A Simple Language Independent Approach for Distinguishing Individuals on Social Media

Guangyuan Piao
Department of Computer Science, Maynooth University
Maynooth, Co Kildare, Ireland
guangyuan.piao@mu.ie

ABSTRACT

Nowadays, the large-scale human activity traces on social media platforms such as Twitter provide new opportunities for various research areas such as mining user interests, understanding user behaviors, or conducting social science studies in a large scale. However, social media platforms contain not only individual accounts but also other accounts that are associated with non-individuals such as organizations or brands. Therefore, distinguishing individuals out of all accounts is crucial when we conduct research such as understanding human behavior based on data retrieved from those platforms. In this paper, we propose a language-independent approach for distinguishing individuals from non-individuals with the focus on leveraging their profile images, which has not been explored in previous studies. Extensive experiments on two datasets show that our proposed approach can provide competitive performance with state-of-the-art language-dependent methods, and outperforms alternative language-independent ones.

CCS CONCEPTS

• **Human-centered computing** → **Social media**; **Social networking sites**; • **Computing methodologies** → **Supervised learning by classification**.

KEYWORDS

Account Classification, Deep Learning, Social Media Analysis

ACM Reference Format:
Guangyuan Piao. 2021. A Simple Language Independent Approach for Distinguishing Individuals on Social Media. In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media (HT '21), August 30-September 2, 2021, Virtual Event, Ireland.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3465336.3475092

1 INTRODUCTION

Social media platforms such as Twitter[1] have been widely used in different research areas to study users from various perspectives in a large scale based on the big data generated by user activities on those platforms. For example, research areas such as predicting substance usage [10], mining user interests [14, 15], and understanding user visiting behaviors [16] or personalities [7]. Although those studies usually assume all accounts retrieved from social media platforms are individuals, McCorriston et al. estimated that 9.4% of accounts on Twitter are non-individual ones (e.g., brands or organizations) [11] . Therefore, distinguishing individual users from retrieved accounts on social media platforms is crucial for studying different user behaviors such as mining user interests or understanding substance usage, and deriving conclusions out of those studies.

Previous studies for distinguishing individuals from non-individuals can be classified into two categories based on whether an approach is language *dependent* or *independent*. For example, language-dependent approaches utilize textual information, such as social posts and/or the profile description (biography) of a user in addition to a set of statistical features, e.g., the number of followees and followers on Twitter, to classify whether a given account belongs to an individual. As one might expect, the profile description of an account can provide crucial information to distinguish individuals. For example, we can assume that an account belongs to an individual if its profile description contains words such as "*my*", "*I*", "*Dad*", "*Mum*", etc. However, this line of approaches depends on language and the majority of the previous works have been focused on English users. Although English is the most popular language on Twitter, it is used in only 32% of all Twitter messages[2]. In contrast to relying on textual information, recent studies [3, 4] have proposed leveraging statistical features for classifying accounts on social media platforms such as Twitter.

Our focus in this paper falls into the second category, i.e., language-independent approaches for classifying individual accounts. To this end, we leverage the visual content of a user (i.e., profile image), which is critical information but has not been explored in previous studies. The intuition behind our approach is that the profile image of a user should be a good indicator for the classification of accounts. Our main contributions include:

- We propose a simple Language-Independent Individual Classification approach (Section 3), named `LIIC`, to classify social media accounts into individuals and non-individuals, with the focus on leveraging their profile images.
- We evaluate our approach with several state-of-the-art approaches using two datasets with ground truth labels in Section 4, and show that `LIIC` can achieve competitive performance in classifying Twitter accounts compared to

[1] https://twitter.com/home

[2] shorturl.at/sTVZ9

HT '21, August 30-September 2, 2021, Virtual Event, Ireland

ACM ISBN 978-1-4503-8551-0/21/08.
https://doi.org/10.1145/3465336.3475092

language-dependent approaches.

- Through an ablation study in Section 5, we further reveal that profile images are indeed an important indicator for classifying individual user accounts, which have not been explored in previous studies.

