Clustering Legends of Runeterra archetypes

Goal

Given a list of `n` Legends of Runeterra decks, how to automatically group them into **archetypes**, *ie* groups of similar decks?

Clustering decks as archetypes allows for better calculation of metrics like win-rate and play-rate. This is a problem that comes back for every card game for every card game I have ever played, and is usually “simply” solved by having humans sort the decks themselves.

But what if this clustering could be automated?

Existing clustering algorithms

Clustery analysis is a complex and heavily researched domain. Algorithms usually exhibit `O(n**3)` complexity.

TODO -> Explain that we’ll try hierarchical but not k-means. Select a soft-clustering archetype to try too.

Preparing data

Games selection

As Legends of Runeterra does not offer access to ranks through its API, I use TrueSkill to identify the best players on a given server and parse them in descending skill order. I only use ranked results to determine skill ratings.

I tested the algorithm by verifying that the best players identified by TrueSkill were indeed in Master rank. We can do that by checking account names, which is the only type of rank data available.

This allows me to parse games for the ~20 000 best players per server, which returns ~35 000 ranked games per day per server. As there are 3 servers in total, we have ~100 000 games per day of data.

SQL query

The data parsed from Riot’s API is susceptible to regular model changes, and I’ve therefore saved it as `JSONB` in `Postgres`.

Here is how to query all decks for a given patch, get the number of games played with each, and order them by that value:

```
SELECT
  COUNT(*),
  jsonb_array_elements(data->'info'->'players')->>'deck_code'
FROM lor_game
WHERE data->'info'->>'game_version' = 'live-green-3-13-42'
GROUP BY jsonb_array_elements(data->'info'->'players')->>'deck_code'
ORDER BY COUNT(*) DESC
```

Games	Deck code
14807	CICQCAQDAMAQKBQBAEDAMFIEAIDBMGRGHICQCAYCBQHSKKACAEBAMLIBAYDB4AA
11885	CMCQCAQAAIAQIAADAECAO6IBAUAAYAYGA4DQSCQFAEAQACIBAUAA4AIGA4EAEAQAAQDQGBAHHNFEYAYBAEAA6AIEA5WQCBQHBM
9272	CUFACAIEGYAQEBR4AEBQKBQBAQEQ2AIFAEDACBQCAMAQMAYMAEDAYAQCAIBQGBADAEBQEDBIAAAQCAIDFY
7688	CMBQCAQAAIBAIB3HQIAQKBQHA4EASCQOAUAQEAABAECAAAQCAEAASDYCAQDQ2OYCAUAAYDQA
6112	CICACAQDAMAQKBQBAQBAMFQ2EY5AKAIDAIGA6JJIAIAQEBRNAIDAMFI6AEAQMBQ4
…	…

Deck codes to deck lists

Riot gives us deck codes, but we want a list of cards to easily compute the distance.

