# **Finding synergies with network analysis: Using Neo4J to identify the best Pokémon teams.**

In this article we…

* Use network analysis to identify which individuals make the best teammates for each other in a competitive game.
* Discuss the indicators for measuring team performance and teammate cooperation.
* Explain importing data from JSON and performing community detection with Neo4J.

Tags: Neo4J, CYPHER, JSON, Pokémon, Smogon.

## Summary

Smogon.com is a platform for online competitive Pokémon battles, which releases a JSON of data regarding matches every month. This is a rich body of data regarding player behaviour in an eSport, which allows us to analyse how competitors strategize. In this article, we explore the following question: “What is the best team I could pick?” where a team consists of six Pokémon. To answer this, we extract data from matches from average and high ranking players, and compare how these two groups choose teams differently. As a result, we identify clusters of Pokémon that high-achieving players tend to choose from. Although there is no one ‘best’ team, because every strategy has a counter-strategy, we can identify how high performing players select their teams, and we can identify the top teams in the current competitive environment.

## Introduction

The term ‘Metagame’ means ‘a game about games’ and it refers to how competitors in any kind of game make strategic choices to set themselves up to win a series of matches. Examples of this include how Major League Baseball clubs draft their teams for a season, or how coaches plan their team’s training regime. In each case, the objective is not to win a single match, but to win a set of games.

eSports are fertile ground for exploring metagame because they take place in a digital space, meaning with the right tools, we can extract data about almost anything that occurs in the game.

Another interesting feature of eSports is that the game’s developers can and do actively intervene in the parameters of the game in an effort to optimize the game’s ‘balance.’ [‘Balance’](https://books.google.dk/books/about/Game_Balance.html?id=FblPvgAACAAJ&redir_esc=y) here refers to how certain strategies in a competitive match may be simply better than others. The result of ‘unbalance’is a very boring competitive game, where only a few strategies are viable options. Players and fans want a game where there is an interesting variety of strategies to employ, and counters to those strategies. When the game is balanced, there is diversity and an interesting variety of matches to enjoy.

In analysing metagame, we can examine the diversity of viable strategies and analyse how players reach those strategies. From this analysis, we can answer questions such as ‘what is the best choice of strategy given the current state of the metagame?’

In this article, we explore two questions about the metagame at Smogon.com:

* Examine how highly successful players, choose their teams differently to average players. We assume that the teams chosen by the highest performing players in the game are the ‘best’ teams.
* Examine how highly skilled players from 2020 choose teams differently compared to 2015, when the metagame was ‘balanced’ by Smogon.com differently.

## Data

In this article, we demonstrate metagame analysis through network analysis, using data from ‘Pokémon Showdown’ managed by Smogon.com.

We aim to create a series of network graphs, in which we display the tendency of Pokémon to be selected together across different levels of player rankings. The variables in the data are summarised below:

### Variables

* **A measure of player skill**: In this analysis we use [‘Elo’ scores](https://en.wikipedia.org/wiki/Elo_rating_system) to estimate player skill. The basic concept of ELO is explained in the following example: new players start with 1,500 Elo points. If a player wins a game, they receive 25 points plus a variable amount which is larger if the opponent they beat has a higher Elo score. Conversely, the losing player loses 25 Elo points plus more if they lost to a lower-rated opponent. Over time, we expect players with higher Elo scores to be skilled at winning more, tougher games, than lower rated players.
* **A measure of two Pokémon being chosen together:** For this, we use ‘teammate weight,’ the change in posterior probability of one Pokémon’s selection given another selection. For example, if the Pokémon ‘Scizor’ appears on 40% of all teams in our data set, but appears on 60% of all teams with ‘Excadrill’ then we can say that Scizor has a ‘teammate weighting’ of 20 with Excadrill. These weightings can also be negative, which would indicate that teams with Pokémon X tend to actively avoid also choosing Pokémon Y.