2 RELATED WORK

In this section, we review related works which are classified into language-dependent and language-independent ones.

Language-dependent approaches. This line of approaches exploits textual content such as social posts or profile descriptions of users for feature engineering or learning latent representations via deep learning approaches [5, 20, 21]. For example, Oentaryo et al. used content, social, and temporal features and investigated several machine learning approaches such as random forests and gradient boosting [6], and showed that the gradient boosting classifier provides the best performance [13]. Wood-Doughty et al. proposed using a character-based Convolutional Neural Network (CNN) [9] to learn the representation of a user's name and incorporated profile features such as the ratio of followers to friends together for classifying individual accounts [20].

Language-independent approaches. This line of approaches uses statistical features such as the social network structure and the posting frequency of a user without relying on textual content for classifying individual accounts [3, 4, 18, 19]. For example, Tavares et al. used a naive Bayes classifier with features related to the time distribution between social posts [18]. More recently, Daouadi et al. proposed a set of comprehensive features incorporating profile and activity related ones, and used gradient boosting regression trees and random forests for classifying individual accounts on Twitter [3, 4].

Despite the appreciable body of previous studies, the profile image of an account, which might be a critical indicator for the classification, has not been explored. In this work, we close the gap and focus on leveraging profile images for classifying individuals and use other profile related features only if those images are not retrievable or are default ones. Our approach can be considered as one of the language-independent approaches as this approach does not require textual content such as social posts or profile descriptions.

3 LIIC: LANGUAGE INDEPENDENT INDIVIDUAL CLASSIFIER

In this section, we introduce our Language Independent Individual Classifier (LIIC) and its components in Section 3.1, and provide the training details of LIIC in Section 3.2.

3.1 LIIC Architecture

Figure 1 illustrates an overview of the `LIIC` architecture. `LIIC` uses three types of input such as the *profile image*, *screen name*, and *profile features* of a user, and leverages three different types of neural networks to learn the representations of each input for the binary classification of the target account (*individual* or *non-individual*).

The main assumption of `LIIC` is that the profile image of an account should be a good indicator for distinguishing individuals.

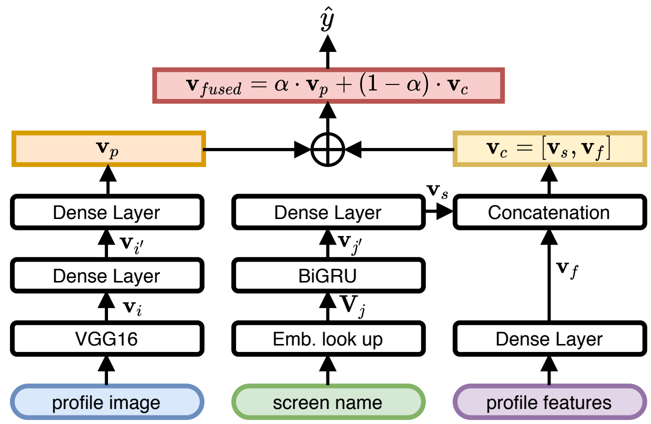

Figure 1: Overview of our proposed approach – `LIIC`, which uses CNN, BiGRU, and FNN for three different types of input, i.e., profile image, screen name (e.g., *@ACMHT*), and profile features (e.g., the number of friends), respectively.

For example, an individual user tends to use his/her face photo while non-individual accounts for companies or conferences tend to use their organization logos instead. However, using profile images only can limit the capacity for classifying accounts, e.g., when the retrieved profile images are default ones from Twitter, or a profile image can not be retrieved using its URL obtained from the Twitter API[3]. Therefore, we also use screen names and profile features in addition to profile images.

