Online Information Review

A user's personality prediction approach by mining network interaction behaviors on Facebook

Tsung-Yi Chen Meng-Che Tsai Yuh-Min Chen

A user's personality prediction approach by mining network interaction behaviors on Facebook

Tsung-Yi Chen
Department of Information Management, Nanhua University, Chiayi County, Taiwan, and
Meng-Che Tsai and Yuh-Min Chen
Institute of Manufacturing Information and Systems, National Cheng Kung University, Tainan, Taiwan

Received 24 August 2015
Revised 4 March 2016
Accepted 14 March 2016

Abstract

Purpose – For an enterprise, it is essential to win as many customers as possible. The key to successfully winning customers is often determined by understanding the personality characteristics of the object of communication in order to employ an effective communication strategy. An enterprise needs to obtain the personality information of target or potential customers. However, the traditional method for personality evaluation is extremely costly in terms of time and labor, and it cannot acquire customer personality information without their awareness. Therefore, the manner in which to effectively conduct automated personality predictions for a large number of objects is an important issue. The paper aims to discuss these issues.

Design/methodology/approach – The diverse social media that have emerged in recent years represent a digital platform on which users can publicly deliver speeches and interact with others. Thus, social media may be able to serve the needs of automated personality predictions. Based on user data of Facebook, the main social media platform around the world, this research developed a method for predicting personality types based on interaction logs.

Findings – Experimental results show that the Naïve Bayes classification algorithm combined with a feature selection algorithm produces the best performance for predicting personality types, with 70-80 percent accuracy.

Research limitations/implications – In this research, the dominance, inducement, submission, and compliance (DISC) theory was used to determine personality types. Some specific limitations were encountered. As Facebook was used as the main data source, it was necessary to obtain related data via Facebook's API (FB API). However, the data types accessible via FB API are very limited.

Practical implications – This research serves to build a universal model for social media interaction, and can be used to propose an efficient method for designing interaction features.

Originality/value – This research has developed an approach for automatically predicting the personality types of network users based on their Facebook interactions.

Keywords Facebook, Social media, DISC theory, Interaction feature, Personality predicting

Paper type Research paper

1. Introduction

All businesses need to attract and retain customers, and the most direct means of achieving this is by promoting the goods and services they offer. Such efforts rely on effective communication strategies, which are more likely to be successful if more is

The authors would like to thank the Ministry of Science and Technology of the Republic of China, Taiwan for financially supporting this research under Contract No. MOST 103-2221-E-343-003-MY3.

1468-4527
DOI 10.1108/OIR-08-2015-0267

known about the personality traits of the target audience. Personality traits determine how people behave, interact, and communicate with others (Knapp and Daly, 2002).

Personality information is of great value in the business arena. In addition to sales promotion behavior, personality information has been used in studies on pedagogy (Blignaut and Naude, 2008), interpersonal relationships (Baron and Wagele, 1995; Alessandra and O'Connor, 1998; Rosenberg and Silvert, 2012), and job performance. In the future, personality information may be used for the construction of adaptive systems, e-commerce, and recommendation systems. The fundamental reason for such extensive applications is that personalities determine the attitudes and responses of people to external environments.

There are many types of personality models and assessments for different applications. Traditionally, a subject's personality is assessed using either a questionnaire or observations made by experts. If a business needs to obtain personality-related information from large numbers of existing or potential customers, such approaches are obviously inefficient and impractical. Thus, it is necessary to develop a method that automatically obtains such information.

Fortunately, the rise of social media in recent years has provided many opportunities to address this issue, because online platforms such as Facebook (Ross *et al.*, 2009) and Twitter provide spaces in which users can reveal details about their attitudes (Golbeck *et al.*, 2011) and interactions with others. Thus, it is possible to use such sites to obtain useful information, from which certain personality traits can be inferred.

Some previous studies (Golbeck *et al.*, 2011; Adali and Golbeck, 2012; Adali *et al.*, 2012; Bai *et al.*, 2012; Moore and McElroy, 2012; Seidman, 2013; Ortigosa *et al.*, 2014) have shown that a user's personality score is related to the data recorded about them on social media. An earlier study also showed that a person's behavior can reveal their personality. However, these previous works did not undertake a close examination of the relationship between personality and a user's interactions with others, which can provide critical clues in traditional assessments.

The goal of this research is to develop an approach for automatically predicting the personality of network users from Facebook, currently the most popular social networking and media site (Ross *et al.*, 2009).

This research has some specific limitations. As Facebook is the data source for this research, the author had to obtain related data via Facebook's API (FB API). However, the data types accessible via FB API are very limited. For example, interaction logs cannot be obtained via FB API. Therefore, a separate program was provided to participants of the experiment to allow them to provide related data semi-manually.

2. Literature review

2.1 Facebook and personality

Social media are tools and platforms on which people share their opinions, views, and experiences with others. Ubiquitous social media (Jim Wu *et al.*, 2015) allows users to build social communities within their own account to disclose various detailed information and views to multiple people worldwide. Web users can use social media to create and share content pertaining to different subjects, exposing activities, opinions, feelings, and thoughts (Lima and Castro, 2014). Today, online social media such as YouTube, Google+, Facebook, Twitter, Instagram, Pinterest, and so on (Wikipedia, 2015a, b; Ross *et al.*, 2009) attracts many data scientists who wish to better understand behaviors and trends (Lima and Castro, 2014).

Facebook was developed by Mark Zuckerberg in 2004, and was initially only available to students enrolled at Harvard University. It has since grown to become the most popular social media site in the world. Facebook offers users the opportunity for self-presentation (Jim Wu *et al.*, 2015). In 2005, Facebook became available to the public at large, and by 2012 the total number of Facebook members exceeded one billion. Today, there are some 1.4 billion Facebook members. Thus, Facebook has a large number of users who are widely distributed across the world (Wikipedia, 2015a, b).

A distinguishing feature of Facebook is that it provides a variety of operating functions, which are some of the most complex among all of the available social media. By providing diversified functions, Facebook allows users to operate their personal profiles at will (Zhao *et al.*, 2008; Nadkarni and Hofmann, 2012) or interact with friends and strangers by sharing information such as text, audio, and video. One of Facebook's core functions is the "Timeline," allowing users to post messages or audio/video data to their own or others' spaces. Each of the posted messages can obtain a response. The latest data release or response is synchronized to the home pages of friends. Another core function is the "Like" button, which allows others to "Like" the status, photo, or comment of a person, business, public figure, or concept. Other functions include check in, album, note, transfer of messages, events, and invitations to others. All these functions allow users to build relationships over the internet more easily, regardless of time and space restrictions.

With the variety of functions mentioned above, Facebook simulates a virtual social environment, allowing users to interact with others online. Facebook not only retains the digital data of users, but also provides APIs to allow external developers to use such digital data (with user consent) for various applications and studies.

For such a virtual social environment in which users are the basic units, people have begun to study the correlation between users and their behavior in the environment. For example, some scholars started researching the correlation between the analysis of a user from a psychological perspective and the user's behavior on Facebook. Back *et al.* (2010) indicated that Facebook user data faithfully reflects the real personality characteristics of Facebook users. Golbeck *et al.* (2011) attempted to analyze the basic data fields published by subjects on Facebook using the five factor model (FFM), and were able to predict FFM scores with considerable accuracy through machine learning. Ross *et al.* (2009) investigated how the FFM of personality related to Facebook use. Some studies have focused on the correlation between the personality model test scores of the subjects and the features of their operating behavior on Facebook (Moore and McElroy, 2012; Seidman, 2013; Lee *et al.*, 2014). Ortigosa *et al.* (2014) predicted FFM scores by analyzing a small amount of interactive information generated by Facebook users, and proved that it is feasible to predict personality characteristics based on interaction features.

2.2 Personality model and dominance, inducement, submission, and compliance (DISC) theory

Personalities (also known as personal characters or behavior styles) refer to the consistent psychological dispositions or features inherent in humankind (Wikipedia, 2015a, b). Youyou *et al.* (2015) stated that personality is a key driver behind people's interactions, behavior, and emotions, and noted that judging others' personalities is an essential skill for successful social living. By interacting with external environments at different times and in different places, an individual's thoughts and behavior are affected by their psychological disposition or features, that is, the unique external behavior of an individual is a manifestation of his/her inner personality. In the *Handbook of Interpersonal Communication* (Knapp and Daly, 2002), the differences in individual personalities give rise to differences in communication modes.

To date, scholars have continued to research various personalities, and have developed diverse theories on personality categories. In recent times, common personality models such as FFM, the Enneagram of Personality (often called the "Enneagram"), the Myers-Briggs type indicator (MBTI), and DISC theory have all been proposed and discussed.

FFM is widely used in contemporary psychological studies, and is considered the most suitable personality model for interpreting personality characteristics (McCrae and Costa, 1997; Funder, 2000). FFM is oriented toward five factors, and evaluates these in each subject via questionnaire surveys without definitely categorizing the subject. The Enneagram definitely classifies people into nine types (or possibly 18 types if composite relationships are considered). However, the Enneagram is not an orthodox theory of personality psychology, as it lacks a full theoretical justification, and is therefore somewhat controversial. In general, the Enneagram is similar to a classification method based on the integration of experiences. MBTI was proposed by (a Swiss psychologist) in his work on *Psychological Types*. MBTI classifies humankind into 16 types by dichotomizing four orientations, and is widely used across the world. DISC was proposed by Marston (1928), a US Psychologist, in *Emotions of Normal People*. DISC classifies people according to four behavior styles with the distinctive characteristics of DISC.

The four personality models are distinct from one another, and their corresponding test results are applicable in different scenarios. For example, FFM provides quantitative scores for five factors, and is frequently used by researchers to calculate correlation results. MBTI is widely used for choosing occupations, whereas DISC is used to determine a strategy for communication with a target. For this research, DISC is chosen as the personality model for the scenario in which an enterprise can implement adaptive marketing strategies suited to customers with different personality types. DISC theory is elaborated in the following sections.

