# What Can a Swiped Word Tell Us More? Demographic and Behavioral Correlates from Shape-Writing Text Entry

Désirée C. A. Lemarquis[1*], Bereket A. Yilma[1] and Luis A. Leiva[1*]

[1]University of Luxembourg, Luxembourg.

*Corresponding author(s). E-mail(s): desiree.lemarquis.001@student.uni.lu; luis.leiva@uni.lu;
Contributing authors: bereket.yilma@uni.lu;

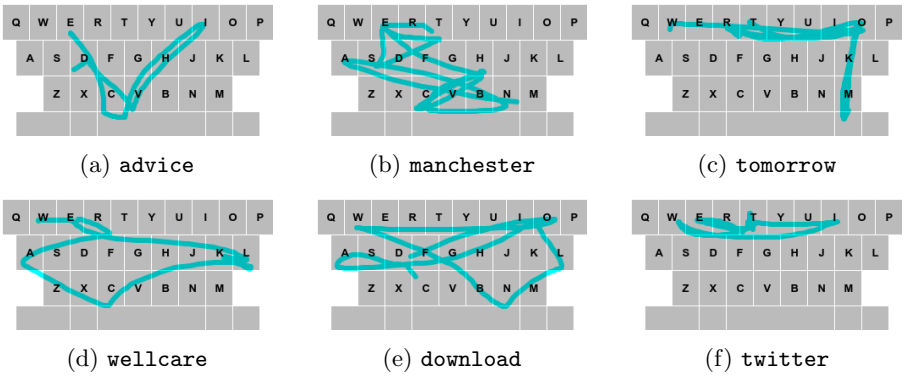
**Abstract**

Shape-writing (aka gesture typing or swiping) is a word-based text entry method for touchscreen keyboards. It works by landing the finger on (or close to) the first character of the desired word and then sliding over all the other character keys without lifting the finger until the last word character is reached. This generates a trajectory of swiped characters on the keyboard layout which can be translated to a meaningful word by a statistical decoder. We hypothesize that swiping carries rich information about the user, such as demographic (e.g. age or gender) and behavioral (e.g. swiping familiarity or input finger) information. To test our hypothesis, we trained several sequence classifiers using different recurrent neural network architectures to predict demographic and behavioral correlates of users from swipe trajectories. We show that our sequence classifiers are always performing better than a random classifier, therefore we conclude that cognitive and motor control mechanisms are embodied and reflected in swipe trajectories, validating thus our research hypothesis. Taken together, our results have implications for user privacy. Currently swiping is supported by all mobile vendors and has millions of users, so people may be inadvertently profiled at an unprecedented granularity. Future work should consider new ways of addressing these issues without impacting the user's swiping experience.

**Keywords:** Shape-writing; Gesture Typing; Biometrics; Neural Networks

# 1 Introduction

Shape-writing, also known as gesture typing, swype input, swipe to text, or just *swiping* (for short), is a prevalent mobile text entry method currently supported by all mobile vendors. Contrary to regular touch typing, where the user touches one key at a time and lifts up the finger to enter one character, swiping is a *word*-based text entry method: The finger lands on (or close to) the first key of the desired word and then, without lifting the finger from the keyboard, it traverses (the vicinity of) all the keys until reaching the last character of the word, generating a trajectory of touch points as a result. See Figure 1 for some examples.



(a) `advice` (b) `manchester` (c) `tomorrow`

(d) `wellcare` (e) `download` (f) `twitter`

**Fig. 1**: Examples of swiped word trajectories.

Swiping performance has been researched in small-scale experiments [1], in large-scale studies as one of many text methods [2], or using proprietary, undisclosed datasets [3]. Collecting such data is challenging, because most mobile keyboards are vendor-locked and do not offer an API for collecting such data [4].

Until now, there was no public swiping dataset where researchers could access raw movements dynamics such as the gesture path drawn on top of the keyboard or the time lapsed between consecutive swiped keys. Recently, Leiva et al. [4] released "How We Swipe", the largest dataset (over 1,300 participants) for conducting research on mobile swiping together with first observations of correlates of typing performance. Large-scale analyses of mobile interaction are relatively rare and mostly undertaken by commercial organizations and mobile vendors that ship soft keyboards. There are notable exceptions [2, 5–7] but they did not investigate swiping behavior.

In this work we analyse the dynamics of swiping trajectories to model and predict different demographic and behavioral correlates, including age, gender, nationality, English level, swiping familiarity, swiping hand, swiping finger, and

dominant hand. In total, we trained 200+ different classifiers with 30 different hyperparameter variations for each predicted target. We found that swipe trajectories actually carry information about demographic and behavioral correlates. To the best of our knowledge, we are the first to perform this analysis in this kind of data.

The rest of this article is organized as follows. In Section 2, we present a comprehensive overview of related works in shape writing and user profiling from demographic and behavioral movement data. In Section 3, we present the dataset we have analyzed and the methodology we have followed. In Section 4 we discuss the results of our experiments. In Section 5, we present a comprehensive summary highlighting limitations and future research directions. Finally, Section 6 closes this article with a summary of the main findings.

## 2 Related Work

Invented by Zhai and Kristensson in 2003 [8], swiping has become a widely adopted text entry method on mobile devices. It well suits touch-based interaction, and provides a high typing speed since it relaxes the requirement of precisely acquiring each (small) key on a soft keyboard. To date, this text entry method is supported by major commercial soft keyboards including Google's Gboard, Microsoft' SwiftKey, and iOS's built-in keyboard on iPhones.

The research community has carried out a large amount of research on swiping techniques. For example, swiping has been extended to support mid-air text entry [9], eyes-free input [10], ring-based input [11], phone-tilting based input [12]. In addition to text entry, swiping has also been extended to support command input [13–15], by gesturing a shortcut related to the command name. Such a method shows advantages in learnability compared with hotkey-based command input [15].

There is a large body of work that demonstrates that our digital footprints, including e.g. the websites we visit or the text we enter in social media, may help derive personally identifiable information like gender, age, location, or even political orientation. Existing literature related to online privacy provides insights around topics such as information leakage while surfing the web using desktop computers [16] or mobile devices [17]. In the following we discuss related work aimed at exploiting this information in the context of human movement data, such as handwriting and gesturing, since they are the closest work to swiping.

### 2.1 Biometrics Prediction from Movement Data

Leiva et al. [18] showed that accurate detection of human movements has more to do with *how* users write, rather than *what* they write. They concluded that it is better to use recurrent models than convolutional models for biometric classification, as recurrent models capture the inherent movement dynamics. However, they only explored one shallow architecture, based on Bidirectional Gated Recurrent Unit (BiGRU) cells, in the context of binary classification

of human vs. fake handwriting. Therefore, it is unclear what design choices over the different hyperparameters (e.g. number of hidden layers, type of recurrent cell, number of hidden units, learning rates, etc.) may affect classification performance. In this work we perform an exhaustive investigation in this regard.

Leiva et al. [19] showed that our mouse movements may reveal who we are. They developed a computational model based on BiGRU cells and were able to classify the age and gender of the users with reasonable precision (near 70%) using a few lines of code. Recent work by White et al. [20] and Gajos et al. [21] could detect neurodegenerative disorders from mouse cursor movements, showing how our "digital phenotypes" could be used as adjunctive screening tools.

Chen et al. [22] were the first to notice a strong relationship between gaze position and cursor position during web browsing. Mueller and Lockerd [23] investigated the use of mouse tracking to visualize and (manually) infer the users' interests. Since then, researchers have noted the utility of mouse cursor analysis as a low-cost and scalable proxy of eye gaze [24–26]. Several works have investigated closely the utility of mouse cursor data in web search [27–29] and web page usability evaluation [30–32], two of the most prominent use cases of this technology. Mouse biometrics is another active research area that has recently shown how to identify an individual by analyzing their mouse movements in controlled settings [33, 34]. This is the research line we focus on in this work.

