

The Possibilities of Classification of Emotional States Based on User Behavioral Characteristics

M. Magdin*, D. Držík, J. Reichel, Š. Koprda

Constantine the Philosopher University in Nitra, Faculty of Natural Sciences, Department of Informatics, Tr. A. Hlinku 1, 949 74 Nitra (Slovakia)

Received 10 September 2020 | Accepted 12 November 2020 | Published 24 November 2020



ABSTRACT

The classification of user's emotions based on their behavioral characteristic, namely their keyboard typing and mouse usage pattern is an effective and non-invasive way of gathering user's data without imposing any limitations on their ability to perform tasks. To gather data for the classifier we used an application, the Emotnizer, which we had developed for this purpose. The output of the classification is categorized into 4 emotional categories from Russel's complex circular model - happiness, anger, sadness and the state of relaxation. The sample of the reference database consisted of 50 students. Multiple regression analyses gave us a model, that allowed us to predict the valence and arousal of the subject based on the input from the keyboard and mouse. Upon re-testing with another test group of 50 students and processing the data we found out our Emotnizer program can classify emotional states with an average success rate of 82.31%.

KEYWORDS

Emotion, Behavioral Characteristics, Valence, Arousal, Classification.

DOI: 10.9781/ijimai.2020.11.010

I. INTRODUCTION

THE classification of user's emotional state belongs to the most debated topics in the areas of pedagogy, psychology and computer science. In computer science, the problem of classification is mostly encountered with cognitive systems or human-computer interaction (HCI). In the recent years, multiple methods have arisen to describe and recognise the emotional states of human subjects [1]. These methods are not only based on visual cues such as the facial expressions or gestures captured by a webcam [2], auditory cues – pitch, and tone of voice [3], physiological measurements (skin resistance, heartbeat rate, body temperature), but also on behavioural characteristics. Standard methods usually represent an invasive way of gathering data, during these the user is aware of the ways the data is gathered, which can distort the results. As an example of this influence we can mention face expression recognition, the extraction of a region of interest and subsequent classification of emotional state based on instructed expressions. Some authors have already pointed out this phenomenon [4], [5].

Methods of classifying the user's emotional state based on behavioural characteristics are relatively new and very topical area of scientific research even nowadays. Students of applied informatics focus their expertise mainly on the field of programming. The speciality of teaching programming (methodology) is that they must not only be able to program a specific application from the ground up but they must also be able to read the code, look for errors and correct them. Based on pedagogical research (interviews with students and questionnaires) we found out that reading and correcting the wrong

code is often a stressful factor for some students that evokes different emotions (amount of anger, joy, frustration, etc.). In order to help them with the educational process and set a suitable learning style for them, we programmed and implemented the Emotnizer application into the educational process.

We have divided the paper into several sections. The Related Work section presents the theoretical bases of emotional state classification, basic models of emotions and their authors, as well as an analysis of the current state. We examined the researchers' focus on emotions based on behavioural characteristics, namely when using the keyboard and mouse of a computer. The Material and Methods section describes our methods used in Emotnizer, an application which collects raw data by capturing real-time keyboard and mouse events using the JNativeHook library. The application groups this data, and stores it for further use using a relational database system contained in a small SQLite library. In the Experiment section, we propose creating a reference database for classifying the emotional state based on user's behavioural characteristics. In the Results section, we present the results of the experiments and a description of the calculation of the overall success of the classification of the emotional state. The rationale for the results and their confrontation with the results published by other authors is given in the Discussion and Conclusion section.

II. RELATED WORK

Emotions are complex psychological processes caused by chemical changes in the brain associated with subjective experiences (the way we experience emotions), physiological reactions (the way our bodies react to emotions) and behavioural or expressive reactions (the way we respond to emotions) [6].

* Corresponding author.

E-mail address: mmagdin@ukf.sk

The classification of emotions is how we can distinguish one emotion from another. There are two unique models for expressing emotions:

- categorical model - in which it is usually mandatory to select one emotion (or emotional category) from the set of emotions, which is best indicated by the mediated feeling. In these models, the emotions are precisely marked - categorised. The most significant categorical model is the Ekman model of classification [7], [8], which is currently referred to by many authors [2], [9], [10].
- dimensional model - definitions of human emotions are usually based on two or three emotional dimensions. Among the majority of these models there is a degree of valence involved as well (pleasurable - unpleasant) and a degree of arousal (arousing - subduing). The most influential model currently used is Russel's complex circuit model [11]. Dimensional emotional models suggest that all affective states are caused by the neurophysiological system [12].

An interesting issue that stands out in emotional classification is the usage of currently available technology [13]. Most commonly the technology used to collect emotional data is facial expression recognition using widely available web-cameras. With this approach, the recognition is realised in three basic phases:

1. face detection,
2. extraction of the region of interest,
3. classification of emotional state.

This procedure was established by Kanade in 1973 and it is still in use to this day [14]-[17].

The facial expression recognition technology is based on the knowledge that the classification of emotional state based on the visible facial expressions of emotion predicts the emotional reactions of people induced by presented imagery. Psychological studies show that experienced emotions can be influenced by various visual stimuli [18]. These influenced emotions are set on a gradient of emotional valence (low level - negative emotion, high level - positive emotion). By visual stimuli we understand the level of contrast, colour, texture, the position of an object on the scene, etc. [19], [20].

Other technologies for emotional recognition are physiological data acquisition using sensors, such as electrocardiograph (ECG), blood volume pulse (BVP), electroencephalograph (EEG) and other electrical signals, e.g. GSR - galvanic skin response, for detection and recognition of emotional states. It is known that the sources of the most reliable physiological signals for emotional recognition and its processing are EEG and GSR [18], [21], [22].

