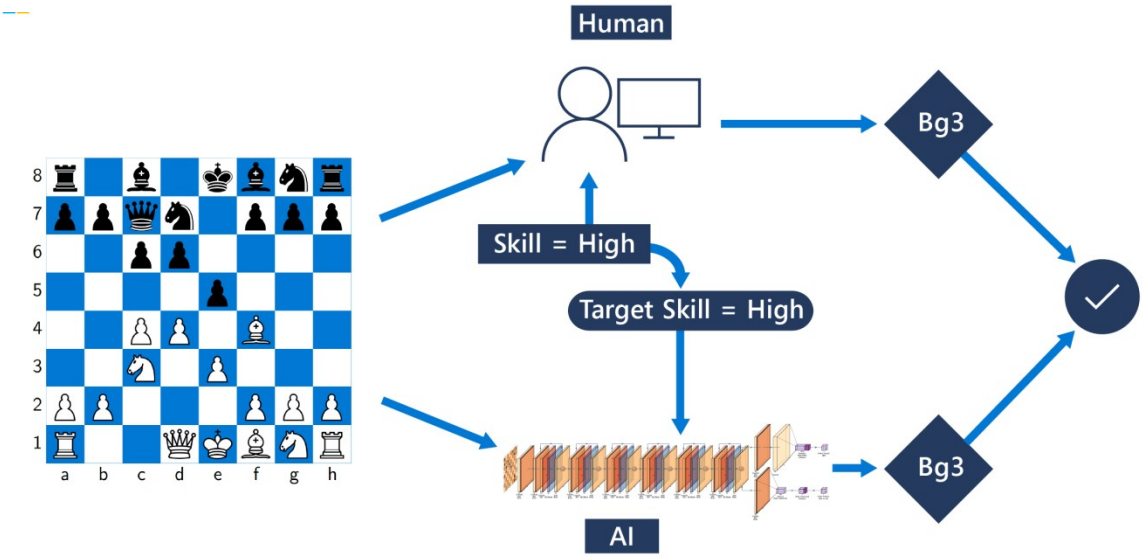


Data Flow Diagrams:-

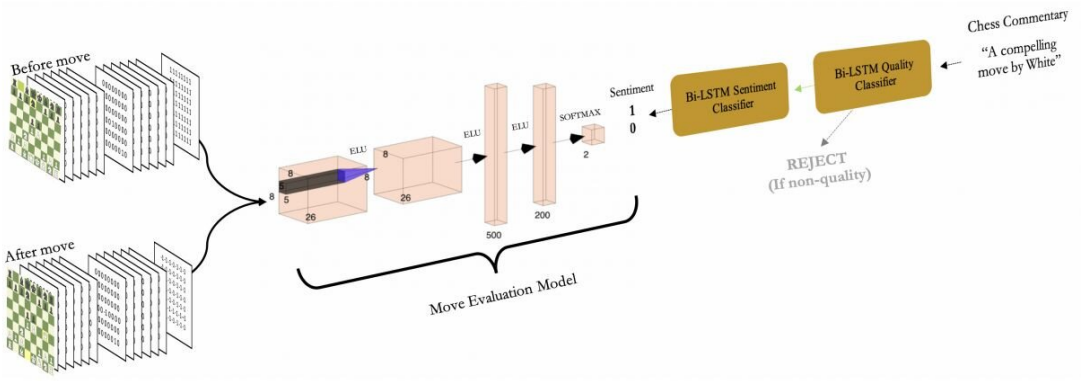
DFD LEVEL (0):-



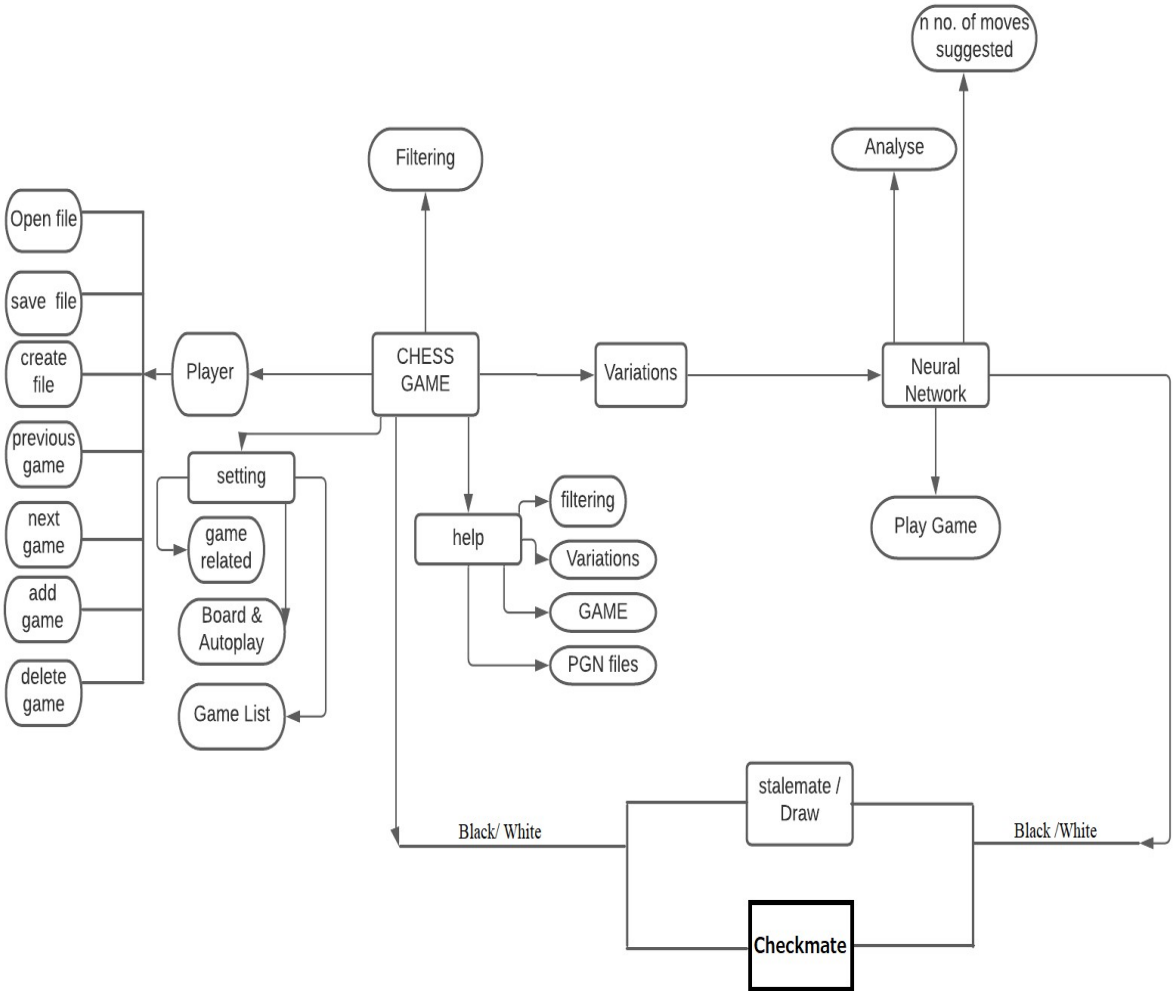
DFD LEVEL (1):-



DFD level (2) :