We will use lor-deckcodes for that, and will then transform card IDs into card names using our own database.

```python
# Preparing the data with our ORM
import dotenv

dotenv.load_dotenv(verbose=True, override=True)

# Connecting to the database
from neotokyo import db

session = db.connection.ghost_session_maker.session_maker()

# We checked for the latest patch value by looking at the most recent games
latest_version = "live-green-3-13-42"

# Querying the JSONB column
import sqlalchemy

# We also get the faction at it'll be useful later
game_version = db.models.LorGame.data["info"]["game_version"].as_string()
deck_code = sqlalchemy.func.jsonb_array_elements(
    db.models.LorGame.data["info"]["players"]).op("->>")("deck_code")
factions = sqlalchemy.func.jsonb_array_elements(
    db.models.LorGame.data["info"]["players"]).op("->>")("factions")
games_count = sqlalchemy.func.count()

# We only select decks with 10 or more games to reduce the amount of data
latest_patch_decks = session.query(
    deck_code,
    factions,
    games_count,
).filter(game_version == latest_version).group_by(
    deck_code,
    factions,
).order_by(games_count.desc()).having(games_count >= 10).all()

# Checking the # of decks
print(f"Found {len(latest_patch_decks):,} different decks for the latest patch")
```

```
Found 17,176 different decks for the latest patch
```

```python
# We use a utility to convert deck codes to deck lists, then turn card codes into
card names
import lor_deckcodes

# Adding a small utility to compute card names
card_cache = {}


def get_card_name(code: str) -> str:
  if code not in card_cache:
    card_cache[code] = session.get(db.models.LorCard, code).data["name"]

  return card_cache[code]


decks_data = {}

for deck_code, factions, games_count in latest_patch_decks:
  deck = lor_deckcodes.LoRDeck.from_deckcode(deck_code)

  # We store decklist as a set of 40 unique strings as it lets use the
intersection operator
  # We tried with unique integers but saw no significant performance improvement
  cards = set()
  for card in deck.cards:
    for i in range(card.count):
      cards.add(f"{i}-{get_card_name(card.card_code)}")

  decks_data[deck_code] = {
      "count": games_count,
      "cards": cards,
      "factions": factions,
  }

# We show a sample "decklist" to check the format
from pprint import pprint

first_deck_code = next(iter(decks_data))
print(f"Deck code:\t{first_deck_code}")
print(f"Factions:\t{decks_data[first_deck_code]['factions']}")
print(f"Games count:\t{decks_data[first_deck_code]['count']}")
print(f"Deck lenght:\t{len(decks_data[first_deck_code]['cards'])}")
print(f"Sample cards:")
pprint(list(decks_data[first_deck_code]["cards"])[:10])
```

```
Deck code:      CICQCAQDAMAQKBQBAEDAMFIEAIDBMGRGHICQCAYCBQHSKKACAEBAMLIBAYDB4AA
Factions:       ["faction_Bilgewater_Name", "faction_Noxus_Name"]
Games count:    17042
Deck lenght:    40
Sample cards:
['2-Jagged Butcher',
 '2-Noxian Fervor',
 '1-Twisted Fate',
 '0-Jagged Butcher',
 '1-Legion Saboteur',
 '0-Legion Grenadier',
 '0-Precious Pet',
 '2-Miss Fortune',
 '1-Zap Sprayfin',
 '2-Legion Saboteur']
```

Rule-based clustering

Basic idea

- Define the **distance** between two decks as the number of different cards between them
- Define **archetypes** as groups of decks satisfying the following rule:
  - **Rule**: All decks must have 26 cards in common with every other deck in the archetype (65%)
  - The archetype is the group of overlapping cards
- For each deck, compute the biggest possible cluster “centered” on that deck
  - We add decks one by one by adding the ones with shortest distance to the cluster each step, defined as the maximum distance to a deck in the cluster

Issues

- With a direct implementation, computational time was through the roof at **~24h with ~17,000 decks**
  - Complexity `O(n**3)` does that
- Even with multiple optimisations and shortcuts the method was still way too heavy and took multiple hours for each iteration

Local optimisation

High complexity came from needing to check clusters for all decks first. The way to make the process much faster is by introducing a way to quickly create clusters as soon as we spot a particularly high “concentration” of similar decks.

My second idea was therefore to do a first pass with a more stringent rule, but add clusters as soon we find a “sizable” cluster. I chose the limit of any cluster containing at least **1% of the decks**.

Pseudocode:

- We iterate on decks in descending popularity (# games)
  - More popular decklists should be good average representatives of their archetypes
- For each deck, we add other decks with at least 33 cards in common
  - This creates pre-clusters of cards that all share at least 26 cards
- Then we iterate on all other decks and add them in ascending distance to the cluster
- As soon as we find a cluster representing at least 1% of the dataset, we add it to our existing clusters
- If we didn’t find a 1% cluster, we add the biggest cluster we found and start our iteration again with the remaining decks

Deck factions

Narrator’s voice: it wasn’t enough.

Even with this huge speed-up, the process stopped identifying big clusters after 8 iterations and moved at a sluggish pace afterwards.

But there is another property we can easily use for our ~17,000 decks: their factions. Each deck in Runeterra has 2 factions it is associated to, and an archetype will never deviate from 2 factions. There are some outliers where an archetype is mostly mono-faction with a splash, but splitting those decks in different archetypes is actually better for us to identify interesting new strategies.

