EmoKey: An Emotion-aware Keyboard for Mental Health Monitoring

Surjya Ghosh
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
Surjya.Ghosh@cwi.nl

Bivas Mitra
IIT Kharagpur
West Bengal, India
bivas@cse.iitkgp.ac.in

Pradipta De
Georgia Southern University
Satesboro, USA
pde@georgiasouthern.edu

Abstract

The rate of mental health disorders is rising across the globe. While it significantly affects the quality of life, early detection can prevent fatal consequences. Existing literature suggests that mobile-based sensing technology can be used to determine different mental health conditions like stress, bipolar disorder. In today's smartphone-based communication, a significant portion is based on instant messaging apps like WhatsApp; thus providing the opportunity to unobtrusively monitor the keyboard interaction pattern to track mental state. We, in this paper, capture the emotion self-reports from suitable probing moments and trace keyboard interaction patterns (not actual data) to train a personalized machine learning model for tracking multiple emotion states. We design, develop and implement an Android-based smartphone keyboard *EmoKey*, which determines four emotion states (*happy, sad, stressed, relaxed*) based on the keyboard interaction pattern. We evaluate *EmoKey* with 22 participants in a 3-week in-the-wild study, which reveals that it can detect the emotions with an average accuracy (AUCROC) of 78%.

Author Keywords

Emotion self-report; Emotion tracking; Smartphone; Keyboard interaction

CHI '20 Extended Abstracts, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6819-3/20/04.
https://doi.org/10.1145/3334480.3383019

CCS Concepts

•**Human-centered computing** → **Empirical studies in HCI;** *Smartphones;*

Introduction

The problem of mental stress and depression is becoming prevalent across the globe. According to the World Health Organization (WHO) report, approximately 320 million people suffering from depression, which is almost 4.4% of the world's entire population [16]. The situation is more alarming in lower-income countries (regions like South-East Asia, Africa), which contains roughly 40% of this population. While the situation is worse, early diagnosis and counseling can help to overcome the problem of mental and depressive disorders to great extent [15, 4]. However, capturing the manifestations of mental disorder is challenging and often gets unnoticed until the problem is in advanced state.

The ubiquity of sensor-rich smartphones in our daily lives and the ability to continuously monitor smartphone data can facilitate early detection of depressive disorders. Existing literature indicates that from smartphone usage it is possible to detect different mental health conditions like stress, bipolar disorder [15, 3, 2]. Among different activities performed using smartphones, engagement with different instant messaging apps like WhatsApp, FB messenger contains a significant portion and led to the development of input interaction-based emotion detection applications [8, 7, 5]. These applications typically deploy a machine learning based model for emotion detection, which is built by correlating momentary emotion self-reports and input interaction patterns. However, identifying suitable probing points for momentary self-report collection is challenging as it demands user attention [10, 6, 13, 20, 12]. Moreover, these applications often lack the provision of communicating the mental condition to the stakeholders, who can detect and intervene early to limit the progression of mental disorders.

We, in this paper, propose an emotion-aware smartphone keyboard *EmoKey*, which determines multiple emotion states based on keyboard interactions [11]. It deploys an on-device personalized machine learning model, which leverages on different typing signatures and determines multiple emotion states. It identifies different typing blocks (sessions) as users perform text entry, collects emotion self-reports via Experience Sampling Method (ESM) [14] and correlates these with the typing features to build the emotion detection model. It identifies suitable emotion capturing moments like the ones when the user completes text entry in an application and going to start using the next. Additionally, it encompasses a user interface, which keeps track of user emotions over time. We evaluate *EmoKey* in a 3-week in-the-wild study involving 22 participants and observe that *EmoKey* can determine four emotion states (*happy, sad, stressed, relaxed*) with an average accuracy (AUCROC) of 78%. All but *relaxed* emotions are identified with an average accuracy of close to 80%, thus showing the promise of monitoring mental health from text input interactions.