Objective measurements of attentional processes are increasingly being used to explain or predict user behavior. A mouse click is often preceded by several interactions such as scrolling, hovers, movements, etc. and thus can lead to a better overall understanding of the user's thought process. In what follows, we review research efforts that have focused on mouse cursor analysis to infer user interest, visual attention, emotions, and demographic variables like gender or age, on a desktop setting. We hypothesize that swiped word trajectories carry the same rich information about the user.

## 2.2 Inferring User Interest

For a long time, commercial search engines have been interested in how users interact with Search Engine Result Pages (SERPs), to anticipate better placement and allocation of ads in sponsored search or to optimize the content layout. Early work considered simple, coarse-grained features derived from mouse cursor data to be surrogate measurements of user interest [35, 36]. Follow-up research transitioned to more fine-grained mouse cursor features [37, 38] that were shown to be more effective. These approaches have been directed at predicting open-ended tasks like search success [39] or search satisfaction [29]. In a similar vein, Huang et al. [24, 40] modeled mouse cursor interactions and extended click models to compute more accurate relevance judgements of search results. Mouse cursor position is mostly aligned to eye

gaze, especially on SERPs [41, 42], and that can be used as a good proxy for predicting good and bad abandonment [43].

## 2.3 Inferring Visual Attention

Mouse cursor tracking has been also used to survey the visual focus of users in sponsored search, thus revealing valuable – and at the same time sensitive – information regarding the distribution of user attention over the various SERP components. Despite the technical challenges that arise from this analysis, previous work has shown the utility of mouse movement patterns to measure within-content engagement [44] and predict reading experiences [45, 46]. Lagun et al. [47] introduced the concept of motifs, or frequent cursor subsequences, in the estimation of search result relevance. Similarly, Liu et al. [29] applied the motifs concept to SERPs and predicted search result utility, searcher effort, and satisfaction at a search task level. Boi et al. [48] proposed a method for predicting whether the user is looking at the content pointed by the cursor, exploiting the mouse cursor data and a segmentation of the contents in a web page. Lastly, Arapakis et al. [27, 49] investigated user engagement with direct displays on SERPs and provided further evidence that supports the utility of mouse cursor data for measuring user attention at a display-level granularity.

## 2.4 Inferring Emotional State

Although the connection between mouse cursor movements and the underlying psychological states has been a topic of research since the early 90s [50, 51], some studies have investigated the utility of mouse cursor data for predicting the user's emotional state. For example, Zimmermann et al. [52] investigated the effect of induced affective states on the motor-behavior of online shoppers and found that the total duration of mouse cursor movements and the number of velocity changes were associated to the experienced arousal. Kaklauskas et al. [53] created a system that extracts physiological and motor-control parameters from mouse cursor interactions and then triangulated those with psychological data taken from self-reports, to analyse correlations with users' emotional state and labour productivity. In a similar line, Azcarraga et al. [54] combined electroencephalography signals and mouse cursor interactions to predict self-reported emotions like frustration, interest, confidence and excitement. Yamauchi et al. [55] studied the relationship between mouse cursor trajectories and generalized anxiety in human subjects. Lastly, Kapoor et al. [56] predicted whether a user experiences frustration, using an array of affective-aware sensors.

## 2.5 Inferring Demographics

Yamauchi et al. [57] examined the extent to which mouse cursor movements can help identify the gender and the experienced feelings of users who were watching short film clips. Although this work provides early evidence on the utility of mouse cursor data for advanced online user profiling, it suffers from

certain limitations that we address in this work. First, the experimental setting has limited generalizability, since the adopted perception task is not very well connected to typical activities that users perform online, such as web search. Second, the data used in their predictive modeling task include multiple samples per participant randomly assigned to the training and test data partitions, hence there may be information leakage that artificially inflated model performance. In our analysis, we limit the training samples to exactly one mouse cursor trajectory per participant and test our models on unseen individuals.

Kratky et al. [58] recorded mouse cursor movements in an e-commerce website and engineered a set of meta-features to predict the user gender and age group. Their classifier was trained on several days of data per participant. Although the training and test collections had disjoint sets of participants, it was stated that the reported results were overly optimistic since researchers could not verify their ground-truth data [58]. In contrast, as discussed later, our dataset was collected from high-quality crowdworkers so we are confident that the ground-truth information is correct.

In a similar vein, Pentel et al. [59] used data from six different external sources, including e.g., keystroke data and feedback questionnaires, and handcrafted features proposed in earlier works [35, 36, 43] to train predictive models that could identify the users' age and gender. However, because their approach relies mainly on ad-hoc data, it is less scalable and more difficult to implement than the approach we propose in this paper, which takes as input *raw* mouse cursor data. Moreover, Pentel et al. reported optimistic performance scores, which may be due to information leakage across data partitions, and omit important classification metrics such as precision, recall, and AUC. To account for their modeling approach, as well as that proposed by Kratky et al. [58], we implement the same classifier and test it in our setting.

### 2.6 Summary

There is a vast research literature about modeling user behavior using movement data. How we move our mouse provides a surrogate signal for gaze fixation, and therefore reveals our focus of attention, which can be used to learn *latent* interests. However there is no previous work in this regard using swipe trajectories. This is because of the lack of public datasets, which are difficult to collect in practice. Fortunately, Leiva et al. [4] have recently released a swiping dataset that enables researchers to create sophisticated Machine Learning models. To the best of our knowledge, we are the first to tap into this dataset to create different biometric-related classifiers.

## 3 Experiments

Based on the research literature, we hypothesize that cognitive and motor control mechanisms are embodied and reflected, to some extent, in swipe trajectories. If our classification results prove to be better than random, this work will open a new research avenue for advanced user profiling.

## 3.1 Dataset

The swiping dataset we used in this study is publicly available at https://osf.io/sj67f/. It covers both movement-level and task performance-level data, and was collected in a crowdsourcing study using a JavaScript-based virtual QWERTY soft keyboard. The dataset comprises 100K+ English words swiped by 1,338 users. Each word was swiped in the context of either a memorable sentence, drawn from the EnronMobile phrase set [60], or a random 4-words sentence, drawn from Google's Trillion Word Corpus[1] and the Forbes 2019 Global 2000 list.[2] All words are lowercased with no punctuation symbols.

Swipe logs include the following information: event name (e.g. `touchstart`, `touchmove`), event timestamp, `x` and `y` coordinates, touch radius, and rotation angle, where available. The logs also include the keyboard size, the prompted word, and whether it was swiped correctly or not. We deliberately ignore the wrongly swiped words, as they were clear outliers, as described in the dataset [4]. There are 11,295 unique words correctly swiped which we consider for analysis. Additionally, the dataset provides the following user metadata: swipe familiarity, age, gender, nationality, browser language, device pixel ratio, screen size (height and width), English level, dominant hand, swipe hand, swipe finger, and mobile vendor. In this paper, we predict all the biometric-related variables: swipe familiarity, age, gender, nationality, English level, dominant hand, swipe hand, and swipe finger.

For the task of training our classifiers, we performed a light preprocessing on the dataset. First, we iterated through the 1,338 user logs and gathered the raw trajectories of `mousemove` $x, y$ coordinates and associated timestamps $t$ for each of the individual correctly swiped words. Each raw trajectory has the following format: $(x, y, t)_1, \ldots, (x, y, t)_n$ where $(x, y, t)_i = (x_i, y_i, t_i)$. Then, we computed the offsets of the raw points, as it makes our sequence models scale-invariant and independent of the device's screen size: $(\Delta x, \Delta y, \Delta t)_i = (x_i - x_{i+1}, y_i - y_{i+1}, t_i - t_{i+1})$. Eventually, our final dataset contains a total of 94,841 swiped word trajectories from 1,308 unique participants.

## 3.2 Predictive Targets

Figure 2 shows the class distributions of the targets we set to predict in this article. From the figures we can see that some targets are more imbalanced than others. This can potentially bias to our sequence classifiers, so we created splits of the data as balanced as possible, as explained next.