The main hurdle in current attempts of emotional state classification using subject's speech pattern is not only the lack of reference databases (such as RAVDESS, SAVEE, EMO-DB) but mainly the variability of the pitch and intensity, which can differ greatly across various human cultures [23], [24]. The standard emotion classification from speech typically employs only these two approaches: discrete categories or emotional dimensions. Morgan determines that also gender was found to impact the ratings of emotional speech and must be considered alongside stimulus factors in the design of future studies of emotion [25].

Nonverbal communication is a manifestation of human behaviour and involves various ways of transferring information, such as human activity, hand movements, facial expressions, etc. It is often used as an essential means of expressing attitudes, feelings and emotions. It plays an important role in multimodal interactions between the human and computer systems [26].

The issue of the classification of emotional state based on behavioural characteristics is related to the current state of professional

literature which would be focused on this area. All research is based mainly on Epp, Lippold and Mandryk [27], who were trying to recognise emotional states based on the keyboard typing dynamics, by logging the duration of keypresses and typing latency. Based on the user's computer activity level, the program encouraged the user to self-evaluate throughout the day. The emotional state questionnaire contained several 5-point questions based on Likert scale regarding the current emotional state of the user for each of the 15 recognised emotional states. The best results of this experiment include two-stage classifiers for self-confidence, hesitancy, nervousness, relaxation, sadness and fatigue with an approximate accuracy of 77.4%. The results of the experiment also show a degree of recognition of anger and arousal with an accuracy of 84% [27].

A similar experiment was carried out in 2015 by Lee using the knowledge gained from Epp, Lippold and Mandryk [27]. The experiment was carried out on a group of 52 people, aged 20 to 26 years (44 men and 8 women). During the experiment subjects wore headphones and were ordered to write "748596132" immediately after hearing each recording from the International Affective Digitized Sounds 2nd edition (IADS-2). The experiment was conducted based on a simple dimensional view of emotions, which assumes that emotion can be defined by matching values on two different strategically chosen dimensions - valence and arousal. The Self-Assessment Manikin - SAM method was used to assess these two dimensions of emotional space. The results of this fixed target text experiment support the hypothesis that keystroke duration and latency are affected by arousal. A shorter key press time ($106.70 \text{ ms} \pm 23.80$) is shown when arousal is high compared to the keypress duration when arousal is low ($108.76 \text{ ms} \pm 24.52$). This result indicates an increase in the duration of key presses when people are tired, sad or bored [28].

In his work, Pentel stated that there are several studies on mouse movement and emotions that suggest the existence of a link between the two. He carried out 2 experiments to confirm his hypothesis. In his first experiment, he created a simple online computer game that collected data from a user's mouse. The screen was filled with random buttons labelled with numbers from 1 to 24. The user's task was to click all the buttons in the correct order as quickly as possible. As it turned out, some buttons were harder to find and caused emotions of confusion in users, thereby allowing the identification of mouse patterns associated with this mental state. It was found that users who reported a confusion rate in the Self-Assessment Scale of 5 to 7 (their confusion rate was high) scored lower in locating buttons speed than those who reported values of 1 to 3 on the Likert Scale. The machine learning algorithm model C4.5 predicted confusion using this data with an accuracy of 84.47%. The result of this experiment was that about half of the users were unable to specify exactly what they were feeling [29].

For this reason, in his second experiment, Pentel used the same data collection procedure as in the previous experiment. Besides, he has collected new data from 400 more game playthroughs. In total, the experimental group consisted of 282 individual users between the ages of 12 and 52. There were up to 21984 collected records representing mouse movement between two button clicks. In this experiment, three types of data were stored (coordinates, button sizes, and colours). Users have reported a few different kinds of emotions: confusion, frustration, shame and satisfaction. The strongest emotions were confusion and frustration, and these were associated with tasks during which the user was unable to find another button for a long time. Pentel has tested four popular machine learning algorithms: Logistic Regression, Support Vector Machine, Random Forest and C4.5. The most successful algorithm was Random Forest with a 93% success rate [29].

III. MATERIAL AND METHODS

The main purpose of this paper is to describe the development of the software providing automatic recognition of the user's emotional state based on the keyboard and mouse input. The Emotnizer application can recognize 4 basic types of emotions according to the dimensions of arousal and valence. The application was programmed in Java. For data collection from keyboard and mouse, it takes advantage of:

JNativeHook - Java library, which allows the program to globally read all the keyboard and mouse events.

SQLite - small, fast, standalone, highly reliable and full-featured relational database system contained in a relatively small library written in C.

JFreeChart - graphing library for Java. Supports the creation of pie charts (2D and 3D), bar charts (horizontal and vertical, regular and stacked), line charts, point charts, etc.

If we had aimed to collect and save all the available data from every mouse event, in an unsynchronized way, we would have gathered an incredible amount of data that would have no informative value. It is raw data, which needs to be grouped by action type (scrolling, mouse-click, etc.) and unit of time (data collection period).

A. Mouse Events

Mouse click frequency – using the `nativeMousePressed` method (from the JNativeHook library), which calls our thread method at the moment of pressing a mouse button and increases the number of the variable by one. As this thread sleeps for 5,000 milliseconds (5 seconds), during this time it only counts the number of mouse clicks. When the thread wakes up it calculates the difference of the number of mouse clicks and the amount of time the thread slept (5 seconds).

