CSC494 - Ranking Twitter Users in the Core Detection Algorithm

Anthony Duong

September 2022 - December 2022

Contents

1 Introduction

This report discusses the exploration of the ranking functions used in the Core Detection Algorithm during the period of September 2022 to December 2022. A few ranking functions were developed and explored including the Influence One, Influence Two, and Intersection Ranking functions. At the end of the period, an improvement in the results of the Core Detection Algorithm was seen using the Intersection Ranking function.

1.1 Background

The Ranking Function is used by the Core Detection Algorithm. Here is the abstract pseudo-code for the Core Detection Algorithm:

Algorithm 1 The Core Detection Algorithm

```
procedure DetectCore(initialUser)
    neighbourhood ← local neighbourhood of initialUser
    clusters ← list of clusters from neighbourhood
    chosenCluster ← arbitrary cluster from clusters
    ranking ← Ranking Function(chosenCluster)
    prevUser ← initialUser
    currUser ← top user of ranking
    top10 ← top 10 users of ranking
    while currUser is not prevUser do
        neighbourhood ← local neighbourhood of currUser
        clusters ← list of clusters from neighbourhood
        chosenCluster ← cluster that is most similar to top10 from clusters
        ranking ← Ranking Function(chosenCluster)
        prevUser ← currUser
        currUser ← top user of ranking
        top10 ← top 10 users of ranking
    end while
    return top10
end procedure
```

As you can see, the Ranking Function is used after a cluster is chosen from the local neighbourhood of a Twitter user. The Ranking function then takes as input this chosen cluster of users and ranks the users relative to each other. The top-ranked user determines the local neighbourhood for the next iteration of the algorithm ($currUser$). In the final iteration, the top ten users is the output of the Core Detection Algorithm. The output of the Core Detection Algorithm is then input to the Core Expansion Algorithm, which its purpose is to find and output all members of a community based on the ten users given.

1.2 Existing Ranking Functions

Prior to September of 2022, two ranking functions existed: the Production Ranking function and the Consumption Ranking function. The two ranking functions assign a score to each user in the cluster and ranks users relative to their score. For both of these functions, a user is higher ranked if they have a higher score. The Production Ranking function scores each user with the count of retweets that their tweets get within the cluster. The Consumption Ranking function scores each user with the count of their retweets within the cluster. It was determined that the two ranking functions on their own did not sufficiently capture the idea of a "core user," which is why other ranking functions were explored.

2 Explored Ranking Functions

In this section, we discuss the ranking functions that were explored during the period of September 2022 to December 2022.

2.1 Influence One

The ranking function known as "Influence One" scores each user with (approximately speaking) their average number of retweets that they get for a tweet. The scoring function is more precisely defined by the algorithm below:

Algorithm 2 Influence One's scoring function

```
1:  procedure ScoreUser(user)
2:      score ← 0
3:      for directFollower of user do
4:          scoreFromFollower ← 0
5:          for tweet of user do
6:              if RETWEETED(tweet, directFollower) then      ▷ RETWEETED procedure is defined below
7:                  scoreFromFollower ← scoreFromFollower + 1
8:              end if
9:          end for
10:         score ← score + scoreFromFollower
11:     end for
12:     numUserTweets ← number of tweets from user
13:     score ← score/numUserTweets
14:     return score
15: end procedure
16: procedure Retweeted(tweet, user)
17:     if tweet is an original tweet then
18:         return user retweeted tweet
19:     else                                                  ▷ tweet is a retweeted tweet
20:         originalTweet ← original of tweet
21:         return user retweeted originalTweet at a later time than tweet
22:     end if
23: end procedure
```

A user is higher ranked by the Influence One ranking function if they are assigned a larger score.

It is important to note line 21 here. First, let us define "UserA" as the user of *tweet* and "UserB" as *user*. If UserB retweets *originalTweet* later than UserA, then it is not necessarily true that UserB retweeted the original **because** of UserA, i.e., UserB saw that UserA retweeted *originalTweet*, which caused UserB to retweet it.

2.1.1 Explorations of Influence One

On line 3, only the direct followers (in the cluster) of the user are considered. At one point, all users in the cluster with a following path to the user were considered, but it was found that considering solely direct followers gave a very similar ranking of the top users. This is exemplified by Table 1 below.

