Who’s in the Gang? Revealing Coordinating Communities in Social Media

Derek Weber¹,²* and Frank Neumann¹
¹School of Computer Science, University of Adelaide, Australia
²Defence Science and Technology Group, Adelaide, Australia
*derk.weber@adelaide.edu.au, dst.defence.gov.au

Abstract—Political astroturfing and organised trolling are online malicious behaviours with significant real-world effects. Common approaches examining these phenomena focus on broad campaigns rather than the small groups responsible. To reveal networks of cooperating accounts, we propose a novel temporal window approach that relies on account interactions and metadata alone. It detects groups of accounts engaging in behaviours that, in concert, execute different goal-based strategies, which we describe. Our approach is validated against two relevant datasets with ground truth data. See https://github.com/weberdc/find_hccs for code and data.

I. INTRODUCTION

Modern information campaigns are participatory, exploiting online social networks (OSNs) by convincing users to become “unwitting agents” and relying on platform algorithms to spread desired narratives [1]. The use of political bots to influence the framing and discussion of issues in the mainstream media (MSM) remains prevalent [2]. Relevant research has focused on campaign detection [3], [4], identification of bots [5], and dissemination groups [6], but there is an increasing shift to look at how motivated actors coordinate their activities [7]–[9].

We present a new approach to detect groups engaging in potentially coordinated activities, revealed through anomalous levels of coincidental behaviour. Links in the groups are inferred from behaviours that, with intent, are used to execute a number of identifiable coordination strategies. We validate our new technique on various datasets and show it successfully identifies coordinating communities.

Our approach infers ties between accounts based on activity to construct latent connection networks (LCNs), in which highly coordinating communities (HCCs) are detected. We use a variant of focal structures analysis (FSA) [10] to do this, because of its focus on revealing influential sets of users rather than individuals. A window-based approach is used to enforce temporal constraints.

Comparison of two relevant datasets, including labeled ground truth, with a randomised dataset provides validation. These research questions guided our evaluation:

RQ1 How can HCCs be found in an LCN?
RQ2 How do the discovered communities differ?
RQ3 How consistent is the HCC messaging?
RQ4 Are the HCCs internally or externally focused?

Further discussion of related work, the approach taken and analyses are available in [11].

II. COORDINATION STRATEGIES

Online influence relies on two primary related mechanisms: dissemination and engagement. OSNs share a variety of features that enable these. Dissemination aims to maximise the reach of a message through repetition, targeting contentious social and political issues to cause outrage, or at least distract from other messaging. Using hashtags, reposting (e.g., Twitter retweets, Facebook shares, or Tumblr reposts), commenting on popular forums (e.g., Facebook Pages, subreddits, Twitter replies) or mentioning highly connected accounts are ways to disseminate a message. If the message is well crafted, it may entice others to spread it further, engaging with the content. Engagement is a subset of dissemination that explicitly solicits a response, targeting individuals or communities with rhetorical techniques that garner responses (e.g., making inflammatory comments, and sensationallly presented biased or false reporting). For greatest effect teams of accounts will coordinate their actions strategically and avoid direct interaction, so their links must be inferred.

A number of online coordination strategies have been observed in the literature making use of both dissemination and engagement, including:

1) Pollution: flooding a community with repeated or objectionable content, causing the OSN to shut it down [7];
2) Boost: reposting content to make it trend [4], [6];
3) Bully: groups of individuals harassing another individual or community [12]; and
4) Metadata Shuffling: groups of accounts changing metadata to hide their identities [13].

A clarification of the present problem is, therefore:

To identify groups of accounts whose behaviour, though typical in nature, is anomalous in degree.

III. METHODOLOGY

The major OSNs share a number of features, primarily in how they permit users to interact. By focusing on these commonalities, it is possible to develop approaches that generalise across the OSNs that offer them.

Traditional social network analysis relies on long-standing relationships between actors. On OSNs these are typically friend/follower relations. These are expensive to collect and
Algorithm 1: Extract HCCs (FSA_V)

Input: L = (V, E): An LCN, θ: HCC threshold
Output: H: Highly coordinating communities

1: E′ ← MergeMultiEdges(E)
2: g_mean ← MeanWeight(E′)
3: louvain_communities ← ApplyLouvain(L)
4: Create new list, H
5: for l ∈ louvain_communities do
6: Create new community candidate, h = (V_h, E_h)
7: Add heaviest edge e ∈ l to h
8: growing ← true
9: while growing do
10: Find heaviest edge e ∈ l connected to h not in h
11: if MeanWeight(E_h) > g_mean then
12: new_mean ← MeanWeight(Concatenate(E_h, e))
13: if new_mean < g_mean or
14: new_mean < (old_mean × θ) then
15: Add e to h
16: else
17: Add h to H
18: if MeanWeight(E_h) > g_mean then
19: Add h to H
20: end

quickly degrade in meaning if not followed with frequent activity. By focusing on active interactions, it is possible to understand not just who is interacting with whom, but to what degree. This provides a basis for constructing (or inferring) social networks, acknowledging they may be transitory.