After Marston proposed DISC theory, Boyd (1994) further improved the model and developed a questionnaire survey. The new model maintained four personality types, among which "inducement" was replaced with "influence" and "submission" with "steady." Boyd believed that DISC should be located in a four-quadrant space comprising two perpendicular axes. The vertical axis represents "pace," whereas the horizontal axis represents "priority." People with high pace are self-confident and always attempt to change everything, whereas those with low pace are self-restrained and conservative. Regarding priority, task-oriented people are devoted to ongoing tasks, prefer individual work, are relatively reserved, and like to hide personal moods. Human-oriented people attach importance to interpersonal relationships, and are unwilling to be fettered by conventions (Boyd, 1994). The four behavior styles have distinctive personality characteristics. Figure 1 and Table I show the general descriptions of the personality characteristics and possible behavior characteristics in the corresponding social media posts, respectively (MBAlib, 2015; Rosenberg and Silvert, 2012).

The results of the DISC test can be applied to various commercial fields such as recruitment, employee evaluation, career planning, and leadership (MBAlib, 2015). In recent years, many enterprises have applied DISC theory to sales promotions. Sales people are trained to use appropriate sales talk by observing the DISC types of their customers in an attempt to boost the probability of a transaction.

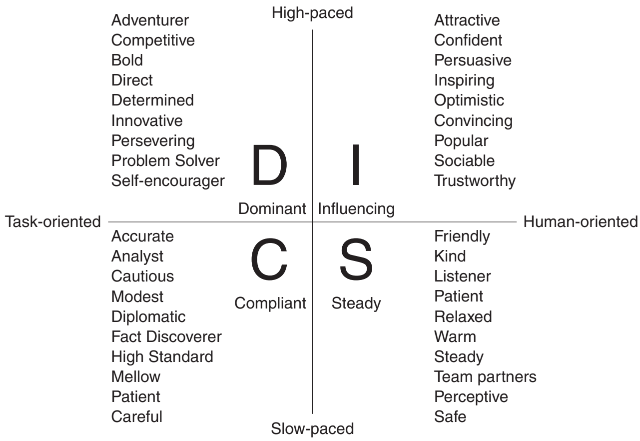

Figure 1. DISC quadrant and general descriptions

Type D	Type I	Type S	Type C
Steadfast and direct	Enthusiastic	Mild and sincere	Unobvious emotion
Self-confident	Excited	Gentle and friendly	Pay attention to details
Expert-like speak	Exaggerated	Depressed	Logical
Inordinate	Cheerful and vibrant	Unobvious personal assertion	Quote references
Instill views into others	Optimistic	Unobvious characteristic	Self-controlled
Strong and egocentric	Playful	Soft attitude	Like to analyze
	Attract attention		

Table I. Possible behavior characteristics in the corresponding social media posts

3. Personality prediction

3.1 Adaptive marketing support systems (AMSS)

In future, the automated personality prediction method proposed in this paper may be applicable to AMSS. Figure 2 shows the configuration plan of an AMSS.

The AMSS modules and their functions are briefly described as follows:

(1) Cloud service interface – software as a service: AMSS has a personalized webpage (user interface) that provides virtual consultancy services for enterprises that wish to determine the personality types of their customers. Enterprises need only "commit" the Facebook interaction log of customers in batches, and AMSS sends the data to the backend for calculation and prediction, and then makes a "response" to the personality types of the submitted objects. AMSS cannot provide the Facebook interaction log of an enterprise's target customers, and the enterprise must access the interface to obtain customer data.

(2) Import service of training data (webpage or application): AMSS creates a separate window to allow the inflow of new training data. The window may be a personality test webpage or application (App) within which the FB API needs to be embedded. The window serves to obtain the standard personality type results and Facebook data of users. This service is capable of steadily and continuously importing training data into a database (DB). As the volume of imported training data increases, AMSS can be adjusted continuously to improve the prediction accuracy.

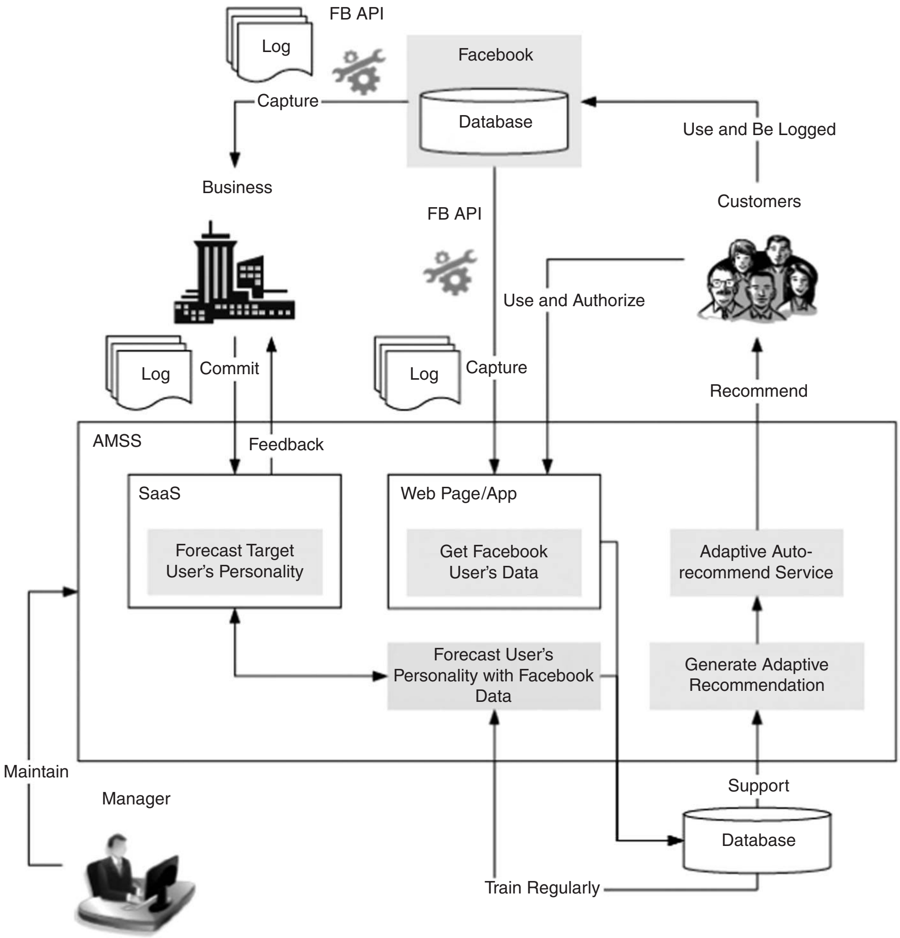

Figure 2.
Configuration plan for AMSS

(3) Automatic recommendation services for adaptability: this service can be enabled when the DB reaches a certain volume of data. An information presentation template is customized for people with different personality types. Enterprises that use this service can employ an appropriate information presentation mode to recommend its products or services to users in the DB.

(4) Personality type predictions based on Facebook data: the functions specified in (1) and (3) are major AMSS functions that require support by the personality type prediction program, and can be viewed as the application of personality type methods. Therefore, the first step in developing the AMSS is to develop the prediction method and implement it as an operable program.

(5) DB of the personality type prediction system: the DB stores manually or automatically marked interaction logs used by the personality type prediction function for regular training in order to update the classification rules and improve the classification program with time.

3.2 User interactive model on Facebook

Most of the user interactions on Facebook concern certain objects relating to other users, for example, "Alice likes Bill's photo" or "Cathy shared Denny's link," as shown in Figure 3.

Based on observations of Facebook user interactions, the four-tier social media personal single interaction model (SIM) shown in Figure 4 was created. The four tiers of SIM are: "actor-tier," which represents a Facebook user; "behavior-tier," which represents the Facebook interactions conducted by said user; "target user (TU)-tier," which represents the target (also a Facebook user) of the interactions conducted by the user; and "target object (TO)-tier," which represents the Facebook item that is the object of the interactions conducted by the user. An object cannot exist independently, and must therefore be owned by a TU. Therefore, the "TO-tier" is below the "TU-tier." The "behavior-tier," "TU-tier," and "TO-tier" comprise their respective elements on Facebook, as presented in Figure 4.

Not all conditions of the arbitrary combination of elements will occur, although the possible user interactions are listed in Table II. Table III lists the elements of each tier of the SIM on Facebook, and gives the abbreviations for such elements.

3.3 Methodology of feature design

"Feature design" is the first issue that should be considered when features serve as the criterion for classification. How to accurately design useful features depends on the extent to which the designer understands the classification objectives, and even some intuitive elements. It is possible to reduce the manual judgment costs of independent feature designs by researchers as follows: first, a large number of default features (DFs) are generated in a structured manner; then, useful key features (KFs) are chosen from numerous DFs through "feature selection."