Considering Following Path			Considering Direct Followers Only		
Rank	User	Score	Rank	User	Score
1	DavidDuvenaud	2.68	1	DavidDuvenaud	2.58
2	TimSalimans	2.58	2	Luke_Metz	2.48
3	Luke_Metz	2.56	3	dpkingma	2.35
4	dpkingma	2.35	4	TimSalimans	2.33
5	RickyTQChen	2.26	5	YSongStanford	2.01
6	YSongStanford	2.13	6	RickyTQChen	1.87
7	huangcza	1.62	7	huangcza	1.56
8	dustinvtran	1.61	8	balajiln	1.53
9	balajiln	1.55	9	dustinvtran	1.51
10	ArnaudDoucet1	1.48	10	ArnaudDoucet1	1.45

Table 1: The data considers one of *DavidDuvenaud*'s clusters (contains 82 users). On the left are the top ten users of the cluster based on their Influence One score where following paths are considered. On the right are the top ten users of the cluster based on their Influence One Score where solely direct followers are considered.

The impact of the later retweets of retweets was compared with the retweets of original tweets on the Influence One score was looked into. Meaning, we compared the frequency of line 21 returning *true* to line 18 returning *true*. The impact of the type varied based on user and cluster, thus it could not be generalized as to which one had a greater effect. For example, in the appendix, Table 7 shows that most users in that cluster received a higher contribution to the Influence One score from their original tweets, whereas Table 8 shows that most users in that cluster received a higher contribution from their retweets.

A modification of the Influence One scoring was explored where contributions from followers were weighted. More precisely, on line 10 of the algorithm, instead of $score \leftarrow score + scoreFromFollower$, it would be $score \leftarrow score + scoreFromFollower * weight$, where $weight$ was the number of followers that have retweeted $directFollower$ at least once. However, the resulting top 10 were not much different from the unmodified version. See Table 9 and Table 10 in the appendix for examples of these insignificant differences. In both examples, the top 10 list only differs by 2-3 users with most of the change coming from the bottom of the list.

Note in the modification above that the $weight$ variable is actually the "spread" of the $directFollower$. The concept of the "spread" of a user was used elsewhere during the exploration of the ranking functions. "Spread" of a user is defined as the number of followers that have retweeted that user at least once. Usually, the "spread" is measured on local data.

2.2 Influence Two

The ranking function known as "Influence Two" scores each user with the sum of all percentages of all direct follower's retweets which can be roughly credited to that user. The scoring function is more precisely defined by the algorithm below:

Algorithm 3 Influence Two's scoring function

```
1: procedure ScoreUser(user)
2:     score ← 0
3:     for directFollower of user do
4:         numRetweets ← 0
5:         for tweet of user do
6:             if RETWEETED(tweet, directFollower) then    ▷ RETWEETED procedure is as defined in "Influence One" section
7:                 numRetweets ← numRetweets + 1
8:             end if
9:         end for
10:        totalRetweets ← total number of retweets of directFollower
11:        scoreFromFollower ← numRetweets/totalRetweets
12:        score ← score + scoreFromFollower
13:    end for
14:    return score
15: end procedure
```

A user is higher ranked by the Influence Two ranking function if they are assigned a larger score.

A modification of the Influence Two scoring was explored where percentages in both directions are considered. To explain, the score of *user* is an accumulation of scores determined by direct followers' interactions with *user*'s tweets. The modification also takes into account the reverse direction, the *user*'s interaction with *directFollower*'s tweets. This would be computing lines 4 to 11 with *user* and *directFollower* in reverse positions. Then, the modification would only add the minimum between *scoreFromFollower* and the reverse computation. This modification comes from the possible idea that "core users" should have "strong" relationships with other community members; this could be captured by a frequent mutual retweeting of each other's tweets, which is what this modification is attempting to capture. The results of this modification are discussed in the Discussion section.

2.3 Intersection of Production, Consumption, Influence One, and Influence Two Ranking

The Intersection Ranking function ranks users by taking the minimum number, n, such that the user is in the top-n of the Production, the Consumption, the Influence One, and the Influence Two rankings. The lower minimum is higher ranked.