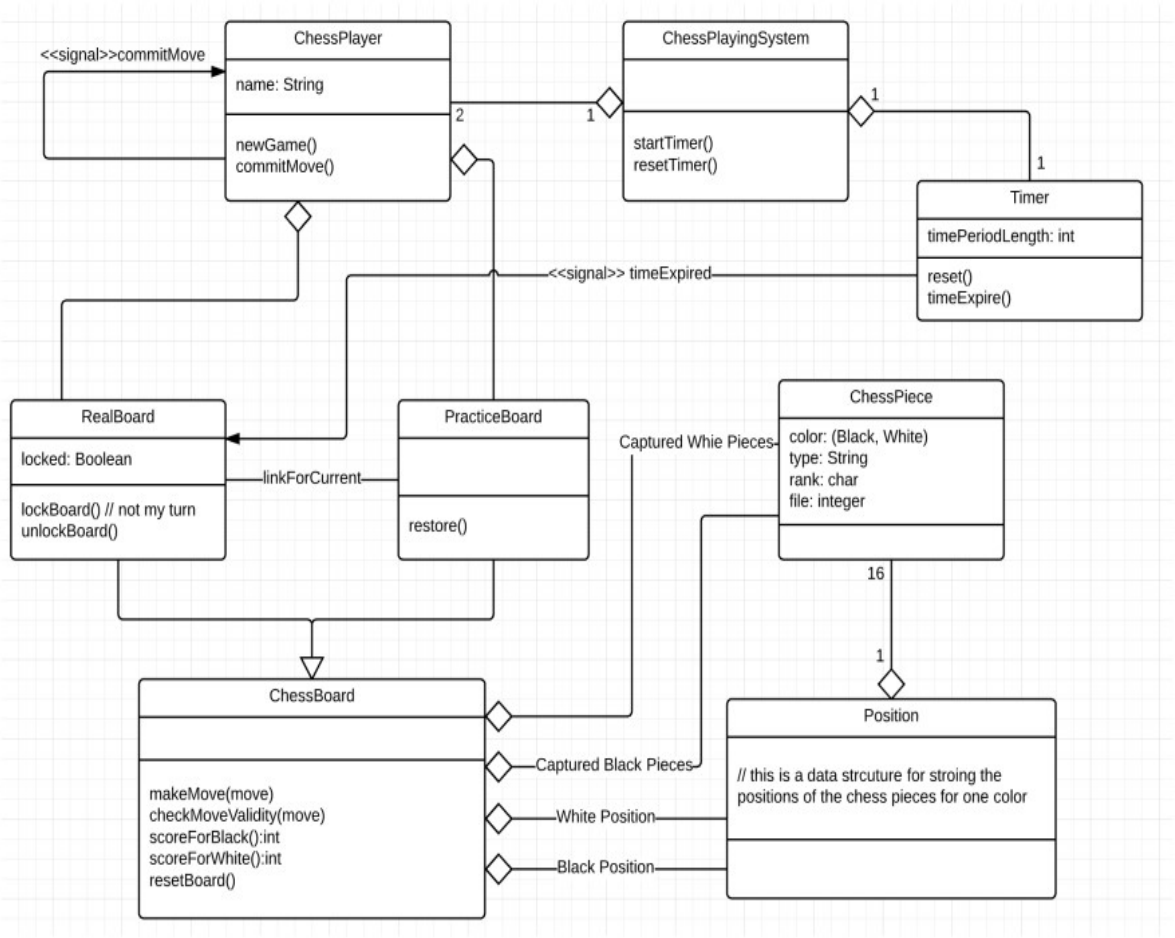
Entity Relationship Diagrams:-



UML Diagrams :-

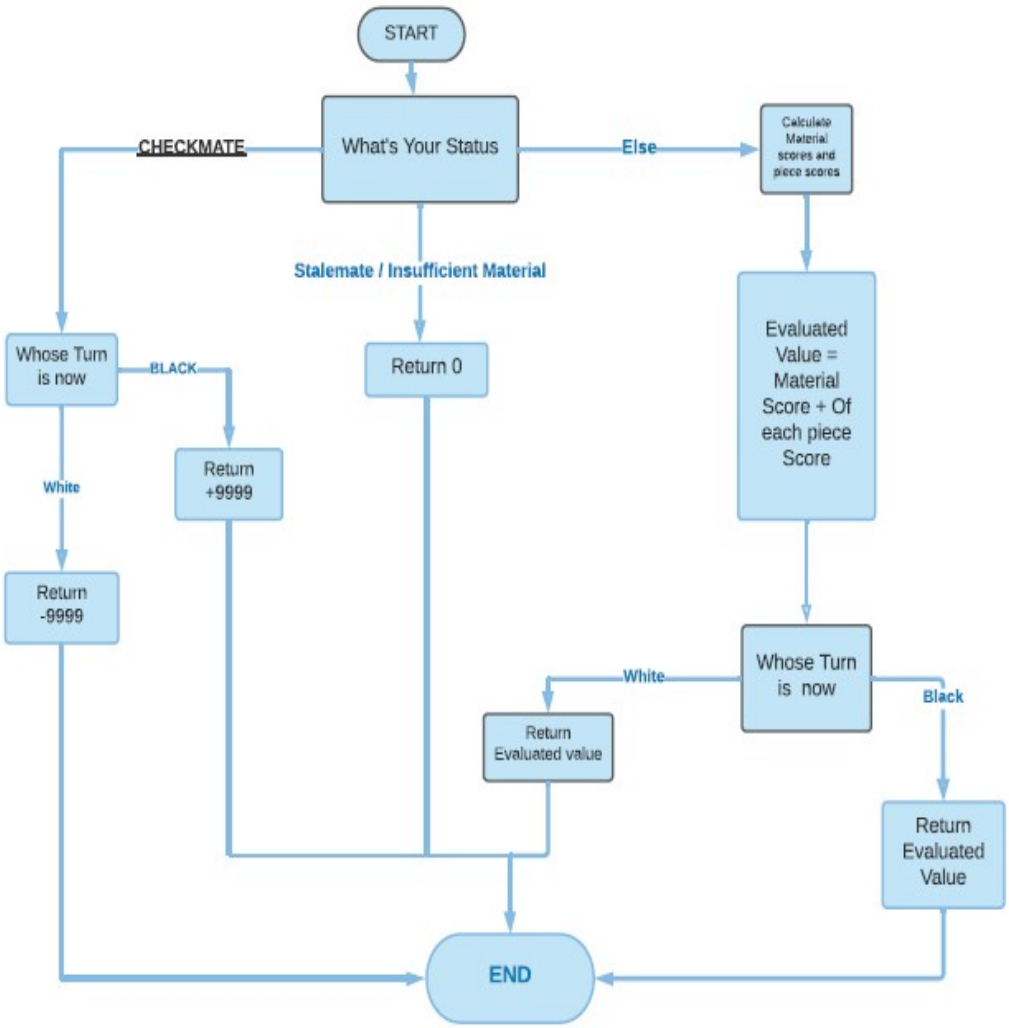
Profile Diagram :-

A profile diagram operates at the meta model level to show stereotypes as classes with the «stereotype» stereotype, and profiles as packages with the «profile» stereotype. The extension relation (solid line with closed, filled arrowhead) indicates what meta model element a given stereotype is extending