EmoKey Design

The design principles of *EmoKey* are based on the schematic as shown in Fig. 1. We define a text entry session as the time period one stays on the single application without changing the same. In Fig. 1, a user starts text entry on WhatsApp at `t1` and continues to do so till `t2`. The elapsed time between `t1` to `t2` is defined as the session, where each black bar denotes a single key pressing event. Once user completes typing in WhatsApp and changes the application, an ESM probe is issued to record the emotion self-report as perceived in this session. The same is performed when the user performs typing in Hangout session (t3 - t4). Later, from each of the typing sessions, different typing features are extracted, which are correlated with the corresponding emotion self-reports to build the emotion detection model. During model construction, the emotion self-reports are manually collected while after model construction, the emotions are predicted based on the typing interactions performed in a session. In both the cases, the emotions are to be uploaded to the background repository, which can provide a means to track the mental health condition.

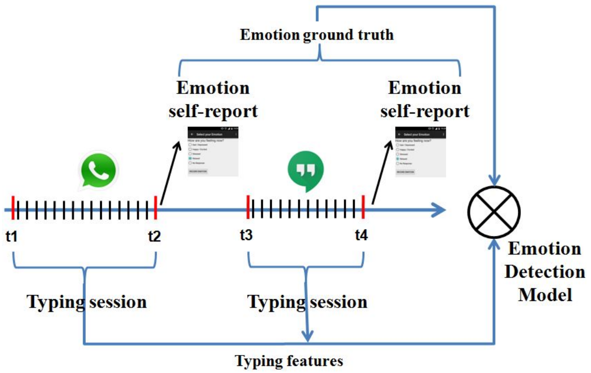


Figure 1: Scenario of typing based emotion detection during text entry. Elapsed time between t1 and t2 is considered a session, when user performs text entry in WhatsApp. Different typing sessions [(t1 - t2), (t3 - t4)] are identified, typing features are extracted from these sessions and correlated with the corresponding emotion self-reports collected via ESM probe to build the emotion detection model.

The scenario described above calls for following design capabilities - (a) tracing user's typing activity (b) collecting emotion self-reports from the users timely (c) building an on-device emotion detection model correlating the typing features and emotion self-reports and (d) providing a mechanism to record and maintain the emotion states of the users over time.

EmoKey Implementation

We show the architecture of *EmoKey* in Fig. 2. It has following major components.

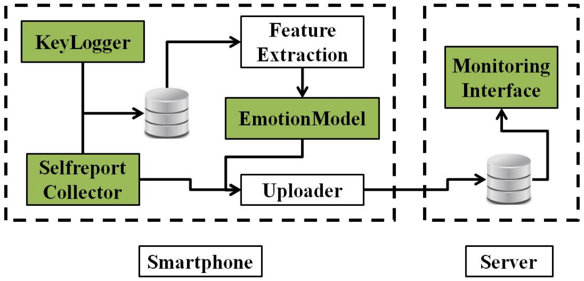


Figure 2: *EmoKey* architecture; key components are highlighted

KeyLogger traces the typing activity. It is implemented as QWERTY keyboard using Android Input Method Editor (IME) facility (Fig. 3). It records the current timestamp, associated application name, any non-alphanumeric character typed during every key press event. To ensure user privacy, we do not store or record any alphanumeric character.

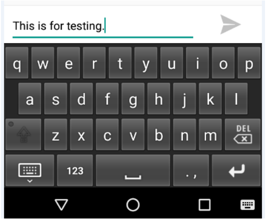
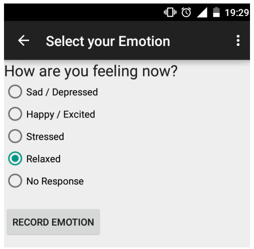

This is for testing.

Figure 3: *EmoKey* keyboard


19:29

Select your Emotion

How are you feeling now?