For example, if a user, UserA, is ranked 4th in Production, 5th in Consumption, 3rd in Influence One, and 2nd in Influence Two, then the user is in the top-5 of all four rankings. Now, take another hypothetical user, UserB, that is ranked 6th in Production, 4th in Consumption, 2nd in Influence One, and 3rd in Influence Two. Then, UserB is lower ranked than UserA because UserB is, at best, only in the top-6 of all four rankings.

The idea behind this Intersection Ranking is that a "core user" should have a high rank in all four of these rankings.

3 Discussion

During the period of September 2022 to December 2022, the Core Detection Algorithm was run many times with the different ranking functions discussed in the previous section using several different initial users. The results of a select few of these runs and their interpretations will be discussed in this section.

First, for a brief discussion of the initial users used. Several initial users were used to varying degrees of success to target specific communities.

- *timnitGebru*, *fchollet*, *david_madras*, and *brandondamos* were input as initial users to target the Machine Learning (ML) community. *fchollet*, *brandondamos* were initial users that led to the most relevant groups of users.
- *Karen_Chess1* and *TheImmortalGame* were input as initial users to target the Chess community.
- *SpceNrdOutdoors* and *Astro_Sandy* were input as initial users to target the Space community.

The above information can be used as a recommendation for selecting initial users for the Core Detection Algorithm.

3.1 Influence One Ranking Results

In the table below are the top 10 users of the final iteration when using the Influence One ranking function running with initial user *fchollet*.

Rank	User	Score	Tweet Count
1	DavidDuvenaud	1.26	23
2	TimSalimans	1.17	18
3	dpkingma	0.95	42
4	Luke_Metz	0.93	59
5	YSongStanford	0.87	47
6	RickyTQChen	0.68	63
7	dustinvtran	0.66	77
8	balajiln	0.65	71
9	baaadas	0.54	28
10	sedielem	0.48	164

Table 2: The final top 10 users of the Core Detection Algorithm when using the Influence One ranking function with initial user *fchollet*. In the rightmost column is the number of tweets that that user tweeted.

The top 10 users come from one of *DavidDuvenaud*'s clusters which contains 82 users. In this cluster, 25% of users have a tweet count less than 100.25, 50% of users have a tweet count less than 488, and 75% less than 1323. Based on this distribution, the users in Table 2 have a low tweet count compared to the rest of the cluster. This can be seen more clearly in Figure 1 below which plots each user's Influence One score with respect to their tweet count.

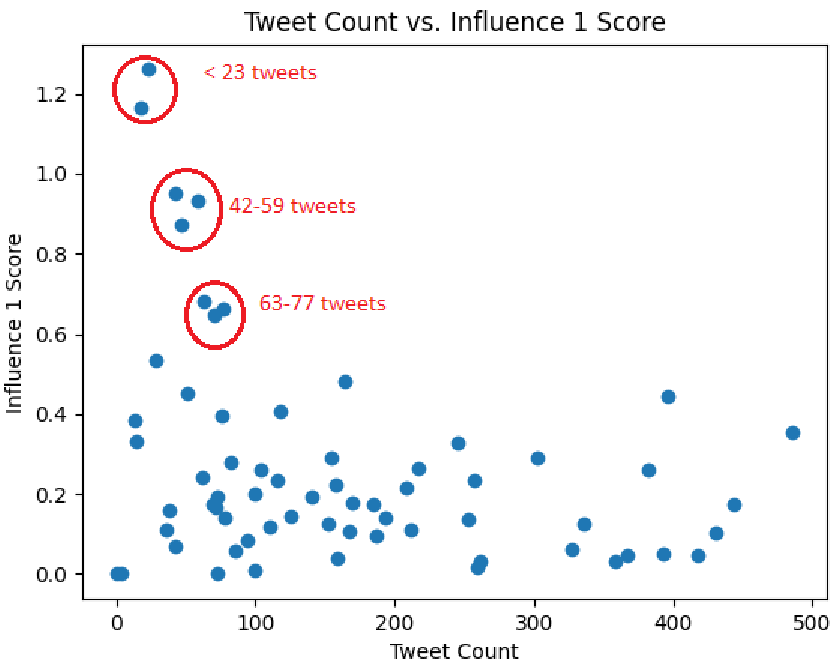

Figure 1: The data considers one of *DavidDuvenaud*'s clusters (contains 82 users). Here, the top 8 users have a tweet count less than 100. Note that users with tweet counts greater than 500 are not shown as they have a low Influence One score.

