Capturing Deep Dynamic Information for Mapping Users across Social Networks

Chiyu Cai[1, 2] Linjing Li[1, 3] Weiyun Chen[4] Daniel Zeng[1, 2, 3]
[1] The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing, China
[3] Shenzhen Artificial Intelligence and Data Science Institute (Longhua), Shenzhen, China
[4] School of Management, Huazhong University of Science and Technology, Wuhan, China
{caichiyu2014, linjing.li, dajun.zeng}@ia.ac.cn, chenweiyun@hust.edu.cn

***Abstract*—Nowadays, it is common that a netizen creates multiple accounts across social platforms. Mapping accounts across platforms could facilitate various applications in security. Existing methods usually focus on profile and network based features. In this paper, we concentrate on capturing dynamic information of social users and present a deep dynamic user mapping model to identify the accounts across platforms. The proposed model captures dynamic latent features from three aspects including posting pattern, writing pattern, and emotional fluctuation. We also develop a matching network that fuses dynamic and traditional features to identify accounts. To the best knowledge of ourselves, this is the first trial that applies deep neural network in mapping users with dynamic information. Experiments on real world dataset demonstrated the effectiveness of the proposed method.**

I. Introduction

In recent years, owing to the diversity, superb feature and strong attraction of social media, there is a trend that netizens create multiple accounts on different platforms. Mapping users across social networks devotes to identify the accounts belong to the same user in different platforms. From a security point of view, mapping users can help control rumors, monitor public opinion, analyze or track cyber criminals[1]. Thus mapping users across platforms is a highly demanded and valuable research question.

In this paper, we concentrate on capturing dynamic information of social users, and proposed a deep dynamic user mapping model (DDUM) to identify the accounts across different platforms. To capture user dynamic information, DDUM explores latent dynamic features from three aspects using three encoders. The posting pattern encoder takes endogenous and exogenous factors which affect user online posting behavior into accounts. The writing pattern encoder develops a pop culture attention mechanism and devices a memory network to model writing pattern. The emotional fluctuation encoder learns representation using attention based Bi-LSTM. DDUM integrates all dynamic representations and traditional information, and develops a matching network to identify accounts. We conduct a series of experiments on real world dataset, the experiment results demonstrated the effectiveness of the proposed model.

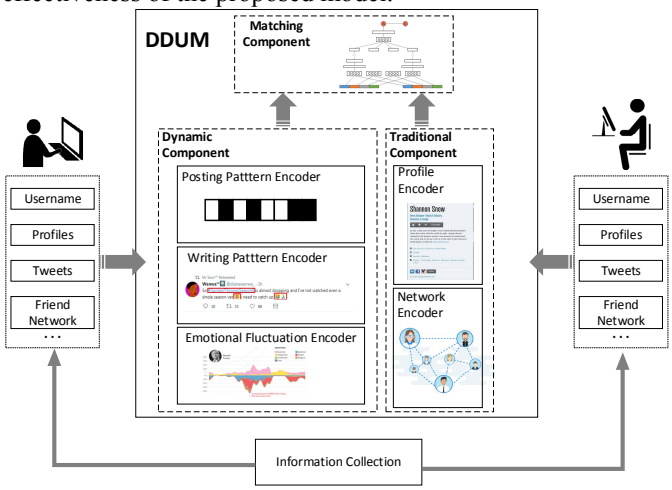

Fig. 1. The overview of Deep Dynamic User Mapping Model

II. Deep Dynamic User Mapping Model

The goal of the proposed model is to match users across social platforms via capturing dynamic information. The architecture of the proposed model is shown in Fig. 1. The DDUM contains two major components, the traditional and the dynamic information component. The former is designed for encoding profiles and networks. The latter is consist of three encoders to encoding posting pattern, writing pattern, and emotional fluctuation. The matching network is devised to measure the similarity of feature representations of different accounts.

A. Posting Pattern Encoder

The factors that impact posting behavior can be coarsely partitioned into two types, endogenous and exogenous factors. The endogenous factor we considered is circadian rhythm which is believed to the cause of why some individuals prefer to surf the internet in the morning houses while others prefer evening hours. Exogenous factor is environment influence. For instance, people with different occupations behave differently on weekends or holidays. Many people spend more time on social networks, but the service industry practitioners are busy at

working. In addition, we analyze the posting patterns from two aspects the original posting and the retweeting, which also show the individual preference. We employ the posting pattern encoder as introduced in [2], and build a posting pattern representation pp_u.

B. Writing Pattern Encoder

We develop a pop culture attention writing model (PAWM) to encode the writing pattern representation. PAWM is mainly composed of two parts: a pop culture attention module and a memory module. In the pop culture attention module, we incorporate an pop culture representation v_{pc} in the attention mechanism to consider the influence of pop culture ingredients in user writing pattern. The attention mechanism maps a hidden embedding h_t which produced by Bi-LSTM into a latent writing pattern vector p,

$$c_t = \tanh\left(W_h h_t + W_{pc} v_{pc} + b_{pc}\right), \quad (1)$$

$$\alpha_t = \frac{exp(c_s^{\mathrm{T}} c_t)}{\sum_t exp(c_s^{\mathrm{T}} c_t)}, \quad (2)$$

$$p = \sum_t \alpha_t h_t, \quad (3)$$

where c_s is a context vector which indicates the informative words for writing patterns, it is randomly initialized and learned by training.