Mouse button hold duration – time difference between a mouse button is pressed and ends at its release. A couple of methods are being used - `nativeMousePressed` and `nativeMouseReleased`. At the moment of the mouse button press the `nativeMousePressed` method calls our thread method and passes the button code as a method parameter (left, right or scroll-wheel button) and the current time in milliseconds. At the release of the mouse button, the `nativeMouseReleased` is invoked, which calls the thread method to calculate the mouse button press duration. It also has 2 parameters: button code and the current time in milliseconds. Subsequently, the calculation is performed - the difference of 2 values (new current time value minus the old current time value). However, since the thread sleeps for 5 seconds the result will be the sum of all button hold times divided by the number of times the thread wakes up. The result therefore will be an arithmetic mean calculated according to (1). This result is stored in the database.

$$Score = \frac{\sum_{i=1}^n (new_i - old_i)}{n} \quad (1)$$

To capture the scroll-wheel activity we use the `nativeMouseWheelMoved` method. The principle is the same as in the previous methods - we create a thread that has a predefined sleep time of 5000 milliseconds (5s). If the method detects the scroll wheel movement it calls the thread method and passes a method parameter of positive or negative rotation (scroll up or down) to the thread. When the thread wakes up it will have several positive and negative scrolls. The following pair of values is considered as output:

- the total amount of scrolling – the sum of these numbers in absolute values divided by sleep time (in our case it is 5 seconds), e.g. positive rotation would be 15 and negative rotation -5 . The result would look like this: $(abs(15) + abs(-5)) / 5 = 4$,
- the sum of counts – we add up the number of positive rotations

with the number of negative rotations, e.g. $15 + (-5) = 10$. The further this number is from zero, the more decisive it was for the user to choose to go up or down the screen. But the closer it is to zero, the user feels more indecisive (moved chaotically).

The speed of the cursor can be determined by the number of pixels per seconds that the cursor passes from point A to point B. We take advantage of methods:

- `nativeMouseClicked` – method, which outputs the current coordinates of the cursor when it moves,
- `nativeMouseDragged` – method, which outputs the current cursor coordinates, similar to `nativeMouseClicked`, while the cursor is dragged, not just moved. These situations occur when the user presses and holds the mouse button while moving the cursor.

The first movement of the mouse cursor sets the starting coordinates of the first point using the next thread method. The thread sleeps every 100 milliseconds (0.1 seconds), wakes up, sets the coordinates for the new point, calculates the distance according to (2) between them using the Pythagorean theorem and overwrites the coordinates of the first point with the new coordinates.

$$|AB| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \quad (2)$$

Distance calculations are added in the meantime to the local variable, and after the thread repeats this 50 times, the sum of all distances is divided by the data collection time (5 seconds). The calculation of the distance between of two points on the screen, A $[x_A; y_A]$, B $[x_B; y_B]$ which is represented by a 2D space, a plane, is determined as the size of the hypotenuse of the right triangle ACB (Fig. 1).

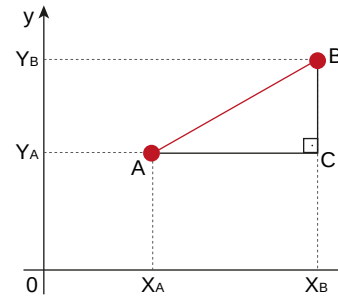


Fig. 1. The distance between of two points in a 2D plane.

B. Keyboard Events

Similarly to the raw mouse events data, raw keyboard events data have to be grouped by the type of action and unit of time.

To determine the frequency of keystrokes or the number of keystrokes per second we use the `nativeKeyPressed` method which calls a specific thread method and increases the value of a variable by one, similar to recording the mouse click frequency. The thread sleeps for 5 seconds, wakes up and divides the number of keystrokes by sleep time (5 seconds). As there is a much greater number of keys on a keyboard than there are buttons on a computer mouse, we've decided to divide the keys into three basic categories:

- modifier keys – Shift (character capitalisation), Ctrl and Alt (keyboard shortcuts), AltGr, Win (summoning Start menu and keyboard shortcuts in Windows), Fn (volume, brightness control, etc.),
- erasing keys – Delete (deletes character after the cursor) and Backspace (deletes the character before the cursor),
- all other keys – alphanumerical keys and any other key apart from modifiers and deletion keys.

We use two methods from the JNativeHook library to determine

key holding period (a time period which starts when a key is pressed and ends when it is released):

- `nativeKeyPressed` - at the moment a key is pressed it calls the thread method with the key code and current time in milliseconds as its method parameters.
- `nativeKeyReleased` - is executed when a pressed key is released and calls a method which passes the key code and current time in a millisecond as parameters to the thread method.

Since we can have multiple keys pressed at the same time, the key code parameter is very important for the calculation. When the key is released the codes are matched in the list, the timestamp of the keypress is found, subtracted from the key release timestamp and the result is converted from milliseconds to seconds. Partial results are obtained during thread sleep. When the thread is awakened the total result is obtained when the sum of all key holding period is divided by the number of all key holds. And since all keys are divided into three basic categories into modifying, erasing and all other keys, the hold time results are divided into these categories.

Pause between keystrokes is a period that starts when the last key is released and lasts until the next key is pressed. Fig. 2 shows the key holding period as well as the pause between keystrokes. To determine the pause period we used the thread we created, which calculates the frequency of keystrokes and the key holding period. After the key was released we checked whether the list of held keys is empty, and if so, we saved the value of time in milliseconds in a local variable. When a keystroke occurred we checked if it happened during the pause or whether the last key release timestamp was written in front of it. If this was true the difference between the two values in milliseconds was sent to the thread that counted the sum of all the pairs and the number of all the pairs. After the thread awoke the average value was calculated from the sum of the pauses and the number of them.

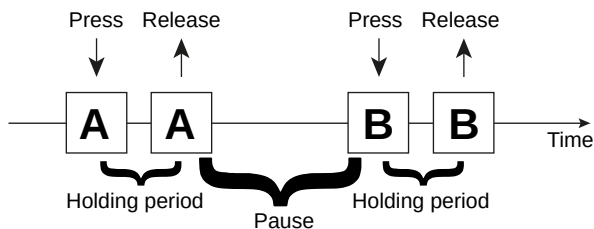


Fig. 2. A pause between keystrokes.