This trend can also be seen in one of *chesscom*'s clusters, which is visualized in Figure 2 in the appendix.

A possible interpretation of a user with a high Influence One score is that the user carefully produces a concentrated amount of refined content, which is justified from the low tweet counts seen. This is in contrast to a user that has a large volume of tweets/retweets.

3.2 Influence Two Ranking Results

In the table below are the top 10 users of the final iteration when using the Influence Two ranking function running with initial user *fchollet*.

Rank	User	Score	Follower Count
1	DeepMind	9.13	675030
2	NandoDF	5.96	92135
3	shakir_za	2.66	41101
4	hugo_larochelle	2.49	105495
5	weballergy	2.40	4923
6	_rockt	2.07	21189
7	MirowskiPiotr	1.95	5877
8	kchonyc	1.92	41638
9	FeryalMP	1.81	8346
10	arkitus	1.80	6489

Table 3: The final top 10 users of the Core Detection Algorithm when using the Influence Two ranking function with initial user *fchollet*. In the rightmost column is the total number of followers that user has on Twitter.

In the table below are the top 10 users of the final iteration when using the Influence Two ranking function running with initial user *Karen_Chess1*.

Rank	User	Score	Follower Count
1	chesscom	3.77	373801
2	STLChessClub	3.53	36001
3	fionchetta	2.94	16090
4	chess24com	2.62	142841
5	FIDE_chess	2.54	202161
6	ChesscomLive	2.09	62955
7	davidllada	2.08	31995
8	GrandChessTour	2.03	30598
9	itherocky	1.96	6921
10	ECUonline	1.88	15514

Table 4: The final top 10 users of the Core Detection Algorithm when using the Influence Two ranking function with initial user *Karen_Chess1*. In the rightmost column is the total number of followers that user has on Twitter.

From Table 3 and Table 4, we can see that the Influence Two ranking function favours users with a large number of followers (*DeepMind*, *hugo_larochelle*, *chesscom*, *FIDE_chess*).

The tendency of Influence Two Ranking to favour these large-number-of-followers users can be an issue as it is often the case that these users have a low Consumption score. In the Table 3, *DeepMind* has a Consumption score of just 3 in that cluster. In an attempt to solve this issue, the modification of Influence Two where both directions of interaction are considered (as discussed in Influence Two section) was explored. In the table below are the top 10 users of the final iteration when using this modified Influence Two ranking function running with initial user *Karen_Chess1*.

Rank	User	Score	Follower Count
1	chesscom	1.04	376100
2	WorldChessHOF	0.92	4180
3	STLChessClub	0.85	36215
4	QBoutiqueSTL	0.77	929
5	ChesscomLive	0.77	63141
6	ChampChessTour	0.69	19630
7	chess24com	0.67	143125
8	FIDE_chess	0.64	202455
9	itherocky	0.62	6921
10	USChessWomen	0.59	2754

Table 5: The final top 10 users of the Core Detection Algorithm when using the modified Influence Two ranking function with initial user *Karen_Chess1*. In the rightmost column is the total number of followers that user has on Twitter.

With the modification, the large-number-of-follower user *chesscom* was not eliminated when targeting the Chess community. A run with this modification was attempted with *fchollet* as the inital user, but the algorithm took too long to run and results were not obtained.

A stronger attempt to resolve the issue of users with a low Consumption score was made by introducing the Intersection Ranking.

3.3 Intersection Ranking Results

In the table below are the top 10 users of the final iteration when using the Intersection ranking function running with initial user *fchollet*.

Top	User
7	hugo_larochelle
17	LiamFedus
21	poolio
29	DaniloJRezende
29	jaschasd
30	marcgbellemare
36	sedielem
37	ZoubinGhahrama1
40	MarlosCMachado
45	huangcza

Table 6: The final top 10 users of the Core Detection Algorithm when using the Intersection ranking function with initial user *fchollet*. In the left column is the number for which the user is in the top-n of the Production, Consumption, Influence One, and Influence Two rankings. The cluster these top 10 came from was a size of 154.