Profile image representation. Given profile images as input, it is natural to use a CNN to extract image features. Instead of training a CNN from scratch, we use a pre-trained CNN model – VGG16 [17] to extract the image features, where the last layer of VGG16 is used for generating an image representation vector $\mathbf{v}_i \in \mathbb{R}^{20548}$. Afterwards, $\mathbf{v}_i$ is fed into two dense layers to output the final representation $\mathbf{v}_p \in \mathbb{R}^{256}$:

$$\mathbf{v}_{i'} = f(\mathbf{W}^{(i)}\mathbf{v}_i + b_i) \tag{1}$$

$$\mathbf{v}_p = f(\mathbf{W}^{(i')}\mathbf{v}_{i'} + b_{i'}) \tag{2}$$

where $\mathbf{W}^{(i)}$ and b_i are a weight matrix and a bias term for the input $\mathbf{v}_i$, $f(\cdot)$ is an activation function (we empirically adopt the *ReLu* [12] in the rest of the paper), and $\mathbf{v}_{i'} \in \mathbb{R}^{1024}$ is the output from the intermediate dense layer.

Screen name representation. Screen names (usernames) such as *@ACMHT* on social media platforms consist of a sequence of allowed characters. Given this sequential input, we use bidirectional Gated Recurrent Units (GRUs) [2] which are good at processing sequence data. To this end, we first use character embeddings to convert the input characters of a screen name, which can be fed into bidirectional GRUs. That is, the sequence of characters for a given screen name is converted into $\mathbf{V}_j \in \mathbb{R}^{50 \times 100}$, where 50 is the fixed length of a screen name (padded with zeros in front), and 100 is the dimension of character embeddings. The bidirectional GRUs then output the representation $\mathbf{v}_{j'} \in \mathbb{R}^{128}$ for the given character

[3]https://developer.twitter.com/en/docs/twitter-api

embeddings. Finally, $\mathbf{v}_{j'}$ is fed into a dense layer to output the final representation $\mathbf{v}_s \in \mathbb{R}^{128}$ for the given screen name as follows:

$$\mathbf{v}_s = f(\mathbf{W}^{(j')}\mathbf{v}_{j'} + b_{j'}) \tag{3}$$

where $\mathbf{W}^{(j')}$ and $b_{j'}$ are a weight matrix and a bias term.

Profile feature representation. Finally, we also extract a set of features (13 in total) from user profile information which can be retrieved via the Twitter API. Table 1 shows the description about those features. As some of the features such as the number of followers or friends can have different scales compared to other features which can make the training of our model difficult, we first scaled those features using a logarithmic transformation, i.e., $\mathbf{v}_{k'} = \log_{10}(\mathbf{v}_k)$. Here, $\mathbf{v}_k \in \mathbb{R}^{13}$ denotes the initial values of those features. $\mathbf{v}_{k'}$ is then used as an input to a dense layer to output $\mathbf{v}_f \in \mathbb{R}^{128}$ as shown below using Equation 4.

$$\mathbf{v}_f = f(\mathbf{W}^{(k')}\mathbf{v}_{k'} + b_{k'}) \tag{4}$$

where $\mathbf{W}^{(k')}$ and $b_{k'}$ are a weight matrix and a bias term.

Concatenation. After obtaining the latent representations of the screen name and the set of features of a user (i.e., $\mathbf{v}_s$ and $\mathbf{v}_f$), those two representations are concatenated together into $\mathbf{v}_c \in \mathbb{R}^{256}$ as follows.

$$\mathbf{v}_c = \left[\mathbf{v}_s, \mathbf{v}_f\right] \tag{5}$$

Fusion. As our intuition is that the profile image of a user account is critical for distinguishing individuals from non-individuals, we deliberately pay full attention for the profile image information if possible, and only use the other information (i.e., screen names and profile features) when the profile image has not been changed from the default one or the image cannot be retrieved via the given image URL from the Twitter API. This can be formulated as follows:

$$\mathbf{v}_{fused} = \alpha \cdot \mathbf{v}_p + (1-\alpha)\cdot \mathbf{v}_c \tag{6}$$

$$\alpha = \begin{cases} 0, & \text{if default profile image or not available} \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

where $\mathbf{v}_{fused} \in \mathbb{R}^{256}$ is the final vector that is used for predicting the score of being classified as an individual.