### Data Sources

Our source for the above variables is provided for us by Smogon.com. Each month, Smogon publishes a JSON file with data separated by both skill tier and game type. For our analysis, we use:

* The ‘Generation 8, Over Used’ (gen8OU) Game type: This is the most well balanced competitive ladder. In this game type, some Pokémon which are too powerful are not permitted, this way Players can choose from a number of viable strategies instead of being forced to pick only the small handful of ‘Overpowered’ choices. ‘Generation 8’ means the players could choose from all Pokémon available up to the 8th release of Pokémon. Those familiar with the series may know that roughly 150 new Pokémon are introduced each generation.
* We are comparing two JSON datasets: the average skill tier (which is what a starting player would compete against) and the highest skill tier.
* We can also see how the choice of strategies changed over time, by comparing the highest skill tier from 2015, with that of 2020.

The data used in this article is open access and available here:

* [High skill players: Gen8OU at 1825+ ELO, for October 2020](https://www.smogon.com/stats/2020-10/chaos/gen8ou-1825.json)
* [Average skill players: Gen8OU at 0-1500 ELO for September 2020](https://www.smogon.com/stats/2020-09/chaos/gen8ou-1500.json)
* [High Skill players from 2015: Gen5OU at 1760+ ELO for October 2015](https://www.smogon.com/stats/2015-10/chaos/gen5ou-1760.json)

## Methodology

To visualise how players in a skill tier are selecting their teams, we created a network graph and then performed a community analysis to identify when certain clusters of Pokémon are being chosen together to a significant degree. In order to carry out this analysis we set up a neo4j sandbox, and then created an analysis pipeline written in CYPHER (c.f. appendix).

The steps we take to perform this analysis are summarised below:

1. We import raw data from JSON and create a node for each Pokémon in the data set.
2. From the same JSON data, we create an edge (i.e. relationship) called ‘teammate’ between every Pokémon to every other Pokémon. That relationship will have a float property, which is the ‘teammate weight’ variable (c.f. Section ‘Variables’).
3. We remove all relations with a teammate weighting below an arbitrary threshold, in this case 40. This is because it is difficult to visualise graphs where every node is related to every other node, and we are only interested in relationships where Pokémon are very likely to be picked together.
4. We implement [Louvain community detection](https://en.wikipedia.org/wiki/Louvain_method), which is similar to hierarchical clustering, to identify sets of Pokémon that are more frequently chosen together. To do this, we scale the teammate weighting variable to a float between 0 and 1.
5. Finally, we visualise the graph with communities separated by colour.

## Results

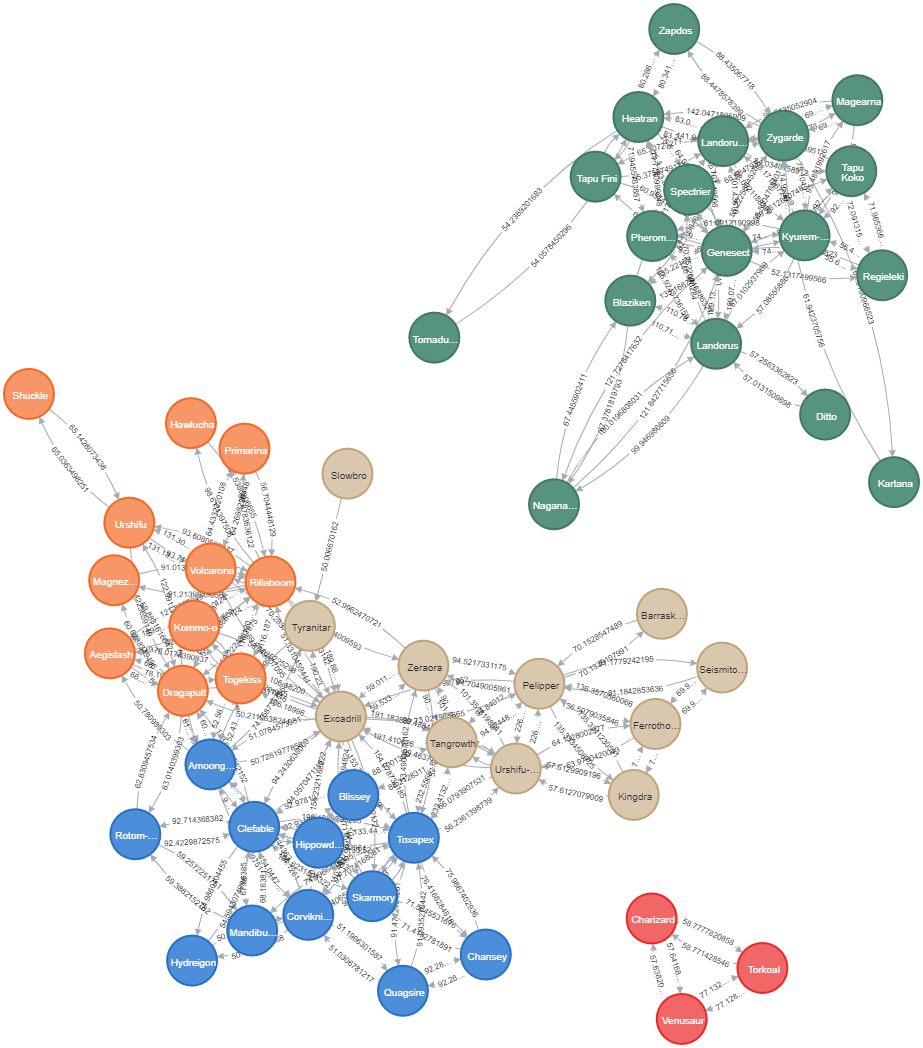
In figure 1, among highly skilled players, we can see some clearly defined strategies for team drafting.

