Identifying Post Types and Representative SNS User Types Based on Public Posting Activities and Profiles

Wenbo SU† Bin DU‡ and Mizuho IWAIHARA‡

†早稲田大学大学院情報生産システム研究科

‡808-0135 福 岡 県 北 九 州 市 若 松 区 ひ び き の 2-7-N251

E-mail: †suwenbo@toki.waseda.jp, ‡binduwaseda@akane.waseda.jp, ‡iwaihara@waseda.jp

Abstract Billions of users share their ideas through posting photos and texts on social network services (SNSs). They are interested in various topics, usually have different sentiment tendencies and posting activities. Identifying user types and characterizing user behaviors can help predict users' interests and improve recommendation systems. In this paper, we propose a clustering model to identify post types and representative user types. The clustering results are evaluated by clustering stability and quality. We also propose a cluster labeling algorithm to characterize significant features and name each user type automatically. In experiments, we collect users from several Facebook public groups, extract text posts and profile information from their public timelines as our dataset. In the post level, our post clustering model shows a good performance on clustering stability and quality, and five post types are identified. In the user level, clustering is executed with varying cluster numbers, and our cluster labeling algorithm is applied to extract important user types. The experiments show the dynamic changes of user types when we change the cluster number and data subsets. The results indicate that five user types are most representative. We also investigate user type distribution on different interest groups. We observe that the identified five representative user types are found in every interest group. However, the percentage of each user type is considerably different between interest groups.

Keyword Sentiment Analysis, Clustering Stability, User Profiling, Cluster labeling, K-means

1. Introduction

Social network services (SNSs) make it possible to connect people across political, economic, and geographic borders. They allow users to create a public profile and provide space for users to express their opinions, share contents, and upload photos/videos. In recent years, SNSs become a major platform for sharing users' ideas. For example, on Facebook, there are more than 1.32 billion active users, who upload about 300 million posts per day. In SNSs, users are interested in various topics, usually have different sentiment tendencies and posting activities. Identifying user types and characterizing user behaviors can help predict users' interests, detect depression and improve recommendation systems.

In this research, our objective is to discover user types from their sets of profile attributes and posting behaviors, which can be collected from their public timeline posts and profile pages on Facebook. we propose a clustering model to identify post types and representative user types. First, in the post level, we cluster the posts into different post types based on their common features, including subjectivity and polarity scores, length of posts and count of interactions to reveal users' posting behavior. Second, in the user level, we cluster the users into different user types based on features aggregated from their timeline posts as well as activity features extracted from profile pages.

In our clustering model, we apply K-means as the core clustering algorithm in our clustering model, because it runs fast on large dataset and is suitable for the post and user data distribution. The clustering results are evaluated by clustering stability and quality.

Characterizing significant features is important for distinguishing and describing the user types. However, clustering is unsupervised learning, there is no golden standard for feature selection or measuring the feature importance. To solve this problem, we propose a cluster labeling algorithm to characterize significant features and name each user type automatically.

In our experiment, we collect users from several Facebook public groups, extract text posts and profile information from their public timelines as our dataset. Our feature set captures emotional tendencies of users' posts, through sentiment analysis as well as posting activities. On the other hand, our feature set is independent from the semantics of interests or topics, such as keywords for interests, which enables us to compare user types over users of semantically unrelated interests, such as pet lovers and political enthusiasts.

In the post level, our post clustering model shows a good

performance on clustering stability and quality, and five post types are identified. In the user level, clustering is executed with varying cluster numbers and data subsets, and our cluster labeling algorithm is applied to extract important user types. Our experiments show the dynamic changes of user types when we change the cluster number and data subsets. The results indicate that five user types are most representative.

For the most representative five user types, we also examine how they are distributed along the interest groups. We observe that the percentage of each identified user type in different interest groups are quite different.

The rest of this paper is organized as follows. Section 2 explains the current research on SNS user analysis, clustering evaluation and cluster labeling. Section 3 describes our methods to collect and extract experiment datasets from Facebook. Section 4 explains about our clustering model, evaluation methods and cluster labeling algorithm. Section 5 presents identified post types, user types and evaluation results. Finally, Section 6 presents a conclusion and our future work.