Summary description of all recorded events:

`KPS_avg_ABC`: the average of keystrokes per millisecond for the group of keys containing all the alphanumerical keys and all the other keys apart from modifiers and erasing keys,

`KPS_avg_Ctrl`: the average of keystrokes per millisecond for the group of modifier keys (Ctrl, Shift, Alt, etc.),

`KPS_avg_DEL`: the average of keystrokes per millisecond for the group of erasing keys,

`DTK_avg_ABC`: the average of key holding periods in milliseconds for the group of keys containing all the alphanumerical keys and all the other keys apart from modifiers and erasing keys,

`DTK_avg_Ctrl`: the average of key holding periods in milliseconds for the group of modifier keys (Ctrl, Shift, Alt, etc.),

`DTK_avg_DEL`: the average of key holding periods in milliseconds for the group of erasing keys,

`pause_avg`: the average of key holding periods in milliseconds for the group of modifier keys,

`pause_max`: maximal pause of all the keystroke pauses in

milliseconds,

`CPS_avg`: the average of mouse clicks in milliseconds,

`CPS_max`: the maximal value of all the mouse clicks in milliseconds,

`DTM_avg`: the average of all the mouse holding periods in milliseconds,

`DTM_max`: maximum of all the mouse holding periods in milliseconds,

`DM_avg`: average count of all the pixels the mouse cursor passes on screen in 1000 milliseconds,

`DM_max`: maximal count of all the pixels the mouse cursor passes on screen in 1000 milliseconds

`DiffChar`: the difference in character count of two text strings,

`SAM1`: rate of valence (from -1 to 1), obtained by users in self-assessment,

`SAM2`: rate of arousal (from -1 to 1), obtained by users in self-assessment,

IV. EXPERIMENT – PREPARATION OF A REFERENCE DATABASE FOR THE CLASSIFICATION OF EMOTIONAL STATE BASED ON USER BEHAVIOURAL CHARACTERISTICS

The experiment aimed to create a reference database of user behavioural characteristics for experimental classification of their emotional state.

Procedure for experimenting:

1. creation of a group to propose a reference database,
2. creation of quality measurement procedures (creation of user profile),
3. collection and processing users data,
4. comprehending data,
5. data analysis and interpretation of results,
6. description of used methods, descriptive statistics, multiple regression.

We asked 52 people (47 males and 5 females) of an average age of 21.46 years to undergo profiling by our Emotnizer application. The experiment was supervised by a psychologist. Not all participants met all the requirements, for example, they did not transcribe the entire text, and therefore we have excluded these erroneous data from the overall testing. The total number of people who met all requirements fell to 50 (45 men and 5 women) of an average age of 21.52 years. Most of the students study at the university in the field of applied informatics, therefore they can work with computers very well. Of the total number of people, 38% said they could typewrite. Also, 20% reported using a computer for less than 5 hours a week, 44% using a computer for 6 to 20 hours a week, and 36% using a computer for more than 20 hours a week. Up to 96% of people said they prefer to use the mouse wheel rather than the Home, End, etc. keys. as you scroll vertically across the screen. In this step we created the source data for the reference database (user profile). Based on this database it is possible to determine the user's behavioural characteristics, i.e. how they work with the text (read it, rewrite it) and compare it with the data obtained in the experiment.

To obtain the data, we used a trio of tests, which were prepared for us by a psychologist:

1. Questionnaire tasks to select options,
2. Rewriting the text,
3. Classification of the user's emotional state using self-assessment.

The choice questionnaires required the users to first enter their

username, age, gender, dominant hand (right or left), hours of computer use per week, etc. While users were solving this task, their work with the computer mouse was recorded.

The users' second task was to transcribe a predefined text (Fig. 3), which was supposed to evoke an emotion of anger or fear in the user. The text consisted of over 800 characters, including spaces. The users' task was to read the text and rewrite it correctly in the dialogue box. All events were measured and recorded when transcribing the text.

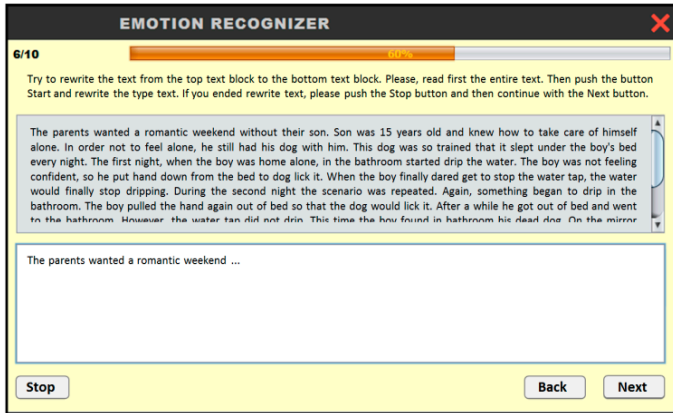


Fig. 3. Text transcription during user profile creation in Emotnizer.

Finally, we calculated the number of errors the users made when rewriting the text. The number of errors was measured using the matrix principle based on the Levenshtein distance between two strings (compared to the original text with the text rewritten by the user). The result was the number of characters that differed from the predefined text.

The users' third task was to evaluate their emotional state with the Self-Assessment Manikin, where they should determine the valence rate (SAM1) and arousal rate (SAM2) on a scale of 1 to 5 they felt during the transcription.

V. RESULTS OF EXPERIMENT

After the data collection phase we had the results from the test group. The following table (Table I) shows the descriptive characteristics recorded in the database. We have selected one of the participants for the demonstration. In the database, descriptive characteristics are stored in a single row for each user. For a clear view we chose to split the characteristics below one another.