○ Sad / Depressed
○ Happy / Excited
○ Stressed
● Relaxed
○ No Response

RECORD EMOTION

Figure 4: Self-report UI

Self-reportCollector collects the emotion self-report from the user at the end of every session. It collects the emotion self-reports based on Experience Sampling Method (ESM), the most common approach for collecting self-reports in behavioral studies [14, 1]. It probes the user as soon as the user completes typing in a session and changes the application. But as this would lead to responding to many probes, we restrict the number of probes using LIHF ESM schedule [9]. The key idea of this scheduling is that instead of issuing probes for every session, we accumulate closely occurring session (within 30 minutes) and issue a single probe. The response obtained via this probe is tagged to all these sessions. The self-report collection UI is shown in Fig. 4. We concentrate on four discrete emotions - *happy, sad, stressed, relaxed*. We select one dominant emotion from each of the four quadrants of the Circumplex model [17] so that they are non-overlapping and user can distinguish them well during self-reporting. We also keep the provision of skipping self-reporting by selecting the *No Response* option. By default, when the UI is displayed, this option gets selected. The user needs to select a emotion and record the same to provide the emotion self-report.

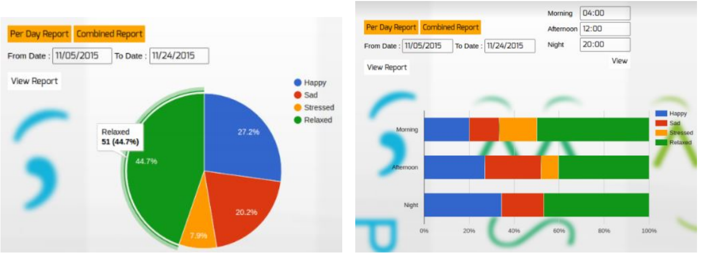
EmotionModel comes into play once the training period is over. The emotion detection model is constructed by correlating the emotion self-reports with the keystroke features as noted in Table 1. We develop personalized emotion detection model as individual typing pattern vary [8, 7]. We use Random Forest classifier to build the models.

From every typing session, we extract the features. We use typing speed as a feature. For every session, we compute the time interval between consecutive tap events, defined as Inter-Tap Distance (ITD). We compute the average of all ITDs present in a session and denote it as *Mean Session ITD (MSI)*. However, it is observed that if two sessions are tagged within short time span, there may be an effect of previous emotion on current one [19], resulting a set of overlapping ITDs. As a result, *MSI* alone is not very effective in distinguishing emotion. To overcome this issue, we use *RMSI*. All ITDs present in a session are clustered into two groups and the average of all ITDs present in the major cluster is denoted as *RMSI*. We also use the amount of backspace and delete keys present in a session along with percentage of special characters (non-alphanumeric character) typed in a session. Additionally, we use session length and session duration also as features. We also use last emotion self-report as a feature to build the model, because emotion states persist over time and current emotion may often be influenced by the previous one [19, 8]. During emotion model construction, we obtain this label from the previous emotion self-report. However, when the model is operational, we use the predicted emotion for last session as the feature value for the current session.

Category	**Feature Name**
Keystroke Features	Mean Session ITD (MSI)
	Refined Mean Session ITD (RMSI)
	Number of special characters
	Number of backspaces (or delete)
	Session duration
	Session text length
Auxiliary Features	Last ESM Response

Table 1: Features used for emotion classification

MonitoringInterface provides the facility to track the user emotion states. During training phase, the user reported self-reports and in deployment phase, the predicted values of user emotion along with timestamp are uploaded to a central repository. It is currently implemented for administrator only, as a result the designated person can login



From Date : 11/05/2015 To Date : 11/24/2015

Morning 04:00
Afternoon 12:00
Night 20:00

(a) Summary report

(b) Detail report

Figure 5: Emotion monitoring interface

to this web-based interface and track the emotion of any participant on a specific date or for the given date range. It also displays the variability in emotion across different time-periods in a day. We show the interface in Fig. 5.