LCNs are built from inferred links between accounts. Supporting criteria include retweeting the same tweet (co-retweet), using the same hashtags (co-hashtag) or URLs (co-URL), mentioning the same accounts (co-mention), or joining the same ‘conversation’ (a tree of reply chains with a common root tweet) (co-conv).

A. The LCN / HCC Pipeline

The key steps to extract HCCs from raw social media data are shown in Figure 1.

Step 1. Convert social media posts P to common interaction primitives, I_alt. This step removes extraneous data and provides an opportunity for the fusion of sources.

Step 2. From I_alt, filter the interactions, I_C, relevant to the set C = \{c_1, c_2, ..., c_q\} of criteria (e.g., co-mentions and co-hashtags). Illustrated in Figure 1b are the filtered mentions (in orange) and hashtag uses (in purple), ordered by timestamp.

Step 3. Infer links between accounts given C, ensuring links are typed by criterion (e.g., links between accounts in Figure 1c). The result, M, is a collection of inferred pairings. We use \( \beta_{I_{u,v}} \) for the number of inferred links between accounts u and v due to criterion c ∈ C.

Step 4. Construct an LCN, L, from the pairings in M. This network L = (V, E) is a set of vertices V representing accounts connected by weighted edges E of inferred links. These edges represent evidence of different criteria linking the adjacent vertices. The weight of each edge \( e_{I_{u,v}} \) ∈ E between vertices representing u and v for each criterion c is \( w^c_{I_{u,v}} \) (i.e., \( \beta_{I_{u,v}}^c \)).

To create a simple weighted network for easier processing, the multi-edges must be collapsed, however the edge weights are incomparable (e.g., retweeting the same tweet is not equivalent to using the same hashtag). For practical purposes, the edge weights can be combined with simple or weighted summation for simple community extraction, but implementations may maintain information of this step (e.g., the contribution of evidence for each criteria) for later analysis.

Some criteria may result in highly connected LCNs, even if its members never directly interact. The final step filters out these coincidental connections.

Step 5. Identify the highest coordinating communities, H, in L (Figure 1e), using a community detection algorithm appropriate to the scale of the data [14]. We propose FSA_V (Algorithm 1), a variant of FSA [10], which augments FSA’s focus on revealing influential sets of accounts with edge weights without the computational cost of recursion or a stitching phase, resulting in a simpler algorithm. FSA_V divides L into communities using the Louvain algorithm [15] and builds candidate HCCs within each, starting with the ‘heaviest’ (i.e., highest weight) edge (representing the most evidence of coordination). It then attaches the next heaviest edge until the candidate’s mean edge weight (MEW) is no less than \( \theta (0 < \theta \leq 1) \) of the previous candidate’s MEW, or is less than L’s overall MEW. In testing, edge weights appeared to follow a power law, so \( \theta \) was introduced to identify the point at which the edge weight drops significantly; \( \theta \) requires tuning. A final filter ensures no HCC with a MEW less than L’s is returned.

This algorithm prioritises edge weights while maintaining an awareness of the network topology by examining adjacent edges, something ignored by simple edge weight filtering. Our goal is to find sets of strongly coordinating users, so it is appropriate to prioritise strongly tied communities while still acknowledging coordination can also be achieved with weaker ties (e.g., 100 accounts paid to retweet a single tweet).

Using FSA_V, the complexity of the entire pipeline is low order polynomial, \( O(n^2) \), due primarily to the pairwise comparison of accounts to infer links in Step 3, which we constrain with temporal window size.

B. Addressing the Temporal Window Size

Temporal information is a key element of coordination, and thus is critical for effective coordination detection. Frequent posts within a short period may represent genuine discussion or deliberate attempts to game trend algorithms [8]. We treat the post stream as a series of discrete windows to constrain detection periods. An LCN is constructed from each window
(Step 4), and these are then aggregated and mined for HCCs (Step 5). As we assume posts arrive in order, their timestamp metadata can be used to sort and assign them to windows.