Based on the regular behavior patterns presented by SIM, the author referred to previous research findings (Golbeck *et al.*, 2011; Adali and Golbeck, 2012; Adali *et al.*, 2012; Ortigosa *et al.*, 2014) to design a default feature set (DFS) comprising up to 612 DFs. This DFS can be applied to a personality prediction method. The methods for

Source: Facebook

Figure 3. Sample of users' interactions on Facebook

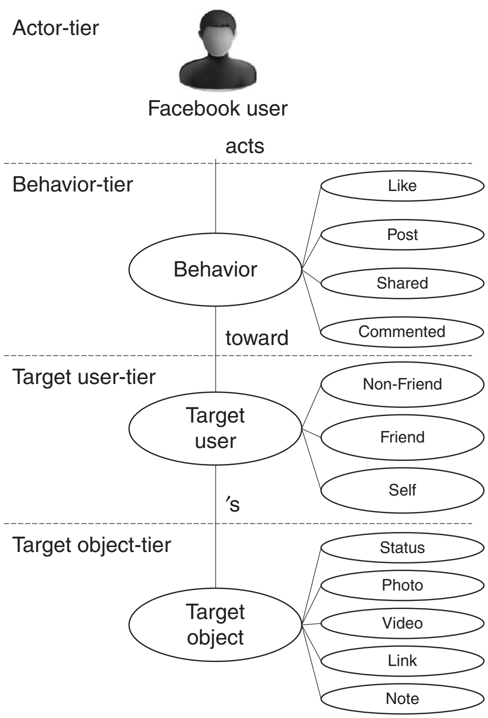

Figure 4. Four-tier social media personal single interaction model (SIM) – example based on Facebook

Table II. Possible user interactions on Facebook

Behavior	TO / TU	Status	Photo	Video	Link	Note	Null
Like	Non-Friend	v	v	v	v	v	v
	Friend	v	v	v	v	v	
	Self	v	v	v	v	v	
Post	Non-Friend	v	v	v	v		
	Friend	v	v	v	v		
	Self	v	v	v	v	v	
Share	Non-Friend	v	v	v	v	v	
	Friend	v	v	v	v	v	
	Self	v	v	v	v	v	
Comment	Non-Friend	v	v	v	v	v	
	Friend	v	v	v	v	v	
	Self	v	v	v	v	v	

Tier	Element	Abbreviation
Behavior (Σ_B)	Like	L
	Post	P
	Share	S
	Comment	C
Target user (Σ_U)	Non-friend	Nf
	Friend	Fd
	Self	Sf
Target object (Σ_O)	Status	St
	Photo	Pt
	Video	Vd
	Link	Lk
	Note	Nt

Table III. Elements and abbreviations of each tier of the SIM on Facebook

generating these DFs share the same logical structure as the tree diagram in Figure 5. "A user" at the root node represents the Facebook user (a Facebook user in the SIM actor-tier) that serves as an observation object. Nodes other than the root node then represent feature classes, and leaf nodes represent feature types.

In this research, features were designed as follows: four scope types (global, TU, TO, and target user-object (TU-O)) were selected to calculate the features that correspond to the four behavior types (like, post, share, and comment) available in SIM. Here, "scope" refers to the extent to which the object of an interaction is considered. The following describes the four scope types:

(1) Global: features within the global scope only consider the status of the current "behavior type," regardless of "TOs" or "target articles." For example, the global features of the "like" behavior only consider the status of the "like" behavior. Statistics of all feature values are measured based on interactions with all behavior types, such as "like."

(2) TU: features within the TU scope need to consider the status of the current "behavior type" and "TOs," but do not consider the "target articles." For example, the TU features of the "like" behavior consider the matching status between the "like" behavior and the defined "TO type" during interactions. "TO types" are defined as friends (Fd), non-friends (Nf), and self (Sf). In the feature instances within the TU scope of the "like" behavior, there are some related features for which the "like" behavior is executed for friends or non-friends, as well as the features for the executed "like" behavior.

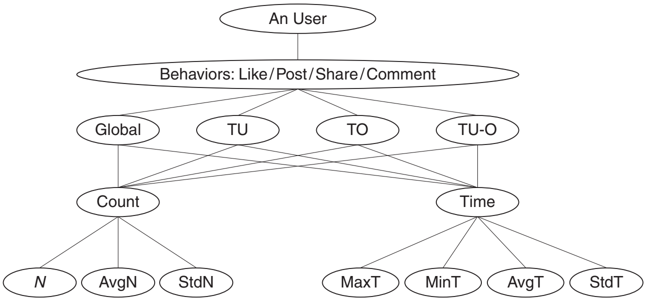

Figure 5. Structure model of interaction features

(3) TO: TOs are similar to TUs, except that the features within the TO consider "target articles" instead of "TOs." For example, the TO features of the "like" behavior consider the various occurrence conditions of the combination between the "like" behavior and the defined "target article type." "Target article types" are defined as dynamic essays or general posts (Ps), photos (Pts), videos (Vds), links (Lks), and network blogs (Nts).

(4) TU-O: features within the TU-O scope need to consider the combinations between "behavior type," "TO type," and "target article type." For example, the TU-O features of the "like" behavior include combinations with a "like" during interactions, such as "Ps or Pt or Vd or Lk or Nt" belonging to "Fd or Nf or Sf."

Features within each scope may be further classified into two types, namely, count and time. Count denotes that the feature values must be calculated by counting, such as the basic number of times (*N*), an average value (AvgN), or standard deviation (StdN). Time implies that the feature values must be calculated according to some time range, and may be further classified as the maximum time range (MaxT), minimum time range (MinT), average time range (AvgT), and standard deviation of time range (StdT).

The various feature types listed above can have one or multiple instances. Table IV presents some features of the "like" behavior generated using this method. Because of the varying lengths of interaction records in the sample data, the "*N*" type (shown in dark gray in Table IV) is transformed into the "Nw" type (shown in the data series below the dark gray data series in Table IV). The feature value measurements are transformed from a pure number to their proportion in the parent body.

3.4 Interaction feature classifier (IFC)

Classification according to feature values is a common approach. Such classification methods include two stages: "training" and "application" (or "classification"). The IFC shown in Figure 6 includes four steps, the first three are for training, and the final step executes the classification:

(1) Collection of sufficient training data: to implement this step, this research developed a data collection program for the personality test function, "ILReaper." The test content can be determined by the personality model. The DISC evaluation form in *The Platinum Rule* (Alessandra and O'Connor, 1998) is used in this research, because DISC is being used as the personality model. After a subject uses this program, the server receives the personality test scores and personality type classification results, and simultaneously obtains the user Facebook interaction log.

(2) Screening of KFs: in this research, not all DFs from the DFS contribute to the personality classification. Therefore, it is necessary to select a subset of appropriate DFs to train the classifier. This process is called "feature selection." To implement this step, DF values must be calculated for all training data. Feature selection can then be conducted in various ways. In this research, different methods were tested to obtain screening results, and those that were applicable were chosen as the feature selection methods for the system.

(3) Classifier training and key feature set (KFS) adjustment: the set of KFs obtained in Step (2) is used to train the classifier model. If the test results show that the classification accuracy is unsatisfactory, we return to Step (2) and adjust the KFS or use another classifier. In this research, multiple classifier algorithms were tested, and their classification accuracies were compared.

Behavior type	Target type	Count type	Abbreviation type	Description	Abbreviation
Like	Global	Cout	*N*	The number of **Like**s A made	N_L
				The weight of number of **Like**s A made	Nw_L
				The number of **individual user** A made **Like**s	N_UL
				The weight of number of **individual user** A made **Like**s	Nw_UL
			AvgN	Average of **Like**s A made for **individual user**s. (N_L/N_UL)	AvgN_L
			StdN	Standard Deviation of number of Likes A made to **individual user**s	StdN_LU
		Time	MaxT	Maximum time interval between adjacent **Like**s A made	MaxT_L
			MinT	Minimum time interval between adjacent **Like**s A made	MinT_L
			AvgT	Average time interval of adjacent **Like**s A made	AvgT_L
			StdT	Standard Deviation of time interval between adjacent **Like**s A made	StdT_L
	Target user, TU	Count	*N*	The number of **Like**s A made to **non-friends**	N_LNf
				The weight of number of **Like**s A made to **non-friends**. Scope: All interactions	Nw_LNf
				The weight of number of **Like**s A made to **non-friends**. Scope: All Like interactions	Nw_LLNf
				The number of **Like**s A made to **friends**	N_LFd
				The weight of number of **Like**s A made to **friends**. Scope: All interactions	Nw_LFd
				The weight of number of **Like**s A made to **friends**. Scope: All Like interactions	Nw_LLFd
				The number of **Like**s A made to **self**	N_LSf
				The weight of number of **Like**s A made to **self**. Scope: All interactions	N_LSf
				The weight of number of **Like**s A made to **self**. Scope: All Like interactions	N_LSf
				The number of distinct **non-friend**s A made **Like**s	N_NfUL
				The number of distinct **friend**s A made **Like**s (N_Fd: the number of all A's friends)	N_FdUL
			AvgN	Average of **Like**s A made for **non-friend**. (N_LNf/N_NfUL)	AvgN_LNfL
				Average of **Like**s A made for **all A's friend**. (N_LFd/N_Fd)	AvgN_LFd
				Average of **Like**s A made for **A's Liked friend**. (N_LFd/N_FdUL)	AvgN_LFdL
			StdN	Standard Deviation of number of Likes A made to non-friends	StdN_LNfUL
				Standard Deviation of number of Likes A made to friends	StdN_LFdUL
		Time	MaxT	Maximum time interval between adjacent **Like**s A made to **non-friend**s	MaxT_LNfL
				Maximum time interval between adjacent **Like**s A made to **friend**s	MaxT_LFdL
				Maximum time interval between adjacent **Like**s A made to **self**	MaxT_LSfL
			MinT	Minimum time interval between adjacent **Like**s A made to **non-friend**s	MinT_LNfL
				Minimum time interval between adjacent **Like**s A made to **friend**s	MinT_LFdL
				Minimum time interval between adjacent **Like**s A made to **self**	MinT_LSfLk
			AvgT	Average time interval between adjacent **Like**s A made to **non-friend**s	AvgT_LNfL
				Average time interval between adjacent **Like**s A made to **friend**s	AvgT_LFdL
				Average time interval between adjacent **Like**s A made to **self**	AvgT_LSfLk
			StdT	Standard Deviation of time interval between adjacent **Like**s A made to **non-friend**s	StdT_LNfL
				Standard Deviation of time interval between adjacent **Like**s A made to **friend**s	StdT_LFd
				Standard Deviation of time interval between adjacent **Like**s A made to **self**	StdT_LSf
	Target object, TO	Count	*N*	The number of **Like**s A made to **post**s	N_LPs
				The weight of number of **Like**s A made to **post**s. Scope: All interactions	Nw_LPs
				The weight of number of **Like**s A made to **post**s. Scope: All Like interactions	Nw_LLPs
				The number of **Like**s A made to **photo**s	N_LPt
				The weight of number of **Like**s A made to **photo**s. Scope: All interactions	Nw_LPt
				The weight of number of **Like**s A made to photos. Scope: All Like interactions	Nw_LLPt
				The number of **Like**s A made to **video**s	N_LVd
				The weight of number of **Like**s A made to **video**s. Scope: All interactions	Nw_LVd
				The weight of number of **Like**s A made to videos. Scope: All Like interactions	Nw_LLVd
				The number of **Like**s A made to **link**s	N_LLk
				The weight of number of **Like**s A made to **link**s. Scope: All interactions	Nw_LLk
				The weight of number of **Like**s A made to links. Scope: All Like interactions	Nw_LLLk
				The number of **Like**s A made to **note**s	N_LNt
				The weight of number of **Like**s A made to **note**s. Scope: All interactions	Nw_LNt