2. Related Work

SNS user analysis is not a new research topic. In fact, several studies have been published on analyzing behaviors and profiles of users. Several studies focus on discovering features to predict user attributes, such as user income prediction [9], age and gender prediction [8]. While other studies investigate users' sentiment patterns. For example, Gutierrez [7] showed that Twitter users consistently stayed in one sentiment profile cluster, at least over a 30-day period. However, it is necessary to further investigate typical and/or stable sentiment clusters.

Applying clustering methods using users' features can detect potential cluster structure and identify different user types. However, how to evaluate the clustering result and select good models is not a trivial problem. The obvious reason is that, as opposed to supervised classification, there is no ground truth against which we could test our clustering results. Typical objective functions in clustering formalize the goal of attaining high intra-cluster similarity and low inter-cluster similarity. Based on the intra-cluster distances and the inter-cluster distances, a group of validity indices were proposed, such as the DB-index [5], Silhouette index [10] and Dunn-index [6].

In recent years， a new method to select good clustering models has become increasingly popular: Selecting the number of clusters based on clustering stability. Instead of defining "what is a clustering," the basic philosophy is simply that a clustering should be a structure on the data set that is "stable." Von Luxburg [11] gave a high-level overview about the existing literature on this new approach. He also concluded a rough scheme for model selection based on clustering stability. We adopt clustering stability in our scheme.

On the other hand, to interpret the clustering results, picking descriptive, human-readable labels for the clusters produced by a clustering algorithm is necessary. Most researches study cluster labeling in the fields of natural language processing and information retrieval. For example, Carmel et al. [4] investigated cluster labeling enhancement by utilizing Wikipedia. However, these researches studied cluster labeling mainly on text data and labels are chosen from occurring words. While our data is very different with text data. Instead, we propose a cluster labeling algorithm which combines Logistic Regression coefficients with clusters' centers to label each cluster.

3. Data Collection and Sentiment Analysis

The experimental data are collected from Facebook, which is currently one of the most popular SNS providers.

3.1 Selecting users from Facebook Interest groups

Users in our dataset are sampled from 15 public Facebook interest groups. These groups have more than one thousand members and focus on diverse interest topics, such as Business, Politics, Pets, Music, Sports, etc. We sampled active users, such that writing at least one post on the interest group timeline in the last one month, from these groups through Facebook Graph API. After dropping users who set their timeline pages or profile pages as private, we obtained 1611 users totally.

Table 1: Number of users from various interest groups

No.	Group Interest Field	Number of users
1	Books	73
2	Politics	59
3	Foods	125
4	University	52
5	Pets	86
6	Music	83
7	Business	158
8	Investment	142
9	Games	96
10	Programming	154
11	Education	73
12	Travel	141
13	Military	114
14	Sports	117
15	Superstar	138
Total		1611

For clustering users, we need to collect users' publicly available profile information and posting dataset. Since we have collected the list of active users of the public Facebook groups, we just need to develop a crawler that collects each user's timeline page and profile page.

We used Python and Selenium framework [1] to automate the Chrome browser to open a page, log into Facebook, access the user's timeline based on the user ID, scroll and save the page until the posts reach the year of 2016.

Since we collect only public posts, we do not need permission from users to access their data. However, we keep users' identities anonymous in this paper.

3.2 Extracting Text Posts and Analyzing Sentiment

We extract all text posts which were published in 2017 from users' timeline pages. After dropping non-English posts, we finally obtain 138,810 posts.

The sentiment analysis tool TextBlob [2] is utilized to analyze the sentiment tendencies of posts. TextBlob is a Python library to calculate sentiment score of text, producing a polarity and a subjectivity score for an English text.

The polarity score is a float within the range [-1.0, 1.0] where -1.0 is very negative and 1.0 is very positive. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective. Polarity score reveals how the expressed opinion of the post is inclined toward positive, negative or neutral. Subjectivity score can tell us the degree of how the text is objective or subjective. Sentiment can be evaluated based on the presence of affect words such as happy, sad, afraid, and bored.

4. Proposed Method

In this section, we explain the clustering model, evaluation methods and cluster labeling algorithm.