IV. EVALUATION AND VALIDATION

Our approach was evaluated by searching for Boost by cotweet and other strategies in two datasets, while varying window sizes (\(\gamma\)). FSA\(_V\) was compared against two other community detection algorithms when applied to the LCNs built in Step 4 (aggregated). We then validated the resulting HCCs through content, temporal and network analysis.

The two real-world Twitter datasets selected (Table I) represent two primary collection techniques: filtering a stream of posts using keywords direct from the OSN (DS1) and collecting the posts of specific accounts (DS2):

**DS1** Tweets relating to a regional Australian election in March 2018, including a ground truth subset (GT); and

**DS2** A large subset of the Internet Research Agency (IRA) dataset published by Twitter in October 2018\(^1\).

DS1 consists of tweets collected during a regional Australian election over an 18 days period in early 2018. Nine hashtags and 134 candidate and party Twitter handles were used as filter terms and reply and quote chains were retrieved post-collection. The candidates’ activity over the same period was collected separately, forming the ground truth. DS2 consists of the IRA tweets posted in 2016, the year of the period was collected separately, forming the ground truth. DS2 consists of tweets posted in 2016, the year of the US Presidential election, where evidence of coordination was expected to be greatest. The data were collected, held and analysed in accordance with University of Adelaide ethics protocol H-2018-045.

Window size \(\gamma\) was set at \(\{15, 60, 360, 1440\}\) (in minutes) and the three community detection methods used on the aggregated LCNs were:

- FSA\(_V\) (\(\theta=0.3\));
- \(k\) nearest neighbour (\(k\)NN) with \(k=ln(|V|)\) (cf. [4]);
- a simple threshold retaining the heaviest 90% of edges.

Values for \(\theta\) and the threshold were based on experimenting with values in \([0.1, 0.9]\), maximising the MEW to HCC size ratio, using the \(\gamma=15, 1440\) DS1 and DS2 LCNs.

### A. Results

1) **HCC Detection (RQ1): Detecting different strategies.**

The three detection methods all detected HCCs when searching for Boost (co-retweets), Pollute (co-hashtags), and Bully (co-mentions) (Table II). Notably, \(kNN\) consistently builds a single large HCC, highlighting the need to filter the network prior to applying it (cf. [4]). The \(kNN\) HCC is also consistently nearly as large as the original LCN for DS2, perhaps due to the low number of accounts and the fact that every edge of the retained vertices is retained, regardless of weight. It is not clear, then, that \(kNN\) is producing meaningful results.

**Varying window size.** Different strategies may be executed over different time periods, based on their aims. Boosting a message to game trending algorithms requires the messages to appear close in time, whereas some forms of Bullying exhibit only consistency and low variation (mentioning the same account repeatedly). Polluting a user’s timeline on Twitter can also be achieved by frequently joining their conversations over a sustained period. Varying \(\gamma\) searching for Boost, we found different accounts were prominent over different timeframes (Table III); the overlap in the accounts detected in each timeframe differed considerably even though the number of HCCs stayed relatively similar. HCC sizes seemed to follow a power law; most were very small but a few were large.

**HCC detection methods.** Similarly, HCCs discovered by the three community extraction methods (Table IV) exhibit large discrepancies, suggesting that whichever method is used, tuning is required to produce interpretable results. This is evident in the literature, e.g., Cao et al. conducted significant pre-processing when identifying URL sharing campaigns [4]. Here we present the variation in results while controlling methods and other variables and keeping the coordination strategy constant, as our focus is the method’s effectiveness.

2) **HCC Differentiation and Consistency (RQ2/RQ3):**

The HCC detection methods used relied on network information; in contrast we examine content, metadata and temporal information to validate the results. We contrast DS1 and DS2 results with GT (cf. [9]) and a RANDOM dataset (cf. [4]), constructed by randomly assigning non-HCC accounts from