By first creating list of decks per factions, we are able to split the clustering problem in ~100 smaller problems, which is 1,000,000 times faster under `O(n**3)` complexity.

```python
# Defining our distance calculation

# Creating a cache
from collections import defaultdict
distances_cache = defaultdict(dict)

# Distances will get computed on-demand and cached
def get_distance(dc_1: str, dc_2: str) -> str:
  # We always take them in growing lexicographic order
  d, dd = sorted([dc_1, dc_2])

  if dd not in distances_cache[d]:
    distances_cache[d][dd] = 40 - len(decks_data[d]["cards"]
                               & decks_data[dd]["cards"])

  return distances_cache[d][dd]

# Checking some values
print(f"Self distance (should be 0):\t{get_distance(first_deck_code,
first_deck_code)}")
```

```
Self distance (should be 0):    0
```

```python
# Creating faction groups
factions_decks = defaultdict(list)

for deck_code in decks_data:
  factions_decks[decks_data[deck_code]["factions"]].append(deck_code)

print(f"The faction with the most decks is", max(factions_decks, key=lambda x:
len(factions_decks[x])))
```

```
The faction with the most decks is ["faction_BandleCity_Name",
"faction_ShadowIsles_Name"]
```

```
# This is the most important parameter as it defines how "compact" our clusters
are
MAX_DISTANCE = 20
CUTOFF = 5/100

# This will be a list of lists containing deck codes
clusters = []

for faction in factions_decks:
  print("############################################################")
  print(f"{len(clusters)} clusters found so far")
  print(f"Starting parsing of {faction=}")

  remaining_decks = factions_decks[faction]
  print(f"\tParsing {len(remaining_decks)} decks")

  # This will iterate untils all decks in the faction are in a cluster
  while len(remaining_decks) > 0:
    message = None

    biggest_cluster = []

    # We iterate on each remaining deck and treat it as a possible cluster center
    for deck_code in remaining_decks:

      # We add all the decks <= MAX_DISTANCE/2
      # We know they're connected together as d(b,c) <= d(a,b) + d(a,c)
      cluster = {
          d for d in remaining_decks
          if get_distance(deck_code, d) <= MAX_DISTANCE / 2
      }

      # We pre-select eligible nodes as the ones <= MAX_DISTANCE from our center
      eligible_decks = {
          d for d in remaining_decks
          if MAX_DISTANCE / 2 < get_distance(d, deck_code) <= MAX_DISTANCE
      }

      # Then we check decks that can be added to the cluster
      while True:
        minimum_distance = float("inf")
        minimum_deck = None

        for eligible_deck_code in eligible_decks:
          # We use a list comprehension to find the maximum distance
          # TODO don't use max, we want to break ASAP if the deck is too far
          cluster_distance = max(
              get_distance(eligible_deck_code, cluster_deck_code)
              for cluster_deck_code in cluster)

          if cluster_distance < minimum_distance:
            minimum_distance = cluster_distance
            minimum_deck = eligible_deck_code

        # One step of iterations is over
        if minimum_deck is None:
          # If we didn't find a new minimum deck, we stop
          break

        else:
          # Adding the deck we found
          cluster.add(minimum_deck)

          # We remove the deck from the remaining decks
          eligible_decks.remove(minimum_deck)

      # Here, we have the biggest cluster centered around deck_code

      # We save the cluster if it's the biggest one found so far
      if len(cluster) > len(biggest_cluster):
        biggest_cluster = cluster

      # If we find more than X% of remaining decks that way, we select this
cluster to speed up the process
      if len(cluster) >= len(remaining_decks) * CUTOFF:
        break

    # We finished one: we add the biggest cluster
    clusters.append(biggest_cluster)

    # Select the decks not in a cluster yet
    remaining_decks = [d for d in factions_decks[faction] if not any(d in c for c
in clusters)]
```

```
############################################################
0 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Noxus_Name"]'
        Parsing 507 decks
############################################################
36 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Shurima_Name"]'
```

```
        Parsing 719 decks
############################################################
71 clusters found so far
Starting parsing of faction='["faction_Jhin_Name", "faction_Noxus_Name"]'
        Parsing 327 decks
############################################################
88 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name",
"faction_ShadowIsles_Name"]'
        Parsing 1015 decks
############################################################
127 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_MtTargon_Name"]'
        Parsing 562 decks
############################################################
223 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Bilgewater_Name"]'
        Parsing 193 decks
############################################################
230 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Ionia_Name"]'
        Parsing 107 decks
############################################################
261 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name",
"faction_ShadowIsles_Name"]'
        Parsing 638 decks
############################################################
321 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_Piltover_Name"]'
        Parsing 352 decks
############################################################
386 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_Noxus_Name"]'
        Parsing 407 decks
############################################################
429 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_MtTargon_Name"]'
        Parsing 195 decks
############################################################
458 clusters found so far
Starting parsing of faction='["faction_Piltover_Name", "faction_Shurima_Name"]'
        Parsing 598 decks
############################################################
507 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name",
"faction_Bilgewater_Name"]'
        Parsing 259 decks