4.1 Clustering Model

1) Post features for clustering

From users' timeline page, we obtain 138,810 English text posts. Instead of detecting occurring keywords, we only observe scalar features of posts from three aspects: sentiment, text length, and interactions. Finally, we extract four features from each post, as shown in Table 2. Polarity score and subjectivity score indicate the sentiment tendency of the post. Text length indicates whether the post is relatively long or short. Average interaction indicates how many interactions occurred with other users on this post.

Table 2: Post features

Post Features	Description
Polarity score	Indicates how the words of the post are inclined toward positive, negative or neutral.
Subjectivity score	The degree of how the post is objective or subjective.
Text length	The character length of the post text.
Interaction count	The sum of the count of received likes, comments and shares.

2) User features for clustering

Certain user information such as the number of friends 8can be extracted from users' timeline pages and profile pages. These activity features indicate activeness level of users on various aspects. We extract two features: the friend count and post frequency. A user's friend count indicates the social network size of the user, while post frequency can reflect his/her activeness on SNS.

On the other hand, tendencies of posting of one user, such as sentiment, needs to be aggregated from the user's posts. We need to collect enough posts of one user and aggregate them into user-level scalar features, to compare his/her posting tendencies with other users. Then these features are combined with other user-level features, to find user types through the user-level clustering. We extract three features from user's posts: the average polarity score and the average subjectivity score, both show the user's sentiment tendency, while the average interaction reflects how influential the user's posts are.

Table 3: User features

Activity Features	Description
Average polarity score	The average polarity score of all posts one user published.
Average subjectivity score	The average subjectivity score of all posts one user published.
Average interaction	The average number of interactions of all posts one user published.
Number of friends	Indicates the size of the social graph the user has.
Post frequency	Reflect the user's activeness on SNS.

3) Feature preprocessing and normalization

Since the range of values of raw data varies widely, the feature functions need to be normalized before clustering. After removing outlier values, we apply log10 dumping to all the features except polarity score and subjectivity score. Then we normalize the attribute values by standard deviation. After normalization shown in (1) below, each feature variable is transformed into a standard normal with expected value 0 and variance 1 [3].

$$x_{new} = \frac{x - x_{mean}}{Deviation} \qquad (1)$$

We further transform all the feature values into the range [0.0, 1.0] by the sigmoid function shown in (2). Now all the feature values are in the same range and have approximately equal impact on the clustering algorithm.

$$S(x_{new}) = \frac{1}{1+e^{-x_{new}}} \qquad (2)$$

4) Clustering algorithm

In our clustering, we use the K-means algorithm with Euclidean distance to cluster the post and user data. K-means clustering is popular for cluster analysis, performing well on large datasets. By the K-means algorithm, we first cluster the posts into different post types based on the four post features in Table 2, including subjectivity and polarity scores, length of posts and number of interactions. Second, we cluster the users based on the five features presented in Table 3, to discover representative user types.

4.2 Evaluation Methods

Selecting cluster number K is important in discovering clustering of desirable properties. In our case, we evaluate clustering results on varying cluster numbers in two ways: comparing clustering instabilities, and comparing the DB-index scores.

1) Clustering stability.

We assign user types to clustering results, but the results could be unstable depending on sample datasets. In order to find universal characterization of user/post characterizations that are stable or common among various post or user subsets, we pay attention to clustering stability. Von Luxburg [11] discussed how to choose most stable K for K-means, under the circumstances where the dataset is slightly modified, by ways such as subsampling, and (random) reduction of dimensions. Clustering stability can be evaluated as follows [11]:

i. For $k = 2, \dots, k_{max}$

a) Subsample b_{max} sub-datasets S_b ($b = 1, \dots, b_{max}$) from the original dataset S.

b) Run K-means clustering on each sub dataset S_b, and obtain the cluster center C_b.

c) For $b, b' = 1, \dots, b_{max}$ ($b \neq b'$), merge each pair of sub-datasets S_b, $S_{b'}$ into S_B. First, assign points in S_B to the closest center point in C_b, and obtain result R_b. Second, assign points in S_B again to the closest center point in $C_{b'}$, and obtain result $R_{b'}$.

d) Compute Jaccard distance $d(R_b, R_{b'})$ on each pair of results R_b, $R_{b'}$.

e) Compute instability as the mean Jaccard distance between all pairs of results R_b, $R_{b'}$.

$$Instab(k) = \frac{1}{b_{max}} \sum_{b,b'=1}^{b_{max}} d(R_b, R_{b'}) \qquad (3)$$

ii. Choose the parameter K which gives the lowest instability.