The lowest ranking of the top 10 is 45 and the cluster is of size 154, this means that the Production, Consumption, Influence One, and Influence Two scores are all above the median of the cluster. The top 10 users having all scores above the median has been observed consistently; this is demonstrated by Table 12 and Table 13 in the appendix, which show final results with other initial users. Due to the Intersection Ranking enforcing high Production and Consumption scores relative to the cluster, it was an improvement over the individual Influence rankings.

The result for *fchollet* is also interesting because all users of the top ten have a direct connection to the Google project Deep Mind. This implies the top ten are a cohesive collection of users, meaning they are likely to all be a part of the same community. A signal that contradicts the above is the fact that some of the top 10 have a low "spread in core set," which can be seen in Table 11 of the appendix. The "spread in core set" value of a user is the number of other users on the table that both follow that user and have retweeted at least one of that user's tweets. A high "spread in core set" is supposed to indicate that a user is highly interactive with the other users.

3.3.1 Results of an Explored Modification

A modification of the Intersection Ranking was explored where Influence One was omitted, so that it was an intersection of Production, Consumption and Influence Two rankings. This was motivated by the possible idea that a user with a high Influence One score may not be a significant indicator that the user is a "core user."

The result of this modification, running it on initial user *fchollet*, is shown in the table below.

Top	User
3	_rockt
8	NandoDF
11	egrefen
14	hardmaru
15	ZoubinGhahrama1
15	sarahookr
18	kchonyc
20	JeffDean
24	hugo_larochelle
28	shakir_za

The cluster, which this top ten originated from, contained 161 users.

The appearance of user *JeffDean* in the result is significant here because of the following. There is the proposition that many communities are a part of a hierarchy of communities. For example, the Computer Science community at UofT is a part of the larger Arts & Science community at UofT, which is a part of the larger University of Toronto community. Assuming this proposition, then there are users that have significant interactions with both the more specific community and the larger community.

JeffDean is considered to be one of these users that is on the edge of two communities. This may be an issue for the Core Expansion Algorithm as, by including *JeffDean* in the initial input, it may include users from both communities rather than only the specific community.

4 Possible Future Work

Although improvements have been made to the Core Detection Algorithm via exploring different ranking functions, the algorithm can be improved to deliver more accurate and consistent results. The following are some proposed ideas (based on limited evidence) for how the improvement can be made.

In the Influence rankings, given two users: UserA and UserB, UserA is given credit for UserB's retweet, if UserA is followed by UserB and UserA retweeted the original tweet before UserB did (this is line 21 of Algorithm 2). This can be improved by trying to find a more causal link for retweets of retweets. If this is not possible, removing credit for retweets of retweets should be considered because this may be biasing the ranking functions towards users that retweet earlier, which is not a strong indicator of whether a user is a "core user."

If the "spread in core set" is determined to be a reliable way to evaluate the quality of a top-ten, then we can select users in a cluster in such a way that maximizes the "spread in core set" of the top 10 users in the final iteration.

It could help to determine characteristics of a "core user" and to create a ranking function that would capture that characteristic. Adding this ranking function to the Intersection Ranking can make the Intersection ranking function more selective.

Currently, the results of the Core Detection Algorithm vary quite a bit when targeting the ML community. It could be that creating local neighbourhoods solely on one user introduces a large amount of variation because the final output also comes from the local neighbourhood of one user. Since the local neighbourhood of a user is determined by their following list, the final top-ten all must be followed by a single user. Thus, by taking into account multiple user's following list, relevant people will be less likely to be excluded from the final iteration. This is how it could be beneficial to create local neighbourhoods based on multiple users instead.

5 Conclusion

We have introduced three ranking functions: Influence One, Influence Two, and Intersection. Various properties of Influence One have been explored and it was found that it favours users with a low tweet count. Influence Two was shown to favour accounts with a large number of followers. The Intersection Ranking function showed to output the most cohesive top-ten, thus making it an improvement over any individual ranking function discussed in this report.