############################################################
533 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_MtTargon_Name"]'
        Parsing 89 decks
############################################################
562 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_Shurima_Name"]'
        Parsing 549 decks
############################################################
591 clusters found so far
Starting parsing of faction='["faction_Freljord_Name",
"faction_ShadowIsles_Name"]'
        Parsing 743 decks
############################################################
691 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_ShadowIsles_Name"]'
        Parsing 306 decks
############################################################
720 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Ionia_Name"]'
        Parsing 155 decks
############################################################
734 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_MtTargon_Name"]'
        Parsing 212 decks
############################################################
777 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_Shurima_Name"]'
        Parsing 307 decks
############################################################
815 clusters found so far
Starting parsing of faction='["faction_Noxus_Name", "faction_ShadowIsles_Name"]'
        Parsing 679 decks
############################################################
920 clusters found so far
Starting parsing of faction='["faction_ShadowIsles_Name", "faction_Shurima_Name"]'
        Parsing 489 decks
############################################################
1007 clusters found so far
Starting parsing of faction='["faction_Piltover_Name",
"faction_ShadowIsles_Name"]'
        Parsing 422 decks
############################################################
1070 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_Piltover_Name"]'
```

```
        Parsing 424 decks
############################################################
1124 clusters found so far
Starting parsing of faction='["faction_MtTargon_Name", "faction_Noxus_Name"]'
        Parsing 28 decks
############################################################
1137 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Freljord_Name"]'
        Parsing 199 decks
############################################################
1230 clusters found so far
Starting parsing of faction='["faction_MtTargon_Name", "faction_Piltover_Name"]'
        Parsing 206 decks
############################################################
1287 clusters found so far
Starting parsing of faction='["faction_Shurima_Name"]'
        Parsing 430 decks
############################################################
1331 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_ShadowIsles_Name"]'
        Parsing 312 decks
############################################################
1415 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_ShadowIsles_Name"]'
        Parsing 75 decks
############################################################
1422 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_ShadowIsles_Name"]'
        Parsing 528 decks
############################################################
1509 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Ionia_Name"]'
        Parsing 56 decks
############################################################
1535 clusters found so far
Starting parsing of faction='["faction_Noxus_Name", "faction_Piltover_Name"]'
        Parsing 327 decks
############################################################
1594 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Shurima_Name"]'
        Parsing 265 decks
############################################################
1616 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Noxus_Name"]'
        Parsing 49 decks
############################################################
1631 clusters found so far
Starting parsing of faction='["faction_Noxus_Name", "faction_Shurima_Name"]'
        Parsing 139 decks
############################################################
1666 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_Noxus_Name"]'
        Parsing 436 decks
############################################################
1744 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Demacia_Name"]'
        Parsing 191 decks
############################################################
1772 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Freljord_Name"]'
        Parsing 179 decks
############################################################
1810 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_MtTargon_Name"]'
        Parsing 499 decks
############################################################
1904 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Shurima_Name"]'
        Parsing 406 decks
############################################################
1918 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Piltover_Name"]'
        Parsing 484 decks
############################################################
1995 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_Shurima_Name"]'
        Parsing 198 decks
############################################################
2003 clusters found so far
Starting parsing of faction='["faction_MtTargon_Name",
"faction_ShadowIsles_Name"]'
        Parsing 220 decks
############################################################
2051 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Freljord_Name"]'
        Parsing 121 decks
############################################################
2081 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_MtTargon_Name"]'
        Parsing 46 decks
############################################################
2096 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Demacia_Name"]'
```

```
        Parsing 49 decks
############################################################
2111 clusters found so far
Starting parsing of faction='["faction_MtTargon_Name", "faction_Shurima_Name"]'
        Parsing 95 decks
############################################################
2147 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Ionia_Name"]'
        Parsing 269 decks
############################################################
2213 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Piltover_Name"]'