The **familiarity** target describes how well a user is used to swiping and contains 5 classes in the original dataset: "everyday" (22%), "often" (15%), "sometimes" (25%), "rarely" (27%) and "never" (11%). By looking at the class distributions, we can see that the class "often" and "never" are twice as small as the other classes. Therefore, we merged the "often" class with "everyday"

[1] https://github.com/first20hours/google-10000-english
[2] https://www.forbes.com/global2000/

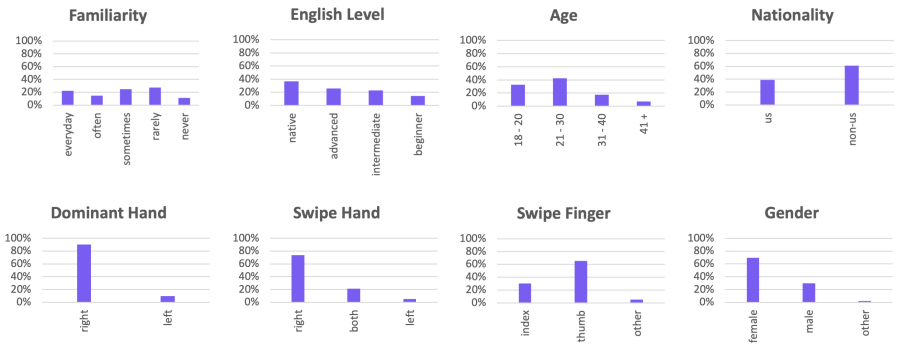



**Fig. 2**: Targets and their class distributions in the original swiping dataset.

and the "never" class with "rarely". Eventually we ended up with 3 classes for our experiments: "everyday" (37%), "sometimes" (25%) and "rarely" (38%).

The **English level** target has 4 classes in the original dataset: "native" (34%), "advanced" (27%), "intermediate" (24%), and "beginner" (15%). In this case the class distributions are fairly balanced except "beginner", however the class semantics are important to preserve in this case. Leiva et al. [4] reported that "intermediate" and "beginner" English speakers were significantly slower and committed more errors than any of the other users, therefore we did not modify the original class distributions for the English level target.

The **age** target has 4 classes in the original dataset: "youth" (18–20 years, 33%), "young" (21–30 years, 42%), "adult" (31–40 years, 18%) and "senior" (41+ years, 7%). Since these class distributions are rather imbalanced, we redefine the age brackets as follows: "youth" (18–20 years, 33%), "young" (21–28 years, 37%), "adult" (29+ years, 30%).

The **nationality** target describes whether a user is a US citizen, given that US users were over-represented in the original dataset. The class distributions are: 39% "US" and 61% "non-US" users. Since there are two classes, we did not modify the original class distributions.

The **dominant hand** target describes the hand the user uses the most. The class distributions are: 90% "right-handed" and 10% "left-handed" users. Since there are two classes, we did not modify the original class distributions.

The **swipe hand** label describes if the user likes to use the right, left, or both hands to swipe. The original dataset has 3 classes: 74% "right hand", 21% "both hands", and 5% "left hand". Due to their significant difference in size compared to the right hand class, we decided to merge the "left hand" and "both hands" classes. Eventually we ended up with 2 classes for our experiments: "right hand" (74%), "other hand" (26%).

The **swipe finger** label describes which finger the user prefers to swipe with. The original dataset has 3 classes: "index finger" (65%), "thumb" (30%), and "other finger" (5%). Given that the latter class is substantially smaller

that the rest, we left it out and consider 2 classes in our experiments: "index finger" and "thumb".

The **gender** target includes "female" (69%), "male" (30%), and "other" (1%). Given that the latter class is substantially smaller that the rest, we left it out and consider 2 classes in our experiments: "female" and "male".

We can see that, even after this careful class redistribution procedure, some of our targets still have notable imbalanced classes. To address this challenge we used cost-sensitive models via class weighting, as described in the next section.

## 3.3 Model Architectures

Since swipe trajectories are sequences of coordinates, we have chosen to train sequence models. Sequence models are Machine Learning models to handle sequential data, where each observation is dependent or contextually related to the previous one, such as text streams, audio, time-series data, etc. Recurrent Neural Networks (RNN) are by far the most popular algorithms used in sequence models. RNNs consist of standard recurrent cells, shown in Figure 3.

The typical feature of the RNN cell is a cyclic (or *loop*) connection, which enables the model to update the current state based on past states and current input data. Formally, the standard recurrent cell is defined as follows:

$$h_j = \Phi(W_h h_{j-1} + W_z z_j + b) \tag{1}$$

$$o_j = h_j \tag{2}$$

where $z_j = (x, y, t)_j$ denotes the $j$th vector of the input signal $\mathbf{z} = (x, y, t)_{j=1,\cdots,\mathbf{z}}$ at timestep $j$ (i.e., the index at which a point coordinate has happened), $h_j$ is the hidden state of the cell, and $o_j$ denotes the cell output, respectively; $W_h$ and $W_z$ are the weight matrices; $b$ is the bias of the neurons; and $\Phi$ is an activation function.

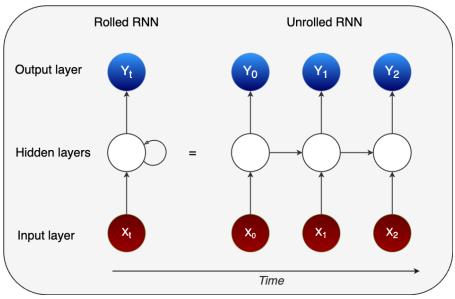



**Fig. 3**: Standard RNN architecture.

Standard recurrent cells have achieved success in many sequence learning problems such as natural language processing [61], action recognition [62], or

image captioning [63]. However, the standard recurrent cells are not capable of handling long-term dependencies. To solve this issue, the LSTM and GRU cells were developed [64, 65]. They improve the capacity of the standard recurrent cell by introducing different gates, which we briefly describe as follows.

On the one hand, the LSTM cell is defined as follows:

$$G_i = \sigma(W_u[h_{j-1}, z_j] + b_i) \tag{3}$$
$$G_f = \sigma(W_f[h_{j-1}, z_j] + b_f) \tag{4}$$
$$G_o = \sigma(W_o[h_{j-1}, z_j] + b_o) \tag{5}$$
$$c_j = G_i \odot \tilde{c}_j + G_f \odot c_{j-1} \tag{6}$$
$$\tilde{c}_j = \Psi(W_c[h_{j-1}, z_j] + b_c) \tag{7}$$
$$h_j = G_o \odot \Psi(c_j) \tag{8}$$

where $c_j$ is a hidden state responsible for the long-term memory, $\tilde{c}_j$ is a candidate state responsible for controlling the cell-state data, $W_*$ are weight matrices, $b_*$ are biases, $G_*$ denote cell gates (i: input, f: forget, o: output), and $\Psi$ and $\sigma$ are activation functions. The $\odot$ operator denotes the Hadamard (element-wise) product.

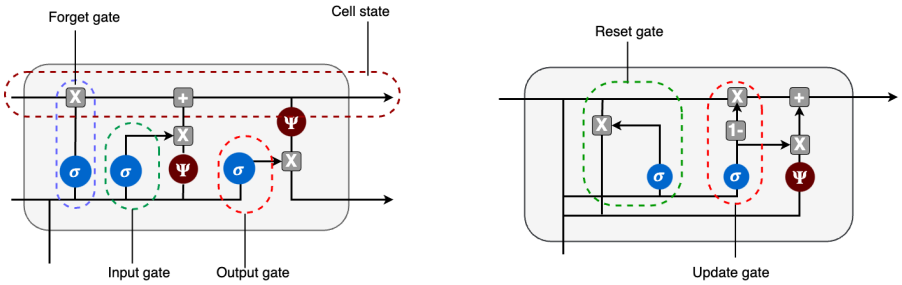
As noted, the LSTM has two kinds of hidden states [66]: a "slow" state $c_j$ that keeps long-term memory, and a "fast" state $h_j$ that makes decisions over short periods of time. The forget gate $G_f$ can decide what information will be thrown away from the cell state. In practice, the activation functions $\Psi$ and $\sigma$ are hyperbolic tangent and sigmoid, respectively, however other non-linear functions have been promoted in the research literature; see the next section for a brief discussion.