TABLE I. DESCRIPTIVE CHARACTERISTICS OF A SINGLE USER

id	name	age	gender	hand	time
10	Vladimir	22	1	1	1
abbreviations	mouse wheel	fingering	SAM1	SAM2	DiffChar
1	0	1	0	0	8
KPS_avg_ABC	KPS_avg_Ctrl	KPS_avg_DEL	KPS_avg_ABC	KPS_avg_Ctrl	KPS_avg_DEL
5662.07	372.73	355.56	91.01	166.05	84.72
Pause_avg	Pause_max	CPS_avg	CPS_max	DTM_avg	DTM_max
128.66	547.86	312.5	600	163.41	554.5
DM_avg	DM_max				
89.76	301				

From these descriptive characteristics we have created average descriptive statistics for 50 users and continued with multiple

regressions. We could not use linear regression because of the independence of the parameters and their number. In multiple regression we focused on processing SAM1 (representing valence) and SAM2 (representing arousal) depending on all recorded events. Multiple regression calculation was performed using MS Excel (Data Analysis tool). We reduced the valence to two states: positive, negative (arousal was reduced similarly) from the original 5-point scale due to the neutral impact of the obtained data on the statistical evaluation.

According to Russel's model, the emotional state of happiness occurs when the user reaches a high value of valence and arousal. State of relaxation is defined by low arousal and high valence. The emotional state of anger is characterized by high arousal and low valence, while sadness is characterized by low arousal and high valence. In the reduction we evaluated happiness and state of relaxation as positive valence, state of anger and sadness as negative valence. Similarly, we have reduced the state of arousal.

When comparing the input parameters we got descriptive statistics for individual coefficients. The Table II shows the results of multiple regression for SAM1 and SAM2 (Table II).

TABLE II. RESULTS OF MULTIPLE REGRESSION FOR SAM1 AND SAM2

	Coefficients for SAM1	Coefficients for SAM2
Intercept	-3.418	1.967
DiffChar	-0.008	-0.023
KPS_avg_ABC	0	0
KPS_avg_Ctrl	0.001	-0.001
KPS_avg_DEL	0.001	0
DTK_avg_ABC	0.01	-0.005
DTK_avg_Ctrl	-0.001	-0.001
DTK_avg_DEL	-0.001	0
pause_avg	0.005	-0.002
pause_max	0	0
CPS_avg	-0.001	0
CPS_max	0	0
DTM_avg	-0.002	-0.001
DTM_max	0	0
DM_avg	0	0.002
DM_max	0	0

These data were multiplied according to (3), where β are coefficients calculated by multiple regression and X are parameters from the keyboard and mouse. The results are predicated of SAM1 and SAM2.

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_n X_{ni} \quad (3)$$

To determine the overall success of the classification we applied the SAM1 and SAM2 predicates to the Emotnizer application we had designed. We verified the overall success of the classification under the guidance of a psychologist by providing another 50 users (average age of 21.15) a PDF file with instructions to use the application. In this file, we presented detailed instructions for creating a profile, and we also focused on recognizing emotions by this application. Users have been notified in advance to read this PDF instruction file first. After reading and understanding, they were tasked to click the Start button and start performing tasks. After pressing the Start button the program randomly chose 5 words from a set of 20 words. The users' task was to open a web browser and look up the first word generated. Users were tasked to open the first search result and copy at least 4 continuous sentences from there into any text editor. They were tasked to transcribe this copied text below itself. As a result, there should be two identical texts below each other in the text editor (the

reference database was created in the same way - users transcribed the predefined text into the dialogue box). In this case the choice of text was up to the user, the user could search for any text unlike the profile creation task. The same was true for the second word chosen by the application. For the last three words users had to think of a short (one sentence) definition that describes the word and the user had to write it in a text editor. After completing all tasks, users were tasked to click the Stop button.

Upon task completion, a program dialogue box appeared in which the users were shown a visual (using a graph) representation of their emotional recognition results by the program. SAM1 and SAM2 values on a scale of -1 to 1 were recalculated to the percentage of all emotion categories according to formula (4). As an example, we chose a formula to calculate the share for the emotion of happiness.

$$y = \left| \frac{(-1 - SAM1)}{2} \right| * \left| \frac{(-1 - SAM2)}{2} \right| \quad (4)$$

Finally, users were asked to objectively evaluate the success rate of detected emotions for all categories of emotions (happiness, anger, sadness and relaxation status) on a scale of 1 to 9 (9 is best) and submit the results (Fig. 4).

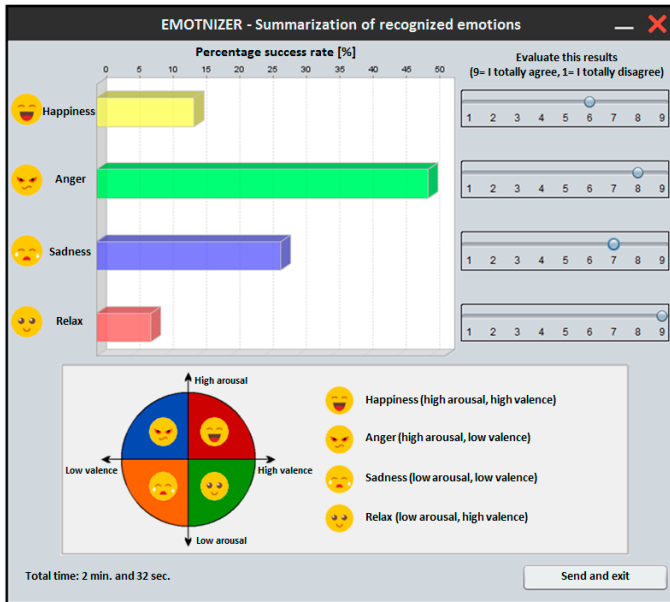


Fig. 4. Summary of recognised emotions.

Table III shows the descriptive characteristics we saw after testing. We have selected one of the participants for the demonstration.