Figure 1: Teammate selection among high skill players, October 2020

* **In blue**, we have a community of 12 options, of which Clefable, Hippowdon and Toxapex are the most central. Experienced players can identify why this is a strong team: Clefable is a powerful Pokémon, but is weak to steel and poison. Hippowdon is an excellent counter to most steel and poison types, including the popular counter to Clefable, Skarmory. Toxapex is a poison type suited for countering \*other\* Clefable-based teams.
* **In brown**, Excadrill appears to be the Pokémon many teams are built around. However, Excardill may also be found in Clefable-based teams, often as a substitute for Hippowdon (hence the relatively lower teammate weight of Hippowdon<->Excadrill).
* **In orange**, Riabloom is another popular choice to build a team around. An interesting observation in this community can be seen in the top left section: Shuckle appears to feature most often in Riabloom based teams, and only when also paired with Urshifu. This tells us there’s an interesting synergy to be found between Schuckle and Urshifu.
* **In green**, Genesect is the most central node. Interestingly no nodes connect this community to the other communities, which tells us this team strategy is mutually exclusive with the other choices.
* **In red,** we have another isolated cluster, but of only three Pokémon.This indicates that these three frequently appear together, but the remaining Pokémon on their team are a mix of many different choices, with no teammate weights above our threshold.

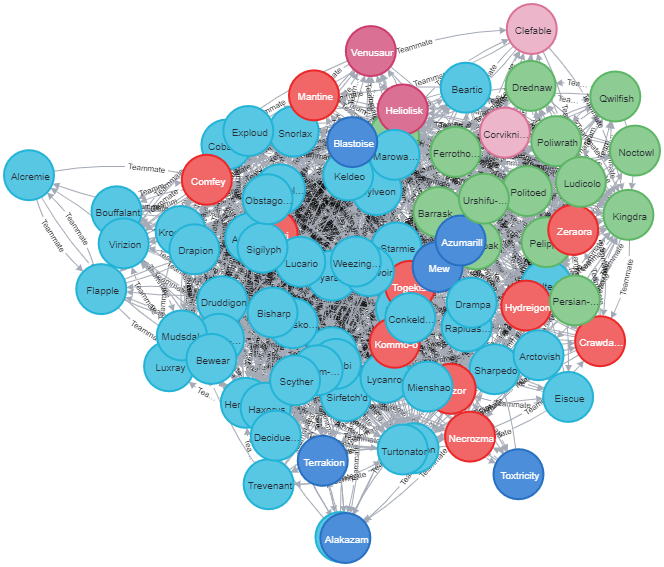


Figure 2: Teammate selection among average players, September 2020

In figure 2, we create the same network graph as in figure 1, but for the average skilled players. From figure 1 and 2 we are then able to compare team selection strategies between average and highly skilled players. By comparing the two figures we find:

* There is much less separation of communities, meaning the average players are mixing Pokémon from different communities much more frequently. As a result, there are less clearly defined sets of Pokémon being organised into typical teams. This may be indicative of players making more experimental choices, whereas more skilled players have a keener sense of the best combinations of teammates.
* The most central nodes in these communities are different to those of the higher skilled tiers. This indicates that average players *are not* making the same teams as higher skilled players, rather, higher skilled players choose totally different teams to average players. This tells us that team selection before a match, not just skilled decisions in a match, is a key factor in success.