Prediction. Finally, a prediction score $\hat{y}$ is calculated using Equation 8, and the target account can be classified as 1 (individual) if $\hat{y} \geq 0.5$ or 0 otherwise,

$$\hat{y} = \sigma(\mathbf{W}^{(fused)}\mathbf{v}_{fused} + b_{fused}) \tag{8}$$

where σ denotes the *sigmoid* function $s(x) = \frac{1}{1+e^{-x}}$, and $\mathbf{W}^{(fused)}$ and b_{fused} are a weight matrix and a bias term..

Table 1: The set of features (13) from the profile of a user.

Feature Description
Whether the user provided a URL in their profile
Whether the user has changed the background or theme of the profile
Whether the user has a verified account
Whether the user provided a description about him-self/herself
Whether the use has changed the profile image
The No. of Twitter Lists that the user belongs to
The No. of followers
The No. of friends
The No. of total posts
The No. of favorites
The length of screen name
The ratio between the No. of friends and followees
The ratio between the No. of friends and the sum of the No. of friends and followees

3.2 Training

We used Tensorflow 2.3.0 to implement LIIC. For training LIIC, we use the Adam update rule [8] and a batch size of 1,024 to train the model on the training set with a learning rate of 0.001 to minimize the binary cross entropy loss $\mathcal{L}$, which is defined as follows.

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \cdot \log \hat{y_i} + (1-y_i)\cdot \log(1-\hat{y_i})\right] \tag{9}$$

where y_i and $\hat{y_i}$ denote the ground truth and predicted labels for i-th instance, respectively. N refers to the total number of instances for training.

To resolve the overfitting problem, we divided the training set and used 1/3 of it as our validation set, and used the rest of the data for training. An early stopping strategy is adopted with the validation set, which stops the training if there is no improvement of accuracy on the validation set.

4 EVALUATION

In this section, we discuss the datasets (Section 4.1) and the set of compared methods (Section 4.2) for evaluating LIIC followed by the experimental results in Section 4.3.

4.1 Datasets

Here, we describe the training set used for training LIIC and two other datasets for testing. Table 2 shows the statistics about the three datasets.

The *training* dataset is from Wood-Doughty et al. [20]. Instead of manual labeling of individuals and non-individuals, the authors constructed the dataset using an automated method based on weak supervision for the discovery and labeling of those accounts. For example, the authors in [20] identified Twitter lists[4] containing non-individual accounts which include terms such as "*companies*" or "*businesses*", and those containing individual accounts which include key terms such as "*friends*" and "*families*" using a search engine. By using this approach, a large dataset can be obtained without manual labeling effort which is time-consuming. From manual

[4] https://help.twitter.com/en/using-twitter/twitter-lists

Table 2: Statistics of datasets used for training and testing.

Dataset	Total (#)	Ind. (#)	Non-Ind. (#)
Training	186,874	159,989	26,885
Test (Ours)	989	624	365
Test (Humanizer)	15,809	140,50	1,759

investigation of 200 accounts out of the large dataset obtained using this approach, the authors in [20] found that the accuracy is also high (98%). We used the same dataset and crawled all accounts that are accessible via the Twitter API except those ones that either do not exist or do not allow access anymore at the time of our experiments. After all, the training set contains 186,874 accounts/instances where 85.6% (159,989) are individual accounts and 14.4% (26,885) are non-individual ones. The idea is that using this large dataset with a small portion of noisy labels is able to train better deep learning approaches compared to using a small manually labeled dataset as shown in [20].

We constructed the first test set – *Test* (*Ours*) in Table 2 – based on the same idea from Wood-Doughty et al. [20], and retrieved 989 Twitter accounts. Afterwards, we further manually investigated and adjusted incorrect labels where 10 (1%) account labels have been adjusted. Among the 989 accounts, 63.1% (624) are individual accounts and the rest (365) are non-individual ones.

The second test dataset – *Test* (*Humanizer*) in Table 2 – is from McCorriston et al. [11] where all the accounts are manually annotated using Amazon Mechanical Turk[5]. After filtering those accounts that are not accessible, the dataset contains 15,809 accounts in total where 88.9% (14,050) are individual accounts and the rest (1,759) are non-individual ones.