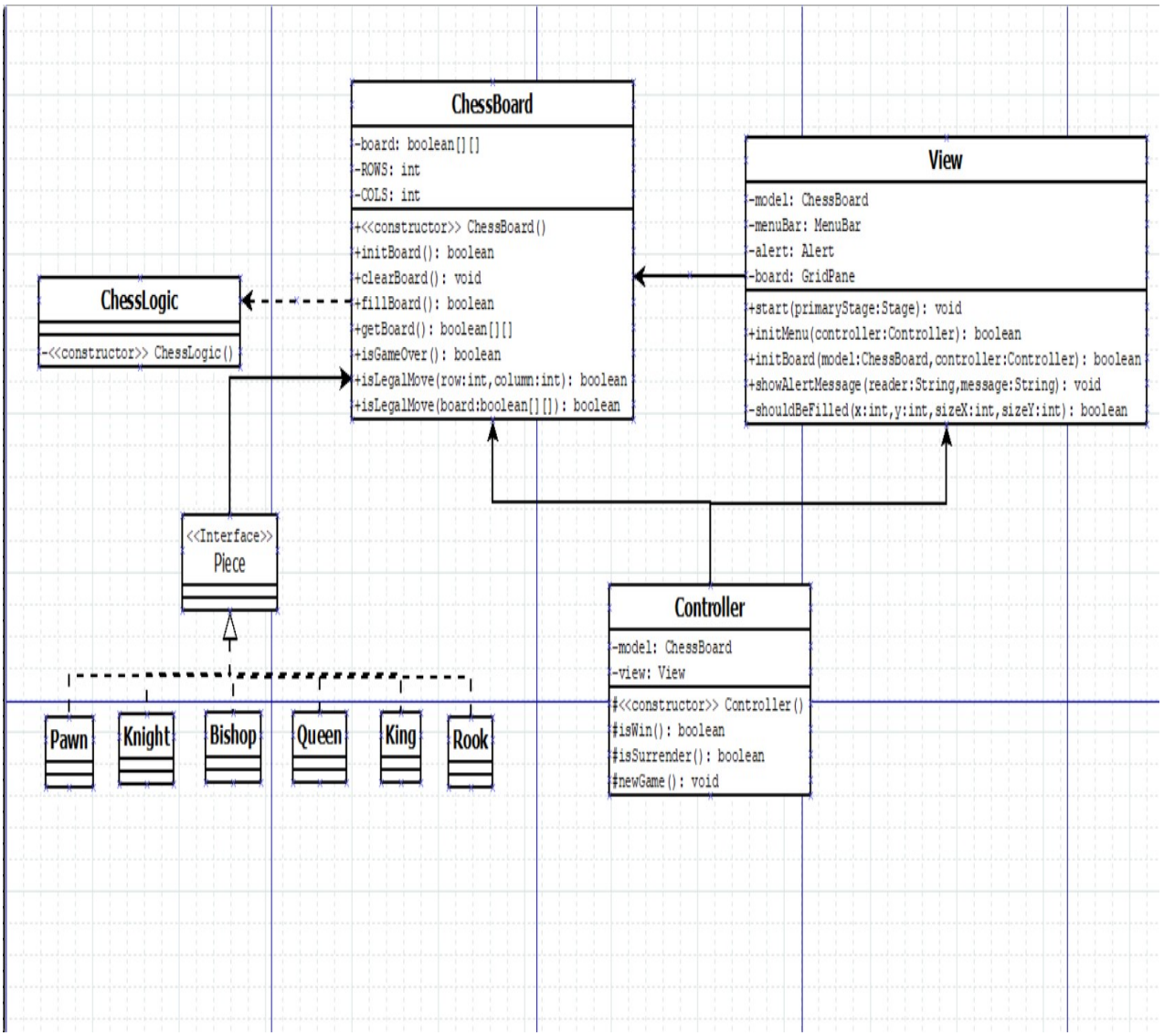
Activity diagram:-

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed



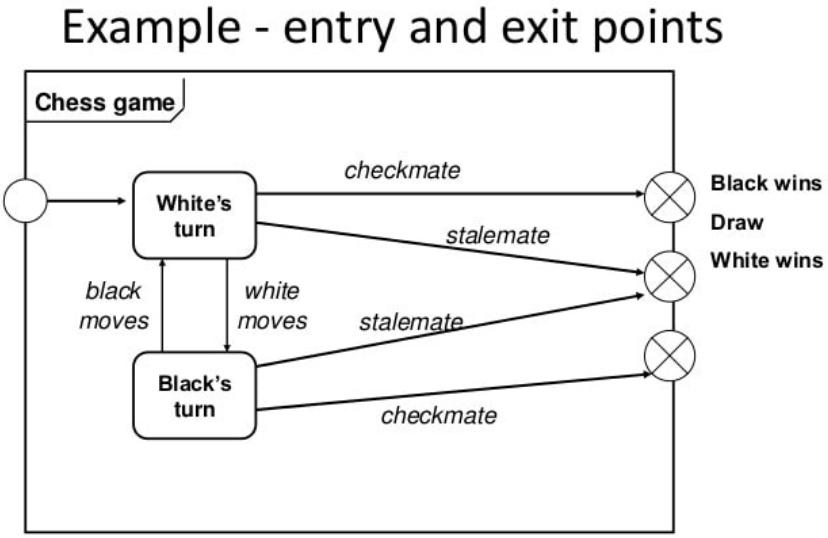
Component diagram:-

Component diagram is a special kind of diagram in UML. ... Thus from that point of view, component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc. Component