On the other hand, the GRU cell is defined as follows:

$$G_u = \sigma(W_u[h_{j-1}, z_j] + b_u) \tag{9}$$
$$G_r = \sigma(W_r[h_{j-1}, z_j] + b_r) \tag{10}$$
$$\tilde{c}_j = \Psi(W_c[G_r \odot h_{j-1}, z_j] + b_c) \tag{11}$$
$$h_j = G_u \odot \tilde{c}_j + (1 - G_u) \odot h_{j-1} \tag{12}$$

where the forget and input gates of the LSTM cell are now combined into a single update gate $G_u$. A reset gate $G_r$ controls which parts of the state get used to compute next cell state. The LSTM and GRU architectures are presented in figure Figure 4. As it can be observed, GRU controls the flow of information like the LSTM cell, but without having to use memory (the forget and output gates in LSTM): the full state vector is output at every time step. Furthermore, performance-wise the GRU cell is on par with the LSTM in many problems [67, 68] but it is computationally more efficient because of its less complex structure. Therefore, our classifier uses GRU cells for its RNN layer.

As can be observed, GRU controls the flow of information like the LSTM cell, but without having to use memory (the forget and output gates in LSTM):




(a) LSTM cell. (b) GRU cell.

**Fig. 4**: Overview of the LSTM and GRU cells.

the full state vector is output at every time step. Furthermore, performance-wise the GRU cell has been shown to be on par with the LSTM in many problems [67, 68] while at the same time being computationally more efficient because of its less complex structure. It remains unclear, however, what would be the best RNN-based architecture configuration (e.g. type of cell, number of layers, number of neurons, etc.) for predicting demographic and behavioral correlates from swipe trajectories. Therefore, we study what is the most suitable configuration.

The hyperbolic tangent activation function is applied in every recurrent layer, since recurrent neural networks do not only get gradients from lower layers but also from previous timesteps. The derivatives of tanh are larger than the other activation functions such as sigmoid or swish. Thus, the cost function is minimized faster by using the hyperbolic tangent activation function. It has a normalized range from -1 to 1 and the output is symmetric around 0 which leads to faster convergence. Formally:

$$\tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \tag{13}$$

Finally, the classification rule of all the classifiers we trained is defined by:

$$\hat{z} = \text{softmax}(W_z \odot h_j + b_z) \tag{14}$$

where softmax($\mathbf{v}$) is the softmax function, which normalizes the output of the last network layer (denoted as $\mathbf{v}$ for the sake of simplicity) to a probability distribution over the predicted output classes $K$:

$$\text{softmax}(\mathbf{v})_i = \frac{e^{v_i}}{\sum_{j=1}^{K} e^{v_j}} \tag{15}$$

for $i = \{1, \ldots, K\}$ and $\mathbf{v} = (v_1, \ldots, v_K)$.

## 3.4 Methodology

Neural networks require inputs with the same shape and size, at least within the same batch. However, the trajectories in the swiping dataset have varying sequence lengths (Mean=72, StDev=64). Thus, we need to apply padding and truncation to the sequence trajectories in order to ensure equally-sized lengths. We chose 200 points as our maximum sequence length, based on the data distribution, thus all sequences were padded or truncated to this maximum capacity.

For the data split ratio, we randomly used 60% for the training set, 20% for the validation set and the remaining 20% for the testing set. Each model was trained and evaluated on the very same data partitions. The testing partition is held out exclusively for model evaluation, as it represents unseen data and is therefore a more challenging but also a more realistic scenario. We chose to work with the Adam optimizer since it is an algorithm for efficient stochastic optimization that only requires first-order gradients with little memory requirement [69]. The loss function to minimize is sparse categorical cross-entropy, since our task is a multi-class classification problem with $C \geq 2$ classes.

### 3.4.1 Model Hyperparameters

We conducted a systematic analysis regarding which architecture hyperparameters affect model performance. The architecture choices include the type of layer (RNN, LSTM or GRU), whether bidirectional or not, number of hidden neurons, number of hidden layers, and optimizer's learning rate.

In bidirectional models, the input flows in both directions such that every component of an input sequence is able to use the information from both the past and present. The model adds for each recurrent layer another recurrent layer with the reversed information flow. The outputs from forward and backward layers are combined by concatenating them.

The learning rate controls how quickly a neural network model adapts to a problem. By choosing smaller learning rates, the model might result in a long training or might get stuck in the process since the changes made to the weights each update are smaller. On the other hand, the model might converge too fast to a sub-optimal solution by choosing larger learning rates. We experimented with learning rate (LR) $10^{-1}$, $10^{-2}$, $10^{-3}$, $10^{-4}$ and $10^{-5}$, 5 variations in total. Note that LR higher than $10^{-1}$ is unlikely to work because of the Robbins-Monro condition: the more LR approaches 1, the less room for the model to learn from backpropagation. Further, when LR $> 1$, stochastic gradient descent is not guaranteed to converge to a minimum [70].

We tested the performance of the different layer types (RNN, BiRNN, LSTM, BiLSTM, GRU and BiGRU), 6 variations in total for each target. The number of hidden layers we tested range from 1 to 4 layers in our model architectures, 4 variations in total. The number of hidden units range from 10 to 150, with a step size of 10, 15 variations in total. Eventually, we tested 30 different hyperparameter variations for each predicted target, totalling a number of 240 different trained classifiers.

### 3.4.2 Model Regularization

We use two popular regularization techniques to prevent overfitting and make our classifiers more generalizable: Dropout and Early Stopping. Dropout removes neurons at random during training, to force different connection paths between neurons and thus avoid the model to take shortcuts (like creating spurious correlations between inputs and outputs). We used Dropout with 0.15 probability in all RNN layers. Early Stopping stops training when no further improvements over a given metric (e.g. classification accuracy, validation loss) are made. We set 40 epochs of patience for Early Stopping while monitoring the F1 score, i.e. if the model does not improve the F1 score over the validation data in 40 consecutive epochs, training stops and the best model weights are retained. The maximum number of training epochs is set to 500 and batch size is set to 32 trajectories.

As discussed in the previous section, our class distributions are imbalanced for most of the targets. To address this challenge we used cost-sensitive models via class weighting. The idea is to give all classes equal importance on gradient updates, regardless of how many samples we have from each class in the training data. This prevents models from predicting the more frequent class more often just because it is more common. Instead, class weighting forces models to learn a more apt representation of each class.

### 3.4.3 Evaluation Metrics

To evaluate model performance, we should note that classification accuracy is not the best metrics to report, since our data is not perfectly balanced. Therefore, we chose the F1 score as our main evaluation metric, since it very popular for imbalanced classification problems. The F1 score (Equation 18) is the harmonic mean of Precision and Recall. Precision (Equation 16) is the ratio of correctly predicted positive instances (True Positives, TP) divided by the total number of predicted positive instances both True (TP) and False Positives (FP). Recall (Equation 17) identifies the number of correct positive predictions among all positive predictions that were possible, including TP and False Negatives (FN). We also report the Area Under the ROC curve (AUC), which determines the discriminatory power of any classifier [71].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{17}$$

$$\text{F1 score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{18}$$

## 4 Results and Discussion

In the following we present the results obtained in our experiments. To test our research hypothesis, we also consider the accuracy of a random classifier


(a) Layer type

(b) Hidden Units

(c) Hidden Layers

(d) Learning Rate

**Fig. 5**: Predicting the user's swiping familiarity from swiped words trajectories.

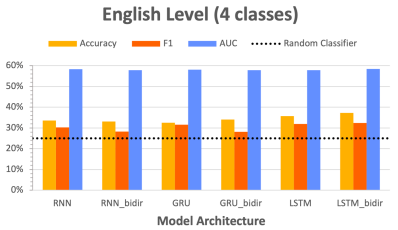
as baseline, denoted with a horizontal black dotted line in all plots, for which we compute the *a priori* distribution of the number of classes $C$, i.e.

$$\text{Random Accuracy } (\%) = \frac{100}{C} \tag{19}$$

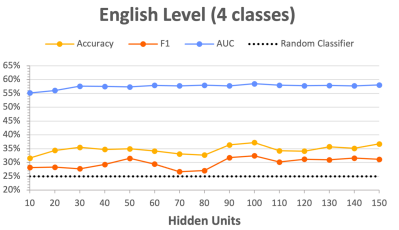
As explained in previous sections, if the models we train are no better than a random classifier, then we must reject our research hypothesis and conclude that swiping data do not reveal demographic or behavioral information about the user. Otherwise, we must validate our research hypothesis and report the best model configuration for each predicted target.

