

Subreddit Comment Gathering

April 6, 2022

1 Setup

```
[ ]: #!pip install gensim
```

```
[ ]: #!pip install python-Levenshtein
```

```
[ ]: #!pip install praw
```

```
[ ]: #!pip install p
```

```
[ ]: #!pip install psaw
```

1.0.1 Imports

```
[2]: import gensim  
import pandas as pd  
import praw  
from psaw import PushshiftAPI  
import datetime as dt  
import matplotlib.pyplot as plt
```

```
[3]: p_reddit = praw.Reddit(client_id='[REDACTED]',  
                           client_secret='[REDACTED]',  
                           username='[REDACTED]',  
                           password='[REDACTED]',  
                           user_agent='[REDACTED]')  
  
p_subreddit = p_reddit.subreddit('leagueoflegends')
```

```
[ ]: #!dir
```

```
[4]: master_table = pd.read_csv('june_to_march_postings.csv', index_col=[0])
```

```
[5]: for i in master_table.columns:  
    print(i)
```

```
created_utc
id
title
url
created
d_
datetime
```

```
[ ]: master_table['created_utc'].iloc[:5]
```

1.0.2 Minor cleanup

```
[ ]: master_table['datetime'] = pd.to_datetime(master_table['created_utc'], unit='s')
```

```
[ ]: master_table = master_table.drop(columns=['Unnamed: 0'])
```

```
[ ]: master_table.to_csv('june_to_march_postings.csv')
```

1.0.3 Data Exploring

```
[ ]: import ast
```

The following is the dictionary components of d_ in the datatable

```
[ ]: for k,v in ast.literal_eval(master_table.iloc[0]['d_']).items():
      print("%-15s, %s" %(k, v))
```

```
[ ]: master_table
```

```
[6]: first_row = master_table.iloc[0]
      submission = p_reddit.submission(id=first_row['id'])
```

```
[7]: # printing just c will give us the comment id for this submission
      for c in submission.comments:
          print(c)
```

```
h1si8vd
h1sil8e
h1sigkk
h1sttyw
h1sqn3p
h1snuvb
h1sqyvc
h1srcys
h1uktmn
h1sr26v
```

```
[8]: # printing c.body gets us the text from the comments
for c in submission.comments:
    print(c.body)
```

Roam by having the wave pushed to enemy turret . I only gank if bot or top is being pushed to our turrets. Make picks before objectives to win easier. You have to come back to your waves not just mid but it prevents turrets getting taken . Keep the lanes split
The first step to roaming is having good wave control
Try not to roam when you think you won't get anything out of it.

When I play mid, the only times I try to roam is when I see a fight in river, or my side of the jungle. Or when I am 100% sure I can get a kill or 2 in another lane.

Don't roam if your unsure about the results.

But this also depends on the champion your playing. If your playing twisted fate or an assassin. One who's job is to roam and find picks. You should be taking every opprotunity you can.

Also obviously shove the wave into their tower first. So that gets the minion kills and the enemy laner loses out on the gold.

Roam less. farm more

1 roam kill is one missed wave. One missed wave is a free assist for your opponent. The longer you are not on your lane the more chances to comeback you give your opponent. Especially if you have streak on yourself and then you accidentally die on a roam. That will definitely throw you far away back
Perma roam can work in specific scenarios but always remember for every wave you miss, you lose both exp and gold(every wave is about 120 gold). Even if you get a kill off your roam, think about how many waves you lost for it. if your other laners are orangutans, don't bother roaming at all.

Generally perma farm can actually win you a lot of games in low elo.

I usually getting fed by CSing. I'm a toplaner and always getting 150 or more gold reward even though I'm just 0/0/0. That usually happens if the whole your team is ahead.

become a kata main

Well you said it yourself. Go farm, don't perma roam, most of the times you're just wasting time running around river doing nothing. They won't run into you like co-op vs AI.

Suppress your lane with your lead, deny their farm either by freezing or shoving and harassing. Play with jungler. Every second you're not CSing those free gold are lost. Don't lose them unless you're sure you're getting something better.

If you get three or four early kills and keep up the CS/min you're gonna just be

really strong in mid game fights say around dragon that'll allow you to naturally carry the game.
so any solution to my problem

Not sure if you are serious. You literally described what you do wrong and already mentionend the solution: cs

1.0.4 Understanding our Indices

```
[9]: shape_0 = master_table.shape[0]
```

```
[10]: import math
interval = math.floor(shape_0/5)
```

```
[11]: indices = list()
i = 0
for j in range(5):
    print(i)
    indices.append(i)
    i += interval

indices.append(shape_0)
print(indices)
```

```
0
28713
57426
86139
114852
[0, 28713, 57426, 86139, 114852, 143567]
```

```
[13]: # An example of how our code is going to be split/indexed
# We'll recombine these collections later

for i in range(3):
    sub_id = master_table.iloc[i]['id']

    print(sub_id)
    temp_submission = p_reddit.submission(id=sub_id)

    for comment in temp_submission.comments:
        print(comment, comment.body)
```

o00m44

h1si8vd Roam by having the wave pushed to enemy turret . I only gank if bot or top is being pushed to our turrets. Make picks before objectives to win easier. You have to come back to your waves not just mid but it prevents turrets getting taken . Keep the lanes split

h1sil8e The first step to roaming is having good wave control
h1sigkk Try not to roam when you think you won't get anything out of it.