(*continued*)

Table IV.
Some features of the "like" behavior

				The weight of number of **Likes** A made to notes. Scope: All Like interactions	Nw_LLNt
		Time	MaxT	Maximum time interval between adjacent **Like**s A made to **post**s	MaxT_LPsL
				Maximum time interval between adjacent **Like**s A made to **photo**s	MaxT_LPtL
				Maximum time interval between adjacent **Like**s A made to **video**s	MaxT_LVdL
				Maximum time interval between adjacent **Like**s A made to **link**s	MaxT_LLkL
				Maximum time interval between adjacent **Like**s A made to **note**s	MaxT_LNtL
			MinT	Minimum time interval between adjacent **Like**s A made to **post**s	MinT_LPsL
				Minimum time interval between adjacent **Like**s A made to **photo**s	MinT_LPtL
				Minimum time interval between adjacent **Like**s A made to **video**s	MinT_LVdL
				Minimum time interval between adjacent **Like**s A made to **link**s	MinT_LLkL
				Minimum time interval between adjacent **Like**s A made to **note**s	MinT_LNtL
			AvgT	Average time interval between adjacent **Like**s A made to **post**s	AvgT_LPsL
				Average time interval between adjacent **Like**s A made to **photo**s	AvgT_LPtL
				Average time interval between adjacent **Like**s A made to **video**s	AvgT_LVdL
				Average time interval between adjacent **Like**s A made to **link**s	AvgT_LLkL
				Average time interval between adjacent **Like**s A made to **note**s	AvgT_LNtL
			StdT	Standard Deviation of time interval between adjacent **Like**s A made to **statuse**s	StdT_LStL
				Standard Deviation of time interval between adjacent **Like**s A made to **photo**s	StdT_LPtL
				Standard Deviation of time interval between adjacent **Like**s A made to **video**s	StdT_LVdL
				Standard Deviation of time interval between adjacent **Like**s A made to **link**s	StdT_LLkL
				Standard Deviation of time interval between adjacent **Like**s A made to **note**s	StdT_LNtL
	Target user-object, TU-O	Count	*N*	The number of **Likes** A made to **non-friend**'s **statuses**	N_LNfSt
				The number of **Likes** A made to **non-friend**'s **photos**	N_LNfPt
				The number of **Likes** A made to **non-friend**'s **videos**	N_LNfVd
				The number of **Likes** A made to **non-friend**'s **links**	N_LNfLk
				The number of **Likes** A made to **non-friend**'s **notes**	N_LNfNt
			AvgN	Average of **Likes** A made to **non-friend**'s **statuses**	AvgN_LNfSt
				Average of **Likes** A made to **non-friend**'s **photos**	AvgN_LNfPt
				Average of **Likes** A made to **non-friend**'s **videos**	AvgN_LNfVd
				Average of **Likes** A made to **non-friend**'s **links**	AvgN_LNfLk
				Average of **Likes** A made to **non-friend**'s **notes**	AvgN_LNfNt
			EtpN	Entropy of number of **Likes** A made to **non-friend**'s **statuses**	EtpN_LNfSt
				Entropy of number of **Likes** A made to **non-friend**'s **photos**	EtpN_LNfPt
				Entropy of number of **Likes** A made to **non-friend**'s **videos**	EtpN_LNfVd
				Entropy of number of **Likes** A made to **non-friend**'s **links**	EtpN_LNfLk
				Entropy of number of **Likes** A made to **non-friend**'s **notes**	EtpN_LNfNt
			N	The number of **Likes** A made to **friend**'s **statuses**	N_LFSt
				The number of **Likes** A made to **friend**'s **photos**	N_LFPt
				The number of **Likes** A made to **friend**'s **videos**	N_LFVd
				The number of **Likes** A made to **friend**'s **links**	N_LFLk
				The number of **Likes** A made to **friend**'s **notes**	N_LFNt
			AvgN	Average of **Likes** A made to **friend**'s **statuses**	AvgN_LFSt
				Average of **Likes** A made to **friend**'s **photos**	AvgN_LFPt
				Average of **Likes** A made to **friend**'s **videos**	AvgN_LFVd
				Average of **Likes** A made to **friend**'s **links**	AvgN_LFLk
				Average of **Likes** A made to **friend**'s **notes**	AvgN_LFNt
			EtpN	Entropy of number of **Likes** A made to **friend**'s **statuses**	EtpN_LFSt
				Entropy of number of **Likes** A made to **friend**'s **photos**	EtpN_LFPt
				Entropy of number of **Likes** A made to **friend**'s **videos**	EtpN_LFVd
				Entropy of number of **Likes** A made to **friend**'s **links**	EtpN_LFLk
				Entropy of number of **Likes** A made to **friend**'s **notes**	EtpN_LFNt
			N	The number of **Likes** A made to **self**'s **statuses**	N_LSfSt
				The number of **Likes** A made to **self**'s **photos**	N_LSfPt
				The number of **Likes** A made to **self**'s **videos**	N_LSfVd
				The number of **Likes** A made to **self**'s **links**	N_LSfLk
				The number of **Likes** A made to **self**'s **notes**	N_LSfNt

Table IV. (*continued*)

			AvgN	Average of **Likes** A made to **self's statuses**	AvgN_LSfSt
				Average of **Likes** A made to **self's photos**	AvgN_LSfPt
				Average of **Likes** A made to **self's videos**	AvgN_LSfVd
				Average of **Likes** A made to **self's links**	AvgN_LSfLk
				Average of **Likes** A made to **self's notes**	AvgN_LSfNt
			EtpN	Entropy of number of **Likes** A made to **self's statuses**	EtpN_LSfSt
				Entropy of number of **Likes** A made to **self's photos**	EtpN_LSfPt
				Entropy of number of **Likes** A made to **self's videos**	EtpN_LSfVd
				Entropy of number of **Likes** A made to **self's links**	EtpN_LSfLk
				Entropy of number of **Likes** A made to **self's notes**	EtpN_LSfNt
		Time	MaxT	Maximum time interval between adjacent **Likes** A made to **non-friend's statuses**	MaxT_LNfSt
				Maximum time interval between adjacent **Likes** A made to **non-friend's photos**	MaxT_LNfPt
				Maximum time interval between adjacent **Likes** A made to **non-friend's videos**	MaxT_LNfVd
				Maximum time interval between adjacent **Likes** A made to **non-friend's links**	MaxT_LNfLk
				Maximum time interval between adjacent **Likes** A made to **non-friend's notes**	MaxT_LNfNt
			MinT	Minimum time interval between adjacent **Likes** A made to **non-friend's statuses**	MinT_LNfSt
				Minimum time interval between adjacent **Likes** A made to **non-friend's photos**	MinT_LNfPt
				Minimum time interval between adjacent **Likes** A made to **non-friend's videos**	MinT_LNfVd
				Minimum time interval between adjacent **Likes** A made to **non-friend's links**	MinT_LNfLk
				Minimum time interval between adjacent **Likes** A made to **non-friend's notes**	MinT_LNfNt
			AvgT	Average time interval between adjacent **Likes** A made to **non-friend's statuses**	AvgT_LNfSt
				Average time interval between adjacent **Likes** A made to **non-friend's photos**	AvgT_LNfPt
				Average time interval between adjacent **Likes** A made to **non-friend's videos**	AvgT_LNfVd
				Average time interval between adjacent **Likes** A made to **non-friend's links**	AvgT_LNfLk
				Average time interval between adjacent **Likes** A made to **non-friend's notes**	AvgT_LNfNt
			EtpT	Entropy of time interval between adjacent **Likes** A made to **non-friend's statuses**	EtpT_LNfSt
				Entropy of time interval between adjacent **Likes** A made to **non-friend's photos**	EtpT_LNfPt
				Entropy of time interval between adjacent **Likes** A made to **non-friend's videos**	EtpT_LNfVd
				Entropy of time interval between adjacent **Likes** A made to **non-friend's links**	EtpT_LNfLk
				Entropy of time interval between adjacent **Likes** A made to **non-friend's notes**	EtpT_LNfNt
			StdT	Standard Deviation of time interval between adjacent **Likes** A made to **non-friend's statuses**	StdT_LNfSt
				Standard Deviation of time interval between adjacent **Likes** A made to **non-friend's photos**	StdT_LNfPt
				Standard Deviation of time interval between adjacent **Likes** A made to **non-friend's videos**	StdT_LNfVd
				Standard Deviation of time interval between adjacent **Likes** A made to **non-friend's links**	StdT_LNfLk
				Standard Deviation of time interval between adjacent **Likes** A made to **non-friend's notes**	StdT_LNfNt
			MaxT	Maximum time interval between adjacent **Likes** A made to **friend's statuses**	MaxT_LFSt
				Maximum time interval between adjacent **Likes** A made to **friend's photos**	MaxT_LFPt
				Maximum time interval between adjacent **Likes** A made to **friend's videos**	MaxT_LFVd
				Maximum time interval between adjacent **Likes** A made to **friend's links**	MaxT_LFLk
				Maximum time interval between adjacent **Likes** A made to **friend's notes**	MaxT_LFNt
			MinT	Minimum time interval between adjacent **Likes** A made to **friend's statuses**	MinT_LFSt
				Minimum time interval between adjacent **Likes** A made to **friend's photos**	MinT_LFPt
				Minimum time interval between adjacent **Likes** A made to **friend's videos**	MinT_LFVd
				Minimum time interval between adjacent **Likes** A made to **friend's links**	MinT_LFLk
				Minimum time interval between adjacent **Likes** A made to **friend's notes**	MinT_LFNt
			AvgT	Average time interval between adjacent **Likes** A made to **friend's statuses**	AvgT_LFSt
				Average time interval between adjacent **Likes** A made to **friend's photos**	AvgT_LFPt
				Average time interval between adjacent **Likes** A made to **friend's videos**	AvgT_LFVd
				Average time interval between adjacent **Likes** A made to **friend's links**	AvgT_LFLk
				Average time interval between adjacent **Likes** A made to **friend's notes**	AvgT_LFNt
			EtpT	Entropy of time interval between adjacent **Likes** A made to **friend's statuses**	EtpT_LFSt
				Entropy of time interval between adjacent **Likes** A made to **friend's photos**	EtpT_LFPt
				Entropy of time interval between adjacent **Likes** A made to **friend's videos**	EtpT_LFVd
				Entropy of time interval between adjacent **Likes** A made to **friend's links**	EtpT_LFLk
				Entropy of time interval between adjacent **Likes** A made to **friend's notes**	EtpT_LFNt

(*continued*)

Table IV.