6 Appendix

6.1 Influence One

User	Original Tweet Score	Retweet Score
DavidDuvenaud	**1.50**	1.08
Luke_Metz	**1.73**	0.75
dpkingma	0.75	**1.60**
TimSalimans	1.17	1.17
YSongStanford	**1.31**	0.71
RickyTQChen	**1.30**	0.57
huangcza	**1.50**	0.06
balajiln	**1.12**	0.40
dustinvtran	0.47	**1.04**
ArnaudDoucet1	**1.23**	0.22
SingularMattrix	**1.00**	0.44
brandondamos	**0.91**	0.20
baaadas	**0.77**	0.33
hugo_larochelle	**0.71**	0.33
sedielem	0.33	**0.58**
ryan_p_adams	**0.68**	0.20
RogerGrosse	**0.50**	0.37
jaschasd	**0.58**	0.26
DaniloJRezende	0.28	**0.29**
Yuhu_ai_	**0.47**	0.10

Table 7: The data considers one of *DavidDuvenaud*'s clusters (contains 82 users). Here, the top 20 users of the cluster are shown and 15 of the 20 have a higher contribution from the original tweets than the retweets. Note that both scores were computed by dividing by the total number of tweets from that user which includes both tweets and retweets.

User	Original Tweet Score	Retweet Score
reda_getachew	**17.25**	13.78
kibrom30	**15.99**	3.16
tigistAA	4.42	**9.30**
zeaxumawit	4.58	**8.04**
hayet_alem	3.11	**9.46**
SteezyFlexTM	2.37	**8.64**
TeklehaymanotG	3.22	**7.34**
FAmdeslasie	**8.24**	2.19
OmnaTigray	**6.15**	3.77
IrobAnina	3.37	**6.21**
Rimey_tig	3.86	**5.47**
SeifGebre	2.32	**6.38**
Yohannes_T_M	1.45	**6.89**
wedi_atse	0.59	**7.41**
HeranTigray	2.38	**4.97**
RAbdiAnalyst	**5.51**	1.76
FitwiDesta	2.50	**4.78**
Fthawi_Hurui	0.60	**6.47**
Yonigussie	1.06	**5.93**
Meron_Geb	2.39	**4.35**
selamina_a	0.60	**6.13**

Table 8: The data considers one of *tigistAA*'s clusters (contains 299 users). Here, the top 20 users of the cluster are shown and 16 of the 20 have a higher contribution from the retweets than the original tweets. Note that both scores were computed by dividing by the total number of tweets from that user which includes both tweets and retweets.

Normal Influence One			Weighted by Followers that Retweet		
Rank	User	Score	Rank	User	Score
1	DavidDuvenaud	1.26	1	DavidDuvenaud	34.48
2	TimSalimans	1.17	2	TimSalimans	27.00
3	dpkingma	0.95	3	Luke_Metz	24.80
4	Luke_Metz	0.93	4	dpkingma	22.38
5	YSongStanford	0.87	5	RickyTQChen	18.63
6	RickyTQChen	0.68	6	YSongStanford	15.89
7	dustinvtran	0.66	7	dustinvtran	15.47
8	balajiln	0.65	8	balajiln	15.08
9	baaadas	0.54	9	RogerGrosse	11.65
10	sedielem	0.48	10	SingularMattrix	11.59

Table 9: The data considers one of *DavidDuvenaud*'s clusters (contains 82 users). On the left are the top ten users of the cluster based on their Influence One score. On the right are the top ten users of the cluster based on their modified Influence One Score where the contribution from a follower is weighted by that follower's number of followers that have retweeted at least once.

Normal Influence One			Weighted by Followers that Retweet		
Rank	User	Score	Rank	User	Score
1	jenniferyuuu	0.86	1	jenniferyuuu	48.71
2	AlirezaFirouzja	0.77	2	LennartOotes	45.22
3	LennartOotes	0.73	3	AlirezaFirouzja	43.06
4	WorldChessHOF	0.62	4	chesscom	30.71
5	chesscom	0.59	5	GMJuditPolgar	28.50
6	Eljanov	0.54	6	Eljanov	27.75
7	Chesser_22	0.54	7	tatasteelchess	26.62
8	johnchess	0.54	8	chessterra	26.00
9	GrandChessTour	0.53	9	Chesser_22	25.86
10	chessleinier	0.52	10	chessleinier	24.90

Table 10: The data considers one of *chesscom*'s clusters (contains 228 users). On the left are the top ten users of the cluster based on their Influence One score. On the right are the top ten users of the cluster based on their modified Influence One Score where the contribution from a follower is weighted by that follower's number of followers that have retweeted at least once. Note that this was not a final result of the Core Detection Algorithm.