The memory module uses an external memory matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ which is consisted of n memory blocks, the size of each block is m. The weight $u_t(i)$ is calculated using a scoring function S:

$$u_t(i) = \frac{exp(S(M_{t-1}(i), p))}{\sum_i exp(S(M_{t-1}(i), p))}, \quad (4)$$

where p is the current input pattern vector and $M_{t-1}(i)$ denotes a previous step memory vector. Function S is formulation as:

$$S(M_{t-1}(i), p) = W_m^3 \tanh(W_m^2 \mathrm{z}(M_{t-1}(i), p) + b_m^1) + b_m^2) + b_m^3 \quad (5)$$

$$\mathrm{z}(me, p) = [me, p, |me - p|, me \circ p, me^{\mathrm{T}} W_z p], \quad (6)$$

where $\circ$ is the element-wise product. The mixture feature set $\mathrm{z}(me, p)$ is designed to capture a variety of similarities between memory vector and input vector.

The update operation is a convex combination of the old memory vector and the current input vector,

$$M_t(i) = u_t(i) a_t + (1 - u_t(i) e_t) M_{t-1}(i), \quad (7)$$

$$a_t = \tanh(W_a p + b_a), \quad (8)$$

$$e_t = \mathrm{sigmoid}(W_e p + b_e), \quad (9)$$

where W_a, W_e, b_a, b_e are parameters to be trained, a_t is an vector represents the current latent writing pattern information we learned from new input, e_t can be viewed as an erase vector as the old knowledge will be forgotten. The read operation is a weighted sum,

$$r_t(i) = \sum_i u_t(i) M_{t-1}(i). \quad (10)$$

The writing pattern representation tp of the input tweet is calculated by:

$$tp = W_o(r_t(i) + p), \quad (11)$$

where W_o is a weight matrix.

For user u, we can obtain a writing pattern matrix through mapping all history tweets into writing pattern vectors. A CNN is employed to extract high level representation wp_u which indicates the user writing pattern representation.

C. Emotional Fluctuation Encoder

To model the emotional fluctuation of users, we pre-train a attention based Bi-LSTM as the emotion classifier. In our work, the emotion categories are divided into five categories, including strong positive (2), positive (1), neutral (0), negative (-1), strong negative (-2). Given a user u, the history tweets posted in day d is denoted as $[twt_{d,1}, twt_{d,2}, \cdots, twt_{d,len}]$, where len is the number of tweets. We feed tweets into attention based Bi-LSTM and obtain the corresponding categories. Then we can calculate the emotion of user u in day d using a weighted sum:

$$ec_{udi} \leftarrow \mathrm{ATLSTM}\left(twt_{d,i}\right), \quad (12)$$

$$ec_{ud} = \sum_i ec_{udi}. \quad (13)$$

For user u, we build the emotion fluctuation embedding ef_u over a period of time D as follows:

$$ef_u = [ec_{u1}, ec_{u2}, \cdots, ec_{uD}]. \quad (14)$$

D. Traditional Information Component

1) Profile Encoder

In this work, we use the profile based features including username, location and description.

Description. We utilize word embedding to convert text into an embedding matrix $\mathbf{D}_i$ in this component. Description text matrix $\mathbf{D}_i$ is encoded into a vector x_i using a convolutional neural network. Then we build a vector x_{mix}:

$$x_i \leftarrow \mathrm{CovNet}(\mathbf{D}_i), \quad (15)$$

$$x_{mix} = \left[x_i, x_i^{\mathrm{T}} \mathrm{M} x_j, x_j\right], \quad (16)$$

where M is a similarity matrix. The vector x_{mix} is then passed through a hidden layer, then we obtain the description matching vector $des_{i,j}$:

$$des_{i,j} = \alpha(w_d \cdot x_{mix} + b_d), \quad (17)$$

where w_d is the weight vector of the hidden layer and $\alpha(\cdot)$ is the non-linearity.

Username. We measure the similarity of username using the Levenshtein (Edit) distance which denoted as $name_{i,j}$.

Location. We directly match the location information from three levels including city, state, and country, the similarity is denoted as $loc_{i,j}$.

2) Network Encoder

Network embedding is an intuitive way to represent the friend network. To represent the network structure, we adopt DeepWalk [3] in work, and learn the network embedding v_s^n.