Table IV.

			StdT	Standard Deviation of time interval between adjacent **Likes** A made to **friend's statuses**	StdT_LFSt
				Standard Deviation of time interval between adjacent **Likes** A made to **friend's photos**	StdT_LFPt
				Standard Deviation of time interval between adjacent **Likes** A made to **friend's videos**	StdT_LFVd
				Standard Deviation of time interval between adjacent **Likes** A made to **friend's links**	StdT_LFLk
				Standard Deviation of time interval between adjacent **Likes** A made to **friend's notes**	StdT_LFNt
			MaxT	Maximum time interval between adjacent **Likes** A made to **self's statuses**	MaxT_LSfSt
				Maximum time interval between adjacent **Likes** A made to **self's photos**	MaxT_LSfPt
				Maximum time interval between adjacent **Likes** A made to **self's videos**	MaxT_LSfVd
				Maximum time interval between adjacent **Likes** A made to **self's links**	MaxT_LSfLk
				Maximum time interval between adjacent **Likes** A made to **self's notes**	MaxT_LSfNt
			MinT	Minimum time interval between adjacent **Likes** A made to **self's statuses**	MinT_LSfSt
				Minimum time interval between adjacent **Likes** A made to **self's photos**	MinT_LSfPt
				Minimum time interval between adjacent **Likes** A made to **self's videos**	MinT_LSfVd
				Minimum time interval between adjacent **Likes** A made to **self's links**	MinT_LSfLk
				Minimum time interval between adjacent **Likes** A made to **self's notes**	MinT_LSfNt
			AvgT	Average time interval between adjacent **Likes** A made to **self's statuses**	AvgT_LSfSt
				Average time interval between adjacent **Likes** A made to **self's photos**	AvgT_LSfPt
				Average time interval between adjacent **Likes** A made to **self's videos**	AvgT_LSfVd
				Average time interval between adjacent **Likes** A made to **self's links**	AvgT_LSfLk
				Average time interval between adjacent **Likes** A made to **self's notes**	AvgT_LSfNt
			EtpT	Entropy of time interval between adjacent **Likes** A made to **self's statuses**	EtpT_LSfSt
				Entropy of time interval between adjacent **Likes** A made to **self's photos**	EtpT_LSfPt
				Entropy of time interval between adjacent **Likes** A made to **self's videos**	EtpT_LSfVd
				Entropy of time interval between adjacent **Likes** A made to **self's links**	EtpT_LSfLk
				Entropy of time interval between adjacent **Likes** A made to **self's notes**	EtpT_LSfNt
			StdT	Standard Deviation of time interval between adjacent **Likes** A made to **self's statuses**	StdT_LSfSt
				Standard Deviation of time interval between adjacent **Likes** A made to **self's photos**	StdT_LSfPt
				Standard Deviation of time interval between adjacent **Likes** A made to **self's videos**	StdT_LSfVd
				Standard Deviation of time interval between adjacent **Likes** A made to **self's links**	StdT_LSfLk
				Standard Deviation of time interval between adjacent **Likes** A made to **self's notes**	StdT_LSfNt

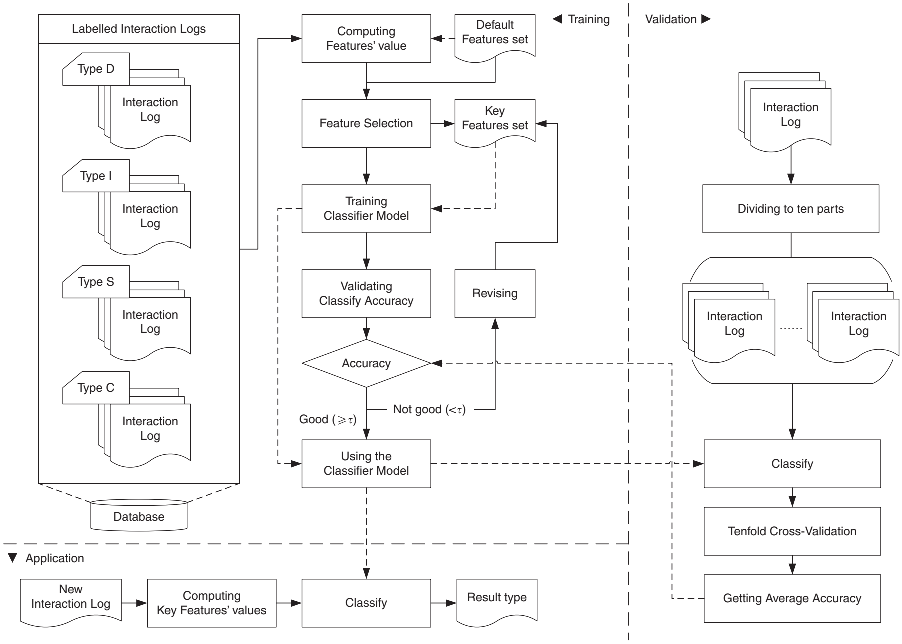

Figure 6.
Process of IFC

(4) Personality classification: in future, the classifier will automatically calculate the user's personality type by obtaining their Facebook interaction log to determine the classifier parameters. As a result, the objective of automatically predicting personality types will be accomplished without recourse to traditional questionnaires.

4. Implementation

For this research, Weka (University of Waikato, 2015) was used to conduct feature selection and train the classifier model. Weka is a Java-based open-source software package developed for data mining and machine learning. Weka offers multiple built-in algorithms that are commonly used for data mining, and is an appropriate tool for verifying the feasibility of this method.

At present, user interaction logs cannot be obtained through any function of the official FB APIs. Thus it is necessary to explore other channels. In this research, the self-developed program "ILReaper" was used to allow subjects to provide their Facebook interaction logs.

If a subject is willing to provide their profile, the Facebook interaction log can be uploaded to an FTP server built for this research. The detailed steps are as follows: log into the Facebook page embedded into the browser; open the "view activity log" page; manually press the end button in the keyboard 60 times; and click the upload button.

The subject needs to press the end button manually. The activity log page of Facebook is a dynamic webpage, and users must scroll to the bottom to read as much content as possible. Experimentally, an additional 30 interaction logs can be read every time a user presses the end button, and displaying the total page requires approximately 0.5-1 s (depending on network speed). To reach a compromise between the required data volume and user convenience, we recommend pressing the end button 60 times to provide 1,000-2,000 interaction logs within a period of 1-1.5 min. These interaction logs typically cover data generated over six months to a year. The entire process does not inconvenience the subject, thus enhancing their willingness to attend the test.

Data uploaded by the subject contains all the source code of the HTML webpages, including the content of the activity log page. In addition to the interaction log section required for this research, the source code contains a lot of unwanted content (e.g. sidebars and advertisements). Therefore, this research resorted to the regular expression (RE) technology commonly used in the field of text mining. RE technology is capable of extracting data from messy webpages, and of constructing each interaction for each user as a SIM instance.

The DFS is generated in a structured manner. Figure 7 shows the algorithm for computing each feature.

After the DF calculation is complete, a DF value table is generated for each subject, and a subject-feature value matrix (shown in Figure 8) is generated for all subjects and DFs. Each element $Value_{i,j}$ in the matrix is the feature value of column $Feature_j$ that corresponds to row $Subject_i$. For example, $Value_{2,1}$ is the feature value of $Feature_1$ that corresponds to $Subject_2$.

The matrix content is transferred into a comma-separated values (CSV) file for use as the Weka input data, and multiple built-in feature selection methods of Weka are used to obtain different recommended KFSs. After obtaining a KFS, the multiple built-in classification algorithms of Weka are used to train the personality type classifier model. The experiment section presents the classification accuracies of the matching between different feature selection methods and classification algorithms.

