2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

Towards Sentiment Analysis for Mobile Devices

Johnnatan Messias, Joao P. Diniz, Elias Soares, Miller Ferreira,
Matheus Araujo, Lucas Bastos, Manoel Miranda, Fabricio Benevenuto
Universidade Federal de Minas Gerais (UFMG), Brazil
{johnnatan, jpaulo, eliassoares, millermarques, matheus.araujo, lucasmbastos, manoelrmj, fabricio}@dcc.ufmg.br

***Abstract*—The increasing use of smartphones to access social media platforms opens a new wave of applications that explore sentiment analysis in the mobile environment. However, there are various existing sentiment analysis methods and it is unclear which of them are deployable in the mobile environment. This paper provides the first of a kind study in which we compare the performance of 17 sentence-level sentiment analysis methods in the mobile environment. To do that, we adapted these sentence-level methods to run on Android OS and then we measure their performance in terms of memory usage, CPU time, and battery consumption. Our findings unveil sentence-level methods that require almost no adaptations and run relatively fast as well as methods that could not be deployed due to excessive use of memory. We hope our effort provides a guide to developers and researchers interested in exploring sentiment analysis as part of a mobile application and can help new applications to be executed without the dependency of a server-side API.**

I. Introduction

A large amount of information shared on mobile devices has opened space for a new wave of mobile applications. There is a large potential for sentiment analysis methods for mobile environments and most of the effort has been concentrated on the development of approaches able to measure the well-being of a smartphone user [1, 2, 3]. There are many other applications, for example, to help users organize the information they read and even analyze opinions extracted from the content they receive [4]. However, little is known about the deployability of these methods in the mobile environment.

Implement sentiment analysis technology on mobile devices is key for many applications. First, sentence-level sentiment analysis methods represent the basis for many other methods to measure user mood and feelings. Second, it allows applications to measure sentiment of user's instant messages.

This paper focuses on measuring the performance of existing popular sentence-level methods to investigate the feasibility of deploying them as part of mobile applications. To do that, we implement 17 sentence-level sentiment analysis methods: AFINN, Emoticons, Emolex, Happiness Index, NRC Hashtag Sentiment Lexicon, OpinionLexicon, OpinionFinder (MPQA), Panas-t, USent, SASA, Sentiment140 Lexicon, SentiStrength, SentiWordNet, Stanford Recursive Deep Model, Umigon, SenticNet, and Vader. All these methods are implemented in the iFeel System [5] and are described in [6]. Then, we evaluate them running each method with an input of 10, 100, 1,000, and 10,000 tweets in English according to the key performance metrics, such as memory usage, battery consumption and run time. To measure these metrics we developed an application that customized the OSMonitor Application, a popular performance monitor available on Google Play.

Our main findings show that it might be hard to use NRCHashtag, OpinionLexicon, USent, Sasa, Stanford on current mobile devices. Another observation is that the lexical methods obtained good results in terms of memory, CPU time, and battery usage.

II. Results

A. Experimental Setup

To measure the performance metrics, we adapted the Android resources Monitor, OSMonitor [7]. All the resources used by the OSMonitor during the experiment are not accounted in our results.

Each experiment consists of running each method with an input of 10, 100, 1,000, and 10,000 tweets in English, extracted from a known dataset [8]. For each method, we ran 31 experiments to report average values with respective confidence intervals of 95% confidence level. Our evaluation consist on evaluating sentiment analysis in two specific moments, during load and during execution. We have used 5 LG G3 smartphone devices to run our experiments.

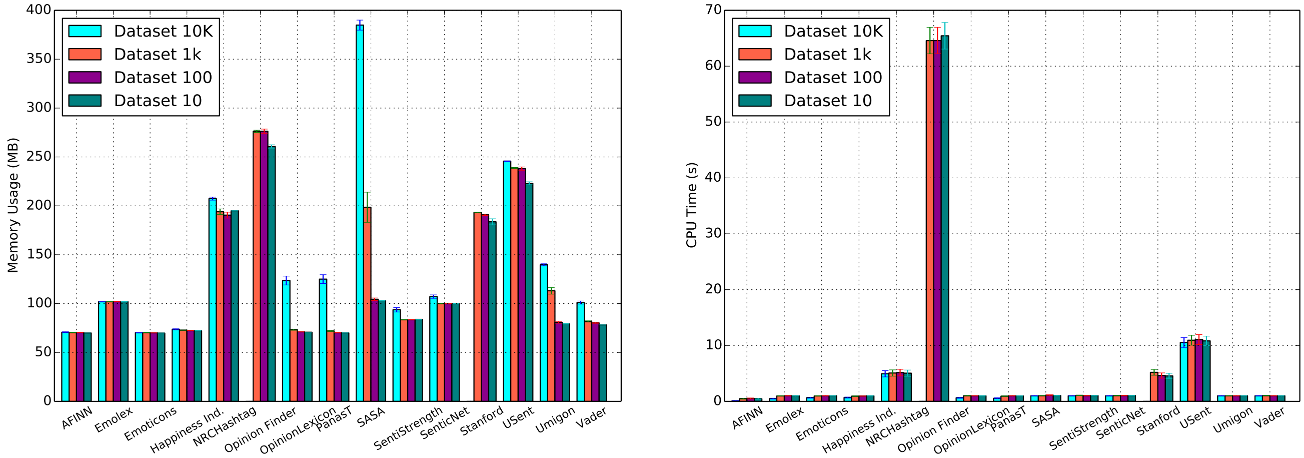
B. Performance Evaluation

We present the performance evaluation in 3 scenarios: (i) battery evaluation; (ii) memory evaluation; and (iii) CPU evaluation.