2) DB-index

The DB-index (Davies–Bouldin index) is an internal evaluation scheme for clustering quality [5]. The DB-index by Euclidean distance is given by:

$$DBI = \frac{1}{k} \sum_{i=1}^{k} max \left(\frac{\overline{C_i} + \overline{C_j}}{\|w_i - w_j\|_2} \right) (i \neq j) \qquad (4)$$

Here, w_i and w_j are the centers of clusters i and j. $\overline{C_i}$ is the mean distance between the points in cluster i and the centroid of cluster i. $\overline{C_j}$ is the mean distance between the points in cluster j and the centroid of cluster j. The lower DB-index is, the better clusters are separated.

4.3 Cluster Labeling Algorithm

Characterizing clustering results is important for examining validity of clusters. We propose a cluster labeling algorithm which gives a label to each post or user type, based on significant features of each cluster. The algorithm consists of two steps: (1) Identifying the two most significant features in each cluster using logistic regression. (2) Labeling each cluster based on the cluster center values of the two identified significant features.

1) Identifying the most two significant features for each cluster

i. Let $\boldsymbol{C_i}, i = 1, .. k$ be a cluster of a result of K-means clustering.

ii. For each cluster $\boldsymbol{C_i}$, we define the characteristic function $\boldsymbol{f_i(x)}$ which returns 1 if data point $\boldsymbol{x}$ is in cluster $\boldsymbol{C_i}$, otherwise $\boldsymbol{f_i(x)} = 0$.

iii. Perform logistic regression on $\boldsymbol{f_i(x)}$ where all the features of point $\boldsymbol{x}$ are used as independent variables.

iv. Choose the two most significant features by EXP(B) score of logistic regression, as the labeling variables of the cluster.

2) Labeling each cluster

Since we normalized all the feature values into the range [0.0, 1.0], we can divide the normalized feature values into three intervals: [0.0,0.4), [0.4,0.6) and [0.6, 1.0]. Then we assign a suitable label term to each feature interval. The

label terms we selected for the post-level features and user-level features are shown in Table 4 and Table 5, respectively.

Table 4: Label terms for post features

Center / Feature	[0.0,0.4)	[0.4,0.6)	[0.6, 1.0]
Polarity score	Negative	Neutral	Positive
Subjectivity score	Objective	Mid-subjective	Subjective
Text length	Short	Mid-long	Long
Interaction number	Boring	Mid-attractive	Attractive

Table 5: Label terms for user features

Center / Feature	[0.0,0.4)	[0.4,0.6)	[0.6, 1.0]
Average polarity score	Serious	Neutral	Joyful
Average subjectivity score	Objective	Mid-subjective	Subjective
Average interaction	Lone	Mid-influential	Influential
Number of friends	Unsociable	Mid-sociable	Sociable
Post frequency	Inactive	Mid-active	Active

Post-level and user-level clusters are labeled with the terms in Tables 4 and 5, based on the cluster centers of their significant features. In this way, we can easily differentiate clusters each other. For example, suppose that the two most significant features of one user-level cluster are Number of friends and Post frequency, where the cluster centers are located in 0.7 and 0.8, respectively. Then using the label terms in Table 5, this user cluster is named as the Sociable-Active user.

5. Experiments

We apply our proposed method on the collected dataset. In both post-level and user-level, all features in Table 2 and Table 3 are very universal features which also exist in other SNS platforms such as Twitter etc. That indicates that it is easy to apply our proposed clustering model on other SNS platforms. On the other hand, although K-means clustering is suitable for our data distribution, we also apply agglomerative clustering for comparison.

As the results in Figures 1-3 show, K-means clustering has a superior performance on both clustering stability and quality. Finally, we identify five post types and five representative user types. For each user type, we also investigate its distribution on different interest groups.