Field Study

We installed the *EmoKey* app in the smartphone of 30 participants (24 male, 6 female, aged between 24 to 33 years) in our university campus. They were asked to use the app for 3 weeks to perform typing and report emotion states. They were instructed that based on their typing activity they will receive emotion reporting survey pop-ups, where they need to report their current emotion state. It was also told that if the pop-up appears at an unfavorable time, when the user is not in a position to report emotion, she can skip the recording by selecting the *No Response* option.

Dataset

During this study period, 3 participants left the study and 5 participants recorded less than 40 labels in total. So, we have discarded data from these participants and finally obtained data from remaining 22 (20 male, 2 female) participants. In total, we have collected close to 135 hours of typing data. We eliminate *No Response* sessions (2.5% of all sessions) as they do not reveal any emotion. Finally, we obtain data from 2705 sessions. It is observed that for most of the users, *relaxed* or *stressed* emotion is the dominant one thus making the distribution of emotion samples skewed. Overall, we have recorded 18%, 9%, 21% and 52% sessions tagged with *happy, sad, stressed* and *relaxed* emotion respectively. We summarize the final dataset in Table 2.

Total typing events	529698
Total typing sessions	2705
Total typing duration	135 Hr.
Per user typing sessions (mean, SD, minimum)	123, 105, 40
Median session duration	98 sec.
Median session length	114

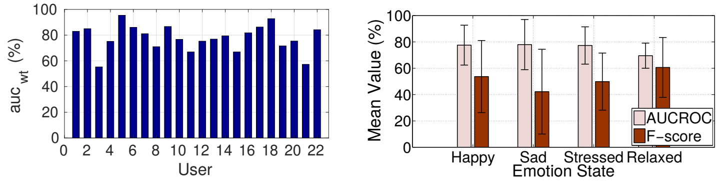
Table 2: Final dataset details

Evaluation

We evaluate different models for emotion detection - (i) L2-regularized Logistic Regression (ii) Support Vector Machine with Radial Basis Function (RBF) Kernel and (iii) Random Forests using 10-fold cross validation. However, best performance is obtained with Random Forests, so we report the results corresponding to this model. We use AUCROC (Area under the Receiver Operating Characteristic curve) and F-score as the performance metric. We compute the weighted average of AUCROC (auc_{wt}) using AUCROC from four different emotions. Let f_i, auc_i indicate the fraction of samples and AUCROC for emotion state i respectively, then auc_{wt} is expressed as, $auc_{wt} = \sum_{\forall i \in \{happy, sad, stressed, relaxed\}} f_i * auc_i$.

Model Performance

We show the classification performance in Fig. 6. We report the user-wise accuracy (auc_{wt}) in Fig. 6a. We obtain an average user-wise accuracy of 78%. It is observed that for more than 45% users the AUCROC is greater than 80% and for all but two users the AUCROC is greater than 70%. The state-wise classification performance (AUCROC, F-score) is reported in Fig. 6b. We observe that all states except *relaxed* state has the AUCROC close to 78%, whereas *relaxed* state has the highest F-score (close to 61%).



(a) User-wise AUCROC

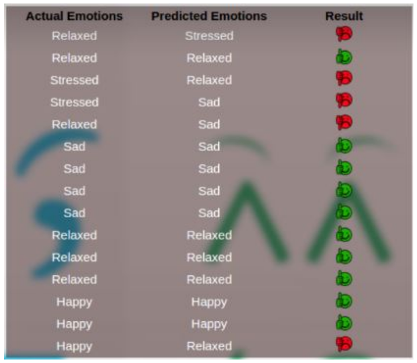
(b) State-wise AUCROC, F-score

Figure 6: Emotion classification performance of the proposed model. Error bar indicates standard deviation.