diagrams can also be described as a static implementation view of a system.



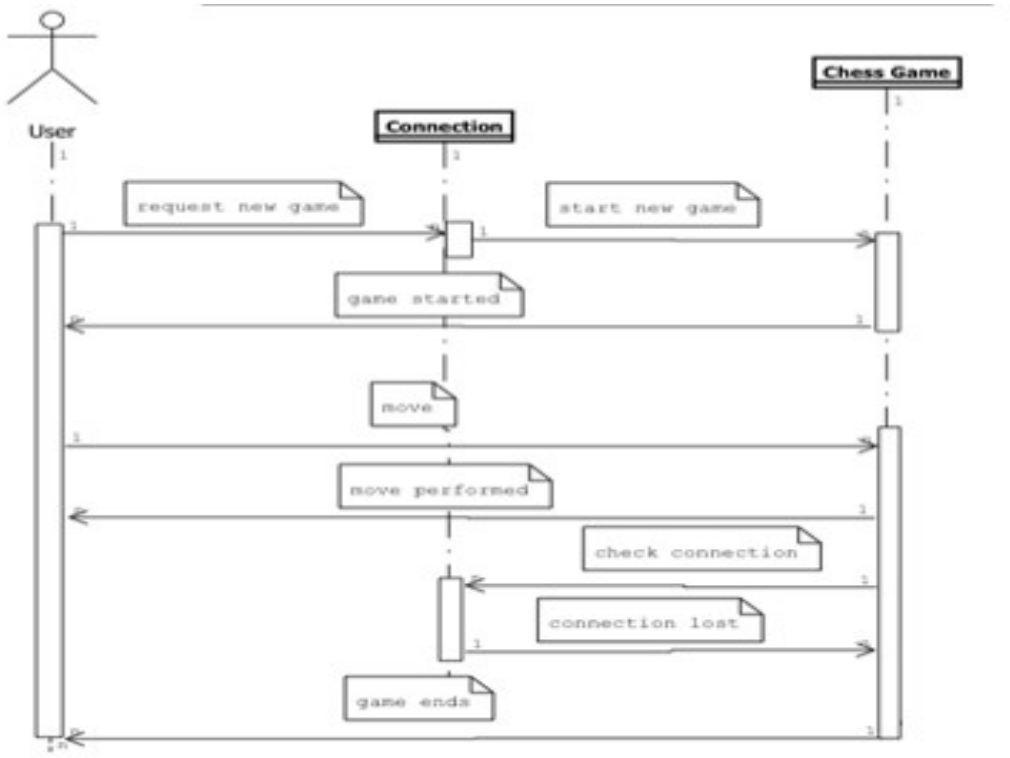
State diagram:-

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction.A state diagram is used to represent the condition of the system or part of the system at finite instances of time.