5.1 Post clustering

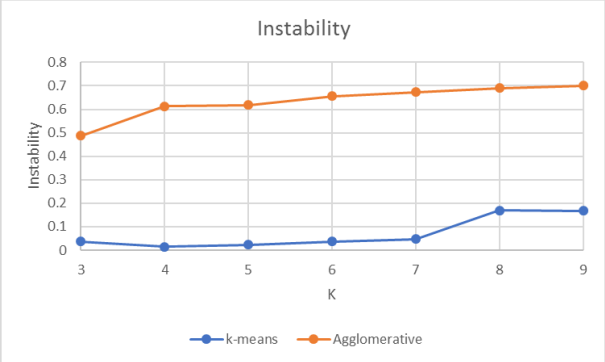

Figure 1: Instability of the post level clustering

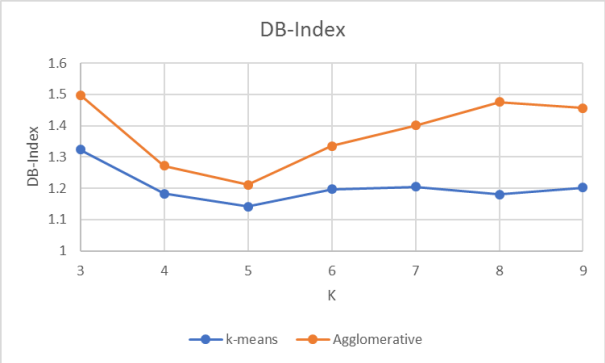

Figure 2: DB-Index of post-level clustering

On the 138,810 collected posts, the four features shown in Table 2 are used for clustering.

To select a suitable cluster number, we evaluate in terms of clustering quality and stability as described in Section 4. We compute instabilities and DB-indexes for changing cluster numbers, from K=3 to 9. The evaluation results are shown in Figure 1 and Figure 2.

The results show that K-means clustering outperforms Agglomerative clustering in both clustering stability and clustering quality. Figure 1 shows that K-means clustering results are more stable than the results by agglomerative clustering. The most stable cluster number is when K = 4, while the clustering results also show a good stability when K is 3 and 5. Figure 2 shows that, when K is equal to 5, we

can obtain the lowest DB-Index value. Considering all the three aspects, K=5 is the best choice for the post level clustering.

Now we apply the cluster labeling algorithm on the post clustering results on K=5. The five identified post types are labeled as:

Post Type 1: Negative-Subjective post
Post Type 2: Positive-Subjective post
Post Type 3: Neutral-Long post
Post Type 4: Objective-Attractive post
Post Type 5: Objective-Boring post

1.2 User-level clustering

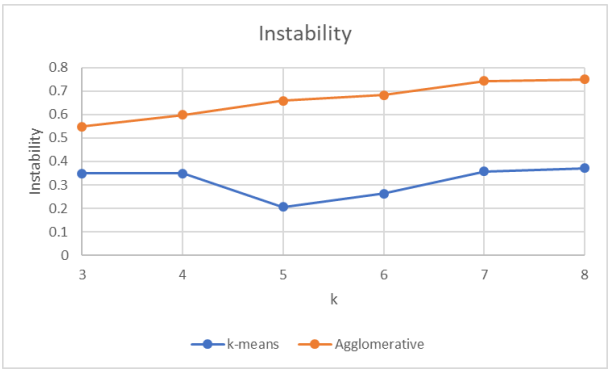

Figure 3: Instability of the user level clustering

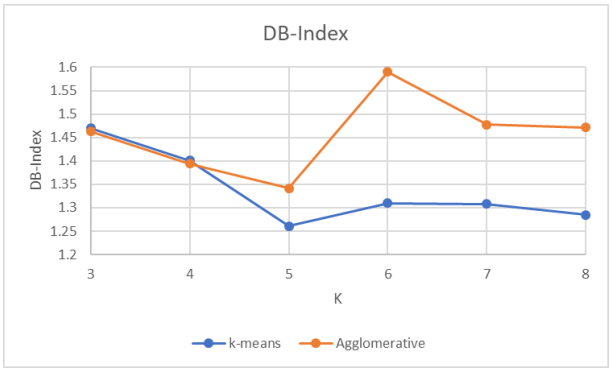

Figure 4: DB-Index of the user level clustering

In the user level clustering, the five features in Table 3 are used. We range K from 3 to 8. The results on clustering stability and quality are shown in Figures 3 and 4.

