NBA All-Star Players Network Analysis
Final Project Report

Name: Chen-Chien Kao Unique Name: cchkao

A) Project Introduction

This project explores NBA All-Star players from 2000 to 2025, focusing on their teammate networks and on-court performance. It identifies the shortest relationship paths between any two all-star players. Based on the lineup data, the project evaluates on-court chemistry to find the best teammate and compares the effectiveness for different All-Star duos. It also provides visualizations that map out the network of connections among these elite players, offering deeper insight into player relationships over the past 25 years.

B) Description of Datasets and Code Structure

B.1) Dataset of the Project
The datasets utilized for this project are from Basketball Reference, a comprehensive database that provides detailed statistics on all NBA players, teams, and games. Data collection was carried out using Python-based APIs. The following datasets are utilized:
1. **All-Star Game Roster:** Contains the list of NBA All-Star players from 2000 to 2025.
2. **Player Statistics:** Includes individual season statistics for each all-star player, such as points per game, assists, rebounds, shooting percentages.
3. **Teammate Connections:** Provides information on the teammates each All-Star has played with throughout their career.
4. **Lineup Information:** Offers season-by-season team rosters and statistical performance when specific All-Star duos are on the court together.

Since the rate of data request to Basketball Reference is limited (roughly 20 times per minutes), fetching data directly from the internet is time-consuming. To address this problem, an on-demand data request approach is adopted to reduce the time cost of data fetching. Besides, once data is fetched, it will be saved in local folder to make that later requests for the same can be served from local folder, significantly reducing the number of the Python API calls.

B.2) Code Structure
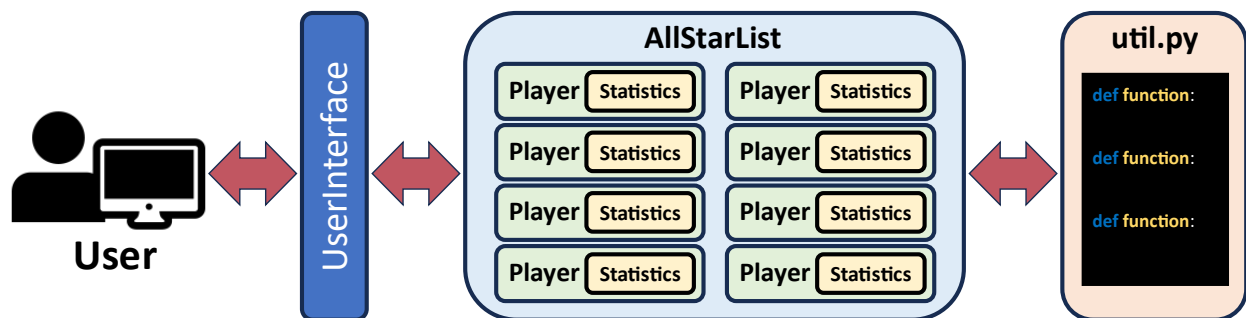In this project, the code is structured in the following:

Figure 1. Code Structure in the Project.

- **class UserInterface:** Deals with the user requests and handles the interaction with the user.
- **class AllStarList:** Collect all player objects for the players in the all-star roster list and execute the command based on the user request from UserInterface.
- **class Player:** Fetch all the data of the player, including the teammate information
- **class Statistics:** Get the statistical data of the player from the Website and store them.
- **util.py:** Provides several useful functions for the projects, including printing the result and searching player names from the website.

C) User Interactions and Function Implementation

This project provides the following types of the user interactions to provide the information and the analysis based on given the condition from the user:

1. **Provide Detailed Player Stats:** The statistics of the specific player given by the user will be shown on the terminal by printing the data in the Statistic class.
2. **Adding the Specific Player for Analyzing**: User can add the specific player who is not in the original database. The program will try to find the player by searching on Basketball Reference based on the name given by user and add it to the list of all-star players. Then the following analysis will take this new player into account.
3. **Searching for the Shortest Degree Between Two Players:** Giving two players' name in the database as input. The program can search for the shortest path by BFS according to the teammate information stored in the Player class.
4. **Finding the Most Connected All-Star Players:** Rank the most connected players based on the information in the dataset by comparing the teammate network of each player.
5. **Finding the Best Partner for Specific Players:** Giving the specific player name, the project will find out the all-star player with the best court chemistry for the given player.
6. **Comparing Player Duos:** Giving the two different duos, the project will evaluate which one is better based on their on-court performance.
7. **Plot the Connection Between the Players:** The program can plot the figure of the teammate connection for these elite players.
8. **Print the Players Stored in the Program:** The program can print all players stored in the program or the all-star roster of the specific year.

D) Observation in the Project

Figure 2 shows the teammate connection among those all-star players. It is obvious that all the connection (edges) in the right-hand is sparser than that in the left-hand side. This difference is due to players (nodes) on the left have been elected as an all-star earlier compared to the players on the right, which implies that they have longer career time to give them more time to form connections with other elite players.
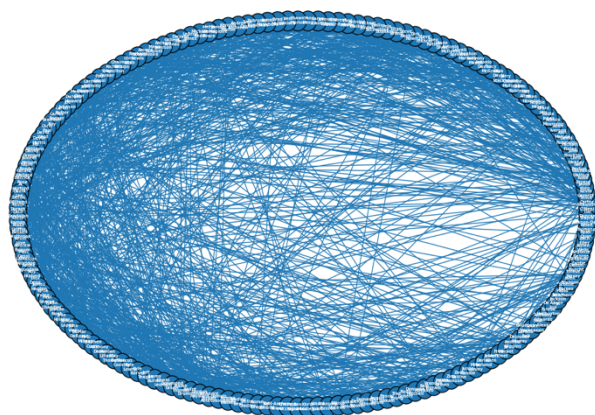

Figure 2. Graph of Teammate Connection.