Big Data Reduction: Lessons Learned From Analyzing One Billion Dota2 Matches

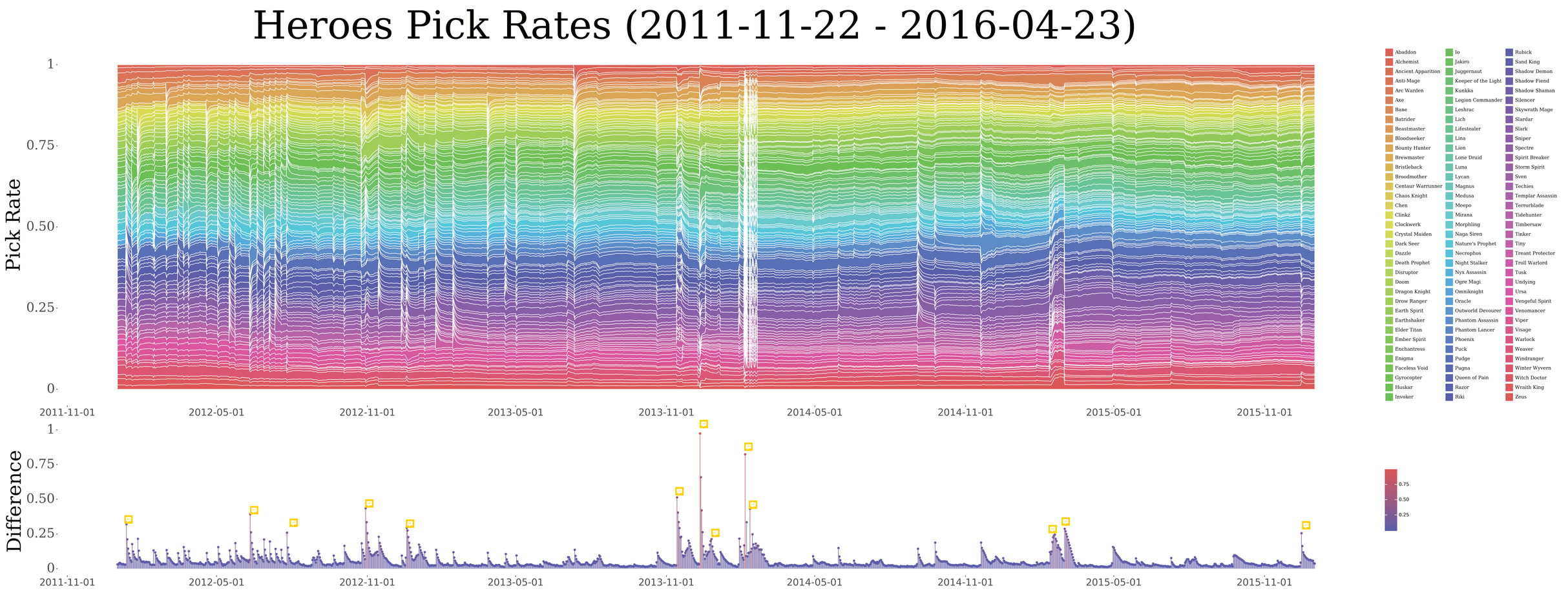
https://git.io/fjJHU

Marco Lussetti
www.marcolussetti.com
marco@marcolussetti.com

Dyson Fraser
www.linkedin.com/in/dysonfraser
dysonfraser@gmail.com

Mila Kwiatkowska
Supervisor

https://git.io/fjJHU



Why Study DOTA2?

Dota 2 is a Multiplayer Online Battle Arena, more commonly known as a MOBA. The core gameplay revolves around teams of five players choosing from a character pool of over one hundred characters and facing off against another five person team in a race to destroy a large structure located in the enemy's base called The Ancient. The publisher of Dota 2, Valve, hosts an annual tournament with a prize poll in the millions of dollars. To keep the game bug free, competitive and fresh, Valve implements frequent patches affecting character balance as well as introducing new playable characters. Thanks to the OpenDota (formally Yasp) project, a very large dataset has been created that has gathered over one billion matches of Dota 2 from 2011 to 2016. We will use this data to predict changes in the metagame due to patches as well as observe the impact of The International. tournament.

Big Data Reduction

The Dataset. The dataset ranges from March 2011 to April 2016 and contains data on 1,191,768,403 (>1 billion) matches that were played during that time<S>. This data is publicly available as a gunzipped CSV file (151GB zipped, 1.2TB unzipped)<S>. At our disposal was only a personal machine with good but not outstanding performance that could not possibly handle the data in its original format. As such, we had to use for Big Data Reduction techniques.

Dimensionality reduction. The curse of dimensionality is a well-known problem where the high number of dimensions present in the dataset cause increasingly high computational burdens[1]. In our case, we have over 50 dimensions in the dataset. We established than only 22 dimensions were needed to achieve our objectives and processed our dataset accordingly. We calculated that on an average day in the dataset, this could have reduced the space needed as much as from 671 MB to 99MB (a 6.7x reduction).

Locality reduction or optimization. We also established that for our purposes of detecting points where the metagame shifted, we need not possess extremely granular data. After all, we are interested in the day or few days when this shift occurs rather than the minute. Thus, we can afford a much coarser locality than what is present in the dataset – we do not need per-match data, just per-day or per-week summaries of what heroes were picked. A sum of the daily picks (and wins/losses) for each hero was what we sought to produce. Condensing the data in this manner produced extraordinary space savings as expected: the potential reduction could be as great as from 99MB to 3KB (35,545x reduction).

