Spying On Runners

Advanced Security Project

Léo Besson Paolo Daolio

22/01/2021

The aim of this project is to infer social links between people based on their running records found on a community french website. The first part of this study was to collect the maximum amount of data from the website. The second part was to analyse this data using similarity metrics to find matching profiles. The produced tool is able to quantify the probability of two people knowing each other and for one person, reavealing all possible related people. The results of this study still have to be tested by contacting the recorded people and verifying the correctness of the infered social links.

1 Introduction

blablabla

2 Scraping

blablabla

3 Data Analysis

All the collected data is processed to guess linked people in the set of collected runners. To do so, we need a function to quantify how probable is the fact that two people know each other based on their race records. This type of function is called similarity metrics. Metrics needs to be analysed to select the correct threshold and to compare how well they perform on this specific application. Finally, the refined metrics are used to build the Python application.

3.1 Selected Similarity Metrics

The metrics chosen for this application are taken from the paper written by Cunche, Kaafar, and Boreli (2012). To infer social link between runners, we need to see the races they have in common in their record and how likely they have been in contact in this race. To measure this second parameter, we use the number of participants in each common race. The less people there are in the race, the most probable it is that the two persons were in contact and may know each other. All the metrics are implemented in the file `analysis_module/metrics.py`

- *Jaccard index*: This metric focuses on the proportion of common races in the record. it is defined as $\mathrm{J}(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}$. It is implemented in the function `jaccard_index`.
- *IDF similarity*: This metric depends on a measure of the rarity of a race defined as $IDF_i = \log \frac{1}{f_i}$ with the frequency of the race f_i the number of participants. Finally the metric is define as followed.

  $$\text{Cosine-IDF}(X, Y) = \frac{\sum_{x \in X \cap Y} IDF_x^2}{\sqrt{\sum_{x \in X} IDF_x^2} \sqrt{\sum_{y \in Y} IDF_y^2}}$$

  It is implemented in the function `idf_similarity`.
- *Adamic similarity*: This metric also depends on the frequency f_i defined as the number of participants in the race i. This metric is defined as $\text{Adamic}(X, Y) = \sum_{i \in X \cap Y} \frac{1}{\log f_i}$. It is implemented in the function `adamic_similarity`.
- *Modified Adamic similarity*: In order to put more weight on the rarity of the races, the Adamic similarity is changed to $\text{Psim-q}(X, Y) = \sum_{i \in X \cap Y} \frac{1}{f_i^q}$. It is implemented in the function `psim_q`.

3.2 Metrics Evaluation

The following part aims at assessing how accurately the previous metrics are at infering social links. As all the metrics return a number, we need to define a threshold such that if the returned value is above the threshold, we consider that the two persons are linked and if the returned value is below the threshold, we consider that the two person are not linked. All the methods described in this part are implemented in `analysis_module/metrics_analysis.py`

The first method followed by the paper written by Cunche, Kaafar, and Boreli (2012) is to separate the database into two sets: a first set where every person is really socially linked to another one and another set where there is no couple of linked people. Then, we run the metric on the whole set and count the number of true positives, true negatives, false positives and false negatives. The closer we get from reality, the most accurate is the metric. As we didn't had the time and ressources to build such databases based on verified testimonies, we made the following simplifying assumption: *two people know each other if and only if they ran in the same club at least one time.* Running the metrics analysis based on this assumption gave erratic results so we realised that this assumption is false and could not replace real data on social links.

As we could not properly conduct this analysis, the choice of the threshold and metric in the final application is left to the user.

3.3 Analysis Python Module

4 Conclusion

blablabla

References

Cunche, Mathieu, Mohamed Ali Kaafar, and Roksana Boreli. 2012. “I know who you will meet this evening! Linking wireless devices using Wi-Fi probe requests.” In *WoWMoM - 13th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks - 2012*. San Francisco, United States. https://hal.inria.fr/hal-00747825.