Input:	Interaction log for a subject, interaction type defined as B, target object type defined as TU, and target article type defined as TO.
Output:	Feature Name vs Feature Value Table (FVT).
Lines 1-40:	Traverse all considered behavior types, and calculate the related Count and Time feature values. The considered behavior types include Like (L), Post (P), Share (S), and Comment (C).
Lines 2-10:	Consider only "Behavior Type" and calculate the feature values within the Global scope. Pay special attention to features Nw and UNw of the Count type. Nw focuses on behavior quantity, and UNw focuses on the non-repetitive object quantity of behavior. Calculate feature values and add the <Feature Name, Feature Value> key assignments to FVT.
Lines 11-39:	Traverse all considered target object types, and calculate the related Count and Time feature values. The considered target object types include Friend (Fd), Non-Friend (Nf), and Self (Sf).
Lines 12-21:	Consider the matching between "Behavior Type" and "Target Object Type," and calculate the feature values of the TU type. The features of the Count type at this level also include Nw_B, indicating the proportion of "This Behavior Type Matched with This Target Object Type" to all interactions of "This Behavior Type." After calculating all feature values, add the <Feature Name, Feature Value> key assignments to FVT.
Lines 22-38:	Traverse all considered target article types, and calculate the related Count and Time feature values. The considered target article types include Post (Ps), Photo (Pt), Video (Vd), Link (Lk), and Note (Nt).
Lines 23-28:	Consider the matching between "Behavior Type" and "Target Article Type," and calculate the feature values of the TO type. The feature values of the TO type differ slightly from those of the TU type. Each article is unique, any behavior acts upon each article only once (with the exception of Comment), and there is no sense in calculating statistical values such as the average and standard deviation. Therefore, the Count features of the TO type only refer to the proportion.
Lines 30-38:	Consider the matching between "Behavior Type," "Target Article Type," and "Target Object Type," and calculate their related quantity and time feature values. The features Nw_BU and Nw_BO of the quantity type at this level are described as follows: Nw_BU represents the proportion of "This Behavior Type Matched with Target Article Type of This Target Object Type" to all interactions of "This Behavior Type Matched with This Target Object Type;" Nw_BO represents the proportion of "This Behavior Type Matched with Target Article Type of This Target Object Type" to all interactions of "This Behavior Type Matched with This Target Article Type." After calculating all feature values, add the <Feature Name, Feature Value> key assignments to FVT.
Line 42:	Return the Feature Name vs FVT.

Figure 7. Algorithm for computing each feature

(*continued*)

```
Input: interaction log IL, behaviors B { L, P, S, C }, target users TU { Fd, Nf, Sf },
target objects TO { Ps, Pt, Vd, Lk, Nt }
Output: feature-value table FVT

For each behv In B do
  Nw[behv] ← Count(behv) / Count(IL)
  UNw[behv] ← DistUsrCnt(behv) / DistUsrCnt(IL)
  Avg[behv] ← Count(behv) / Count_U(behv)
  StdN[behv] ← Std(CntToUserList(behv))
  MaxT[behv] ← MaxTimeInterval(behv)
  MinT[behv] ← MinTimeInterval(behv)
  AvgT[behv] ← TotalTimeInterval(behv) / (Count(behv)-1)
  StdT[behv] ← Std(TimeIntervalList(behv))
  push all F-V into FVT
  For each tUsr In TU do
    Nw[behv][tUsr] ← Count(behv, tUsr) / Count(IL)
    Nw_B[behv][tUsr] ← Count(behv, tUsr) / Count(behv)
    UNw[behv][tUsr] ← DistUsrCnt(behv, tUsr) / DistUsrCnt(behv)
    Avg[behv][tUsr] ← Count(behv, tUsr) / Count_U(behv, tUsr)
    StdN[behv][tUsr] ← Std(CntToUserList(behv, tUsr))
    MaxT[behv][tUsr] ← MaxTimeInterval(behv, tUsr)
    MinT[behv][tUsr] ← MinTimeInterval(behv, tUsr)
    AvgT[behv][tUsr] ←
      TotalTimeInterval(behv, tUsr) / (Count(behv, tUsr)-1)
    StdT[behv][tUsr] ← Std(TimeIntervalList(behv, tUsr))
    push all F-V into FVT
    For each tObj In TO do
      Nw[behv][tObj] ← Count(behv, tObj) / Count(IL)
      Nw_B[behv][tObj] ← Count(behv, tObj) / Count(behv)
      MaxT[behv][tObj] ← MaxTimeInterval(behv, tObj)
      MinT[behv][tObj] ← MinTimeInterval(behv, tObj)
      AvgT[behv][tObj] ←
        TotalTimeInterval(behv, tObj) / (Count(behv, tObj)-1)
      StdT[behv][tObj] ← Std(TimeIntervalList(behv, tObj))

      Nw[behv][tUsr][tObj] ← Count(behv, tUsr, tObj) / Count(IL)
      Nw_B[behv][tUsr][tObj] ← Count(behv, tUsr, tObj) / Count(behv)
      Nw_BU[behv][tUsr][tObj] ←
        Count(behv, tUsr, tObj) / Count(behv, tUsr)
      Nw_BO[behv][tUsr][tObj] ←
        Count_U(behv, tUsr, tObj) / Count(behv, tObj)
      MaxT[behv][tUsr][tObj] ← MaxTimeInterval(behv, tUsr, tObj)
      MinT[behv][tUsr][tObj] ← MinTimeInterval(behv, tUsr, tObj)
      AvgT[behv][tUsr][tObj] ←
        TotalTimeInterval(behv, tUsr, tObj) / (Count(behv, tUsr, tObj)-1)
      StdT[behv][tUsr][tObj] ← Std(TimeIntervalList(behv, tUsr, tObj))
      push all F-V into FVT
    End For
  End For
End For
Return FVT
```

Figure 7.

As its name implies, the IFC needs to use the interaction data as the classification criteria. A Facebook user can enter the "view activity log" page to view his/her previous interaction logs, as shown in Figure 9.

However, although Facebook allows users to enter this page to view their interaction logs, no other method of obtaining these logs is available. Even FB API does not provide such a function. To verify the method proposed by this research, alternative methods of conveniently obtaining the interaction logs with the users' consent are needed. To this end, the "ILReaper" program was developed.

ILReaper is connected to the remote laboratory FTP platform, and is notified that the subject has allowed the program to perform the two-stage operation. The first stage extracts the user interaction logs. The subject must log into Facebook through the program and enter the "view activity log" and "friends" pages. The user's HTML files

$$
\begin{array}{c c}
 & \begin{array}{cccc} \mathit{Feature}_1 & \mathit{Feature}_2 & \cdots & \mathit{Feature}_n \end{array} \\
\begin{array}{c} \mathit{Subject}_1 \\ \mathit{Subject}_2 \\ \vdots \\ \mathit{Subject}_m \end{array} &
\begin{bmatrix} \mathit{Value}_{1,1} & \mathit{Value}_{1,2} & \cdots & \mathit{Value}_{1,n} \\ \mathit{Value}_{2,1} & \mathit{Value}_{2,2} & \cdots & \mathit{Value}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathit{Value}_{m,1} & \mathit{Value}_{m,2} & \cdots & \mathit{Value}_{m,n} \end{bmatrix}
\end{array}
$$

Figure 8.
Subject-feature value matrix

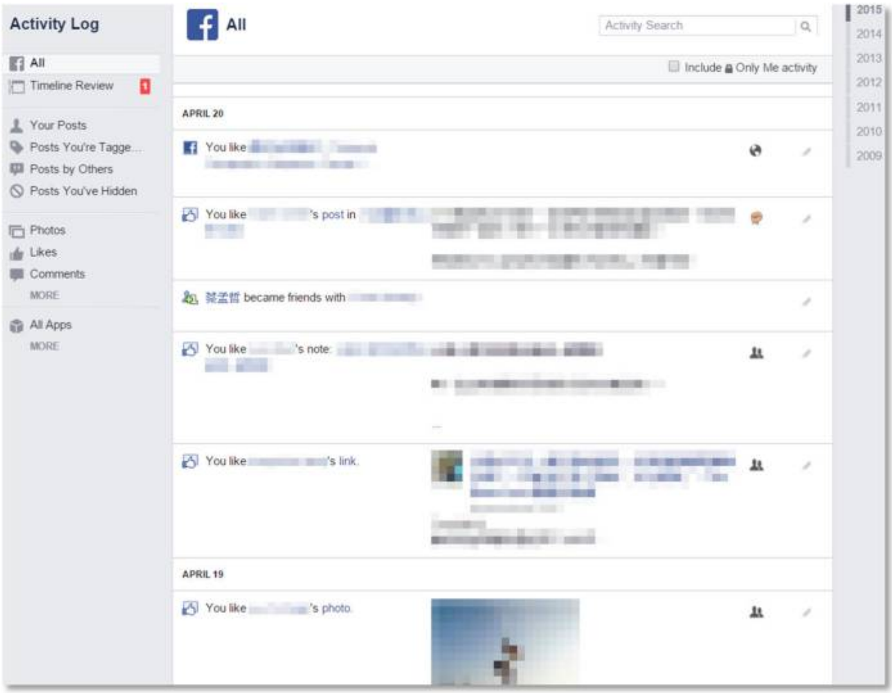

Figure 9.
Facebook activity log page

on these two pages are automatically extracted by the program, and transferred to the FTP platform. The program then moves to the second stage to conduct the DISC questionnaire survey based on the content of *The Platinum Rule*.

Figures 10 and 11 show screenshots of ILReaper. Figure 10 shows the first stage of the program. The subject clicks the "complete the task (Mark 1)" button to upload HTML files. In this figure, only the section with Mark 2 is required for analysis, and the sections with Marks 3, 4, and 5 contain unwanted data. However, this unwanted data are collected with the HTML files. Therefore, the interaction logs and friends list uploaded by the subject are actually messy HTML content, and need to be preprocessed to extract useful data. In this research, a separate program is compiled to solve this problem. Figure 11 shows screenshots of the personality test after the subject has completed data collection. Upon completing the personality test, the program automatically uploads the test results

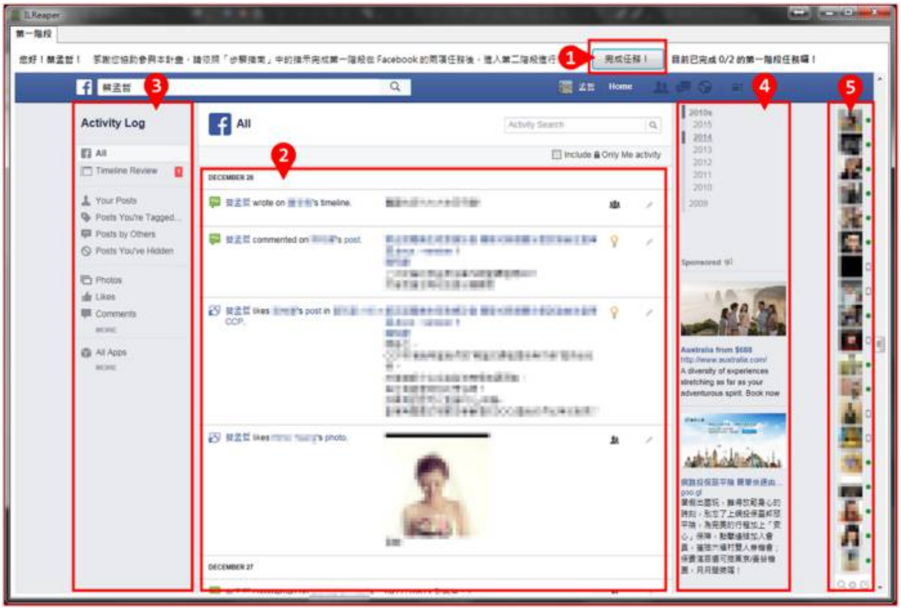

Figure 10.
Screenshots of ILReaper (data collection)

Figure 11.
Screenshots of ILReaper (personality test)

to the FTP platform. The laboratory can then collect the raw data of the subject's personality test results, interaction logs, and friends list.