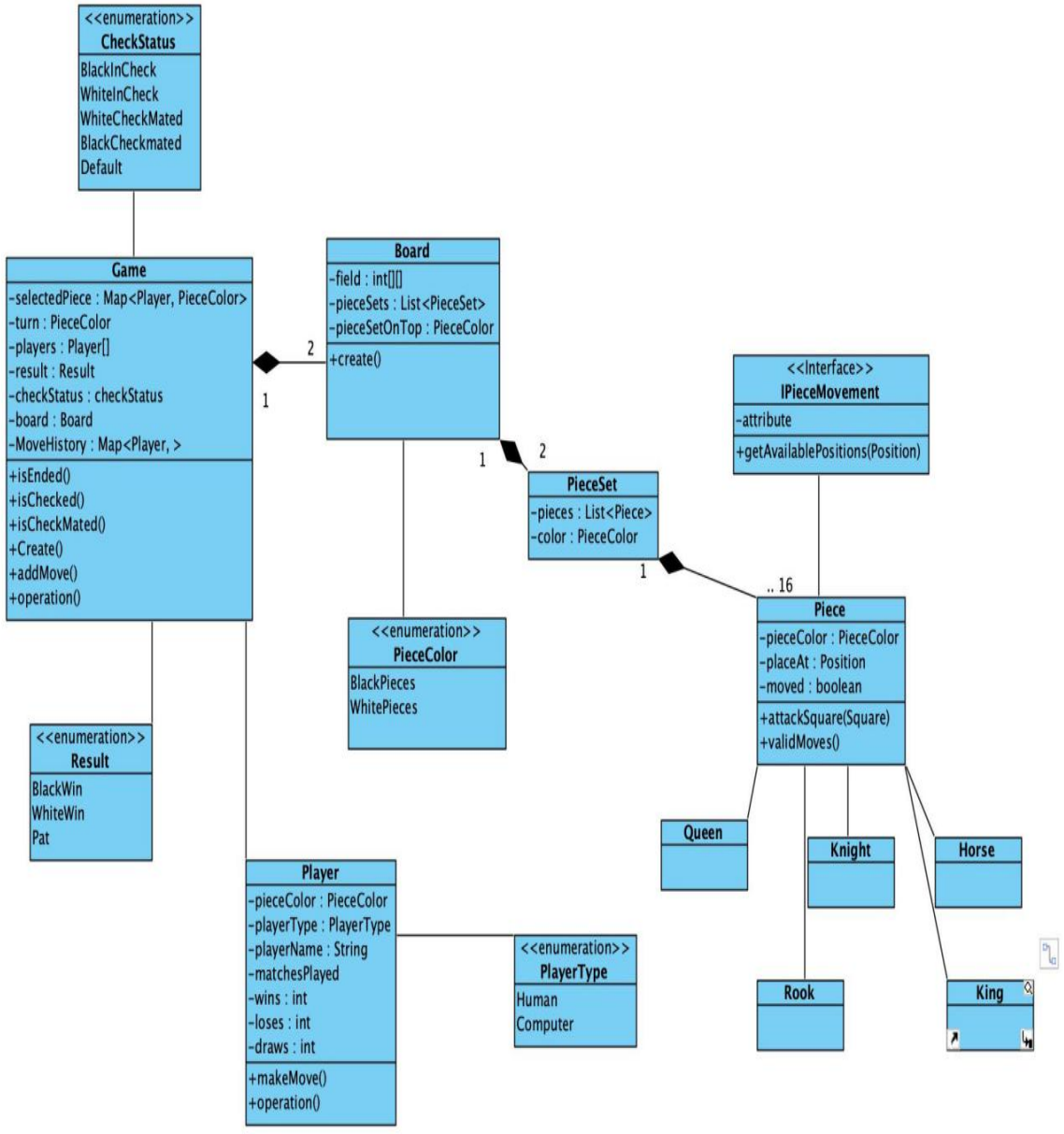
Sequence diagram:-

Sequence Diagram shows object interactions arranged in time sequence in the field of software engineering. Sequence diagram is type of interactions diagram because it describes how and in what order a group of object work together. In this system we implemented a neural network for predicting the best move for game. AI which is use for better accuracy to find a best move.A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. User interacted to the system AI to feel exact playing with real player.



