**A close up of a logo

Description automatically generated**

Project Report

KNOWLEDGE & EXPERT SYSTEMS

Project Title : Dota 2 – Hero Picker

Course Facilitator : Uzair Ahmad

Student Name & IDE : Rajan Walia (100727112)

Peter Kusiak (100399575)

Shemil Hashan (100768825)

**PROBLEM STATEMENT**

**Dota 2 – MOBA PC GAME**

It is a strategy game which is played by roughly 9 million users monthly. In competitive arenas, prize pools as high 34 million is awarded in grand stages. It is a 5 vs 5 game, 2 teams, where each team has to select 5 heroes out of 119 heroes in the pool with different skills. Same hero cannot be picked by both teams. Selection of these heroes directly supports the win probability.

We will specifically look into competitive professional mode of tournaments i.e. Captains Mode.

**A screenshot of a social media post

Description automatically generated**

Ban Phase: Select heroes to be banned. Banned heroes cannot be picked. This done to prevent opponents from picking that hero because some heroes are weak against these heroes

Pick Phase: Pick the heroes you want to play

When you strategize the plan, it is important to ban heroes which will become a problem to our draft and pick strong heroes to win. Some of the hero skills combine to make powerful combos which can turn the course of events. Our expert system will help the users to decide which heroes to ban and pick. This analysis is critical and since there is a timer ticking for picks and bans, making the correct decisions fast is crucial.

**DATA**

Data: We have chosen dotabuff.com data as our knowledge extraction.  
Overall win rates: <https://www.dotabuff.com/heroes/played>

A screen shot of a computer

Description automatically generated

Single hero stats: Example Hero – Pudge - <https://www.dotabuff.com/heroes/pudge>

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

With these data of Best Versus and Worst versus, we can create an expert systems to suggest the next pick or ban.

**IMPLEMENTATION**

**ARRANGE DATA**

All data in current Meta section is arranged as JSON file.

A screenshot of a computer screen

Description automatically generated

Localized\_name / caption - displaying purpose

Name – unique id string

Cat\_wins – win percentage for each skill category

Data – best\_vs – which heroes this particular hero is best against

best\_vs\_advantage – percentage advantage over relevant hero

worst\_vs – heroes weak against

worst\_vs\_disadvantage – percentage disadvantage

This JSON file is our Knowledge Base to be used for selection criteria.

**APP**

A screenshot of a computer screen

Description automatically generated

We have created the application hosting with Flask. In the root route it will present an HTML file for the UI. When a selection happens, /api/suggest route is called to get the recommendations with win probability. When the values are returned, UI is updated. Java Scripts are used to handle web page data and functionality. Index.html is UI and index.js controls UI functionality. We created an instance of Engine to call the recommendation function.

**ENGINE**

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

It initially loads all data of 120 heroes. Then when recommend function is called with ids of my team, enemy team and medal level. We iterate through my team and nested inside for enemy team. Our current selection hero medal values are aggregated together while enemies aggregated separately. Then average is taken for consideration as maximum is not the correct win rate neither minimum. Also, some heroes average less than 50% win rate. In case best is taking my team and enemy team ratio as a percentage for base win rate.

While iterating we check if my team has heroes which are good against enemy current selection, if it exists, positive reward of advantage value is added to win percentage. If our selection has weak heroes against enemy selections, disadvantage value is deducted from win percentage as a negative reward. So the win percentage is calculated with that.

Then we have considered which heroes are best against enemy team with enemy team data worst\_vs. Then they are pooled together followed by aggregating values for each hero then divided by a value larger than max + constant of current advantage aggregates. Then sorted for maximum of that value. Then best up to 5 is selected as recommended.

So, both recommend hero ids and win rate is sent back to app to update.

**GUI**

A screenshot of a cell phone

Description automatically generated

First box is to select level. Dota 2 rank medals Herald, Guardian, Crusader falls under 1. Archon 2, Legend 3, Ancient 4, Diving and Immortal is 5. It is the first selection to make.

Then can choose our team or enemy selection. If we select ours first, no recommendation is given since enemies not filled. If enemy is selected, recommended list is populated. We have to type and make more selections. Based on our selection win percentage and recommendations change. If we have not picked any hero, it will stay at 0.

**DEMO**

Step 1: Open a terminal, go to project folder.

Step 2: pip install -r requirements.txt

This install all required python libraries to run

Step 3: python app.py

This will start the server run

Step 4: Open a browser and go to 127.0.0.1:5000 , index page will be loaded

Step 5: Make the medal level selection by typing 1 – 5 and select option.

Step 6: Enjoy selecting heroes by typing hero names, list will be shown to click and select.

**Example Demo**

A screenshot of a cell phone

Description automatically generated

Figure 1

Figure 1 shows selection of medal as 3. Then enemy Medusa is selected. So the recommendations pop up with Anti-Mage, Broodmother, etc. Since we current picked no hero, win rate is 0.

A screenshot of a cell phone

Description automatically generated

Figure 2

So we make recommended hero Anti-Mage, win percentage shows as 58% and that option is removed from recommended.

FiA screenshot of a cell phone

Description automatically generated

Figure 3

Now we take another suggestion as Broodmother, results in increasing win percentage because both are good against Medusa.

A screenshot of a cell phone

Description automatically generated

Figure 4

Now enemy picks a good counter pick for Anti-Mage, Bloodseeker . So the win percentage goes down significantly.

**FUTURE IMPROVEMENTS**

We have taken data from a frozen time, but these data change rapidly when a new large update come in for Dota, usually once in 3 to 4 months. So, we have to automate the data gathering part for this to work properly.   
  
Also, UI can be improved significantly to cater the player with a good user experience.

If we get chance to analyze each player in my team with their game history, we can improve the recommendation to tailor the players. Dota has an API to get player match data, but it is a bit complex integration which requires high processing power to analyze thousands of games.

**CONCLUSION**

We did not employ a full algorithm from the course but we took the idea of reward and penalty for the suggestion with our own algorithm for selection criteria and calculation. Our final product works on par with our expected level.