Tensorflow vs R: A Comparative Study of Usability

Luís Ricardo Araujo Dias, Rosalvo Ferreira de Oliveira Neto
Universidade Federal do Vale do São Francisco – UNIVASF
Avenida Antonio Carlos Magalhães, 510, Country Club
CEP: 48902-300 – Juazeiro-BA – Brazil
+55 (74) 2102-7636
luis.ric@hotmail.com, rosalvo.oliveira@univasf.edu.br

ABSTRACT
Google released on November of 2015 Tensorflow, an open source machine learning framework that can be used to implement Deep Neural Network algorithms, a class of algorithms that shows great potential in solving complex problems. Considering the importance of usability in software success, this research aims to perform a usability analysis on Tensorflow and to compare it with another widely used framework, R. The evaluation was performed through usability tests with university students. The study led do indications that Tensorflow usability is equal or better than the usability of traditional frameworks used by the scientific community.

CCS Concepts
• Information systems→ Information systems applications → Data mining • Software and its engineering → Software creation and management → Software verification and validation → Empirical software validation.

Keywords
Tensorflow; Usability; Installability.

1. INTRODUCTION
A framework can be described as a collection of classes, interfaces and standards composing an abstract design aimed to solve some problem family in a flexible and extensible way [4].

Google released on November of 2015 Tensorflow, an open source deep learning framework [15] which can express and implement machine learning algorithms, so that it can be used for both research and deployment on commercial products [1].

Tensorflow has presented some advantages over another machine learning frameworks, such as better performance in complex tasks [8], and better compilation time [15]. However, in order to be successful, many factors must be considered, among then, there is usability [10]. Usability helps increasing user productivity, decreasing users' error rate, and increasing user satisfaction. Therefore, it is a decisive criterion when users have to decide about buying or using a software [16].

Considering usability importance on a software success, the potential shown by Tensorflow, and the importance of machine learning algorithms in information systems, this article aims to analyze Tensorflow usability and to compare it with R usability, since it is the most used framework on machine learning [7]. In order to achieve those goals, usability tests were performed with university students.

The tests aimed to analyze two aspects of usability: ease of install, which characterizes the effort required to install a software and can impact on the software portability and operability [6]; and the availability and quality of documentation, either official documentation or tutorials and guides available on the internet, about the frameworks.

The remaining of this paper is organized as follows. Section 2 presents the literature review. Section 3 shows the experimental methodology. Section 4 presents the experimental results and their interpretation. Finally, Section 5 concludes this paper and proposes future works.

2. LITERATURE REVIEW

2.1 Machine learning
Machine learning is a field of study of artificial intelligence focused on studying computer programs that can improve performance through experience [15]. Machine learning algorithms are widely used in modern society, being present, for instance, on image and natural language processing, data mining and information systems applications [8].

Traditional machine learning algorithms face some limitations when processing data with high dimensionality, making it necessary to pre-process the input before it can be used [2]. Therefore, when we face this sort of datasets, the result is highly dependent on human effort, and many times it's not even viable to process them [8]. On the other hand, a class of algorithm known as deep learning algorithms have shown advances in many complex areas, especially when associated with recent hardware advancement and parallel processing [15].

Usually, deep learning algorithms are composed by multiple computational layers, where each layer process input data using a simple, non-linear function, transforming the input into a more abstract representation of the data, what allow deep learning algorithm to provide high quality results even when raw data is used as input [8]. Normally, the output of each layer is used as input data into the following layer, creating a deep architecture where the data possess a more abstract representation with each successive layer [15].

For instance, if we consider the deep neural network shown on Figure 1, the input may be presented as a vector of pixels. The first layer may identify lines, curves, and other primitives. The second layer, in turn, detect patterns on the primitives to identify simple objects, such as eyes, noses, and ears. Finally, the third layer may look for patters on simple objects to identify more complex images, such as faces or facial expressions.