4.2 Compared Methods

To evaluate the performance of `LIIC`, we compare `LIIC` with the following baseline and state-of-the-art approaches.

- `Majority Class` is a straightforward baseline method which always predicts the majority class, i.e., *individual* in our case.
- `Humanizer` [11] uses three types of features such as post content features (e.g., words and hashtags for each class), stylistic features (e.g., the average number of words used per tweet), and structural and behavioral features based on how the account interacts with others (e.g., ratio of retweets to tweets), and trained a LibSVM [1] classifier using those features. We used the implementation of the authors in [11] [6].
- `Demographer` [20] uses a character-based CNN to learn the representation of a user's name, and uses profile-based features such as the ratio of followers to friends as well as the presence of pronouns such as "*my*" and "*our*". We used the implementation of the authors in [4] which is publicly available[7].
- `RandomForest` [4] uses a set of language-independent features, and a Random Forest classifier is used for classifying different account types. We extracted the same set of features that are used in [4], and trained a Random Forest classifier using the training set with three-fold cross-validation for optimizing hyperparameters.

4.3 Results

We evaluate the performance of aforementioned methods in terms of the overall *accuracy* of classification, and *precision*, *recall*, and F_1 score of both individual and non-individual labels. Table 3 and 4 show the results on our and the humanizer datasets, respectively. The last column of each table indicates whether a method is language independent.

As we can see from Table 3, all methods outperform the baseline approach – `Majority Class`. `LIIC` provides competitive performance compared with `Demographer` despite `LIIC` is language independent and does not require any profile description or tweet. Overall, `LIIC` and `Demographer` achieve an *accuracy* of 0.93 followed by `Humanizer` and `RandomForest`. When the focus is the classification of individual, `LIIC` with a *precision* of 0.97 and F_1 score of 0.95 provides the best performance followed by `Demographer`. `Demographer` performs better in terms of *precision* (0.91) when the classification focus is non-individuals compared to `LIIC` (0.88) while `LIIC` performs better in terms of both *recall* and F_1 score.

The results of the humanizer dataset in Table 4 show similar trends with Table 3. For instance, both `LIIC` and `Demographer` achieve an *accuracy* of 0.94 followed by `RandomForest` (0.90). `LIIC` with an F_1 score of 0.97 outperforms `Demographer` when the focus is the classification of individual. When the focus is classifying non-individuals, `LIIC` with an F_1 score of 0.74 also performs better than `Demographer` while `Demographer` provides a higher *precision* (0.91) over `LIIC` (0.88).

The performance on the two datasets indicates that we can achieve comparable performance for distinguishing individual accounts in a language-independent manner using `LIIC`.

5 ABLATION STUDY

To investigate the effectiveness of the components of `LIIC` such as the ones for profile images, screen names, and profile features, we conducted an ablation study with some variants of `LIIC` listed below.

- $\texttt{LIIC}_{\texttt{img}}$ uses V_p instead of V_{fused} in Figure 1 to predict the label $\hat{y}$, which does not use the screen name or statistical features of a user such as the number of friends. This can be seen as an image classifier.
- $\texttt{LIIC}_{\texttt{screenname+features}}$ uses the concatenated vector V_c instead of V_{fused} in Figure 1 for predicting the label $\hat{y}$, which does not use the profile image of a user.

[5] https://www.mturk.com/
[6] http://networkdynamics.org/resources/software/humanizr/
[7] https://bitbucket.org/mdredze/demographer/src

Table 3: Performance of individual classification using compared methods on our dataset with the best-performing scores in bold (except the baseline – `Majority Class`). `Majority Class`, `RandomForest`, and `LIIC` are language independent as they are not using textual information such as profile descriptions or posts.