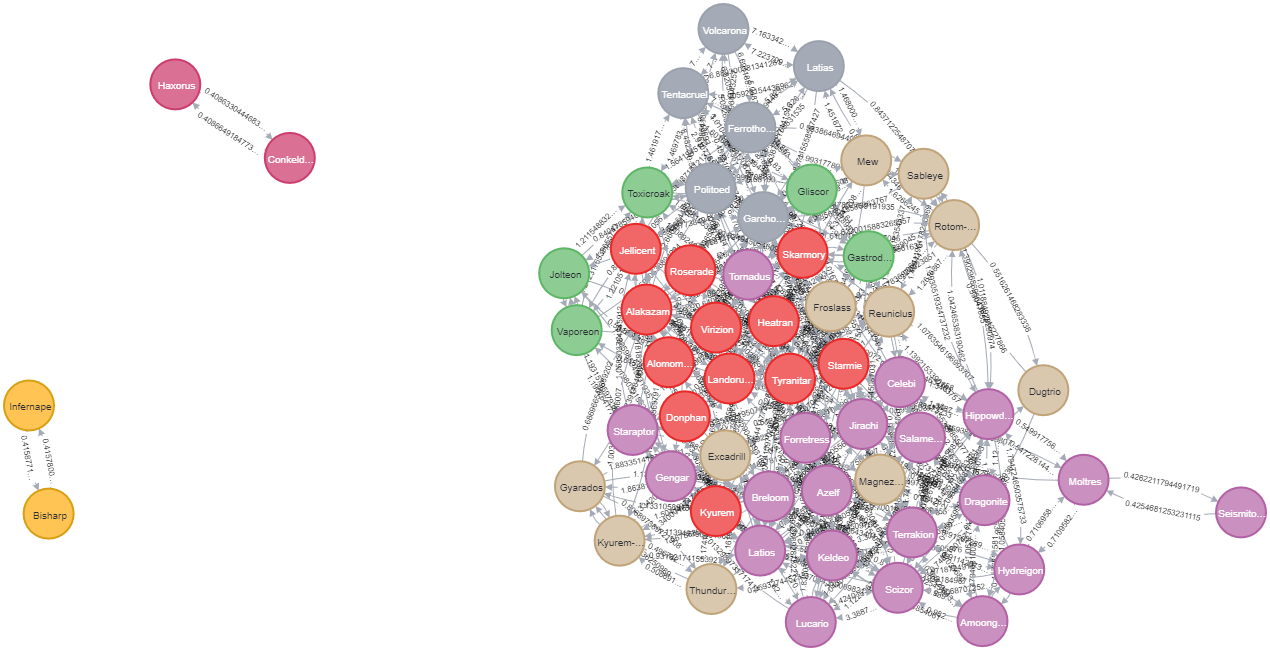


Figure 3: Teammate selection among high skill players, October 2015

It is also interesting to compare what teams are popular in 2020 as opposed to 2015, when generations 6 to 8 were not yet available. In figure 3 we create a network graph for the highly ranked players from 2015. From this figure we once again see that highly ranked players have more clearly defined communities than average tier players.

One interesting observation is that even though Clefable and Excadrill were both available in 2015, they were not pivotal choices for team building, like we see in 2020. This indicates to us how the metagame was shifted over time.

## Conclusion

Metagames in eSports are an excellent topic for analysing strategic choices, because we have a unique opportunity to capture highly detailed information for how players are making choices in response to different strategic contexts. In many games, but also in many other contexts, the way in which individual components work together is just as important as how they operate individually. This emphasis on synergies between actors is an important approach for analysing systems made up of genes, marketing campaigns, or teammates in a game.