During the experimentation process, the methods Sentiment140 Lexicon and SentiWordNet were not able to run on Android. They had an issue on the load dictionary step. It may be overcome by using an Android SQLite implementation for their dictionaries. The methods OpinionFinder and Stanford Recursive Deep Model, for 10k dataset only, they were executed until the battery runs out.

In scenario (i), battery evaluation, the best method for mobile devices would be which consumes less battery. In this analysis, the methods USent and Stanford did not get a good performance. USent spent almost 6% of battery while Stanford spent almost 11.5% for the instances 10k and 1k, respectively.

The next scenario (ii), memory evaluation, some of the sentiment analysis methods required a huge amount of memory RAM. The methods NRCHashtag, USent, and Stanford use almost 100 MB of memory RAM while OpinionFinder uses almost 220 MB during the load process. Further, for the execution process, see Figure 1 (a), the methods NRCHashtag (207 MB), OpinionFinder (276 MB), USent (245 MB), Sasa



(a) Memory Consumption on Execution

(b) CPU Load Time

Fig. 1: Run Time Performance of the CPU and consumption of the Memory Evaluation during the load and execution of 17 sentiment Analysis Methods. Results are average values of 31 executions with confidence intervals of 95% confidence level.

(384 MB), and Stanford (193 MB) also use a lot of memory RAM.

The last scenario, CPU evaluation is correlated to the first scenario (Battery consumption) because the battery life decreases due to a long CPU consumption. The method OpinionFinder needs almost 64 seconds to load the method. Also, USent needs almost 10 seconds. The Figure 1 (b) shows the time, also in seconds, necessary to execute the sentiment analysis methods for all the instances. During the execution, we can note that Stanford consumes almost 1,255 seconds to run the 1k instance. The USent run a 10k in around 713 seconds.

Our results show that some evaluated methods consume many mobile resources as battery, memory, and also CPU time. Therefore, methods such as NRCHashtag, OpinionFinder, USent, Sasa, Stanford are not recommended for mobile devices. These methods use machine learning techniques, only exception for NRCHashtag that is a lexical method. In general, the machine learning methods got the worst results. The lexical methods had a good performance once they are based on dictionaries to compute the sentiment polarity scores. The methods Sentiment140 Lexicon and SentiWordNet were not able to run on Android.

III. Concluding Discussion

This work presents the first of a kind analysis about sentiment analysis in mobile devices, providing a better understand of the performance of 17 existent and popular sentence-level sentiment analysis methods.

Considering all methods in this study that were able to finish the experiment, we show that it might be hard to use NRCHashtag, OpinionLexicon, USent, Sasa, Stanford on current mobile devices. Another observation is that the lexical methods obtained good results in terms of memory, CPU time, and battery usage.

As a final contribution, we release the Android API that implements all the 17 sentiment analysis. The API is available at http://www.ifeel.dcc.ufmg.br/.

Acknowledgment

This work was partially supported by the project FAPEMIG-PRONEX-MASWeb, Models, Algorithms and Systems for the Web, process number APQ-01400-14, and by individual grants from CNPq, CAPES, and Fapemig.

References

[1] M. Burke, C. Marlow, and T. Lento, "Social network activity and social well-being," in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2010.

[2] R. LiKamWa, Y. Liu, N. D. Lane, and L. Zhong, "Can your smartphone infer your mood?" *ACM Workshop on Sensing Applications on Mobile Phones (PhoneSense)*, 2011.

[3] M. De Choudhury, S. Counts, E. J. Horvitz, and A. Hoff, "Characterizing and predicting postpartum depression from shared facebook data," in *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, 2014.

[4] J. Reis, F. Benevenuto, P. Vaz de Melo, R. Prates, H. Kwak, and J. An, "Breaking the news: First impressions matter on online news," in *Proceedings of the 9th International AAAI Conference on Web-Blogs and Social Media*, 2015.

[5] M. Araújo, J. P. Diniz, L. Bastos, E. Soares, M. Júnior, M. Ferreira, F. Ribeiro, and F. Benevenuto, "ifeel 2.0: A multilingual benchmarking system for sentence-level sentiment analysis," in *Proceedings of the International AAAI Conference on Web-Blogs and Social Media*, Cologne, Germany, May 2016.

[6] F. N. Ribeiro, M. Araújo, P. Gonçalves, M. A. Gonçalves, and F. Benevenuto, "Sentibench - a benchmark comparison of state-of-the-practice sentiment analysis methods," *EPJ Data Science*, 2016.

[7] "Osmonitor for android," www.osmonitor.mobi, accessed September 19, 2015.

[8] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring user influence in twitter: The million follower fallacy," in *International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2010.