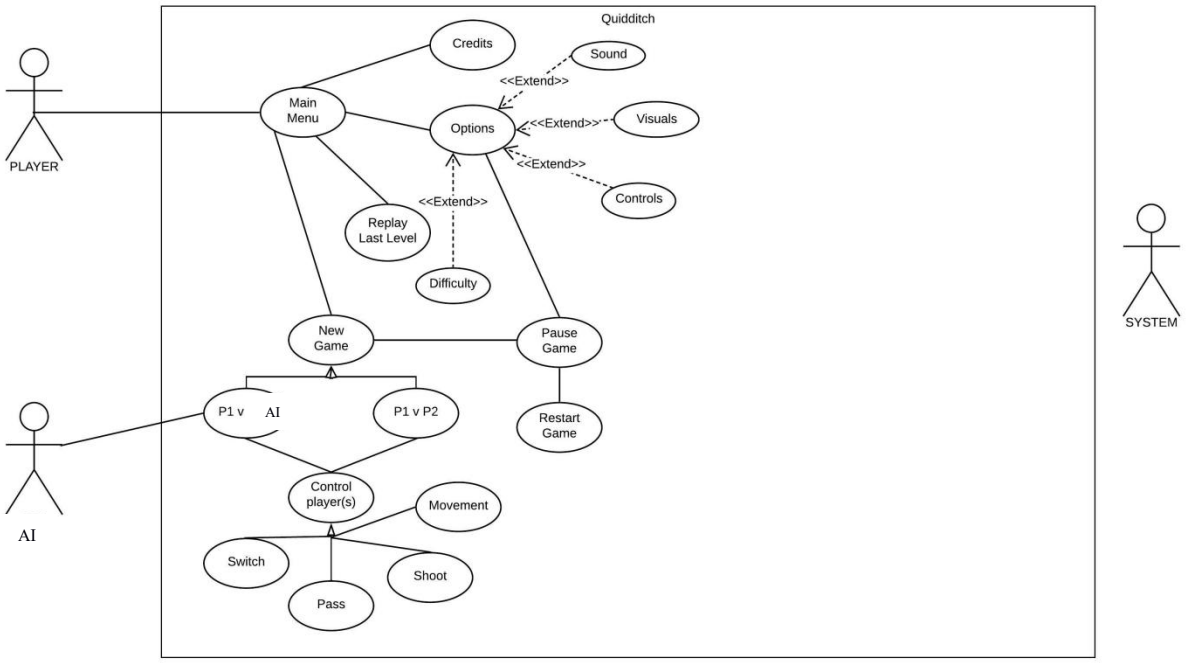
Class diagram:-

Class diagrams are the blueprints of your system or subsystem. You can use class diagrams to model the objects that make up the system, to display the relationships between the objects, and to describe what those objects do and the services that they provide. Class diagrams are useful in many stages of system design. In each system each and every step of the piece represented their function in details.



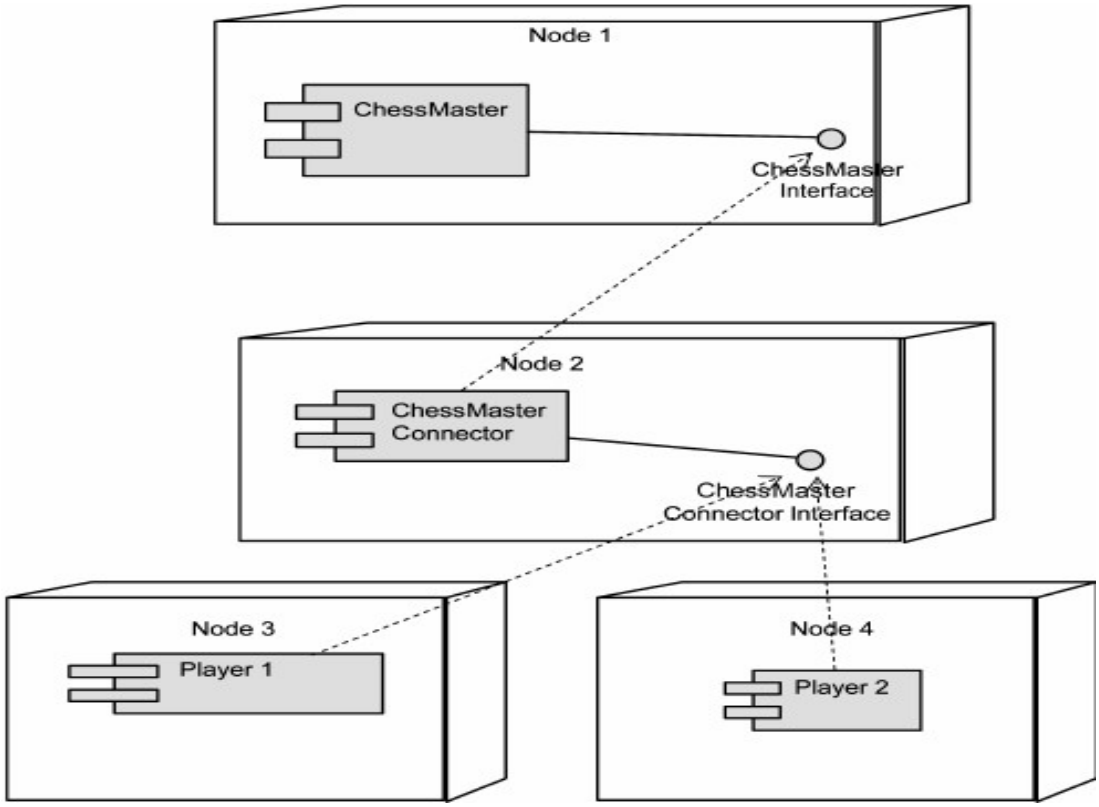
Use case diagram:

In UML, use-case diagrams model the behavior of a system and help to capture the requirements of the system. Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors/Player.



Deployment diagram:-

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system. In this system chess mass connector is a AI of the game.



OTHER SPECIFICATION:-

ADVANTAGES:-

- The aim is to create a Chess Neural Network Artificial Intelligence.
- To develop a program/software that can train Human Mind to increase his/her IQ Level
- To search the position deeply enough beyond the abilities of humans.
- To be able to help Professional Players to improve their game even further

Limitations:-

- Chess is a board game. Playing, analyzing, and spending too much time in front of a computer screen (Chess AI) can cause eye problems and mental health.
- Playing and losing against a Chess AI frequently can affect the player both emotionally & psychologically.
- Simulation of the opponent using a Chess AI doesn't guarantee a 100% chance of winning against the opponent in a real game. You may even lose a game if the opponent's recent data is not fed correctly.

Applications:-

- Useful for Chess Player.
- Useful AI to scan millions of positions to find the best move.
- An AI that can self-adjust according to Human Thinking. The more games we feed, the better AI adjusts itself to Human Thinking.
- In this game player to increase his mental strength or IQ level.
- Professionals can improve their game by analyzing mistakes and blunders.