Once the model is constructed, the predicted emotions are computed and stored on the background server. We also evaluate the model performance by splitting the data in 80-20%, where we build the model using initial 80% and validate on the remaining 20%. We show the prediction result of one representative user in Fig. 7. We obtain an average accuracy of 75% across all users. These results indicate the model can determine multiple emotion states based on typing activities on smartphone.

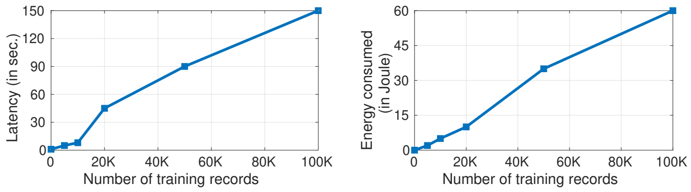
Resource Overhead

Emokey performs both model construction and inference on the smartphone. However, the volume of training data increases with training period. As a result, there may be a performance bottleneck in terms of required model construction time and battery consumption. In order to validate this, we measure latency and power consumption to build the model with different volume of training data. But as our dataset is limited, we synthetically add training records and measure these two parameters. We used OnePlus X (2.3 GHz quad-core Qualcomm Snapdragon 801 3GB RAM) for the experiment.



Actual Emotions	Predicted Emotions	Result
Relaxed	Stressed	👎
Relaxed	Relaxed	👍
Stressed	Relaxed	👎
Stressed	Sad	👎
Relaxed	Sad	👎
Sad	Sad	👍
Sad	Sad	👍
Sad	Sad	👍
Sad	Sad	👍
Relaxed	Relaxed	👍
Relaxed	Relaxed	👍
Relaxed	Relaxed	👍
Happy	Happy	👍
Happy	Happy	👍
Happy	Relaxed	👎

Figure 7: Model validation result of one representative user in the *MonitoringInterface* UI of *EmoKey*.



(a) Training latency

(b) Battery consumption

Figure 8: Measurement of training latency and battery consumption reveals that on-device training is feasible without major performance overhead if the training data is small in size. However, the performance deteriorates significantly with larger training dataset.

We measure (on device) the required time and energy consumed to build the model with varying amount of training records and report the same in Fig. 8a, 8b respectively.

We observe a latency of less than 10 seconds and battery consumption of 5 joules with 10K records. However, both latency and battery consumption increase if the training records are increased significantly (more than 50K records). This also indicates that it may not be feasible to train an emotion detection model on-device with very large training dataset.

Discussion

Our results demonstrate that using only typing characteristics multiple emotions can be inferred. However, we do not obtain very high classification performance. The prediction accuracy can be improved incorporating additional contextual features that come with typing details (e.g. weekday/weekend, application category). Another possible reason for comparatively poor accuracy could be the skewness in the distribution of emotion samples. By adopting specialized machine learning algorithms for skewed data [18], or by balancing the dataset better classification accuracy can be obtained.

Conclusion and Future Work

We design and develop an emotion-aware smartphone keyboard *EmoKey*, which determines four emotions (*happy, sad, stressed, relaxed*) based on text input interactions. It traces users' typing activity and deploys an on-device machine learning model to determine the emotions. Additionally, it provides the facility to monitor user's emotions over time. The evaluation of the *EmoKey* in a 3-week in-the-wild study involving 22 participants reveals that the model can determine the four emotions with an average accuracy of 78%. It also reveals the scope of developing efficient on-device model training algorithms for long-term mental health monitoring.

REFERENCES

[1] Niels Van Berkel, Denzil Ferreira, and Vassilis Kostakos. 2017. The Experience Sampling Method on Mobile Devices. *ACM Computing Surveys (CSUR)* 50, 6 (2017), 93.

[2] Bokai Cao, Lei Zheng, Chenwei Zhang, Philip S Yu, Andrea Piscitello, John Zulueta, Olu Ajilore, Kelly Ryan, and Alex D Leow. 2017. Deepmood: modeling mobile phone typing dynamics for mood detection. In *Proceedings of the ACM SIGKDD*. 747–755.