TABLE III. DESCRIPTIVE CHARACTERISTICS OF A SINGLE USER

Predicting of happiness (%)	Predicting of anger (%)	Predicting of mourning (%)	Prediction of relaxation (%)
51.97	30.82	6.41	10.81
Evaluation of happiness	Evaluation of anger	Evaluation of mourning	Evaluation of relaxation
9	7	8	9

From the predicted emotions and their ratings we calculated the success rate percentage of each emotion for each user. We have recalculated partial success rates as percentage equivalents of the respondents' accuracy of prediction, represented by (5).

$$partialsuccess = (evaluation - 1) * 100/8 \quad (5)$$

By summing all the partial success rates of the emotional state rating we gained the overall success rate for that user. By summing up the overall success rate of individual users divided by the number of users we obtained an overall average success rate of Emotnizer classification of 82.31% (Table IV).

VI. DISCUSSION AND CONCLUSION

Recognizing the emotional state of a user utilizing their behavioural characteristics in particular, while working with a mouse and a keyboard, is a relatively new and not sufficiently explored topic. Every person has a different typing speed and not all people are used to work with computers. Therefore, when creating Emotnizer application we had to work with the so-called user profile. Using the user profile we obtained user behavioural characteristics, and based on multiple regression and re-calculated SAM1 (valence) and SAM2 (arousal) predicates. We designed a classification algorithm for emotional states classification. Emotnizer is currently able to recognize 4 types of emotional states (from Russell's complex circular model): happiness, anger, sadness, and a state of relaxation.

A classifier is generally considered successful if it has at least 80% average success rate of the classification. According to the valence and arousal predicates, Emotnizer is currently able to classify emotional states with a success rate of 82.31%. Epp, Lee and Pentel also achieved similar values in their research [27]–[29]. In his research, Epp only recorded key holding periods and latency, not focusing on how respondents work with a computer mouse. In the resulting classification he found that the classifiers used by him could better identify anger and excitement (84%) but conversely had difficulty correctly determining self-esteem, hesitation, nervousness, relaxation, sadness or fatigue (77.4%). Based on the analysis of data from Table IV (average values for each emotional state) we see that classifiers were able to determine anger with a probability of 85.25%, state of relaxation with a probability of 85.00%, sadness of 80.25% and happiness of 78.75%. Thus, we achieved very similar classification values to Epp. Pentel also addressed the state of confusion in his classification. His C4.5 machine learning algorithm model for data processing predicted confusion with an accuracy of 84.47%, in his second experiment, using logistic regression, Support Vector Machine, Random Forest and C4.5 he reached the success rate of 93%. However, according to Rosenberg and Ekman [30], confusion, frustration or shame cannot be regarded as emotion but rather a manifestation (effect) of a particular emotional state.

We have now solved 4 basic emotions. These emotions are found in both Ekman's classification and Russell's circumplex model of affect. The reason why we have developed a classification algorithm for 4 basic emotions is that the larger the number of emotions, the lower the overall success of the classification algorithm. Epp, Lee and Pentel also had this problem. We want to expand our solution in the future. We assume that it is possible to increase the success of the classification and at the same time recognize (and classify) a greater number of emotional states. For example, with the help of a sufficiently long text it would be possible to observe the course of individual emotions in a certain time interval. Emotions would again be classified on the basis of predicates. The role of the psychologist would be to tag (evaluate) individual words and sentences from the text from the point of view of sentiment, and thus we would be able to clearly identify the degree of success in classifying individual emotional states.

Currently, our next goal is to integrate Emotnizer into the learning process. The whole application was developed in order to determine why students of applied informatics are unsuccessful in solving certain programming tasks. We want to continue to improve the application and use it to determine how students behave when programming:

TABLE IV. RESULTS OF THE EMOTIONAL STATE CLASSIFICATION

id	Time [s]	Prediction of Emotnizer [%]				User's evaluation of Emotnizer [%]				
		Happiness	Anger	Sadness	Relax	Happiness	Anger	Sadness	Relax	Average
1	577	51.97	30.82	6.41	10.81	100	75	88	100	91
2	602	40.32	26.18	13.19	20.31	63	100	88	63	78
3	507	31.68	44.31	14.00	10.01	88	75	100	100	91
4	665	0.00	63.11	36.89	0.00	50	75	100	100	81
5	737	17.80	54.12	21.13	6.95	63	100	75	88	81
6	620	32.91	47.01	11.81	8.27	100	63	75	75	78
7	435	35.14	15.81	15.22	33.84	75	88	75	88	81
8	514	16.12	57.98	20.27	5.64	88	88	75	100	88
9	444	37.10	36.69	13.03	13.18	63	88	75	63	72
10	498	23.64	54.11	15.49	6.76	88	100	75	75	84
11	255	46.43	12.98	8.87	31.72	88	100	88	75	88
12	343	21.69	59.67	13.67	4.97	88	63	100	100	88
13	189	27.88	58.66	9.13	4.34	88	75	100	88	88
14	286	40.06	23.49	13.47	22.97	100	63	88	100	88
15	359	14.60	57.95	21.92	5.52	50	88	13	100	63
16	557	44.77	24.47	10.87	19.89	38	100	100	13	63
17	437	42.86	18.21	11.61	27.32	100	100	100	100	100
18	563	17.96	41.34	28.38	12.33	100	100	100	100	100
19	1135	22.61	0.00	0.00	77.39	50	100	50	88	72
20	571	28.63	37.46	19.22	14.69	100	63	50	100	78
21	1226	24.54	0.00	0.00	75.46	63	100	100	38	75
22	591	46.74	0.00	0.00	53.26	100	100	100	100	100
23	345	15.80	46.89	27.90	9.41	88	100	100	100	97
24	375	17.06	3.36	13.10	66.48	50	100	38	100	72
25	138	55.63	17.76	6.44	20.17	0	100	100	50	63
26	352	39.12	41.30	10.06	9.53	88	100	13	50	63
27	445	45.75	22.90	10.46	20.89	100	50	100	75	81
28	626	28.24	38.48	19.19	14.09	88	63	13	100	66
29	299	26.89	43.06	18.49	11.55	63	63	75	88	72
30	475	24.43	24.69	25.58	25.31	100	100	100	100	100
31	514	22.07	36.99	25.64	15.30	75	50	88	100	78
32	338	51.65	26.15	7.46	14.74	88	50	88	75	75
33	368	36.49	41.84	11.58	10.10	50	50	88	88	69
34	386	30.70	24.11	19.88	25.31	100	100	50	75	81
35	461	5.77	14.08	56.84	23.31	100	100	100	100	100
36	382	35.16	34.74	14.96	15.14	100	100	100	100	100
37	61	35.41	0.00	0.00	64.59	88	100	100	88	94
38	95	42.22	0.00	0.00	57.78	63	100	100	100	91
39	86	14.79	0.00	0.00	85.21	88	100	100	100	97
40	97	20.38	0.00	0.00	79.62	63	100	100	88	88
41	83	34.19	0.00	0.00	65.81	75	100	100	100	94
42	152	50.10	0.00	0.00	49.90	63	100	38	75	69
43	91	52.31	0.00	0.00	47.69	50	100	100	75	81
44	68	28.08	0.00	0.00	71.92	75	63	100	75	78
45	73	36.86	0.00	0.00	63.14	100	100	100	100	100
46	513	33.57	43.52	12.93	9.98	75	100	50	50	69
47	433	18.76	59.16	16.77	5.32	100	88	38	75	75
48	579	26.07	59.92	9.76	4.25	100	100	100	100	100
49	233	24.09	46.14	19.55	10.21	88	25	63	75	63
50	369	32.79	35.25	16.56	15.41	88	63	63	100	78