When I play mid, the only times I try to roam is when I see a fight in river, or my side of the jungle. Or when I am 100% sure I can get a kill or 2 in another lane.

Don't roam if your unsure about the results.

But this also depends on the champion your playing. If your playing twisted fate or an assassin. One who's job is to roam and find picks. You should be taking every opprotunity you can.

Also obviously shove the wave into their tower first. So that gets the minion kills and the enemy laner loses out on the gold.

h1sttyw Roam less. farm more

h1sqn3p 1 roam kill is one missed wave. One missed wave is a free assist for your opponent. The longer you are not on your lane the more chances to comeback you give your opponent. Especially if you have streak on yourself and then you accidentally die on a roam. That will definitely trow you far away back

h1snuvb Perma roam can work in specific scenarios but always remember for every wave you miss, you lose both exp and gold(every wave is about 120 gold). Even if you get a kill off your roam, think about how many waves you lost for it. if your other laners are orangutans, don't bother roaming at all.

Generally perma farm can actually win you a lot of games in low elo.

h1sqyvc I usually getting fed by CSing. I'm a toplaner and always getting 150 or more gold reward even though I'm just 0/0/0. That usually happens if the whole your team is ahead.

h1srcys become a kata main

h1uktmn Well you said it yourself. Go farm, don't perma roam, most of the times you're just wasting time running around river doing nothing. They won't run into you like co-op vs AI.

Suppress your lane with your lead, deny their farm either by freezing or shoving and harassing. Play with jungler. Every second you're not CSing those free gold are lost. Don't lose them unless you're sure you're getting something better.

If you get three or four early kills and keep up the CS/min you're gonna just be really strong in mid game fights say around dragon that'll allow you to naturally carry the game.

h1sr26v so any solution to my problem

Not sure if you are serious. You literally described what you do wrong and already mentionend the solution: cs

o00k4w

o00jk8

2 Comment Gathering

THIS COMMENT BLOCK IS STRICTLY FOR MY PROJECT PARTNERS TO RUN AND GATHER COMMENTS

READ THIS: If you're just opening this notebook, you can run the code from here. **Run these next two cells before running your own respective cell** (*indicated below with headers of your names*).

Additionally, please fill in the `praw.Reddit(...)` where it says `YOUR_CREDS_HERE` with your reddit api information to avoid potential issues with ratelimiting (though when I tried to gather it all myself, I didn't seem to run into any ratelimiting issues). Just in case there is, expect the code to run between 4-8 hours. If not, then maybe 2 hours... or less. Thank you!

```
[5]: import pandas as pd
import praw
import math
from tqdm import tqdm

"""p_reddit = praw.Reddit(client_id=YOUR_CREDS_HERE,
                          client_secret=YOUR_CREDS_HERE,
                          username=YOUR_CREDS_HERE,
                          password=YOUR_CREDS_HERE,
                          user_agent=YOUR_CREDS_HERE)

p_subreddit = p_reddit.subreddit('leagueoflegends')"""

master_table = pd.read_csv('june_to_march_postings.csv', index_col=[0])

shape_0 = master_table.shape[0]
interval = math.floor(shape_0/5)

indices = list()
i = 0
for j in range(5):
    indices.append(i)
    i += interval
indices.append(shape_0)
```

```
[27]: def gather_comments(t, sbmssn, sid):
    for comment in sbmssn.comments:
        if type(comment) is praw.models.reddit.comment.Comment:
            t['submission_id'].append(sid)
            t['comment_id'].append(comment)
            t['comment_text'].append(comment.body)
```

2.0.1 Baolong 1

```
[ ]: b1_comments_table_dict = {  
    'submission_id' : list(),  
    'comment_id' : list(),  
    'comment_text' : list()  
}
```

```
[ ]: for i in tqdm(range(indices[0],indices[1])):  
    sub_id = master_table.iloc[i]['id']  
  
    temp_submission = p_reddit.submission(id=sub_id)  
  
    gather_comments(b1_comments_table_dict,  
                    temp_submission,  
                    sub_id)
```

```
[ ]: b1_comments_dataframe = pd.from_dict(b1_comments_table_dict)
```

```
[ ]: b1_comments_dataframe.to_csv('comments_dataframe_1.csv')
```