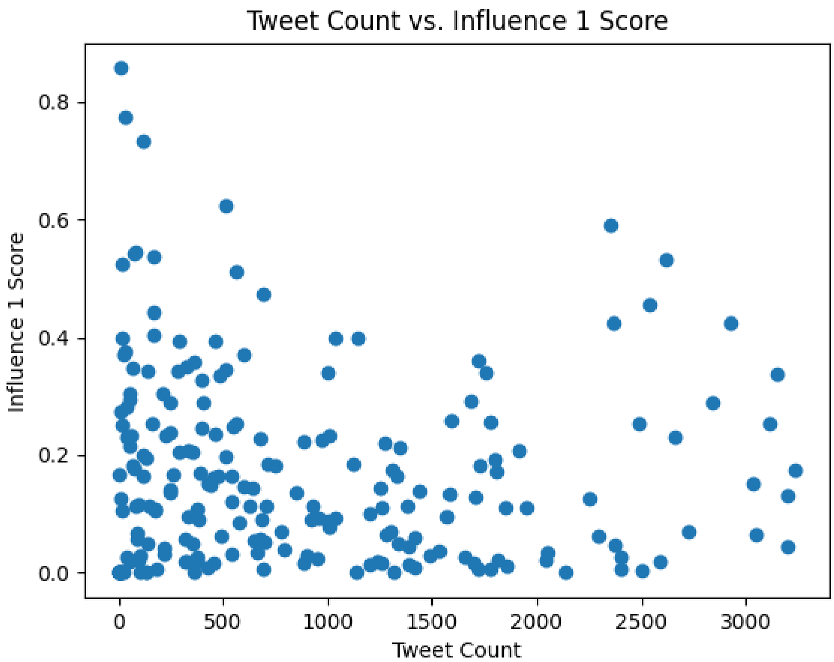

Figure 2: The data considers one of *chesscom*'s clusters (contains 228 users). The tweet count of each user is plotted by their Influence One score.

6.2 Intersection Ranking

Top	User	Followers	"Spread in Core Set"
7	hugo_larochelle	105448	9
17	LiamFedus	6213	6
21	poolio	14430	7
29	DaniloJRezende	35067	3
29	jaschasd	8069	4
30	marcgbellemare	9415	5
36	sedielem	41561	5
37	ZoubinGhahrama1	16025	7
40	MarlosCMachado	6004	5
45	huangcza	3514	2

Table 11: The final top 10 users of the Core Detection Algorithm when using the Intersection ranking function with initial user *fchollet*. The cluster these top 10 came from was a size of 154. In the left column is the number for which the user is in the top-n of the Production, Consumption, Influence One, and Influence Two rankings. The column labeled "Spread in Core Set" is the number of users on the table that both follow and have retweeted that user at least once.

Top	User
4	NASAKennedy
6	NASA
11	NASAArtemis
15	NASA_Johnson
22	NASAAmes
22	NASA_SLS
26	NASAEarth
27	Commercial_Crew
29	NASAGoddard
30	NASA_Marshall
30	NASAWebb

Table 12: The final top 10 users of the Core Detection Algorithm when using the Intersection ranking function with initial user *Astro_Sandy*. In the left column is the number for which the user is in the top-n of the Production, Consumption, Influence One, and Influence Two rankings. The cluster these top 10 came from was a size of 143.

Top	User
7	chesscom
17	FIDE_chess
18	STLChessClub
19	chesscom_in
26	GrandChessTour
28	fionchetta
30	chess24com
32	WorldChessHOF
33	TarjeiJS
38	aicfchess

Table 13: The final top 10 users of the Core Detection Algorithm when using the Intersection ranking function with initial user *TheImmortalGame*. In the left column is the number for which the user is in the top-n of the Production, Consumption, Influence One, and Influence Two rankings. The cluster these top 10 came from was a size of 232.