From the results, we can observe that K-means clustering still outperforms agglomerative clustering in both clustering stability and quality. In Figure 3, we can see that when K equals 5, the clustering results are most stable. Figure 4 indicates that, when K equals five, we can obtain the lowest DB-index on the user-level clustering.

On the other hand, from Figure 4, we can further observe that although we can obtain the lowest instability when K=5, the instability value of the user-level clustering is still higher than the post-level clustering results. This is because the user dataset is much smaller than the post data set. It indicates that it is difficult to find an optimum cluster number for the user-level clustering, indicating that representative user types are not stable, influenced by user subsets. Therefore, we investigate the dynamic changes of user types when we change the cluster number and data subsets, to find stable and/or dominant clusters.

We first apply the cluster labeling algorithm defined in Section 4 to characterize user types on the clusters, for varying cluster numbers. The dynamic changes of user types by different K are shown in Figure 5. Secondly, we fix K to be five, and randomly sample users with the varying set size, from 1/1 to 1/8, and examine the dynamic changes of user types. The results are shown in Figure 6.

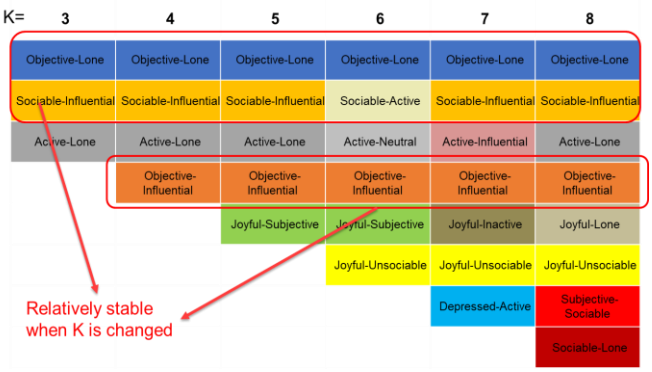

Figure 5: Dynamic changes of user types when K is changed

1/1 dataset	1/2 dataset	1/4 dataset	1/8 dataset
Objective-Lone	Unsociable-Lone	Unsociable-Lone	Objective-Lone
Objective-Influential	Objective-Influential	Objective-Influential	Objective-Influential
Active-Lone	Active-Lone	Active-Lone	Active-Lone
Sociable-Influential	Sociable-Influential	Sociable-Influential	Sociable-Influential
Joyful-Subjective	Joyful-Subjective	Joyful-Subjective	Joyful-Subjective

Figure 6: Five user types on different dataset sizes

From Figure 5, we can observe that the user types Objective-lone, Sociable-Influential, Objective-Influential are quite common in clustering results even when K is changed, indicating that these three user types are stable

and representative. Also, Figure 6 indicates that when fix the user type number as five and change the user set size, the identified five user types are very stable except the Objective-Lone user type which has a slight change on 1/2 and 1/4 dataset.

As a final result, the following five representative user types are selected: Objective-lone, Objective-Influential, Active-lone, Sociable-Influential and Joyful-Subjective. Each user type is described as follows:

User Type 1: Objective-Lone user

This user type tends to write objective posts, but receiving less responses. They are not active in writing posts. Their posts often show serious feelings. Since they have a relatively small number of SNS friends, their posts receive relatively a low number of comments and likes.

User Type 2: Objective- Influential user

This type of users occasionally write objective and serious posts on their timeline. However, they have a relatively large number of SNS friends. Their posts usually receive more comments and feedbacks than the average.

User Type 3: Active-Lone user

This type of users write more posts on their timelines than others. They have a medium number of friends. Their posts are usually neutral in sentiment and receive less comments and likes than the average.

User Type 4: Sociable-Influential user

These users have relatively a large number of SNS friends. They tend to write subjective and positive posts. Also, their posts often receive more comments and likes than others.

User Type 5: Joyful-Subjective user

These uses usually write highly positive and subjective posts. They have a medium number of SNS friends and a medium level of post frequency. Their posts usually receive comments and likes from other users.

5.3 User distribution

For each identified user type, we compare its users' populations in interest groups we sampled.