2.0.2 Baolong 2

```
[38]: b2_comments_table_dict = {  
    'submission_id' : list(),  
    'comment_id' : list(),  
    'comment_text' : list()  
}
```

```
[39]: for i in tqdm(range(indices[1],indices[2])):  
    sub_id = master_table.iloc[i]['id']  
  
    temp_submission = p_reddit.submission(id=sub_id)  
  
    gather_comments(b2_comments_table_dict,  
                    temp_submission,  
                    sub_id)
```

```
100%|      | 28713/28713 [7:57:26<00:00, 1.00it/s]
```

```
[40]: b2_comments_dataframe = pd.DataFrame.from_dict(b2_comments_table_dict)
```

```
[41]: b2_comments_dataframe.to_csv('comments_dataframe_2.csv')
```

2.0.3 Mohammad

```
[ ]: m_comments_table_dict = {  
    'submission_id' : list(),  
    'comment_id' : list(),  
    'comment_text' : list()  
}
```

```
[ ]: for i in tqdm(range(indices[2],indices[3])):  
    sub_id = master_table.iloc[i]['id']  
  
    temp_submission = p_reddit.submission(id=sub_id)  
  
    gather_comments(m_comments_table_dict,  
                    temp_submission,  
                    sub_id)
```

```
[ ]: m_comments_dataframe = pd.from_dict(m_comments_table_dict)
```

```
[ ]: m_comments_dataframe.to_csv('comments_dataframe_3.csv')
```

2.0.4 Aleksa 1

```
[28]: a1_comments_table_dict = {  
    'submission_id' : list(),  
    'comment_id' : list(),  
    'comment_text' : list()  
}
```

```
[24]: #t = master_table.iloc[370:400]
```

```
[25]: #x = 'hello'  
      #print(type(x))
```

```
[26]: #for _id in t['id']:  
      #    t_s = p_reddit.submission(id=_id)  
      #    for c in t_s.comments:  
      #        print(type(c) is praw.models.reddit.comment.Comment)
```

```
[29]: for i in tqdm(range(indices[3],indices[4])):  
    sub_id = master_table.iloc[i]['id']  
  
    temp_submission = p_reddit.submission(id=sub_id)  
  
    gather_comments(a1_comments_table_dict,  
                    temp_submission,  
                    sub_id)
```


100%| | 28713/28713 [7:55:55<00:00, 1.01it/s]

```
[31]: a1_comments_dataframe = pd.DataFrame(a1_comments_table_dict)
```

```
[32]: a1_comments_dataframe
```

```
[32]:      submission_id comment_id \
0          qwart4      hl1locor
1          qwar5c      hl1o8mx
2          qwaqso      hl1p6r4
3          qwaqso      hl3bo65
4          qwaqso      hl4bm4k
...
165809      s5gxcg      hszbkip2
165810      s5gxcg      hszud12
165811      s5gvxi      hsxkd1j
165812      s5gvbx      hsx9imu
165813      s5gu40      hsx9a7p

                                     comment_text
0      This submission has been removed because [text...
1      This submission has been removed because [text...
2      The worst part is that the bounties don't go a...
3      The reason it's not a single objective is beca...
4      Because then a 5v5 team fight would break out ...
...
165809                                     Leona is absurd
165810                                     Rek'sai
165811  Ap twitch doesn't liandries or demonic he goes...
165812  This submission has been removed because [text...
165813  This submission has been removed because [text...

[165814 rows x 3 columns]
```

```
[33]: a1_comments_dataframe.to_csv('comments_dataframe_4.csv')
```

2.0.5 Aleksa 2

```
[34]: a2_comments_table_dict = {
      'submission_id' : list(),
      'comment_id' : list(),
      'comment_text' : list()
    }
```

```
[35]: for i in tqdm(range(indices[4],indices[5])):
      sub_id = master_table.iloc[i]['id']
```

```
temp_submission = p_reddit.submission(id=sub_id)

gather_comments(a2_comments_table_dict,
                temp_submission,
                sub_id)
```

```
100%|          | 28715/28715 [7:54:15<00:00, 1.01it/s]
```

```
[36]: a2_comments_dataframe = pd.DataFrame.from_dict(a2_comments_table_dict)
```

```
[37]: a2_comments_dataframe.to_csv('comments_dataframe_5.csv')
```

3 Final Comment

Each dataframe is about **30 MB**, and contains about **160,000 rows** (*or **comments***) that will be used in the implementation of the **Word2Vec** model which will serve in our binary classification machine learning model. As of today, April 6, 2022, we're collecting the final dataframe and will move forward to finalizing the Word2Vec development in the coming days.