As described above, the file content provided by the user includes a great deal of unrelated data (including sidebars and advertisements) from the two pages. Therefore, the file content needs to be preprocessed correctly so that useful data required for IFC can be extracted. This research developed a separate program called "Facebook Activity Miner" to obtain the desired webpage content using RE. RE works on the following principle: the data pattern to be extracted (or filtered) from the webpages is first observed, and then the desired data are selected by designing pattern word strings. Figure 12 shows screenshots from the Facebook Activity Miner. In this figure, the section with Mark 1 contains the screening options used for debugging during program development. The sections with Marks 2 and 3 are the extracted interaction logs and friends list, respectively.

In addition, Facebook Activity Miner also provides a feature value calculation function, that is, it calculates the feature values of the 612 DFs of each subject in a structured way. Figure 13 presents the calculation results for a single subject.

Figure 7 shows the algorithm for calculating the specific feature values. Table V lists the feature values calculated for the collected interaction logs. The full table would be very large, and so only a small part is shown here.
The above data are saved as a CSV file for use as the Weka input data (Figure 14), and the combinations of different classification and feature selection algorithms are used to conduct an experiment. Table VI lists the experimental results. In the table, the numerical elements represent the classification accuracy for the matching between the classification algorithms on the left and the feature subsets filtered by the upper feature selection.

The results of the classification accuracy test (Table VI) are calculated through tenfold cross-validation. Table VI indicates that the Naïve Bayes classification algorithm has good universal performance. Combined with the feature subsets filtered by the feature selection algorithm "CfsSubsetEval," Naïve Bayes can reach a classification accuracy of 80 percent.

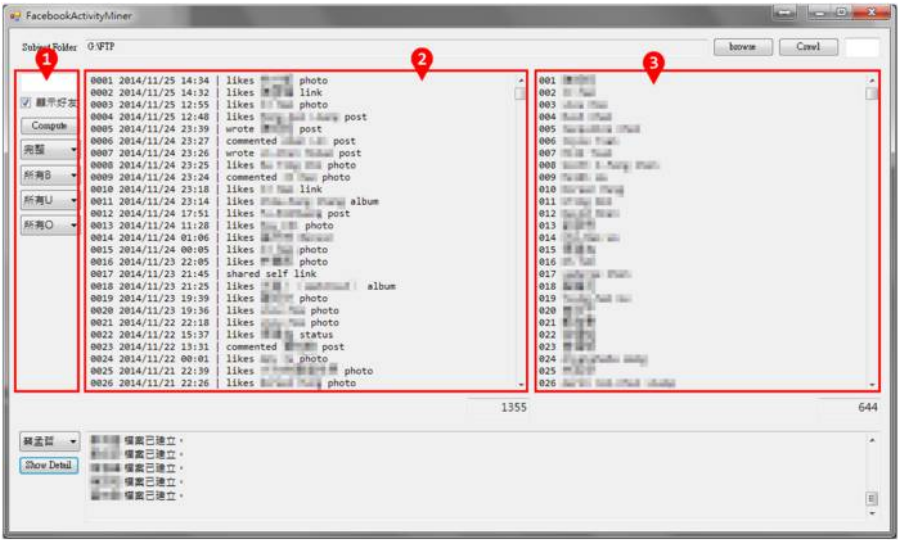

Figure 12.
Extracted interaction logs and friends list

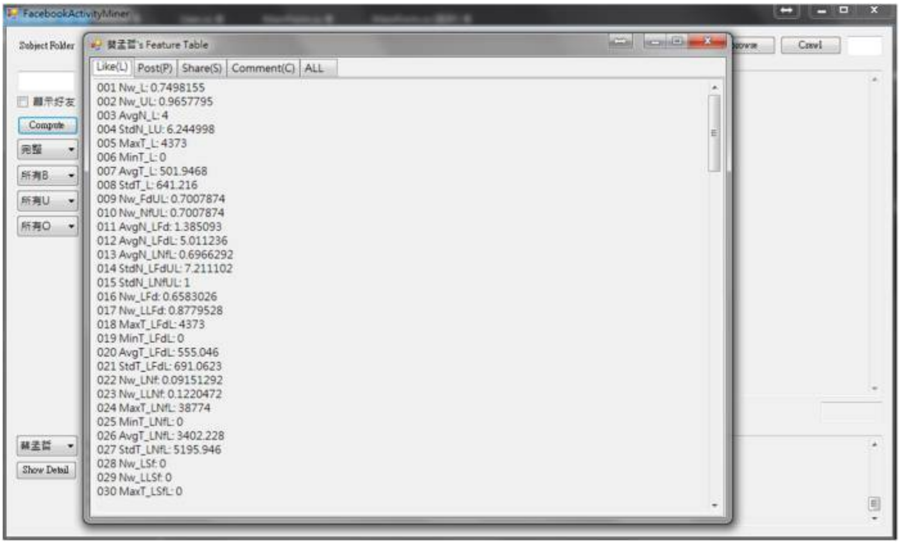

Figure 13.
Calculation results for a subject

Feature abbr.	Subject no. 1	2	3	4	5	…
Nw_L	0.586892	0.807545	0.867427	0.685841	0.814255	…
Nw_UL	0.959732	0.956621	0.984906	0.981013	0.947631	…
AvgN_L	4.853147	3.01432	4.662835	2	2.976316	…
StdN_LU	6.78233	4	8.246211	1.732051	7.28011	…
MaxT_L	2,682	1,746	2,906	4,58,564	5,24,170	…
MinT_L	0	0	0	0	0	…
AvgT_L	323.3972	87.98177	260.4383	3,252.136	965.7584	…
StdT_L	464.9882	237.5121	409.2163	26,306.62	15,597.47	…
Nw_FdUL	0.594406	0.599045	0.639847	0.535484	0	…
Nw_NfUL	0.594406	0.599045	0.639847	0.535484	0	…
AvgN_LFd	2.638037	1.044929	2.756164	0.837607	0	…
AvgN_LFdL	5.058824	3.243028	6.023952	2.361446	0	…
AvgN_LNfL	3.105882	1.784861	1.263473	1.373494	0	…
StdN_LFdUL	6.403124	4.358899	9.273619	1.414214	0	…
StdN_LNfUL	6.164414	2.828427	4.123106	1.414214	0	…
Nw_LFd	0.363636	0.52046	0.717035	0.433628	0	…
Nw_LLFd	0.619597	0.644497	0.826623	0.632258	0	…
MaxT_LFdL	4,438	1,730	3,352	408,838	0	…
MinT_LFdL	0	0	0	0	0	…
AvgT_LFdL	464.0349	129.5252	304.602	4,580.487	0	…
StdT_LFdL	605.6996	279.6855	434.2776	29,933.03	0	…
Nw_LNf	0.223256	0.286445	0.150392	0.252212	0.814255	…
Nw_LLNf	0.380404	0.354711	0.173377	0.367742	1	…
MaxT_LNfL	9,971	2,040	9,955	458,564	524,170	…
⋮	⋮	⋮	⋮	⋮	⋮	⋱
Type	C	C	S	C	S	…

Table V.
Feature values calculated for the collected interaction logs

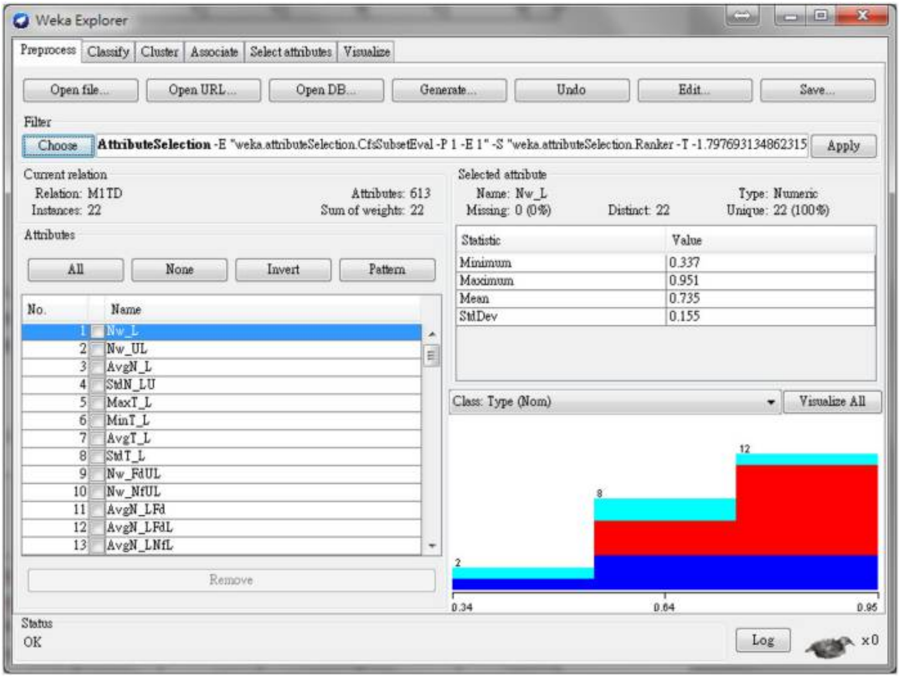

Figure 14.
CSV file of Table V as the input data for Weka

Table VI.
Results of the classification accuracy test

Classification algorithms	Null (%)	Feature selection algorithms CfsSubset Eval (%)	GainRatio Attribute Eval (%)	InfoGain Attribute Eval (%)	OneR Attribute Eval (%)
Naïve bayes	*54.55*	*81.82*	*68.18*	*68.18*	54.55
Bayes net	45.45	77.27	54.55	54.55	45.45
1-IBk (knn)	45.45	77.27	*68.18*	*68.18*	*59.09*
3-IBk (knn)	40.91	68.18	*68.18*	*68.18*	*59.09*
5-IBk (knn)	59.09	63.64	*68.18*	*68.18*	54.55
J48 (DT)	36.36	63.64	54.55	54.55	50
LibSVM	50	50	50	50	50

5. Discussion and conclusion

This section summarizes the findings and contributions of this research, and discusses the limitations and enhancements that should be considered in future research.