From Figure 7 we observe that the identified five representative user types are found in every interest group. However, the percentage of each user type is considerably different between interest groups. For example, the Joyful-Subjective user has a very small percentage in politics group, while it has higher percentages in other interest groups, indicating that users interested in politics often write serious and critical posts, and write less emotional posts, accordant with the characteristics of the political interest group.

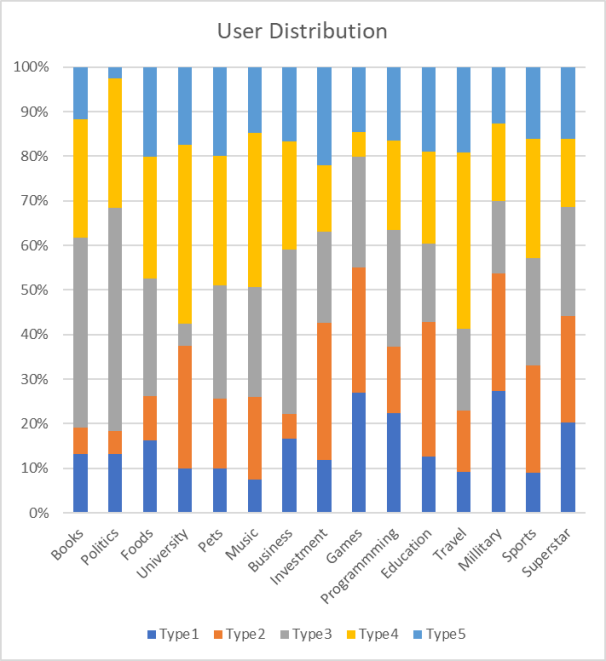

Figure 7: User Distribution on different groups.

6. Conclusion and Future Work

In this paper, we proposed a clustering model for identifying post types and representative SNS user types, based on public posting activities and profiles. We also proposed a cluster labeling algorithm to characterize significant features and name each user type automatically. Since the user types are not having a stable cluster number, we utilized the cluster labelling algorithm and examined dynamic changes of user types when the cluster number and data subsets are changed, to find stable and/or common user types. In experiments, we identified five post types and five user types. At last, we investigate user types' distributions on interest groups users are joining. We find that the identified five representative user types are showing considerably different percentages in these interest groups, showing correspondence with characteristics of the interest groups.

For future work, we plan on measuring similarity between interest groups, in terms of user type distributions, and examining predictability of users' potential interests from their posting styles. We will also try to apply our clustering models and clustering algorithm on other SNS platforms such as Twitter, and compare user type distributions between multiple SNS platforms.

7. Reference

[1] Anon, Selenium - Web Browser Automation. Available at: http://www.seleniumhq.org/ [Accessed January 14, 2017c]

[2] Anon, TextBlob: Simplified Text Processing — TextBlob 0.12.0.dev0 documentation. Available at: http://textblob.readthedocs.io/en/dev/ [Accessed January 14, 2017d].

[3] Bland J M, Altman D G. Statistics notes: measurement error[J]. Bmj, 1996, 313(7059): 744.

[4] Carmel, David, Haggai Roitman, and Naama Zwerdling. "Enhancing cluster labeling using wikipedia." Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009.

[5] Davies D L, Bouldin D W. A cluster separation measure[J]. IEEE transactions on pattern analysis and machine intelligence, 1979 (2): 224-227.

[6] Dunn J C. Well-separated clusters and optimal fuzzy partitions[J]. Journal of cybernetics, 1974, 4(1): 95-104.

[7] Gutierrez F J, Poblete B. Sentiment-based user profiles in microblogging platforms[C]//Proceedings of the 26th ACM Conference on Hypertext & Social Media. ACM, 2015: 23-32.

[8] Marquardt, J. et al., 2014. Age and gender identification in social media. CEUR Workshop Proceedings, 1180, pp.1129–1136.

[9] Preoţiuc-Pietro, D. et al., 2015. Studying user income through language, behaviour and affect in social media. PLoS ONE, 10(9), pp.1–17.

[10] Rousseeuw P J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis[J]. Journal of computational and applied mathematics, 1987, 20: 53-65.

[11] Von Luxburg U. Clustering stability: an overview[J]. Foundations and Trends® in Machine Learning, 2010, 2(3): 235-274.