Source: http://www.rsipvision.com/exploring-deep-learning/

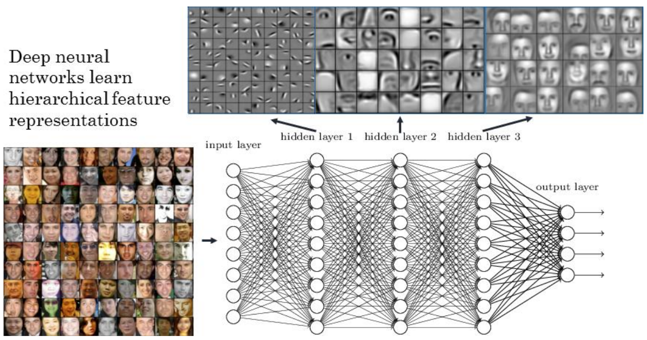

Figure 1. Deep Neural Network Example

2.2 Tensorflow

In this context, Google released Tensorflow, an open source deep learning framework suitable for research and deployment on commercial products [1].

Besides of being suitable for deep learning, Tensorflow has support for GPU processing and has shown good performance while solving complex tasks, as well as some advantages over other machine learning frameworks available on the market [8].

However, in order to be successful, many factors must be considered, among then, there is usability [10]. Usability helps increasing users' productivity and decreasing users' error rate, in addition to contributing to increase user satisfaction. Therefore, it is a decisive criterion when users have to decide about buying or using a software [16].

2.3 Usability

There are many definitions for usability in the literature, Winkler and Pimenta [16] define usability as a technical term to describe the quality of use of an interface. Additionally, Nielsen [12] defines usability as a qualitative attribute related to how easy an interface is to use, and says it is related with all aspects of a system a human may interacts, including installation and maintenance procedures

Usability can be decomposed in five quality components [14]: 1)learnability, how easily a user can perform basic tasks when he or she starts using the software; 2)efficiency, how quickly a task can be performed by the user; 3)memorability, how easily can a user regain proficiency after a period away from the design; 4)error, including error rate, severity and recoverability; and 5)user satisfaction with the design.

This study focus on a single usability component, learnability. More specifically, on how easy it's for a new user to learn how to install and configure the framework, before getting started with coding.

There are many methods available to access a software usability. They can be divided into four main classes: formal, empirical, informal and automatic. However, formal and automatic methods are costly and thus they are rarely used [13]. So, the most common methods for evaluating usability are empirical and informal.

Empirical tests rely on users to evaluate the software. Usually the users perform tasks involving the system being evaluated, while the evaluators try to identify problems by watching the execution, identifying mistakes, timing execution, interviewing the user, and so on [11].

Informal tests, also known as inspection methods, on the other hand, rely on the evaluator own experience and it's usually based on a set of guidelines, checklists, or heuristics [9]. With those methods, the evaluator executes a task and checks the suitability of the software according to the baselines defined. However, these methods are sensitive to the evaluator's expertise [10].

Despite the variety of methods available, empirical tests are the most common, and usually perform better than other methods under the same circumstances [13], although inspections methods are used when there aren't enough users available for testing.

3. MATERIALS AND METHODS

The research question was: "Is there a usability difference between TensorFlow and R?". Empirical usability tests were performed in order to answer this question. The independent variables were: a) eight teams, b) one member per team, c) one functional requirement and d) a meeting for general explanation about the functional requirement.

3.1 Subjects

Teams were composed by students from the discipline called Computational Intelligence from the Computer Engineering course at the Universidade Federal do Vale do São Francisco (UNIVASF). This discipline is offered in the 8th semester and has as prerequisite one discipline: Artificial Intelligence (60 h). The entire population of the study is small, so there was no sample. All 8 students from the class participated in the study. Six subjects were male and two female.

3.2 Apparatus

The experiments were conducted on a laboratory at Univasf, on desktop computers, all with 8Gb memory, intel core i7 CPU, and Windows 7 operating system.

To avoid possible problems caused by programs incompatibility or already installed dependencies, the experiments were conducted on virtual machines with 4Gb memory, 4 CPU cores, and Ubuntu 16.04 operating system.