---

**TABLE I**

<table>
<thead>
<tr>
<th>Dataset Statistics</th>
<th>Tweets (T)</th>
<th>Retweets (RT)</th>
<th>Accounts</th>
<th>T / Account / Day</th>
<th>RT / Account / Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1</td>
<td>115,913</td>
<td>63,164 (54.5%)</td>
<td>20,563</td>
<td>0.31</td>
<td>0.17</td>
</tr>
<tr>
<td>- GT</td>
<td>4,193</td>
<td>2,505 (59.7%)</td>
<td>134</td>
<td>1.74</td>
<td>1.04</td>
</tr>
<tr>
<td>DS2</td>
<td>1,571,245</td>
<td>729,937 (55.6%)</td>
<td>1,293</td>
<td>0.32</td>
<td>0.45</td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>HCCs by Coordination Strategy</th>
<th>Strategy</th>
<th>(\gamma)</th>
<th>Nodes</th>
<th>Edges</th>
<th>Comp.</th>
<th>Nodes</th>
<th>Edges</th>
<th>Comp.</th>
<th>Nodes</th>
<th>Edges</th>
<th>Comp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boost</td>
<td>15</td>
<td>44</td>
<td>112</td>
<td>5</td>
<td>8,855</td>
<td>60,762</td>
<td>419</td>
<td>855</td>
<td>23,022</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>Pollute</td>
<td>15</td>
<td>51</td>
<td>154</td>
<td>2</td>
<td>13,831</td>
<td>1,281,134</td>
<td>75</td>
<td>1,205</td>
<td>85,949</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Bully</td>
<td>40</td>
<td>70</td>
<td>482</td>
<td>1</td>
<td>16,519</td>
<td>1,925,487</td>
<td>222</td>
<td>1,103</td>
<td>37,168</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>GT</td>
<td>15</td>
<td>9</td>
<td>6</td>
<td>3</td>
<td>633</td>
<td>753</td>
<td>167</td>
<td>113</td>
<td>738</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>Pollute</td>
<td>15</td>
<td>9</td>
<td>5</td>
<td>4</td>
<td>135</td>
<td>93</td>
<td>50</td>
<td>24</td>
<td>15</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Bully</td>
<td>60</td>
<td>11</td>
<td>7</td>
<td>4</td>
<td>338</td>
<td>208</td>
<td>119</td>
<td>109</td>
<td>1,123</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>DS1</td>
<td>15</td>
<td>9</td>
<td>6</td>
<td>3</td>
<td>633</td>
<td>753</td>
<td>167</td>
<td>113</td>
<td>738</td>
<td>19</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE III**

<table>
<thead>
<tr>
<th>HCCs by Window Size (\gamma) (Boost, FSA(_V))</th>
<th>Strategy</th>
<th>(\gamma)</th>
<th>Nodes</th>
<th>Edges</th>
<th>HCCs</th>
<th>Min.</th>
<th>Max.</th>
<th>Nodes in common</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1</td>
<td>15</td>
<td>633</td>
<td>753</td>
<td>167</td>
<td>2</td>
<td>18</td>
<td>633</td>
<td>218 93</td>
</tr>
<tr>
<td>DS2</td>
<td>60</td>
<td>619</td>
<td>1,293</td>
<td>151</td>
<td>2</td>
<td>13</td>
<td>-169</td>
<td>208 193</td>
</tr>
<tr>
<td>DS2</td>
<td>360</td>
<td>503</td>
<td>1,119</td>
<td>127</td>
<td>2</td>
<td>19</td>
<td>-</td>
<td>503 350</td>
</tr>
<tr>
<td>DS2</td>
<td>1440</td>
<td>815</td>
<td>2,019</td>
<td>141</td>
<td>2</td>
<td>110</td>
<td>-</td>
<td>815</td>
</tr>
<tr>
<td>DS2</td>
<td>15</td>
<td>113</td>
<td>758</td>
<td>19</td>
<td>2</td>
<td>65</td>
<td>113</td>
<td>34</td>
</tr>
<tr>
<td>DS2</td>
<td>60</td>
<td>77</td>
<td>394</td>
<td>18</td>
<td>2</td>
<td>27</td>
<td>77</td>
<td>62</td>
</tr>
<tr>
<td>DS2</td>
<td>440</td>
<td>98</td>
<td>775</td>
<td>15</td>
<td>2</td>
<td>32</td>
<td>98</td>
<td>36</td>
</tr>
<tr>
<td>DS2</td>
<td>1440</td>
<td>69</td>
<td>380</td>
<td>15</td>
<td>2</td>
<td>25</td>
<td>69</td>
<td></td>
</tr>
</tbody>
</table>

\(^1\)https://about.twitter.com/en_us/values/elections-integrity.html
DS1 to groups matching the distribution of its HCCs (FSA_V, \(\gamma=15\)). As DS2 consisted entirely of bad actors, it was felt non-HCC accounts from DS1 would be more representative of non-coordinating 'normal' accounts.