        Parsing 28 decks
############################################################
2222 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Bard_Name"]'
        Parsing 20 decks
############################################################
2230 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Evelynn_Name"]'
        Parsing 113 decks
############################################################
2243 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Piltover_Name"]'
        Parsing 83 decks
############################################################
2291 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Evelynn_Name"]'
        Parsing 37 decks
############################################################
2300 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_Piltover_Name"]'
        Parsing 14 decks
############################################################
2309 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Noxus_Name"]'
        Parsing 153 decks
############################################################
2337 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_Ionia_Name"]'
        Parsing 103 decks
############################################################
2396 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Piltover_Name"]'
        Parsing 115 decks
############################################################
2432 clusters found so far
Starting parsing of faction='["faction_Ionia_Name", "faction_Jhin_Name"]'
        Parsing 43 decks
############################################################
2442 clusters found so far
Starting parsing of faction='["faction_MtTargon_Name"]'
        Parsing 59 decks
############################################################
2461 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Demacia_Name"]'
        Parsing 43 decks
############################################################
2482 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Evelynn_Name"]'
        Parsing 15 decks
############################################################
2483 clusters found so far
Starting parsing of faction='["faction_Jhin_Name", "faction_Piltover_Name"]'
        Parsing 14 decks
############################################################
2488 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Noxus_Name"]'
        Parsing 12 decks
############################################################
2493 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_Noxus_Name"]'
        Parsing 6 decks
############################################################
2496 clusters found so far
Starting parsing of faction='["faction_ShadowIsles_Name"]'
        Parsing 60 decks
############################################################
2532 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_MtTargon_Name"]'
        Parsing 5 decks
############################################################
2537 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Jhin_Name"]'
        Parsing 4 decks
############################################################
2540 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_Freljord_Name"]'
        Parsing 10 decks
############################################################
2546 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name", "faction_Jhin_Name"]'
        Parsing 7 decks
```

```
############################################################
2552 clusters found so far
Starting parsing of faction='["faction_Demacia_Name"]'
        Parsing 32 decks
############################################################
2565 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Freljord_Name"]'
        Parsing 32 decks
############################################################
2576 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name", "faction_Evelynn_Name"]'
        Parsing 9 decks
############################################################
2581 clusters found so far
Starting parsing of faction='["faction_Bilgewater_Name"]'
        Parsing 17 decks
############################################################
2587 clusters found so far
Starting parsing of faction='["faction_Noxus_Name"]'
        Parsing 12 decks
############################################################
2597 clusters found so far
Starting parsing of faction='["faction_BandleCity_Name"]'
        Parsing 6 decks
############################################################
2601 clusters found so far
Starting parsing of faction='["faction_Demacia_Name", "faction_Jhin_Name"]'
        Parsing 1 decks
############################################################
2602 clusters found so far
Starting parsing of faction='["faction_Freljord_Name"]'
        Parsing 27 decks
############################################################
2617 clusters found so far
Starting parsing of faction='["faction_Ionia_Name"]'
        Parsing 8 decks
############################################################
2622 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Shurima_Name"]'
        Parsing 6 decks
############################################################
2627 clusters found so far
Starting parsing of faction='["faction_Jhin_Name", "faction_MtTargon_Name"]'
        Parsing 6 decks
############################################################
2630 clusters found so far
Starting parsing of faction='["faction_Evelynn_Name", "faction_Ionia_Name"]'
        Parsing 2 decks
############################################################
2632 clusters found so far
Starting parsing of faction='["faction_Piltover_Name"]'
        Parsing 8 decks
############################################################
2639 clusters found so far
Starting parsing of faction='["faction_Bard_Name", "faction_Jhin_Name"]'
        Parsing 1 decks
############################################################
2640 clusters found so far
Starting parsing of faction='["faction_Freljord_Name", "faction_Jhin_Name"]'
        Parsing 2 decks
############################################################
2642 clusters found so far
Starting parsing of faction='["faction_Jhin_Name", "faction_ShadowIsles_Name"]'
        Parsing 2 decks
```

Results

Parameters

| MAX_DISTANCE | CUTOFF | ARCHETYPES | | 14 | 1% | 4167 | | 14 | 5% | 4018| | 16 | 5% | 3247| | 18 | 5% | 2642| | 20 | 5% | |