3.3 Functional requirement

Two frameworks, namely Tensorflow and R, were used to solve a digit identification problem. In order to do it, the subjects were asked to install and configure the framework and to use a given script to solve the problem.

The subjects had access to the internet because the study intends to evaluate the learning process a user usually faces when installing a new framework. However, they were not allowed to communicate with the researcher or other subjects during the experiment, so they had to rely only on the documentation available on the internet to complete the tasks.

3.4 Experimental design

A within-subject experiment design [9], where all subjects performed both tasks, was used. In order to prevent carryover effects [3], subjects were divided into two groups of four people each, based on subjects' experience, so both groups presented similar experience with the technologies involved in the experiment. The first group performed the task using Tensorflow first, and R afterward, while the second group performed the tasks using the frameworks in the opposite order.

3.5 Procedure

The experiment was conducted in a single day with all subjects. Initially all subjects were asked to answer a questionnaire to evaluate their experience with the technologies involved in the experiment. The results were used to determine how the groups were divided.

Once the groups were divided, a brief explanation on the frameworks and tasks was conducted. When all subjects understood the task, they were asked to perform both tasks, according to the order designed to their groups, and, at the end of each task, to fill out a questionnaire to access their opinion about the framework used. Besides the questionnaire, short interviews were conducted at the end of the experiment, aiming to identify the difficulties faced by each subject while they performed the tasks, as well as the possible causes for those difficulties.

4. RESULTS AND DISCUSSION

Regarding the availability of documentation, R has shown better results. Subjects consulted over 8 distinct websites to help them with the tasks, while this number was only 4 for Tensorflow. This result was already expected considering how long each framework is available.

However, R high resources availability was not always beneficial. During the experiment, some participants encountered outdated guides or guides that included unnecessary steps, such as installation of not required dependencies or download server configuration. Besides, because R is available to download from different sources located worldwide, some guides used sources located far from the user location, what has a negative impact on download speed, increasing the time needed to perform the task.

Another difficulty reported by some subjects occurred during the installation of extra packages on R. The first reason causing the difficulties was the fact that the script would execute until the point where a function from the package was required before reporting the error, while on Tensorflow the error reporting would occur when trying to import the packages. The second reason was related to the installation of missing packages, since many guides consulted by the subjects assumed basic knowledge on R in order to be understood, what made some subjects to expend a fairly long time trying to find an adequate guide or trying to figure out the installation process.

Tensorflow, on the other hand, is still lacking in terms of online resources, what led 7, out of all 8 subjects, to use guides from Tensorflow official website to perform the tasks, 5 of them as the only source, and only 1 subject had to do further research after finding the official documentation. So, we have indications that, although Tensorflow still doesn't have much support from the community, the official documentation and support provided have high quality.

Figures 2 and 3 summarize the results of the questionnaires filled by the subjects about availability and quality of documentation for both target frameworks.

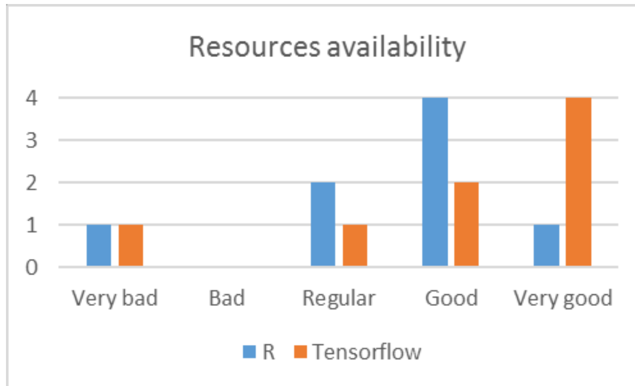

Figure 2. Users evaluation of resources availability

As we can see, Tensorflow showed better results for both resources availability and quality, with 75% and 85.5% positive evaluations, respectively, against 62.5% and 50% for the same variables on R. The two frameworks had only 12.5% negative evaluations for both resources availability and quality.