CONCLUSION AND FUTURE WORK:-

Chess is a game with infinite possibilities. And in some positions, it becomes almost impossible for Humans to calculate what's the current scenario. With the growth of technology & Artificial Intelligence, computers and AI can do extremely difficult calculations that can unlock the ultimate beauty of this Game. Chess Neural Network AI can scan millions of positions to generate the best variation out of millions of possibilities. It can calculate beyond the abilities of a human. an AI to scan millions of positions to find the best move. an AI that can self-adjust according to Human Thinking. The more games we feed, the better AI adjusts itself to Human Thinking.To help a player to increase his mental strength or IQ level. Professionals can improve their game by analyzing mistakes and blunders. To change the game of chess of how it was observed a few years back.The Neural Network, although functional in making predictions, frequently predicts illegal moves, as the range of moves are continuous for ranges of moves that are not legal. I could not create a new solution for this, but thought of a new way to implement a chess AI with a different algorithm: Alpha Beta Pruning , Pure Minimax Search , Zobrist Hashing , greedy Algorithms , Neive Pattern Searching , Move Ordering Search Stay tuned for that!

Appendix A

What is P?

- P is set of all decision problems which can be solved in polynomial time by a deterministic.
- Since it can be solved in polynomial time, it can be verified in polynomial time.
- Therefore P is a subset of NP.

P:Whenever a user types in her password in any organization'ssignin box, the hacker intercepts the password. The threat of such hackers is pervasive.Username is useful to find the particular user and the password for the authorization of the user.Once a password file is stolen, by using the password cracking technique it is easy to capture most of the plaintext passwords

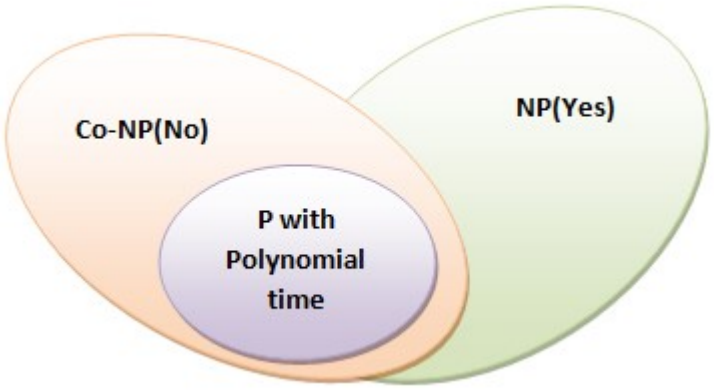


Fig.9.1: Polynomial Time

What is NP?

"NP" means "we can solve it in polynomial time if we can break the normal rules of step-by-step computing".

What is NP Hard?

A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem (nondeterministic polynomial time) problem. NP-hard therefore means "at least as hard as anyNP-problem," although it might, in fact, be harder.

NP Hard:

In this study, we focus on the security issue and deal with fake passwords or accounts as a simple and cost effective solution to detect compromise of passwords. Honeypot is one of the methods to identify occurrence of a password database breach.In this approach, the administrator purposely creates deceit user accounts to lure adversaries and detects a password disclosure, if any one of the honeypot passwords get used

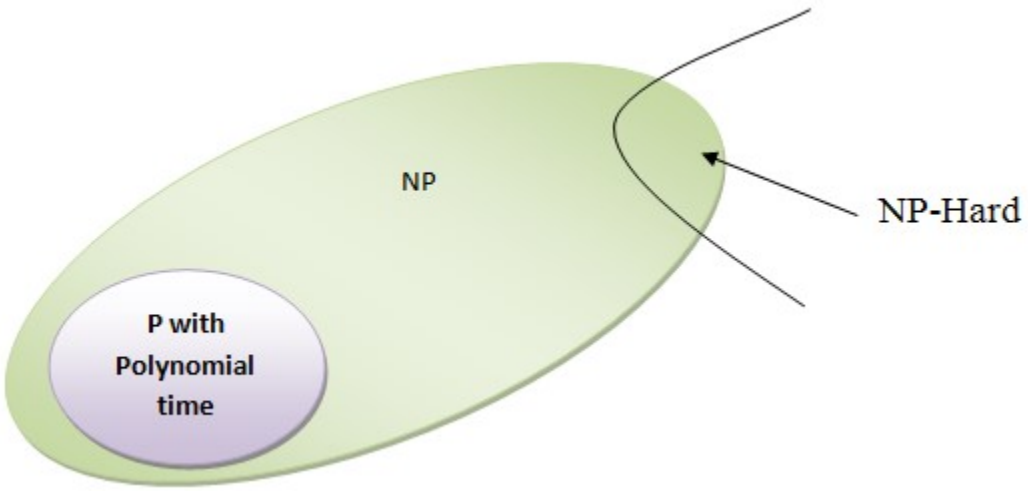


Fig.9.2: NP-Hard Problem

What is NP-Complete?

- Since this amazing "N" computer can also do anything a normal computer can, we know that "P" problems are also in "NP".
- So, the easy problems are in "P" (and "NP"), but the really hard ones are *only* in "NP", and they are called "NP-complete".

- It is like saying there are things that People can do ("P"), there are things that Super People can do ("SP"), and there are things *only* Super People can do ("SP-complete").

NP Complete:

We have study carefully the security of the honeyword system and introduce a number of defect that need to be fitted with before successful realization of the scheme. In this respect, we have pointed out that the strong point of the honeyword system directly depends on the generation algorithm Finally, we have presented a new approach to make the generation algorithm as close as to human nature by generating honeywords with randomly picking passwords that belong to other users in the system. We present a standard approach to securing personal and business data in the system.