In this article, we have made one demonstration of using co-appearance to create network graphs we can analyse to discern the various strategies players employ. We can identify when certain selections of Pokémon may have complementary roles, or when they serve mutually-exclusive roles, and so players should pick one or the other, but not both. Insights into high skill player behaviour like this can help us understand how to achieve success at the game faster and apply that approach to other fields.

## References

* <https://www.smogon.com/forums/threads/everything-you-ever-wanted-to-know-about-ratings.3487422/>
* <https://www.smogon.com/stats/2015-10/chaos/gen5ou-1760.json>
* <https://www.smogon.com/stats/2020-10/chaos/gen8ou-1825.json>
* <https://www.smogon.com/stats/2020-09/chaos/gen8ou-1500.json>
* <https://en.wikipedia.org/wiki/Elo_rating_system>
* <https://en.wikipedia.org/wiki/Louvain_method>

## Appendix: Walkthrough

You can recreate this entire analysis yourself! The below walkthrough may take around 20-30 minutes to complete. This is a beginner friendly walkthrough, so you don’t need any prior experience to follow along. Familiarity with any query language will help you understand what the code is doing.

### Prepare an environment

1. Create a free account with Neo4J and create your own Sandbox [here](https://neo4j.com/sandbox/).
2. Select ‘New Project’ and Select ‘Blank Sandbox.’
3. Select ‘Launch in Browser’
4. At the top of the screen, you will see the command line, enter the following code in chunks (unfortunately, neo4j Sandbox isn’t powerful enough to execute all the code at once!)

### The Code

Execute each commented section separately. The below code is extracting data from one of the selected JSON files. Simply change the URL to a different set from smogon.com/stats, to analyse a different skill tier or time period.

// Create Nodes and Relationships from JSON

WITH "https://www.smogon.com/stats/2020-10/chaos/gen8ou-1825.json" AS url

CALL apoc.load.json(url) YIELD value

UNWIND value.data as d

FOREACH (name in keys(d) | CREATE (Pokémon:Pokémon {id: name, teammates: keys(d[name].Teammates)}))

With value

UNWIND keys(value.data) as name

MATCH (a:Pokémon) WHERE a.id = name

UNWIND a.teammates as tm

MATCH (b:Pokémon) WHERE b.id = tm

CREATE (a)-[r:Teammate {name: a.id +'<->'+b.id, weight: value.data[a.id].Teammates[b.id]}]->(b)

// Cull Relationships where weight is below a threshold

With 40 as threshold

MATCH p=()-[r:Teammate]->() WHERE r.weight < threshold DELETE r

// Scaling weights before community detection algorithm

MATCH ()-[r:Teammate]->() WITH toFloat(max(r.weight)) as max

MATCH ()-[r:Teammate]->() SET r.nweight = toFloat(r.weight) / max

// Create a named graph with gdc package

CALL gds.graph.create(

'myGraph',

'Pokémon',

'Teammate',

{

relationshipProperties: 'nweight'

})YIELD graphName

// Call the Louvian community detetction algorithm on the named graph

CALL gds.louvain.write('myGraph', { writeProperty: 'community', relationshipWeightProperty: 'nweight' })YIELD communityCount

// Name the community after most central node

MATCH (p:Pokémon)

WITH p, p.community as community, size( (p)-[:Teammate]-() ) as centrality ORDER BY community ASC, centrality DESC

WITH community, (head(collect(p))).id as top, count(\*) as size, collect(p.id)[0..6] as likleyTeam, collect(p) as all

ORDER BY size DESC

FOREACH (p IN all | SET p.communityName = top)

// Add the community name as a label to each node, which will then color each node in the visualization

MATCH (p:Pokémon)

CALL apoc.create.addLabels(p,[p.communityName]) yield node RETURN node

// Before visualising, we remove the 'Pokémon' label, so neo4j will color code by community

MATCH (p:Pokémon)

REMOVE p:Pokémon

RETURN p.name, labels(p)

// Visualize the graph

MATCH pkmn=()-[r:Teammate]->() RETURN pkmn