In our experiments we have investigated how model performance metrics vary by manipulating the different hyperparameters: layer type (RNN, GRU, LSTM), bidirectionality, number of hidden layers and hidden units, and optimizer's learning rate. The best performance results obtained through a systematic analysis of the chosen hyperparameters are reported from Figure 5 to Figure 12. As can be observed, in all cases, the trained models' performance is better than that of a random classifier, which indicates that our models are able to learn latent representations about the demographic and behavioral information from swiping trajectories.

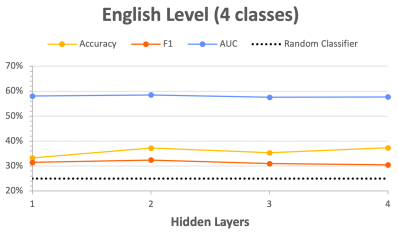
Table 1 summarizes the best performing models and their corresponding hyperparameters. We also provide a random classifier baseline to contextualize



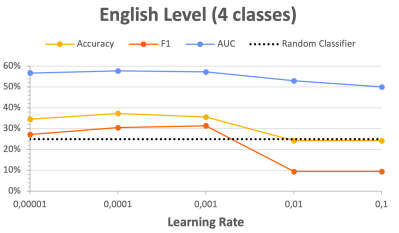

(a) Layer type




(b) Hidden Units

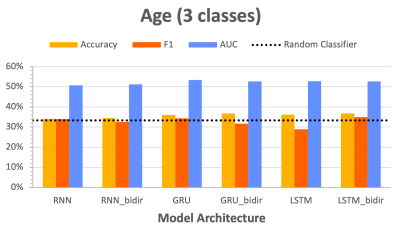



(c) Hidden Layers

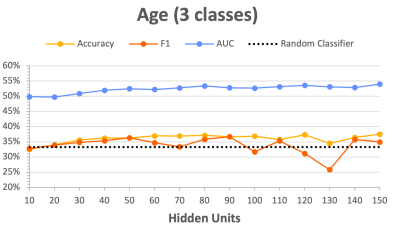



(d) Learning Rate

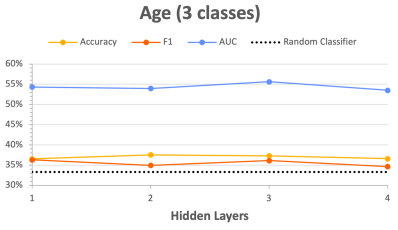
**Fig. 6**: Predicting the user's English level from swiped words trajectories.



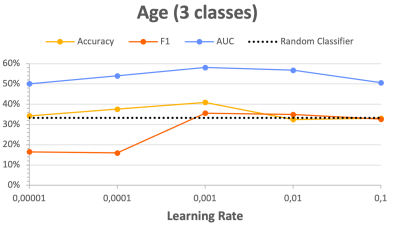

(a) Layer type




(b) Hidden Units

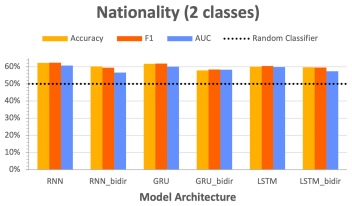



(c) Hidden Layers

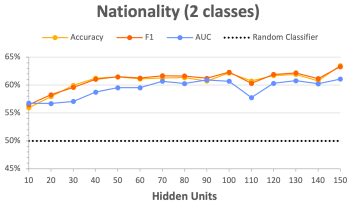



(d) Learning Rate

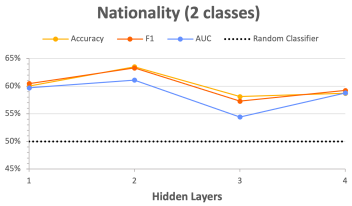
**Fig. 7**: Predicting the user's age from swiped words trajectories.



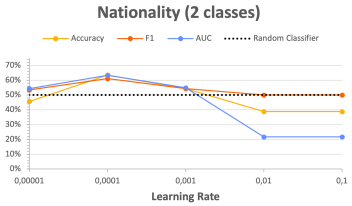

(a) Layer type




(b) Hidden Units

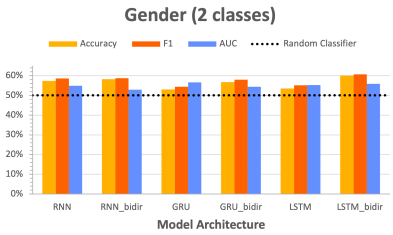



(c) Hidden Layers

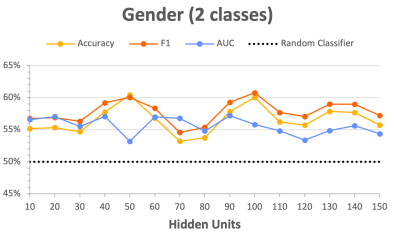



(d) Learning Rate

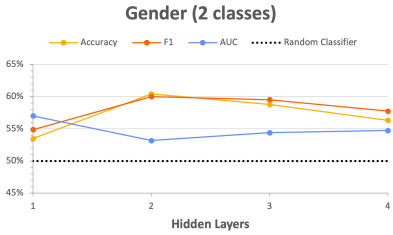
**Fig. 8**: Predicting the user's nationality from swiped words trajectories.



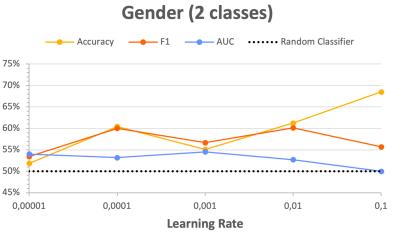

(a) Layer type




(b) Hidden Units

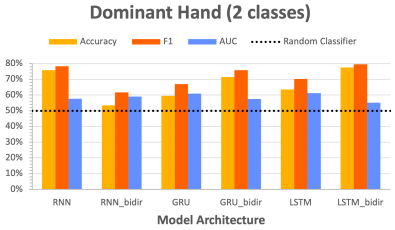



(c) Hidden Layers

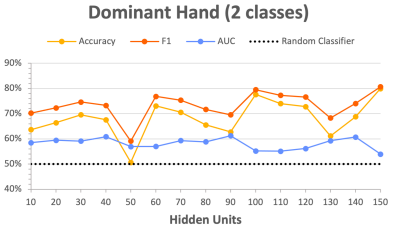



(d) Learning Rate

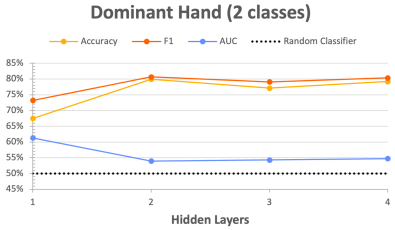
**Fig. 9**: Predicting the user's gender from swiped words trajectories.