1. They must read the task assignment first (induction of emotion).
2. They propose a solution, they start programming (change of emotional state).
3. The application will detect programming errors (number of corrections in the code) and monitor the emotional state of students (whether, for example, frustration does not affect their programming skills and the end result).
4. Based on the results of behavioral characteristics together with a psychologist, we will propose a suitable solution to a possible problem.

ACKNOWLEDGEMENT

This paper was created with the financial support of the projects:

1. KEGA 036UKF-4/2019, Adaptation of the learning process using sensor networks and the Internet of Things
2. UGA VII/13/2020- Identification of student's behavioural characteristics based on text

REFERENCES

- [1] A. Dzedzickis, A. Kaklauskas, and V. Bucinskas, "Human Emotion Recognition: Review of Sensors and Methods," *Sensors (Switzerland)*, vol. 20, no. 3, p. 592, Jan. 2020, doi: 10.3390/s20030592.
- [2] S. Stöckli, M. Schulte-Mecklenbeck, S. Borer, and A. C. Samson, "Facial expression analysis with AFFDEX and FACET: A validation study," *Behavior Research Methods*, vol. 50, no. 4, pp. 1446–1460, Aug. 2018, doi: 10.3758/s13428-017-0996-1.
- [3] M. Magdin, T. Sulka, J. Tomanová, and M. Vozár, "Voice Analysis Using PRAAT Software and Classification of User Emotional State," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 5, no. 6, p. 33, 2019, doi: 10.9781/ijimai.2019.03.004.
- [4] L. Abramson, I. Marom, R. Petranker, and H. Aviezer, "Is fear in your head? A comparison of instructed and real-life expressions of emotion in the face and body," *Emotion*, vol. 17, no. 3, pp. 557–565, Apr. 2017, doi: 10.1037/emo0000252.
- [5] M. Magdin, L. Benko, and Š. Koprda, "A case study of facial emotion classification using affdex," *Sensors (Switzerland)*, vol. 19, no. 9, May 2019, doi: 10.3390/s19092140.
- [6] D. H. Hockenbury and S. E. Hockenbury, *Discovering psychology*. Worth Publishers, 2007.
- [7] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *Journal of Personality and Social Psychology*, vol. 17, no. 2, pp. 124–129, Feb. 1971, doi: 10.1037/h0030377.
- [8] P. Ekman and W. V. Friesen, "Measuring facial movement," *Environ. Psychol. Nonverbal Behav.*, vol. 1, no. 1, pp. 56–75, Sep. 1976, doi: 10.1007/BF01115465.
- [9] M. Bahreini, R. Nadolski, and W. Westera, "FILTWAM - A framework for online affective computing in serious games," in *Procedia Computer Science*, 2012, vol. 15, pp. 45–52, doi: 10.1016/j.procs.2012.10.057.
- [10] G. M. PS Sreeja, "Emotion models: a review," *International Journal of Control Theory and Application*, vol. 10, no. 8, pp. 651–657, 2017.
- [11] J. A. Russell, "A circumplex model of affect," *Journal of Personality and Social Psychology*, vol. 39, no. 6, pp. 1161–1178, Dec. 1980, doi: 10.1037/h0077714.
- [12] J. Posner, J. A. Russell, and B. S. Peterson, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," *Development and Psychopathology*, vol. 17, no. 3, pp. 715–734, Jul. 2005, doi: 10.1017/S0954579405050340.
- [13] M. Khari, A. K. Garg, R. Gonzalez-Crespo, and E. Verdú, "Gesture Recognition of RGB and RGB-D Static Images Using Convolutional Neural Networks," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 5, no. 7, p. 22, 2019, doi: 10.9781/ijimai.2019.09.002.
- [14] T. Kanade, "Picture processing system by computer complex and recognition of human faces," 1974.
- [15] D. Ghimire and J. Lee, "A robust face detection method based on skin color and edges," *Journal of Information Processing Systems*, vol. 9, no. 1,