**Internal consistency.** If HCCs are boosting a message, it is reasonable to assume the content of HCCs members will be more similar internally than when compared externally, to the content of non-members. Treating each HCC member's tweets as a single document, we created a doc-term matrix using 5 character n-grams for terms, and then compared the members' document vectors using cosine similarity. This approach was chosen for its performance with non-English corpora [16], and because using individual tweets as documents produced too sparse a matrix. Visualising the similarities between accounts, grouping them by HCC (Figure 2), the HCCs are discernible as being internally similar. This method ignores the number of tweets HCCs post, so we can draw no conclusions about connections between HCC size and the internal similarity of their content, though more active HCCs (i.e., with more tweets) are more likely to be similar, through co-occurrence of n-grams. The RANDOM groupings demonstrated little to no similarity, internal or external, as expected, while the DS2 HCCs demonstrated high internal similarity, as expected of organised accounts over an extended period.

**Messaging.** In GT and DS2, where the ratio of tweets to accounts is high, similarity of content among HCC members is clear: the large yellow square in the bottom left of Figure 2c shows an HCC where each member's contribution is highly similar to that of the others. The effect is less clear for DS1, given the high number HCCs found by FSA_V, but the speckled yellow areas indicate high similarity between some HCCs - these could be candidates for merging.

**Temporal patterns.** Campaign types exhibit different temporal patterns [3]. We used the same temporal averaging technique as Lee et al. [3] to compare the daily activities of the HCCs found in GT, DS1 and RANDOM (Figure 3a) and weekly activities in DS2 (Figure 3b). The GT accounts were clearly most active at two points prior to the election (around day 15), during the last leaders' debate and just prior to the mandatory electoral advertising blackout. DS1 and RANDOM HCCs were only consistently active at different times: around the day 3 leaders' debate and on election day, respectively. Inter-HCC variation may have dragged the mean activity value down, as many small HCCs were inactive each day. Reintroducing FSA's stitching element to FSA_V may avoid this. In DS2, HCC activity increased in the second half of 2016, culminating in a peak around the election, inflated by two very active HCCs, both of which used many predominantly benign hashtags over the year.

**Hashtag use.** The most frequent hashtags in the most active HCCs revealed the most in GT. It is possible to assign some HCCs to political parties via the partisan hashtags (e.g., #voteliberals), although the hashtags of contemporaneous cultural events are also prominent (Figure 4a). DS1 hashtags are all politically relevant, but are dominated by a single small HCC which used many hashtags very often (Figure 4b). These accounts clearly attempted to disseminate their tweets through using 1,621 hashtags in 354 tweets. Similarly, DS2 hashtags were dominated by a single HCC (using 41,317 relatively general hashtags in 40,992 tweets) and one issue-motivated HCC (Figure 4c). Given DS2 covers an entire year, it is unsurprising that the largest HCCs use such a variety of hashtags that their hashtags do not appear on the chart.

Analysing co-occurring hashtags can help further explore the HCC discussions to determine if HCCs are truly single groups or merged ones. Applied to GT HCC activities (Fig-

---

TABLE IV
HCCs BY DETECTION METHOD (Boost, \(\gamma=15\))

<table>
<thead>
<tr>
<th>Graph Attributes</th>
<th>HCC Sizes</th>
<th>Nodes in common</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSA_V</td>
<td>Min.</td>
<td>Max.</td>
</tr>
<tr>
<td>Nodes</td>
<td>Edges</td>
<td>HCCs</td>
</tr>
<tr>
<td>DS1</td>
<td>633</td>
<td>753</td>
</tr>
<tr>
<td>DS1 kNN Threshold</td>
<td>1,041</td>
<td>1</td>
</tr>
<tr>
<td>DS1 FSA_V</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>2</td>
<td>18</td>
</tr>
<tr>
<td>DS2</td>
<td>113</td>
<td>22,494</td>
</tr>
<tr>
<td>DS2 kNN Threshold</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DS2 FSA_V</td>
<td>85</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>31</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>113</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>675</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>675</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

---

Fig. 2. Similarity matrices of content posted by HCC accounts (\(\gamma=15, FSA_V\)). Each axis has an entry for each account, grouped by HCC. Each cell represents the similarity between the two corresponding accounts' content, calculated using cosine similarity (yellow = high similarity). Each account's content is represented as a vector of 5 character n-grams of all their tweets.

Fig. 3. Averaged temporal graphs of HCC activities (\(\gamma=15, FSA_V\)).
We have presented four coordination strategies and a pipeline-based approach to finding groups engaging in “orchestrated activities” [8], and successfully evaluated them on real-world datasets. Further HCC validation methods, strategy evolution, and real-time applications are to be explored.

REFERENCES