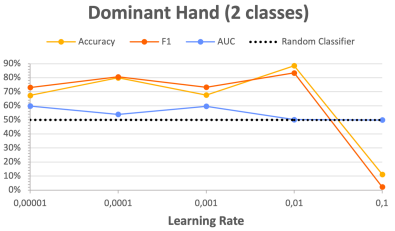

(a) Layer type




(b) Hidden Units

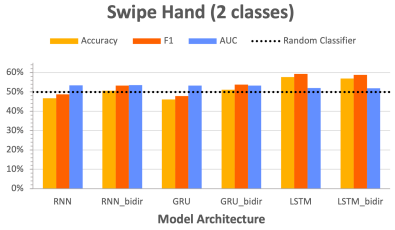



(c) Hidden Layers

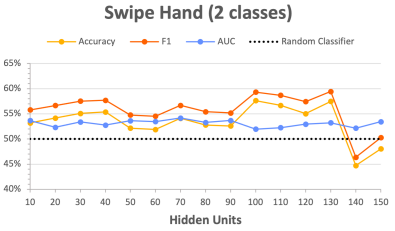



(d) Learning Rate

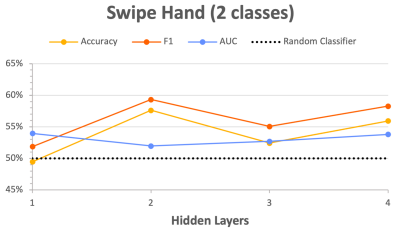
**Fig. 10**: Predicting the user's dominant hand from swiped words trajectories.



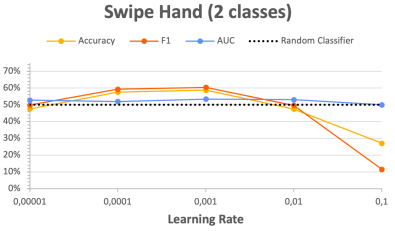

(a) Layer type




(b) Hidden Units

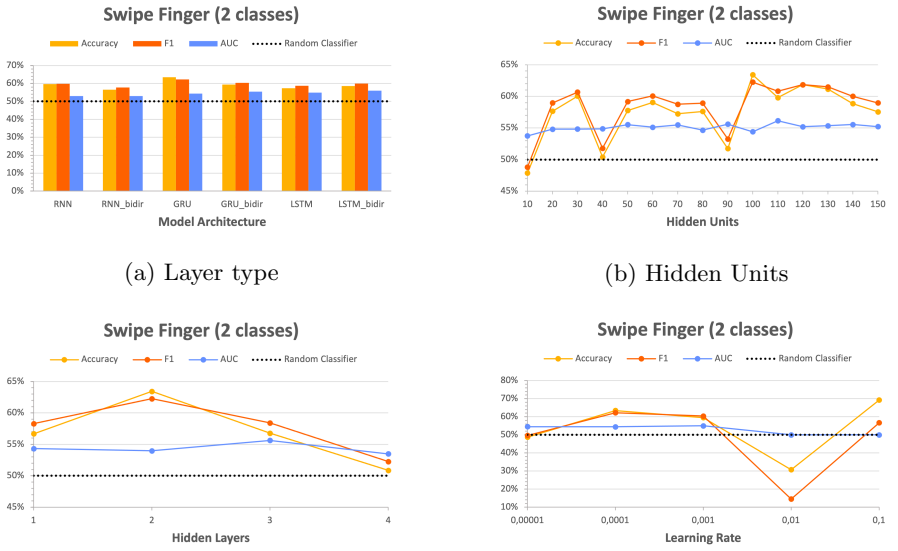



(c) Hidden Layers




(d) Learning Rate

**Fig. 11**: Predicting the user's swiping hand from swiped words trajectories.




(a) Layer type

(b) Hidden Units

(c) Hidden Layers

(d) Learning Rate

**Fig. 12**: Predicting the user's swiping finger from swiped words trajectories.

the importance of the results. As shown in the table, the classifiers performed best on most demographic and behavioural correlates by defining the model to be bidirectional and choosing recurrent layers with LSTM or GRU cells. All models performed better than a random classifier, with an increase from 4% to 10% for the swiping familiarity, age and swiping hand targets, and with an increase from 10 to almost 40% for the English level, nationality, gender, dominant hand and swiping finger targets. Overall, our results show that the trained classifiers were able to identify latent demographic traits from the swiping trajectories, hence validating our hypothesis that swiping carries rich information about the user. This opens interesting research avenues which are yet to be fully explored. For example, new techniques to prevent advanced user profiling should be devised in future work.

Finally, we also report the confusion matrices for each of the best performing models in Figure 13. As we can see, in most cases the trained models were able to successfully discriminate between classes. However, in other cases the models failed to learn properly to discriminate and degenerated as ZeroR classifier, i.e. a model that always predicts the majority class. This only happened to the models that predicted gender, dominant hand, and swiping finger. In any case, the performance of all models is better than that of a random classifier, therefore we validate our research hypothesis and conclude that, overall, demographic and behavioral information can be inferred from swiping trajectories.

| Target | Model | Layers | Units | LR | Rand. Acc. | Model Acc. |
|---|---|---|---|---|---|---|
| Age | BiGRU | 2 | 150 | $10^{-3}$ | 33.33% | 40.91% |
| Gender | BiLSTM | 2 | 50 | $10^{-1}$ | 50% | 68.51% |
| Nationality | RNN | 2 | 150 | $10^{-4}$ | 50% | 63.50% |
| Familiarity | LSTM | 2 | 140 | $10^{-4}$ | 33.33% | 37.87% |
| English level | BiLSTM | 4 | 100 | $10^{-4}$ | 25% | 37.32% |
| Dominant hand | BiLSTM | 2 | 150 | $10^{-2}$ | 50% | 88.57% |
| Swiping hand | LSTM | 2 | 100 | $10^{-3}$ | 50% | 58.74% |
| Swiping finger | GRU | 2 | 100 | $10^{-1}$ | 50% | 69.28% |

**Table 1**: Summary of the best model architectures for predicting demographic (top rows) and behavioral (bottom rows) information from swiping trajectories.

## 5 Limitations and Future Work

In this work we have focused on single-word classification, which is the most challenging scenario since some words might be as short as two characters and therefore the information a classifier can infer is minimal. It would be interesting to predict demographic and behavioral attributes at the sentence level, i.e., by considering all swiped words a user might enter in a row. This could be done, e.g., by concatenating the feature vectors of each word in the sentence. In addition, we have not considered word difficulty in our analysis. This might be an interesting avenue for future work, since it has been shown that experienced shape-writing users tend to be more loose while entering non-familiar words [4]. It may be the case, therefore, that other demographic and behavioral traits are highly correlated with swiping patterns. For example, ageing is marked by a decline in motor control abilities, which is reflected by the users' writing performance. Smith et al. [72] observed that older people incurred in longer movement times, but also more sub-movements and more pointing errors than the young. Unfortunately, we cannot study this effect in the dataset we have analyzed, since most users are aged between 20 and 40 years. More concretely, the dataset is skewed towards young female right-handed participants.

Our sequence models are relatively costly to train because of their recurrent nature and the large amount of swiping data we analyzed. For example, training the non-bidirectional LSTM models took between 1.5 to 10 hours in the High-Performance Computing facilities of our university, whereas the same model with bidirectional layers took between 3 and 20 hours. While these training times might be small for today's standards, real-time processing is clear out of reach at present, which means that we cannot build our models as new data come in. We also have not experimented with attention mechanisms such as local [73] and global [74] attention. Attention has been proved to be effective for recurrent models, but we decided to go for an initial exploration of the


(a) Familiarity (b) English level (c) Age

(d) Nationality (e) Gender (f) Dominant hand

(g) Swiping hand (h) Swiping finger

**Fig. 13**: Confusion matrices for the best performing models reported in Table 1.

feasibility of our approach in the first place. Given that our results are promising, we can even experiment with other architectures such as Transformers in future work.

Furthermore, we have explored sequential models only, as they are better suited to handle the sequential nature of swipe trajectories. Observing the promising results obtained in our study, we believe that other data representations should be explored. For example, if we consider images (pixel values) instead of discrete spatio-temporal time series as input data, then we could try Convolutional Neural Networks (CNNs) as well as more powerful architectures like Vision Transformers (ViT). This is left as an interesting opportunity for future work.