E. Matching Component

We develop a matching network to identify accounts as shown in Fig. 1. Given an account pair ac_i and ac_j, the corresponding representation pair of feature k are denoted as F_i^k and F_j^k. We build the mixture vector v_m^k as:

$$v_m^k = \left[F_i^k, F_j^k, \left|F_i^k - F_j^k\right|, {F_i^k}^{\mathrm{T}} W_s F_j^k\right]. \quad (18)$$

For instance, the mixture vector of posting pattern is v_m^{pp}, i.e.

$$v_m^{pp} = [pp_i, pp_j, |pp_i - pp_j|, pp_i{}^{\mathrm{T}} W_s pp_j] . \quad (19)$$

We feed the vector v_m^k into a fully connect network and obtain a similarity vector v_s^k for feature k. According to this method, we can obtain the similarity vectors including posting pattern similarity vector v_s^{pp}, writing pattern similarity vector v_s^{wp}, emotional fluctuation similarity vector v_s^{ef}, network similarity vector v_s^n. We fuse all similarity vectors and build a matching vector v_{match}:

$$v_{match} = [v_s^{pp}, v_s^{wp}, v_s^{ef}, v_s^n, v_s^p] , \quad (20)$$

$$v_s^p = [des_{i,j}, name_{i,j}, loc_{i,j}] . \quad (21)$$

Finally, we feed the vector v_{match} into a hidden layer, then a classifier is applied to identify accounts.

III. Experiment

A. Dateset

To validate the proposed model, we collect account pairs use a dataset from [4, 5], and supplement through Google Profiles and about.me website which provide a list of other accounts on different platforms. Then, we collect information of each account including profiles, history tweets, friend networks, etc. Table I provides more details.

TABLE I. Summary of the Dataset

Sites	#Accounts	#Tweets
Twitter	2602	963,900
Others		823,700

B. Baselines and Evaluation Metrics

The evaluation metrics we selected are precision, recall, and F1 score. Evaluation is performed through 10-fold cross-validation. For each split, we fit model to 80% data, tuned with 10% data, and tested on the remaining data.

To validate the performance of the proposed model, we compare it with baseline methods which introduced in related works: FRUI [6], MBS [7], Hybrid[8], HYDRA[9].

C. Model Training

The entire model is trained to minimize the cross-entropy error through mini-batch Adagrad which is an adaptive learning rate method. To avoid overfitting, the dropout with probability 0.5 is adopted. During the training process, we take an early stop strategy.

D. Experimental Results

Table II shows the experimental results of baselines and our model. The proposed model outperforms all baselines. The proposed model achieves the best performance with F1 value 0.8232. Among all methods, FRUI performs worst which focuses on relationship network. MBS uses statistics of behavioral patterns which performs better than FRUI. It partly demonstrates that online behavior pattern is an effective feature that can be used in this task. The HYDRA which combines profile, network, and content features performs best in all baselines. The experiment results demonstrate the effectiveness of the proposed model.

TABLE II. Performance of proposed model and baselines

Method	Precision	Recall	F1
FRUI	70.51	63.44	66.79
MBS	77.29	64.56	70.35
Hybrid	75.47	80.84	78.06
HYDRA	84.23	76.59	80.23
DDUM	**85.75**	**79.16**	**82.32**

IV. Conclusion

In this paper, we proposed a deep dynamic user mapping model (DDUM) by fusing dynamic and traditional features and developed a matching network to identify accounts across social networks. DDUM captures dynamic information through three encoders which encoding posting pattern, writing pattern, and emotional fluctuation. The posting pattern encoder is inspired by endogenous and exogenous factors of online behavior. The writing pattern encoder combines the pop culture attention mechanism and memory network to encode representation. To our best knowledge, this work is the first trial which applies deep learning in mapping social users with dynamic information.

Acknowledgment

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0820105 and 2016QY02D0305, the National Natural Science Foundation of China under Grants 71702181, 71621002. Linjing Li is the corresponding author.

References

[1] E. K. Vraga and L. Bode, "I do not believe you: how providing a source corrects health misperceptions across social media platforms," *Information, Communication & Society,* vol. 21, pp. 1337-1353, 2018.

[2] C. Cai, L. Li, and D. Zeng, "Detecting Social Bots by Jointly Modeling Deep Behavior and Content Information," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1995-1998.

[3] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701-710.

[4] P. Jain, P. Kumaraguru, and A. Joshi, "@ i seek'fb. me': Identifying users across multiple online social networks," in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 1259-1268.

[5] A. Malhotra, L. Totti, W. Meira Jr, P. Kumaraguru, and V. Almeida, "Studying user footprints in different online social networks," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, 2012, pp. 1065-1070.

[6] X. Zhou, X. Liang, H. Zhang, and Y. Ma, "Cross-platform identification of anonymous identical users in multiple social media networks," *IEEE transactions on knowledge and data engineering,* vol. 28, pp. 411-424, 2016.

[7] F. M. Naini, J. Unnikrishnan, P. Thiran, and M. Vetterli, "Where You Are Is Who You Are: User Identification by Matching Statistics," *IEEE Trans. Information Forensics and Security,* vol. 11, pp. 358-372, 2016.

[8] A. Kokkos, T. Tzouramanis, and Y. Manolopoulos, "A Hybrid Model for Linking Multiple Social Identities Across Heterogeneous Online Social Networks," in *International Conference on Current Trends in Theory and Practice of Informatics*, 2017, pp. 423-435.

[9] S. Liu, S. Wang, F. Zhu, J. Zhang, and R. Krishnan, "Hydra: Large-scale social identity linkage via heterogeneous behavior modeling," in *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, 2014, pp. 51-62.