5.1 Results and contribution of this research

This research has developed an automated personality type prediction method that classifies DISC personality types based on Facebook user profiles. This research resorted to the commonly used feature classification method: designing the features, screening the features, training the classifier model using the remaining features, and designing the classification method IFC accordingly. Experimental results show that the Naïve Bayes classification algorithm combined with the "CfsSubsetEval" feature selection algorithm produces the best performance, enabling personality types to be predicted with 70-80 percent accuracy.

The contributions of this research are as follows:

(1) Construction of a model for user interactions on social media: the SIM designed for this research cannot only be applied to Facebook, but also to other social media platforms. The current social media differ only in the type and quantity of elements at different levels. Therefore, SIM can also serve as a reference model for related research on other social media interactions. Researchers need only define the elements of the target social media at different SIM levels.

(2) Proposal of a structured design methodology for the features of social media interactions: feature design is always the first problem in traditional feature classification methods. In the past, designing reasonable features involved relying on observations and experience. This research proposed a feature design method for social media interactions that can generate a large number of interaction features in a structured way, and then obtains KFSs via feature selection. The advantage of this method lies in the opportunity to identify features that cannot be designed experimentally. For example, this research identified the classification KF of "StdN_LU," which was intended to indicate the standard deviation of "like" for a non-repetitive user. If features were designed based only on experience or intuition, it is possible that this feature would not have been considered.

(3) Proposal of personality type prediction methods for Facebook users: previous related studies have been limited to examining the correlation between the "operating behavior" or "publish articles" of social media users and their personality test scores, and predicted the range of personality test scores of social media users at most. In contrast, this research considers accurate type predictions for social media users. Experimental results have shown that the IFC method proposed by this research is indeed feasible for Facebook users, although the accuracy is not sufficiently high to satisfy the needs of actual applications because of the limited sample data in the initial stage.

5.2 Limitations and future directions of this research

This research has certain specific limitations. The features used by the IFC method are specifically designed for the interactions of Facebook users. If the IFC method is applied to other social media, the features will need to be redesigned. For details on feature design, please refer to the structured feature design methods proposed by this research. As Facebook was used as the data source platform for this research, the author had to obtain related data via FB API. However, the data types that can be accessed via FB API are very limited. However, the required interaction logs are currently difficult to obtain. This not only requires subjects who are willing to provide their interaction log data, but also requires the subjects to perform certain manual operations.

The section discusses other deficiencies in this research, thus offering suggestions for future study. This research is oriented toward Facebook. Theoretically, the research methods could be applied to other social media with minimal modifications. Considering the differences in the life habits of users of different social media, the feasibility of personality type predictions via other social media requires further research and verification. Theoretically, the personality models used in this research could be applied to other social media, provided that they have clearly defined personality types, such as Enneagram and MBTI. In this research, the DISC classification method, which is used in various commercial activities, was used with related techniques. If other personality models were used in subsequent studies to predict personality types, it will be necessary to modify the test questionnaire used in the training period. The questions should be able to accurately predict the classification results of the subject under the relevant personality model. As a result, a prediction system trained using the methods proposed in this research will be able to predict the different personality types included in the personality model.

To verify the feasibility of personality prediction via interactions, the methods proposed in this research focused on the features of "things done" by Facebook users. This study did not probe deeply into the content of their interactions, such as the body text of responses to others' essays, the actual content of re-posted links to webpages, or image data of posted photos. Examining such content would require technical expertise in various fields (e.g. image analysis). In this regard, this research cannot provide individual deep analysis, and so subsequent studies may wish to take a more in-depth viewpoint.

If future studies continue to focus on Facebook, they will need to continuously track revised information. As Facebook is updated, SIM and the interaction features may also require updates.

References

Adali, S. and Golbeck, J. (2012), "Predicting personality with social behavior", *International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 302-309.

Adali, S., Sisenda, F. and Magdon-Ismail, M. (2012), "Actions speak as loud as words: predicting relationships from social behavior data", *WWW'12 Proceedings of the 21st International Conference on World Wide Web*, pp. 689-698.

Alessandra, T. and O'Connor, M.J. (1998), *The Platinum Rule: Discover the Four Basic Business Personalities and How They Can Lead You to Success*, Grand Central Publishing, New York, NY.

Back, M.D., Stopfer, J.M., Vazire, S., Gaddis, S., Schmukle, S.C., Egloff, B. and Gosling, S.D. (2010), "Facebook profiles reflect actual personality, not self-idealization", *Psychological Science*, Vol. 21 No. 3, pp. 372-374.

Bai, S., Zhu, T. and Cheng, L. (2012), "Big-five personality prediction based on user behaviors at social network sites", eprint arXiv:1204.4809, 7pp.

Baron, R. and Wagele, R. (1995), *Are You My Type, Am I Yours? Relationships Made Easy through the Enneagram*, Harper, San Francisco, CA.

Blignaut, P. and Naude, A. (2008), "The influence of temperament style on a student's choice of and performance in a computer programming course", *Computers in Human Behavior*, Vol. 24 No. 3, pp. 1010-1020.

Boyd, C.F. (1994), *Different Children Different Needs: The Art of Adjustable Parenting*, Questar Publishers Inc., Oregon, OR.

Funder, D.C. (2000), "Personality", *Annual Review of Psychology*, Vol. 52, pp. 197-221.

Golbeck, J., Robles, C. and Turner, K. (2011), "Predicting personality with social media", *Proceedings of the 29th ACM Conference on Human Factors in Computing Systems (CHI), Vancouver, May 7-12*.

Jim Wu, Y.-C., Chang, W.-H. and Yuan, C.-H. (2015), "Do Facebook profile pictures reflect user's personality?", *Computers in Human Behavior*, Vol. 51, Part B, pp. 880-889.

Knapp, M.L. and Daly, J.A. (2002), *Handbook of Interpersonal Communication*, Sage Publications, Berlin.

Lee, E., Ahn, J. and Kim, Y.J. (2014), "Personality traits and self-presentation at Facebook", *Personality and Individual Differences*, Vol. 69, pp. 162-167.

Lima, A.C.E.S. and de Castro, L.N. (2014), "A multi-label, semi-supervised classification approach applied to personality prediction in social media", *Neural Networks*, Vol. 58, pp. 122-130.

McCrae, R.R. and Costa, P.T. (1997), "Personality trait structure as a human universal", *American Psychologist*, Vol. 52 No. 5, pp. 509-16.

Marston, W.M. (1928), *Emotions of Normal People*, Kegan Paul Trench Trubner and Company Limited, New York, NY.

MBAlib (2015), "DISC", available at: http://wiki.mbalib.com/zh-tw/DISC (accessed April 16, 2015).

Moore, K. and McElroy, J.C. (2012), "The influence of personality on Facebook usage, wall postings, and regret", *Computers in Human Behavior Archive*, Vol. 28 No. 1, pp. 267-274.

Nadkarni, A. and Hofmann, G.H. (2012), "Why do people use Facebook?", *Personality and Individual Differences*, Vol. 52 No. 3, pp. 243-249.

Ortigosa, A., Carro, R.M. and Quiroga, J.I. (2014), "Predicting user personality by mining social interactions in Facebook", *Journal of Computer and System Sciences*, Vol. 80 No. 1, pp. 57-71.

Rosenberg, M. and Silvert, D. (2012), *Taking Flight!: Master the DISC Styles to Transform Your Career, Your Relationships … Your Life*, FT Press, NJ.

Ross, C., Orr, E.S., Sisic, M., Arseneault, J.M., Simmering, M.G. and Orr, R.R. (2009), "Personality and motivations associated with Facebook use", *Computers in Human Behavior*, Vol. 25 No. 2, pp. 578-586.

Seidman, G. (2013), "Self-presentation and belonging on Facebook: how personality influences social media use and motivations", *Personality and Individual Differences*, Vol. 54 No. 3, pp. 402-407.

University of Waikato (2015), "Weka", available at: www.cs.waikato.ac.nz/ml/weka/

Wikipedia (2015a), "Social media", available at: http://en.wikipedia.org/wiki/Social_media#Distinction_from_other_media (accessed March 12, 2015).

Wikipedia (2015b), "Facebook", available at: http://zh.wikipedia.org/wiki/Facebook (accessed March 26, 2015).

Youyou, W., Kosinski, M. and Stillwell, D. (2015), "Computer-based personality judgments are more accurate than those made by humans", *Proceedings of the National Academy of Sciences*, Vol. 112 No. 4, pp. 1036-1040.

Zhao, S., Grasmuck, S. and Martin, J. (2008), "Identity construction on Facebook: digital empowerment in anchored relationships", *Computers in Human Behavior*, Vol. 24 No. 5, pp. 1816-1836.

Further reading

Facebook (2015), available at: www.facebook.com/ (accessed February 10, 2015).

Jung, C.G. (2014), *Psychological Types*, Routledge (first published 1921), London and New York, NY.

Corresponding author

Tsung-Yi Chen can be contacted at: tsungyi@mail.nhu.edu.tw