Another interesting opportunity for future work is to consider more balanced splits of the data. Given the number of target classes, there are 1152

possible combinations, not all of which may be well represented, and are certainly not represented equally. We should point out, however, that our study is the first to use the original "How we Swipe" dataset for training Machine Learning models and to get a more realistic estimate of their performance. We also have not considered any interactions between targets, such as gender and age. For example, training only for young females (which happens to be the most represented group of participants) might result in more accurate models.

Finally, we should remark that all mobile vendors provide a swiping service to millions of users. Although, they ask for consent to track swiping information for the purpose of improving the service and enhancing user experience, our study brings to light that extracting personal information of users from swiping data is also within reach. The significantly large amount of swiping data mobile vendors can have access to, allows them to build a powerful demographic and behavioral inference engines. Thus, the users can potentially give up their privacy, without actually consenting to it, only by sharing their swiping information with mobile vendors.

# 6 Conclusion

We have analyzed the dynamics of swiping trajectories from the only publicly available swiping dataset to model and predict different demographic and behavioural correlates. We explored three popular sequence model architectures (RNN, LSTM, GRU) and conducted a systematic analysis regarding which model hyperparameters yield better classification performance. Overall, the classifiers performed best on most demographic and behavioural traits by defining the models to be bidirectional and choosing recurrent layers with LSTM or GRU cells having 100 units each. We noticed that the eight considered targets are challenging to predict, however the results from our experimental evaluations demonstrate that all models perform better than a random classifier, with an increase from 4 to 10% for the swiping familiarity, age, and swiping hand targets, and with an increase from 10 to 40% for the English level, nationality, gender, dominant hand, and swiping finger targets. For the three latter targets (gender, dominant hand, and swiping finger) the trained models tended to predict the majority class.

Taken together, our results show that the classifiers are able to identify latent demographic and behavioral information from swiping trajectories, hence validating our research hypothesis that swiping carries rich information about the user. This finding may have unexpected consequences for user's privacy, since currently swiping is supported by all mobile vendors and has millions of users, so people may be inadvertently profiled at an unprecedented granularity. It is a matter of time for researchers to build more sophisticated Machine Learning models, therefore future work should consider new ways of addressing these issues without impacting the user's swiping experience.

**Acknowledgements.** The experiments presented in this paper were carried out using the HPC facilities of the University of Luxembourg: http://hpc.uni.lu

# Declarations

## Funding

This work was supported by the Horizon 2020 FET program of the European Union through the ERA-NET Cofund funding grant CHIST-ERA-20-BCI-001 and the European Innovation Council Pathfinder program (SYMBIOTIK project, grant 101071147).

## Competing interests

The authors have no competing interests to declare that are relevant to the content of this article.

## Availability of data and materials

The swiping dataset we used is publicly available at https://osf.io/sj67f/.

# Authors' contributions

**D.C.A. Lemarquis**: Software, Writing - Original Draft. **B.A. Yilma**: Methodology, Writing - Original Draft. **L.A. Leiva**: Conceptualization, Methodology, Writing - Reviewing and Editing.

# References

[1] Reyal, S., Zhai, S., Kristensson, P.O.: Performance and user experience of touchscreen and gesture keyboards in a lab setting and in the wild. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 679–688 (2015)

[2] Palin, K., Feit, A.M., Kim, S., Kristensson, P.O., Oulasvirta, A.: How do people type on mobile devices? observations from a study with 37,000 volunteers. In: Proc. Intl. Conf. on Human-computer Interaction with Mobile Devices and Services (MobileHCI), pp. 1–12 (2019)

[3] Quinn, P., Zhai, S.: Modeling gesture-typing movements. Hum.-Comput. Interact. **33**(3) (2018)

[4] Leiva, L.A., Kim, S., Cui, W., Bi, X., Oulasvirta, A.: How we swipe: A large-scale shape-writing dataset and empirical findings. In: Proc. Intl. Conf. on Human-computer Interaction with Mobile Devices and Services (MobileHCI) (2021)

[5] Buschek, D., Bisinger, B., Alt, F.: ResearchIME: A mobile keyboard application for studying free typing behaviour in the wild. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1–14 (2018)

[6] Henze, N., Rukzio, E., Boll, S.: Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 2659–2668 (2012)

[7] Dhakal, V., Feit, A.M., Kristensson, P.O., Oulasvirta, A.: Observations on typing from 136 million keystrokes. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1–12 (2018)

[8] Zhai, S., Kristensson, P.O.: Shorthand writing on stylus keyboard. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 97–104 (2003)

[9] Markussen, A., Jakobsen, M.R., Hornbæk, K.: Vulture: A mid-air word-gesture keyboard. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1073–1082 (2014)

[10] Zhu, S., Zheng, J., Zhai, S., Bi, X.: i'sFree: Eyes-free gesture typing via a touch-enabled remote control. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1–12 (2019)

[11] Gupta, A., Ji, C., Yeo, H.-S., Quigley, A., Vogel, D.: RotoSwype: Word-gesture typing using a ring. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1–12 (2019)

[12] Yeo, H.-S., Phang, X.-S., Castellucci, S.J., Kristensson, P.O., Quigley, A.: Investigating tilt-based gesture keyboard entry for single-handed text entry on large devices. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 4194–4202 (2017)

[13] Kristensson, P.O., Zhai, S.: Command strokes with and without preview: using pen gestures on keyboard for command selection. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1137–1146 (2007)

[14] Alvina, J., Griggio, C.F., Bi, X., Mackay, W.E.: CommandBoard: Creating a general-purpose command gesture input space for soft keyboard. In: Proc. ACM Symposium on User Interface Software Technology (UIST), pp. 17–28 (2017)

[15] Cui, W., Zheng, J., Lewis, B., Vogel, D., Bi, X.: HotStrokes: Word-gesture shortcuts on a trackpad. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI), pp. 1–13 (2019)

[16] Starov, O., Gill, P., Nikiforakis, N.: Are you sure you want to contact us? quantifying the leakage of PII via website contact forms. In: Proc. PoPETs (2016)

[17] Leung, C., Ren, J., Choffnes, D., Wilson, C.: Should you use the app for that? comparing the privacy implications of app- and web-based online services. In: Proc. Internet Measurements Conference (IMC) (2016)

[18] Leiva, L.A., Diaz, M., Ferrer, M.A., Plamondon, R.: Human or machine? it is not what you write, but how you write it. In: Proceedings of the Intl. Conf. on Pattern Recognition (ICPR) (2020)

[19] Leiva, L.A., Arapakis, I., Iordanou, C.: My mouse, my rules: Privacy issues of behavioral user profiling via mouse tracking. In: Proceedings of ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR) (2021)

[20] White, R.W., Doraiswamy, P.M., Horvitz, E.: Detecting neurodegenerative disorders from web search signals. npj Digital Med. **1**(8) (2018)

[21] Gajos, K.Z., Reinecke, K., Donovan, M., Stephen, C.D., Hung, A.Y., Schmahmann, J.D., Gupta, A.S.: Computer mouse use captures ataxia and parkinsonism, enabling accurate measurement and detection. Mov. Disord. **35**(2) (2020)

[22] Chen, M.C., Anderson, J.R., Sohn, M.H.: What can a mouse cursor tell us more? correlation of eye/mouse movements on web browsing. In: Proc. Extended Abstracts on Human Factors in Computing Systems (CHI EA) (2001)

[23] Mueller, F., Lockerd, A.: Cheese: Tracking mouse movement activity on websites, a tool for user modeling. In: Proc. Extended Abstracts on Human Factors in Computing Systems (CHI EA) (2001)

[24] Huang, J., White, R.W., Buscher, G., Wang, K.: Improving searcher models using mouse cursor activity. In: Proc. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR) (2012)

[25] Huang, J., White, R., Buscher, G.: User see, user point: Gaze and cursor alignment in web search. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI) (2012)

[26] Navalpakkam, V., Jentzsch, L., Sayres, R., Ravi, S., Ahmed, A., Smola, A.: Measurement and modeling of eye-mouse behavior in the presence of nonlinear page layouts. In: Proc. The Web Conference (WWW) (2013)