Method	*Accuracy*	Individual			Non-individual			Language Independent
		Precision	*Recall*	F_1	*Precision*	*Recall*	F_1	
`Majority Class`	0.63	0.63	1.00	0.77	0.00	0.00	0.00	x
`Humanizer` [11]	0.81	0.79	**0.95**	0.87	0.88	0.58	0.69	
`Demographer` [20]	**0.93**	0.93	**0.95**	0.94	**0.91**	0.88	0.90	
`RandomForest` [4]	0.77	0.76	0.93	0.84	0.82	0.50	0.62	x
`LIIC`	**0.93**	**0.97**	0.92	**0.95**	0.88	**0.95**	**0.91**	x

Table 4: Performance of individual classification using compared methods on the humanizer dataset [11] with the best-performing scores in bold (except the baseline – `Majority Class`). The results of `Humanizer` is not applicable as it is trained using the humanizer dataset.

Method	*Accuracy*	Individual			Non-individual			Language Independent
		Precision	*Recall*	F_1	*Precision*	*Recall*	F_1	
`Majority Class`	0.89	0.89	1.00	0.94	0.00	0.00	0.00	x
`Humanizer`	-	-	-	-	-	-	-	
`Demographer`	**0.94**	0.95	**0.98**	0.96	**0.82**	0.55	0.66	
`RandomForest`	0.90	0.92	**0.98**	0.95	0.63	0.30	0.41	x
`LIIC`	**0.94**	**0.97**	0.96	**0.97**	0.71	**0.78**	**0.74**	x

Table 5: Performance of variants of `LIIC` by removing different components.

Dataset	Method	*Accuracy*	Individual			Non-individual		
			Precision	*Recall*	F_1	*Precision*	*Recall*	F_1
Ours	`LIIC`	**0.93**	**0.97**	0.92	**0.95**	0.88	**0.95**	**0.91**
	$\texttt{LIIC}_{\texttt{img}}$	0.92	0.93	**0.94**	0.94	**0.90**	0.88	0.89
	$\texttt{LIIC}_{\texttt{screenname+features}}$	0.79	0.77	0.94	0.85	0.84	0.52	0.64
	$\texttt{LIIC}_{\texttt{features}}$	0.72	0.72	0.91	0.80	0.72	0.41	0.52
	$\texttt{LIIC}_{\texttt{screenname}}$	0.74	0.72	0.96	0.82	0.85	0.35	0.49
Humanizer	`LIIC`	**0.94**	**0.97**	0.96	**0.97**	0.71	**0.78**	**0.74**
	$\texttt{LIIC}_{\texttt{img}}$	**0.94**	0.96	0.97	0.96	**0.73**	0.68	0.70
	$\texttt{LIIC}_{\texttt{screenname+features}}$	0.92	0.93	**0.98**	0.95	0.72	0.43	0.54
	$\texttt{LIIC}_{\texttt{features}}$	0.90	0.91	**0.98**	0.95	0.64	0.27	0.38
	$\texttt{LIIC}_{\texttt{screenname}}$	0.88	0.92	0.95	0.94	0.46	0.31	0.37

- $\texttt{LIIC}_{\texttt{features}}$ relies on the set of features only with a fully connected dense layer, and uses V_f instead of V_{fused} in Figure 1 for prediction.
- $\texttt{LIIC}_{\texttt{screenname}}$ relies on the screen name of a user only with a bidirectional GRU, and uses V_s instead of V_{fused} in Figure 1 for prediction.

Table 5 shows the results of those variants of `LIIC` on both our and the humanizer datasets. Overall, `LIIC` achieves the best performance in terms of the *accuracy* and F_1 scores for both individual and non-individual. This indicates that considering all components of `LIIC` is useful to achieve the best performance. It is worth noting that $\texttt{LIIC}_{\texttt{img}}$ which can be seen as an image classifier also provides good performance and outperforms the rest of those variants. This shows that profile images alone indeed provide a good clue for distinguishing individuals, and $\texttt{LIIC}_{\texttt{img}}$ can be generally applied to other social media platforms as it does not rely on features might be tied to a specific platform. The results in Table 5 also show that $\texttt{LIIC}_{\texttt{screenname+features}}$ outperforms $\texttt{LIIC}_{\texttt{features}}$ and $\texttt{LIIC}_{\texttt{screenname}}$. This indicates that incorporating both screen name and features is useful compared to considering each separately.

