Password Anomaly Detection

Mario Murrent

Contents

Introduction

PassSecure The purpose is to develop a simple password application which uses machine learning to detect if a password entered by the user is indeed from a genuine user or not. An application to test the selected algorithm is developed and described. The application is based on the users unique typing rythm.

1. Algorithms

This section will describe all tested algorithms.

1.1 Kullback Leibler divergence

The Kullback Leibler divergence is a non symmetric measure of the difference between two probability distributions P and Q.[1] This method will give you a number between 0 and 1, but the problem is that is does not take into account if the password typing speed varies. In fact that algorithm is not suitable for this specific purpose.

1.2 Manhattan Distance

The Manhattan distance function computes the distance that would be traveled to get from one data point to the other if a grid-like path is followed. The Manhattan distance between two items is the sum of the differences of their corresponding components[2].
The Manhattan distance is according to the paper ”Comparing Anomaly-Detection Algorithms for Keystroke Dynamic” [3] on of the best algorithms.

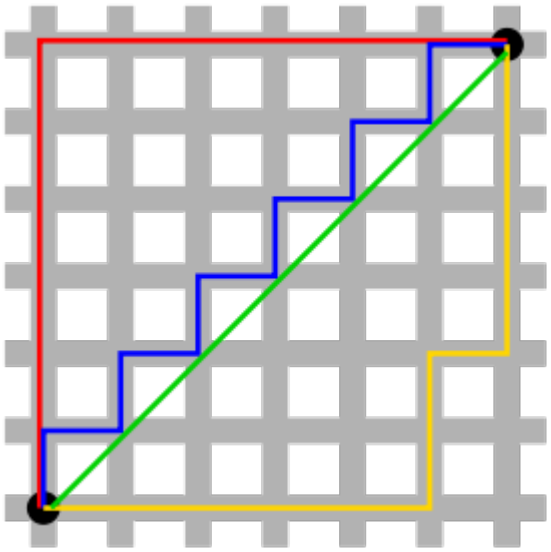

Figure 1. Red: Manhattan distance. Green: diagonal, straight-line distance. Blue, yellow: equivalent Manhattan distances.[2]

2. Implementation Details

During the training mode all necessary data is collected including:

- Total time between first key up and last key up
- Total time between first key down and last key down
- Time between each key up
- Time between each key down
- Average key down time
- Average key up time
- Manhattan distance

The complete training consists of a set of training entries. A training entry is only added when the password is correct. Failures are not taken into account. On adding a new training entry the distance to the training set is calculated and the training set is analyzed. Analyzing a training set means recalculating all average values.

2.1 Used Libraries

- Accord - Image Processing & Machine Learning Framework (`http://accord-framework.net/intro.html`)

2.2 Restrictions

The algorithm is tested with normal user input and not with any password management software.

3. Application

The application is designed to test the used algorithm to detect if a password entered by the user is indeed from a genuine user or not. It is possible to enter multiple users with specific passwords. Every user is evaluated separately. It is also possible to import or export data which has been recorded. The data windows is designed to view data which is used

Figure 2. A screenshot of applications main window.

for evaluation. The application has two modes: Training

Figure 3. A screenshot of the data window.

& Normal. During the training mode all necessary data is collected. In the normal mode the current password is checked against the captured data. A picture on the right side of the login controls indicates whether the password is accepted or not. Partially accepted means that if might not be the user.

4. Summary and outlook

The algorithm used to analyze the keystrokes seems to be quite good. Basically it is possible to check if the password which is entered is from a genuine user or not, but you need alot of training data. The test data used to test the algorithm contains about 1200 traiing entries. The algorithm is tested with a few unit tests which cover some special cases but definitely not all.

The algorithm used just covers some basic cases. Another algorithm should be implemented and compared to the Manhattan Distance algorithm.

References

[1] Kullback leibler divergence. `http://en.wikipedia.org/wiki/Kullback-Leibler_divergence`. Accessed: 2010-09-30.

[2] Manhattan distance. `http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Clustering_Parameters/Manhattan_Distance_Metric.htm`. Accessed: 2010-09-30.

[3] A. J. Figueredo and P. S. A. Wolf. Assortative pairing and life history strategy - a cross-cultural study. *Human Nature*, 20:317–330, 2009.