[3] M. Ciman and K. Wac. 2016. Individuals' stress assessment using human-smartphone interaction analysis. *IEEE Transactions on Affective Computing* PP, 99 (2016), 1–1. DOI: http://dx.doi.org/10.1109/TAFFC.2016.2592504

[4] Matteo Ciman, Katarzyna Wac, and Ombretta Gaggi. 2015. iSenseStress: assessing stress through human-smartphone interaction analysis. In *Proceedings of the 9th International Conference on Pervasive Computing Technologies for Healthcare*. 84–91.

[5] Marc Exposito, Rosalind W Picard, and Javier Hernandez. 2018. Affective keys: towards unobtrusive stress sensing of smartphone users. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 139–145.

[6] Joel E Fischer, Chris Greenhalgh, and Steve Benford. 2011. Investigating episodes of mobile phone activity as indicators of opportune moments to deliver notifications. In *Proceedings of the 13th international conference on human computer interaction with mobile devices and services*. 181–190.

[7] Surjya Ghosh, Niloy Ganguly, Bivas Mitra, and Pradipta De. 2017a. Evaluating effectiveness of smartphone typing as an indicator of user emotion. In *Affective Computing and Intelligent Interaction (ACII), 2017 Seventh International Conference on*. 146–151.

[8] Surjya Ghosh, Niloy Ganguly, Bivas Mitra, and Pradipta De. 2017b. TapSense: Combining Self-report Patterns and Typing Characteristics for Smartphone Based Emotion Detection. In *Proceedings of the ACM MobileHCI*.

[9] Surjya Ghosh, Niloy Ganguly, Bivas Mitra, and Pradipta De. 2017c. Towards Designing an Intelligent Experience Sampling Method for Emotion Detection. In *Proceedings of the IEEE CCNC*.

[10] Surjya Ghosh, Niloy Ganguly, Bivas Mitra, and Pradipta De. 2019a. Designing an experience sampling method for smartphone based emotion detection. *IEEE Transactions on Affective Computing* (2019).

[11] Surjya Ghosh, Sumit Sahu, Niloy Ganguly, Bivas Mitra, and Pradipta De. 2019b. EmoKey: An Emotion-aware Smartphone Keyboard for Mental Health Monitoring. In *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*. IEEE, 496–499.

[12] Joyce Ho and Stephen S Intille. 2005. Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 909–918.

[13] Kostadin Kushlev, Bruno Cardoso, and Martin Pielot. 2017. Too tense for candy crush: affect influences user engagement with proactively suggested content. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 1–6.

[14] Reed Larson and Mihaly Csikszentmihalyi. 1983. The experience sampling method. *New Directions for Methodology of Social & Behavioral Science* (1983).

[15] Hong Lu, Denise Frauendorfer, Mashfiqui Rabbi, Marianne Schmid Mast, Gokul T Chittaranjan, Andrew T Campbell, Daniel Gatica-Perez, and Tanzeem Choudhury. 2012. Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of ACM Conference on Ubiquitous Computing*.

[16] World Health Organization and others. 2017. Depression and other common mental disorders: global health estimates. (2017).

[17] James A Russell. 1980. A circumplex model of affect. *Journal of Personality and Social Psychology* 39, 6 (1980), 1161–1178.

[18] Yuchun Tang, Yan-Qing Zhang, Nitesh V Chawla, and Sven Krasser. 2009. SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39, 1 (2009), 281–288.

[19] Philippe Verduyn and Saskia Lavrijsen. 2015. Which emotions last longest and why: The role of event importance and rumination. *Motivation and Emotion* 39, 1 (2015), 119–127.

[20] Dominik Weber, Alexandra Voit, Philipp Kratzer, and Niels Henze. 2016. In-situ investigation of notifications in multi-device environments. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 1259–1264.