Results of the implementation. These are theoretical results for simulated data, however in our actual implementation, we observed the original dataset of 1.2TB being reduced to 3MB or a 396,514x compression. The increased performance in the real-world application is likely a result of JSON compression and some level of data sparsity that was not present in the hand-crafted test case.

Poor Man's Solution

Java 8 Streams (Bad idea! Keep it simpler!). With such a large amount of data to be processed, it is not viable to load the data in memory, and it is necessary to process it iteratively. Initially, we sought to use Java 8 Streams to do so to avoid having to decompress it ahead of time. However, we quickly discovered that Streams add significant overhead and quickly exhaust a consumer computer's memory. A simple for-loop (well, an iterator really) was much more performant!

CSV Parsers (They matter!). Key to the performance of iterating through such a large dataset. We originally attempted to use the fastest CSV parser available, uniVocity-parsers. However, memory usage from this parser was extremely substantial and would slow down after ~400,000 and crash at ~450,000 records. We fell back to one of the most common and well supported parsers, opencsv, however we found its performance insufficient (around 2,000 lines/s). We found that the second-fastest parser, SimpleFlatMapper, has negligible performance loss but maintains a constant low memory profile. The entire processing could be done with a few hundred megabytes of RAM!

JSON Parsers (Take your pick). As each row contains a JSON field that needs to be parsed, a JSON parser must be employed. We did not see any significant differences in the parsers we briefly auditioned and were able to make our choice by and large based on intuitiveness. We found Jsoniter to be perfectly serviceable.

HashMap Optimization (Doesn't matter). We were concerned about the memory used by the HashMap/Dictionary we intended to store the condensed data in while it was processed, and selected a high-performance library, Trove4J. This library from Palantir stores a THashMap using only (8 * CAPACITY) bytes compared to Java's HashMap that uses (32*SIZE + 4*CAPACITY) bytes. However, the HashMap of either kind did not amount to much memory usage considering the very coarse locality needed for this project.

```
FileReader fileReader = new FileReader(input);
Iterator<String[]> csvReader = CsvParser.iterator(fileReader);
while (csvReader.hasNext())
    parseRow(csvReader.next(), onlyCount);
```

Metagame Shifts

The spikes in the graph above show the change in of the pick rate heroes compared to the average of the last two weeks. The first peak of our graph shows the introduction of four characters into the game as well as fine tuning on preexisting characters. As Dota 2 was in beta until July 2013, the graph looks quite turbulent up until then, this is caused by multiple characters a month being ported from Dota to Dota 2. These smaller peaks show users trying out the new heroes. The larger peaks indicating balance changes to these added characters.

Once Dota 2 was fully released the meta was relatively consistent up until November of that year. A special event featuring newly introduced heroes Earth Spirit and Ember Spirit as well as revamped Storm Spirit. Patch 6.79c released in December and removed the hero Skeleton King from the game, replacing him with Wraith King. Legion Commander was also ported from Dota. Earth Spirit played a big role in the November 2013 spike as he was nerfed this patch.

In early February 2014, the second largest spike occurs with the release of patch 6.80. This patch was focused around balancing as our good friend Earth Spirit was again nerfed along with a few other heroes. Many other heroes were buffed during this patch causing more diversity in picks during matches.

The International normally takes place in August with a few exceptions where it took place in July, it does not make a noticeable impact on our graph. This means that the heroes that the professional players pick during these tournament matches do not influence the broader player base to mimic their choices.

The event that caused the most deviation from previous choices was the release of a new hero. With a few exceptions like the major peak of patch 6.80 that was caused by large balance changes, most peaks come from the introduction of new characters. This shows that adding new characters to a game entices players to pick different heroes more than balancing existing heroes.

Future Developments

Hentia et volorest aut doluptae nos di omnis quiscid erferro eum quos ut laut voloratur ab ius, nihitae moluptaquiam que soluptatiunt aborpor aestem comnihicto estrum res exerferum reptia nobita doloris ipsandit, commolo rrovit alit ellaute molorer uptaepe ruptae et que pelendi gendis dis nonsedit aut res molupta digent vendigent.

Ost, sit, optiore ea sit, suntis ad es sere, omnis denis rehenec totatem porrum solorepe nullit od quae. Ut mos volore restiatumet omni alique vernate ditinitin pla quos sit is nonsecate dia comnis quam, omnis ipicimusam arumqui corpore quaessi tiusto dunte comnitiat.

Uciur, volor sim repe autem aut ut videl maios eicipsus, officiis ium, omnim aut pore volore nem la quisi cum sunt as ut ant vendund empossum ut eatendipsam harum sum

References

Henim qui blacit, sit fugitat.

Pudisse quatur? Ipis aute nihillupta quas entias ipsum landem apiendam, offic tem. Et que nullaut voluptate voluptatur? Vid minctat la dolupitia corruptat.

Ere mi, torepudit et voloreptibus que excerspel idio omnimus nobit, voluptatur?

Omnient voloribus, vollaut eossitate necerore simus doluptas at.

Volorae pa vellenist qui con comniendus, corem ate que sint aut ut utem quat.

Velias isti sapidel estotam ipsum aliciduci iume volore vellaci picitia spicipicture velignist labore modit omnia velitat explitam quaspe verum sernam, cus erum voluptatur aut renimus ut es atatemolore pedigendit inim ne nonsequi dolliquas enducitibus, voluptatqui is arunt vit ut atem et, qui aut omnis rent exceat quas nimaior uptatium rerciatemodi viti commolu ptatia nesto doluptatibus alitatatur, im dis quae. Ullaboriat aut eatem es doles sed quo incto el etur?