POZNAN UNIVERSITY OF TECHNOLOGY

FACULTY OF COMPUTING AND TELECOMMUNICATION
Institute of Computing Science

Bachelor's thesis

BIOMETRIC IDENTIFICATION OF A SMARTPHONE USER USING GRAPH NEURAL NETWORKS

Jakub Grabowski, 151825
Filip Kozłowski, 151823
Krzysztof Matyla, 151778
Igor Warszawski, 151585

Supervisor
dr hab. inż. Szymon Szczęsny, prof. PP

POZNAŃ 2025

Tutaj będzie karta pracy dyplomowej;

oryginał wstawiamy do wersji dla archiwum PP, w pozostałych kopiach wstawiamy ksero.

Contents

Chapter 1

Introduction

TODO for JG: additional introduction to biometrics and their usage.

The project aimed to develop a model, along with a corresponding mobile app, capable of recognizing the user by their biometric data contained mostly within the keystroke data. The users in the study, which was a part of the project, provided their data by entering long stretches of text as testing data. Models were created for each user, with the standard model testing procedures and validations. A subgroup of the study participants was also asked to verify the model in real-life testing by writing short paragraphs in the application, which were sent to the server for user verification.

The scope of the work was to create a mobile application capable of gathering the keystroke data, which could then be used by the server to create Graph Neural Network (GNN) models tasked with recognizing the user as opposed to other possible users. Also in the scope was performing a study on a group of participants who provided the data for the project and participated in the application and model demonstration and testing.

The sources used in this thesis mostly concerned the two following groups: studies of keystroke data models and their effectiveness and the specialist literature on the topic of Graph Neural Networks.

The thesis has the following structure: Chapter 2 consists of some theory concerning biometrics, especially in the context of user input data, with a small literature review about using biometrics for user recognition. Chapter 2.1 contains basic theoretics about Graph Convolutional Networks, which are used for user recognition in the model created for the project. Chapter 3.1 is a brief overview of the project, explaining its components and the relationships between them. Chapter 3.2 describes the mobile application used for user data collection and model validation. Chapter 3.3 contains a description of the Neural Network model used for user recognition, complete with the hyperparameters used in model training and validation. Chapter 3.4 discusses the metrics used in the model testing on data gathered from users and the testing results. Chapter 3.5 concerns the user testing with the help of study participants and the study results. Chapter 4 is a conclusion to the thesis.

Work on this project was divided as follows: Jakub Grabowski created the mobile application, set up and coordinated the project, and researched biometrics for his thesis paper. Filip Kozłowski created the server and integrated the GNN model with it. He also planned and implemented communication between the server and the application. Krzysztof Matyla helped in creating the mobile application, provided testing for various parts of the project, and coordinated user testing. Igor Warszawski planned and implemented the GNN model used on the server. He also tested and validated the results, together with Filip Kozłowski.

Chapter 2

Biometrics in mobile devices - theory

Fundamental to the goal of the project was the use of biometric data in user identification. Biometric data can be defined as measurements of some unique characteristics of an individual. These can largely be divided into two main categories: physiological data, which is the measurement of the inherent characteristics of an individual's body, such as a fingerprint, an iris scan or a face scan, and behavioral data, which measures the person's movements, behaviors, speech patterns etc. [1]

One possible way to extract data from a person's behavior is via *keystroke dynamics*. This type of behavioral biometrics is acquired from a user by means of a keyboard or other typing device and records and extracts features from the way the keyboard is used. Most commonly used and almost universally applicable to any keyboard device is the measurement of timings between each character typed. If the user uses a physical keyboard, it is also convenient to derive the following features [2]:

1. **Hold Time** – time between key press and release
2. **Down-Down Time** – time between first key press and second key press
3. **Up-Up Time** – time between first key release and second key release
4. **Up-Down Time** – time between first key release and second key press
5. **Down-Up Time** – time between first key press and second key release.

With some keyboards it may be more difficult to gather all the possible features. Even basic feature, such as the hold time can prove difficult when using for example GBoard on mobile devices, which does not naturally send key press and key release information to the application. This information can thus only be gathered in approximation or by building another virtual keyboard application. This, however, has its drawbacks. The users are generally used to one type of keyboard (on mobile it may be for example GBoard or SwiftKey), so forcing them to use another type of keyboard may be detrimental.