- pp. 141–156, 2013, doi: 10.3745/JIPS.2013.9.1.141.
- [16] S. H. Yoo, S. K. Oh, and W. Pedrycz, “Optimized face recognition algorithm using radial basis function neural networks and its practical applications,” *Neural Networks*, vol. 69, pp. 111–125, Sep. 2015, doi: 10.1016/j.neunet.2015.05.001.
- [17] L. Surace, M. Patacchiola, E. B. Sönmez, W. Spataro, and A. Cangelosi, “Emotion recognition in the wild using deep neural networks and Bayesian classifiers,” in *ICMI 2017 - Proceedings of the 19th ACM International Conference on Multimodal Interaction*, 2017, vol. 2017-January, pp. 593–597, doi: 10.1145/3136755.3143015.
- [18] K. Eroğlu, T. Kayıkçıoğlu, and O. Osman, “Effect of brightness of visual stimuli on EEG signals,” *Behavioural Brain Research*, vol. 382, p. 112486, Mar. 2020, doi: 10.1016/j.bbr.2020.112486.
- [19] A. Meikleham and R. Hugo, “Understanding informal feedback to improve online course design,” *European Journal of Engineering Education*, vol. 45, no. 1, pp. 4–21, Jan. 2020, doi: 10.1080/03043797.2018.1563051.
- [20] L. Li, X. Zhu, Y. Hao, S. Wang, X. Gao, and Q. Huang, “A hierarchical CNN-RNN approach for visual emotion classification,” *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 15, no. 3s, pp. 1–17, Dec. 2019, doi: 10.1145/3359753.
- [21] N. Y. Oktavia, A. D. Wibawa, E. S. Pane, and M. H. Purnomo, “Human Emotion Classification Based on EEG Signals Using Naïve Bayes Method,” in *Proceedings - 2019 International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, iSemantic 2019*, 2019, pp. 319–324, doi: 10.1109/ISEMANTIC.2019.8884224.
- [22] D. Ayata, Y. Yaslan, M. K.-I. U.-J. Of, and U. 2017, “Emotion recognition via galvanic skin response: Comparison of machine learning algorithms and feature extraction methods,” *Istanbul University – Journal of Electrical and Electronics Engineering*, vol. 17, no. 1, p. 3129, 2017.
- [23] M. Dedeoglu, J. Zhang, and R. Liang, “Emotion classification based on audiovisual information fusion using deep learning,” in *IEEE International Conference on Data Mining Workshops, ICDMW*, 2019, vol. 2019-November, pp. 131–134, doi: 10.1109/ICDMW.2019.00029.
- [24] A. A. Akinrinmade, E. Adetiba, J. A. Badejo, and A. A. Atayero, “Creation of a Nigerian Voice Corpus for Indigenous Speaker Recognition,” *Journal of Physics: Conference Series*, vol. 1378, no. 3, 2019, doi: 10.1088/1742-6596/1378/3/032011.
- [25] S. D. Morgan, “Categorical and dimensional ratings of emotional speech: Behavioral findings from the Morgan emotional speech set,” *Journal of Speech, Language and Hearing Research*, vol. 62, no. 11, pp. 4015–4029, Nov. 2019, doi: 10.1044/2019_JSLHR-S-19-0144.
- [26] S. Gong, H. Mao, Y. Wang, and A. Xu, “Machine learning in human-computer nonverbal communication,” in *NeuroManagement and Intelligent Computing Method on Multimodal Interaction, AICMI 2019*, 2019, pp. 1–7, doi: 10.1145/3357160.3357670.
- [27] C. Epp, M. Lippold, and R. L. Mandryk, “Identifying emotional states using keystroke dynamics,” in *Conference on Human Factors in Computing Systems - Proceedings*, 2011, pp. 715–724, doi: 10.1145/1978942.1979046.
- [28] P.-M. Lee, W.-H. Tsui, and T.-C. Hsiao, “The Influence of Emotion on Keyboard Typing: An Experimental Study Using Auditory Stimuli,” *PLoS One*, vol. 10, no. 6, p. e0129056, Jun. 2015, doi: 10.1371/journal.pone.0129056.
- [29] A. Pentel, “Employing think-aloud protocol to connect user emotions and mouse movements,” in *IISA 2015 - 6th International Conference on Information, Intelligence, Systems and Applications*, 2016, doi: 10.1109/IISA.2015.7387970.
- [30] E. L. Rosenberg and P. Ekman, “Emotion: Methods of study,,” in *Encyclopedia of Psychology, Vol. 3*, American Psychological Association, 2000, pp. 171–175.



D. Držik

He is student at the Department of Informatics. He deals with programming in Java, Python and C. He specialization is using and implementation various algorithm for face detection, acquisition of user behavioral characteristics and classification emotion.



J. Reichel

He was born in Nova Bana, Slovakia, in 1988. He received the M.S. degree in computer science teaching from Faculty of Natural Sciences, Constantine the Philosopher University in Nitra, Slovak Republic, in 2012. In 2016, he received the Ph.D. degree in theory of teaching mathematics at Constantine the Philosopher University in Nitra. From 2016 to present, he is an Assistant Professor with the Department of Informatics, Constantine the Philosopher University, Nitra, Slovakia. His research interests are in the field of Educational Data Mining.



S. Koprda

He has been an employee of the Department of Informatics since 2005. Since 2015 he has been working as an Associate Professor at the Department of Informatics. He has many years of experience in the field of electrical engineering with a focus on digital and control technology. He is an expert in working with microcontrollers, specifically Arduino. In field of microcontrollers he specializes to programming microcontrollers for face recognition, fingerprint reader using the Atmel Studio programming environment and Arduino IDE.



M. Magdin

He works as a professor assistant at the Department of Informatics. He deals with the theory of teaching informatics subjects, mainly implementation interactivity elements in e-learning courses, face detection and emotion recognition using a webcam, speech recognition. He participates in the projects aimed at the usage of new competencies in teaching and also in the projects dealing with learning in virtual environment using e-learning courses.