[27] Arapakis, I., Leiva, L.A.: Predicting user engagement with direct displays using mouse cursor information. In: Proc. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR) (2016)

[28] Chen, Y., Liu, Y., Zhang, M., Ma, S.: User satisfaction prediction with

mouse movement information in heterogeneous search environment. IEEE Trans. Knowl. Data. Eng. **29**(11) (2017)

[29] Liu, Y., Chen, Y., Tang, J., Sun, J., Zhang, M., Ma, S., Zhu, X.: Different users, different opinions: Predicting search satisfaction with mouse movement information. In: Proc. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR) (2015)

[30] Arroyo, E., Selker, T., Wei, W.: Usability tool for analysis of web designs using mouse tracks. In: Proc. Extended Abstracts on Human Factors in Computing Systems (CHI EA) (2006)

[31] Atterer, R., Wnuk, M., Schmidt, A.: Knowing the user's every move: User activity tracking for website usability evaluation and implicit interaction. In: Proc. The Web Conference (WWW) (2006)

[32] Leiva, L.A.: Restyling website design via touch-based interactions. In: Proc. Intl. Conf. on Human-computer Interaction with Mobile Devices and Services (MobileHCI) (2011)

[33] Krátky, P., Chudá, D.: Recognition of web users with the aid of biometric user model. J. Intell. Inf. Syst. **51**(3) (2018)

[34] Lu, H., Rose, J., Liu, Y., Awad, A., Hou, L.: Combining mouse and eye movement biometrics for user authentication. In: Traoré, I., Awad, A., Woungang, I. (eds.) Information Security Practices. Springer, ??? (2017)

[35] Claypool, M., Le, P., Wased, M., Brown, D.: Implicit interest indicators. In: Proc. Intelligent User Interfaces (IUI) (2001)

[36] Shapira, B., Taieb-Maimon, M., Moskowitz, A.: Study of the usefulness of known and new implicit indicators and their optimal combination for accurate inference of users interests. In: Proc. Symposium on Applied Computing (SAC) (2006)

[37] Guo, Q., Agichtein, E.: Exploring mouse movements for inferring query intent. In: Proc. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR) (2008)

[38] Guo, Q., Agichtein, E.: Ready to buy or just browsing? detecting web searcher goals from interaction data. In: Proc. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR) (2010)

[39] Guo, Q., Lagun, D., Agichtein, E.: Predicting web search success with fine-grained interaction data. In: Proc. Intl. Conf. on Information and Knowledge Management (CIKM) (2012)

[40] Huang, J., White, R.W., Dumais, S.: No clicks, no problem: Using cursor movements to understand and improve search. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI) (2011)

[41] Guo, Q., Agichtein, E.: Beyond dwell time: Estimating document relevance from cursor movements and other post-click searcher behavior. In: Proc. The Web Conference (WWW) (2012)

[42] Speicher, M., Both, A., Gaedke, M.: TellMyRelevance! predicting the relevance of web search results from cursor interactions. In: Proc. Intl. Conf. on Information and Knowledge Management (CIKM) (2013)

[43] Diriye, A., White, R., Buscher, G., Dumais, S.: Leaving so soon? understanding and predicting web search abandonment rationales. In: Proc. Intl. Conf. on Information and Knowledge Management (CIKM) (2012)

[44] Arapakis, I., Lalmas, M., Cambazoglu, B.B., Marcos, M.-C., Jose, J.M.: User engagement in online news: Under the scope of sentiment, interest, affect, and gaze. J. Assoc. Inf. Sci. Technol. **65**(10) (2014)

[45] Arapakis, I., Lalmas, M., Valkanas, G.: Understanding within-content engagement through pattern analysis of mouse gestures. In: Proc. Intl. Conf. on Information and Knowledge Management (CIKM) (2014)

[46] Hauger, D., Paramythis, A., Weibelzahl, S.: Using browser interaction data to determine page reading behavior. In: Proc. UMAP (2011)

[47] Lagun, D., Ageev, M., Guo, Q., Agichtein, E.: Discovering common motifs in cursor movement data for improving web search. In: Proc. ACM Conf. on Web Search and Data Mining (WSDM) (2014)

[48] Boi, P., Fenu, G., Spano, L.D., Vargiu, V.: Reconstructing user's attention on the web through mouse movements and perception-based content identification. ACM Trans. Appl. Percept. **13**(3) (2016)

[49] Arapakis, I., Penta, A., Joho, H., Leiva, L.A.: A price-per-attention auction scheme using mouse cursor information. ACM Trans. Inf. Syst. **38**(2) (2020)

[50] Accot, J., Zhai, S.: Beyond fitts' law: Models for trajectory-based HCI tasks. In: Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI) (1997)

[51] Card, S.K., English, W.K., Burr, B.J.: Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys, for text selection on a CRT. In: Baecker, R.M., Buxton, W.A.S. (eds.) Human-computer Interaction. Taylor & Francis, ??? (1987)

[52] Zimmermann, P., Guttormsen, S., Danuser, B., Gomez, P.: Affective computing – a rationale for measuring mood with mouse and keyboard. Int. J. Occup. Saf. Ergon. **9** (2003)

[53] Kaklauskas, A., Krutinis, M., Seniut, M.: Biometric mouse intelligent system for student's emotional and examination process analysis. In: Proc. ICALT (2009)

[54] Azcarraga, J., Suarez, M.T.: Predicting academic emotions based on brainwaves, mouse behaviour and personality profile. In: Proc. PRICAI (2012)

[55] Yamauchi, T.: Mouse trajectories and state anxiety: Feature selection with random forest. In: Proc. ACII (2013)

[56] Kapoor, A., Burleson, W., Picard, R.W.: Automatic prediction of frustration. Int. J. Hum.-Comput. Stud. **65**(8) (2007)

[57] Yamauchi, T., Bowman, C.: Mining cursor motions to find the gender, experience, and feelings of computer users. In: Proc. ICDMW (2014)

[58] Kratky, P., Chuda, D.: Estimating gender and age of web page visitors from the way they use their mouse. In: Proc. WWW Companion (2016)

[59] Pentel, A.: Predicting age and gender by keystroke dynamics and mouse patterns. In: Adj. Proc. UMAP (2017)

[60] Vertanen, K., Kristensson, P.O.: A versatile dataset for text entry evaluations based on genuine mobile emails. In: Proc. Intl. Conf. on Human-computer Interaction with Mobile Devices and Services (MobileHCI), pp. 295–298 (2011)

[61] Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Proc. NeurIPS (2014)

[62] Du, W., Wang, Y., Qiao, Y.: Recurrent spatial-temporal attention network for action recognition in videos. IEEE Trans. Image Process. **27**(3) (2018)

[63] Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., Yuille, A.: Deep captioning with multimodal recurrent neural networks (m-RNN). In: Proc. ICLR (2015)

[64] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8) (1997)

[65] Cho, K., van Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder

for statistical machine translation. In: Proc. EMNLP (2014)

[66] Jozefowicz, R., Zaremba, W., Sutskever, I.: An empirical exploration of recurrent network architectures. In: Proc. ICML (2015)

[67] Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NeurIPS Workshops (2014)

[68] Dey, R., Salemt, F.M.: Gate-variants of gated recurrent unit (GRU) neural networks. In: Proc. MWSCAS (2017)

[69] D.P.Kingma, Ba, J.L.: Adam: A method for stochastic optimization. In: Proc. ICLR (2015)

[70] Ranganath, R., Gerrish, S., Blei, D.M.: Black box variational inference. In: Proc. AISTATS (2014)

[71] Powers, D.M.W.: Evaluation: from Precision, Recall and F-measure to ROC, informedness, markedness and correlation. J. Mach. Learn. Technol. **2**(1) (2011)

[72] Smith, M.W., Sharit, J., Czaja, S.J.: Aging, motor control, and the performance of computer mouse tasks. Hum. Factors **41**(3) (1999)

[73] Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proc. ICLR (2015)

[74] Luong, M.-T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proc. EMNLP (2015)