While the model may be less accurate because of the lack of features, there can be some ways to mitigate it. Some other features can be added, which are largely specific to mobile devices, such as accelerometer data, or a larger sample can be used. A few of those options were considered by the researchers, and the results will be discussed in the next chapters (chapters 3.3 to 3.5).

The keystroke identification can also rely on other data gathered from the keyboard, such as the specifics of letters used, their average frequencies, most common connections between the letter or other statistics [3]. These statistics can be modeled in many ways. If the average Up-Up Time between two keys is gathered from the data, a graph can be formed, having additional features as see fit by the designers. Such graphs were constructed for the Neural Network models constructed in this study, which will be discussed in the next chapter.

TODO for JG: continue the paragraphs.

Chapter 3

Graph Convolutional Networks - theory

TODO for anyone/everyone (probably me and FK): how GNNs work, how GCNs work, how the networks can be constructed.

Chapter 4

Implementing keystroke data on mobile devices

There are many ways to recognise phone user using biometrics, such as scanning fingerprints or facial recognition. It's very useful for security purposes. The ease of use and reliability have made passwords less popular and led to their replacement by biometrics. However, since other biometrics are also available, it is reasonable to test if biometrics derived from writing button press intervals and phone orientation could also be a reliable way to recognise the user. To collect data and test the results, the mobile application was created. The main goal of the application is to gather data with an easy-to-use, intuitive interface, send the data to a server for training purposes, check if the model recognises the user.

As previously stated, State of the Art models can actually perform well* on such data. These models are however usually trained on data gathered from physical keyboards. Additionally, the Neural Network model created for user identification was chosen to be based on Graph Convolutional Networks, which are still rather novel*. Thanks to that, an important part of the project was a study of results and data gathered, which is presented in chapter 3.4 and 3.5.

TODO: find statistics and add them to sources

Chapter 5

Mobile application for data gathering and model testing

The application was written for Android devices supporting Android 8.1 or newer. As of 2024*, more than 93% of Android devices should be compatible. The Android platform was chosen, as it was easier to test on and find a study group of the Android users as opposed to the iOS users (according to * , significantly more people in Poland, where the researchers are based in, use Android devices).

Technology used in the mobile application itself was Jetpack Compose, which is quoted by Google to be "Android's recommended modern toolkit for building native UI" (from site*). Language used was Kotlin. Persisence was achieved by using Android Room, which provided an abstraction layer over SQLite database, which was used for data collection.

TODO for KM: add statistics sources (Internet), some technicals about the inner workings of the app. How the data is stored, what is gathered and when. If you have any doubts, feel free to ask about any methods/composables. I will be updating method descriptions/docs soon – JG

Model View Controller and DataStore...

Data was modeled as...

Data was saved...

Application design...

Training screen...

Testing screen...

Communications with the server...

Data sent to the server and downloaded locally...

Chapter 6

Graph Convolutional Network for user recognition

TODO for IW: anything and everything about the model, the inner structure, the feature extraction can also go there, ask FK how you want to split these subjects up. FK will probably also check in with something here.

Chapter 7

Model fine-tuning and hyperparameters - metrics

TODO for IW: write about the fine-tuning process, the metrics used and why are they used, cross-validations used etc. You can also post some hyperparam statistics here.

Chapter 8

Testing model on users

TODO for xxx: when the tests are done (hopefully a week) we will discuss this.

Chapter 9

Conlusion

TODO for JG: not needed now, will write after the user tests are done.

Bibliography

[1] Hussien AbdelRaouf, Samia Allaoua Chelloug, Ammar Muthanna, Noura Semary, Khalid Amin, and Mina Ibrahim. Efficient convolutional neural network-based keystroke dynamics for boosting user authentication. *Sensors*, 23(10), 2023.

[2] Atharva Sharma, Martin Jureček, and Mark Stamp. Keystroke dynamics for user identification, 2023.

[3] C. Wang, H. Tang, H. Y. Zhu, J. H. Zheng, and C. J. Jiang. Behavioral authentication for security and safety. *Security and Safety*, 3:2024003, 2024.