6 CONCLUSIONS

In this paper, we presented `LIIC` which is a language-independent approach for classifying individual accounts on social media platforms such as Twitter. Despite of the language independence, our results show that `LIIC` can achieve competitive performance compared to other state-of-the-art methods. In addition, the ablation study in Section 4 indicates that the profile image of an account indeed is a good indicator for distinguishing individuals, while using all components of `LIIC` provides the best performance. For future work, we will investigate using different CNN models such as customized smaller models compared to VGG16 for extracting the profile image features and its impact on performance. Our datasets and code can be found here[8].

REFERENCES

[1] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *TIST* 2, 3 (2011), 1–27.

[2] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv:1406.1078* (2014).

[3] Kheir Eddine Daouadi, Rim Zghal Rebaï, and Ikram Amous. 2018. Towards a Statistical Approach for User Classification in Twitter. In *ICMLN*. Springer, 33–43.

[4] Kheir Eddine Daouadi, Rim Zghal Rebaï, and Ikram Amous. 2019. Organization, Bot, or Human: Towards an Efficient Twitter User Classification. *Computación y Sistemas* 23, 2 (2019), 273–280.

[5] Munmun De Choudhury, Nicholas Diakopoulos, and Mor Naaman. 2012. Unfolding the event landscape on twitter: classification and exploration of user categories. In *CSCW*. 241–244.

[6] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.

[7] Haiqian Gu, Jie Wang, Ziwen Wang, Bojin Zhuang, and Fei Su. 2018. Modeling of user portrait through social media. In *ICME*. IEEE, 1–6.

[8] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980* (2014).

[9] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[10] Jing Li, Yan Song, Zhongyu Wei, and Kam-Fai Wong. 2018. A joint model of conversational discourse and latent topics on microblogs. *Computational Linguistics* 44, 4 (2018), 719–754.

[11] James McCorriston, David Jurgens, and Derek Ruths. 2015. Organizations Are Users Too: Characterizing and Detecting the Presence of Organizations on Twitter.. In *ICWSM*. Citeseer, 650–653.

[12] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *ICML*. 807–814.

[13] Richard Jayadi Oentaryo, Jia-Wei Low, and Ee-Peng Lim. 2015. Chalk and cheese in twitter: Discriminating personal and organization accounts. In *ECIR*. Springer, 465–476.

[14] Megha Pandey and Alex Chia Yong Sang. 2015. Capturing the visual language of social media. In *ICME*. IEEE, 1–6.

[15] Guangyuan Piao and John G Breslin. 2018. Inferring user interests in microblogging social networks: a survey. *UMUAI* 28, 3 (2018), 277–329.

[16] Ryosuke Shigenaka, Yan-Ying Chen, Francine Chen, Dhiraj Joshi, and Yukihiro Tsuboshita. 2017. Image-based user profiling of frequent and regular venue categories. In *ICME*. IEEE, 541–546.

[17] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[18] Gabriela Tavares and Aldo Faisal. 2013. Scaling-laws of human broadcast communication enable distinction between human, corporate and robot twitter users. *PloS one* 8, 7 (2013), e65774.

[19] Zijian Wang, Scott Hale, David Ifeoluwa Adelani, Przemyslaw Grabowicz, Timo Hartman, Fabian Flock, and David Jurgens. 2019. Demographic inference and representative population estimates from multilingual social media data. In *The World Wide Web Conference*. 2056–2067.

[20] Zach Wood-Doughty, Praateek Mahajan, and Mark Dredze. 2018. Johns Hopkins or johnny-hopkins: Classifying individuals versus organizations on twitter. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*. 56–61.

[21] Liang Yan, Qiang Ma, and Masatoshi Yoshikawa. 2013. Classifying Twitter users based on user profile and followers distribution. In *DEXA*. 396–403.

[8] https://github.com/parklize/twitter-account-classification