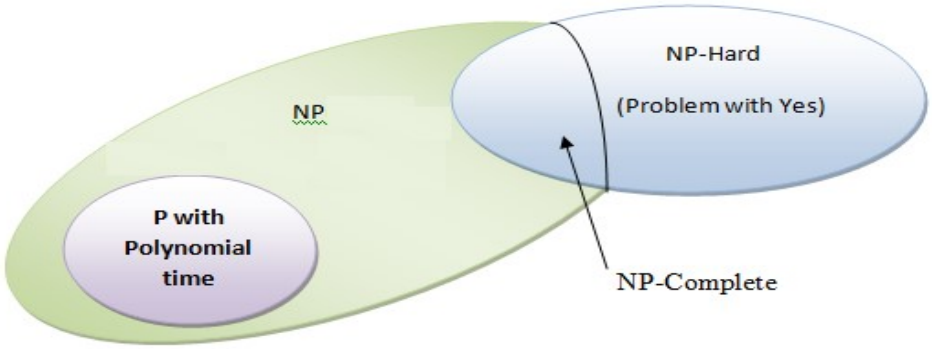


Fig.9.3: NP-Complete Problem.

References

[1] Murray Campbell, A. Joseph Hoane Jr., Feng-hsiung Hsu, “Deep Blue”, IBM, August 1, 2001.

[2] Laurence Fausett, “Fundamentals of Neural Networks Architectures, Algorithms, and Applications”, Dorling Kindersley Pvt. Ltd., South Asia, 2008

[3] Alexandru Godescu, “An information theoretic analysis of decision in computer chess” ETH Zurich December 13, 2011

[4] Diwas Sharma, Udit Kr. Chakraborty, Artificial Neural Network based adaptive chess playing machine. International Journal of Advance Research in Computer Science, Vol 4 no. 4, March-April 2013.

[5] Chess Agent Prediction using Neural Network , John Dalloul , Mark Bechthold , Stanford University.

[6] Chess AI : Completing Paradigms for Machine Learning , Shiva Maharaj, Nick Polson, Alex Turk

[7]Baxter, J., Tridgell, A., Weaver, L.: Learning to play chess using temporal-differences. Mach. Learn. 40(3), 243–263 (2000)

[8]Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: NIPS (2007)

[9]David, O.E., Koppel, M., Netanyahu, N.S.: Genetic algorithms for mentor-assisted evaluation function optimization. In: GECCO (2008)

[10]David, O.E., van den Herik, H.J., Koppel, M., Netanyahu, N.S.: Simulating human grandmasters: evolution and coevolution of evaluation functions. In: GECCO (2009)

[11]David, O.E., Koppel, M., Netanyahu, N.S.: Expert-driven genetic algorithms for simulating evaluation functions. Genet. Program. Evolvable Mach. 12(1), 5–22 (2011)

[12]David, O.E., van den Herik, H.J., Koppel, M., Netanyahu, N.S.: Genetic algorithms for evolving computer chess programs. IEEE Trans. Evol. Comput. 18(5), 779–789 (2014)

[13]Elo, A.E.: The Rating of Chessplayers, Past and Present. Batsford, London (1978)

[14]Hinton, G., Vinyals, O., Dean, J.: Distilling knowledge in a neural network. In: Deep Learning and Representation Learning Workshop, NIPS (2014)

[15]Knuth, D.E., Moore, R.W.: An analysis of alpha-beta pruning. Artif. Intell. 6(4), 293–326 (1975)

[16]Lai, M.: Giraffe: Using deep reinforcement learning to play chess. Master's Thesis, Imperial College London (2015)

[17]Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NIPS (2015)

[18]Romero, A., Ballas, N., Ebrahimi Kahou, S., Chassang, A., Gatta, C., Bengio, Y.: FitNets: hints for thin deep nets. In: ICLR (2015)

[19]Schaeffer, J., Hlynka, M., Jussila, V.: Temporal difference learning applied to a high-performance game-playing program. In: Joint Conference on Artificial Intelligence (2001)

[20]Schaeffer, J., Burch, N., Björnsson, Y., Kishimoto, A., Müller, M., Lake, R.: Checkers is solved. Science 317, 1518–1522 (2007)

[21]Silver, D., et al.: Mastering the game of Go with deep neural networks and tree search. Nature 529, 484–489 (2016)

[22]Tesauro, G.: Practical issues in temporal difference learning. Mach. Learn. 8(3–4), 257–277 (1992)

[23]Wiering, M.A.: TD learning of game evaluation functions with hierarchical neural architectures. Master's Thesis, University of Amsterdam (1995)

[24]Ashton Anderson, Jon Kleinberg, and Sendhil Mullainathan. 2017. Assessing human error against a benchmark of perfection. ACM Transactions on Knowledge Discovery

from Data (TKDD) , Vol. 11, 4 (2017), 45.

[25]John R Anderson, C Franklin Boyle, and Brian J Reiser. 1985. Intelligent tutoring

systems. Science , Vol. 228, 4698 (1985), 456--462.

[26]Tamal Biswas and Kenneth W Regan. 2015. Measuring Level-K Reasoning, Satisficing, and Human Error in Game-Play Data. In 2015 IEEE 14th International Conference on Machine Learning and Applications . IEEE, Miami, FL, 941--947.

[27] The History of Chess AI , Paessler.

[28] Computer Chess , Wikipedia.

[29] Garry Kasparov : Chess , Deep Blue , AI and Putin | Lex Fridman Podcast#46 , Youtube.

[30] Chess and Artificial Intelligence , Chessbase.

[31] The Human side of AI for chess , Microsoft Research Blog.

Plagarism Check Report :-