Figure 3. Users evaluation of resources quality

One can also notice that, even though more resources were available to R, Tensorflow still got better users evaluation on resources availability. That happened because many times, while searching, the subjects stumbled upon resources to solve different problems than the one he or she was looking for, what required them to consult more than one resource to get the job done.

Figure 4 summarize the result of the questionnaires filled by the subjects about the tasks difficulty.

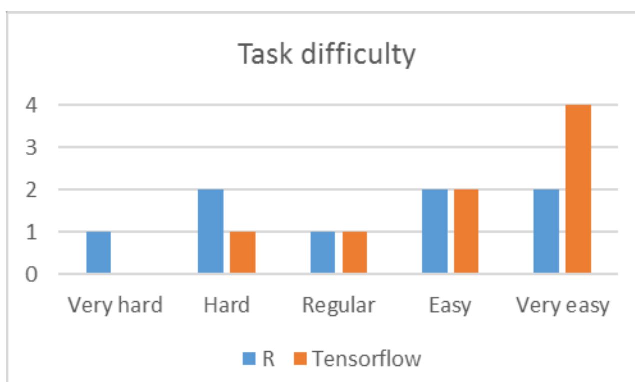

Figure 4. Users evaluation of task difficulty

When it comes to task difficulty, Tensorflows showed even better results. It was considered easy to install and use by 75% of the subjects, and hard by only 12.5%. When we consider R, the numbers were 50% and 37.5% for easy and hard, respectively.

5. CONCLUSION AND FUTURE WORK

The results shown before provide indications that the installation process are quite simple for both frameworks and that, although fairly new, Tensorflow shows good potential for use in machine learning projects.

Moreover, the results indicate that Tensorflow possess greater usability when compared to frameworks already in use by the scientific community, besides having a high quality documentation, what can make it easier for those who are starting in the field, as well as those already inside it, to develop high quality solutions.

Some plausible reasons for the better usability of Tensorflow, when compared with R, are the base language used by the first and its narrower focus.

Tensorflow has some advantages due to the fact it is written in python, such as: First, it's easier to install in many operating systems, because python is part of the default installation for most Unix distributions, what reduce the number of steps of the installation process; Second, people are more interested on learning python than R, as a matter of fact, python was one of the 5 most popular languages in 2016 [5]; Finally, python is becoming one the main languages on machine learning field, being the second most used language on machine learning projects in 2016, besides having the highest growth in the same area for the past two years [6].

Because Tensorflow has a narrower focus than R, it makes sense that Tensorflow is closer to machine learning thinking. Moreover, R embrace many others mathematical and statistical functionalities, and relies on the use of many packages to solve machine learning problems, each of them with specificities and particularities, requiring the users to learn and to install multiple packages according to the problems they have to solve.

Finally, considering the advantages of Tensorflow over R observed in the literature review and in this study, we have indications that Tensorflow has potential to become a successful machine learning framework.

A new study evaluating how easy it's for a new user to start coding with the frameworks would be a great addition to learnability comparison of the frameworks. In addition, studies comparing code generated with each framework could prove valuable, because code legibility and maintainability are important criteria for software quality.

6. REFERENCES

[1] Abadi, M. et al. 2015. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*, http://download.tensorflow.org/paper/whitepaper2015.pdf.

[2] Arel, I., Rose, D. C. and Karnowski, T. P. 2010. Deep Machine Learning – A new Frontier in Artificial Intelligence Research. *IEEE Computational Intelligence Magazine*, 5, 10 (Oct. 2010), 13-18. DOI=http://dx.doi.org/10.1109/MCI.2010.938364.

[3] Charness, G., Gneezy, U. and Kuhn, M. A. 2011. Experimental methods: Between-subject and within-subject design. *Journal of Economic Behavior & Organization* 81, 1 (Jan. 2012), 1-8. DOI=http://dx.doi.org/10.1016/j.jebo.2011.08.009.

[4] Govoni, D. 1999. *Java Application Frameworks*. John Wiley & Sons Inc..

[5] IEEE 2016. *Interactive: The Top Programming Languages 2016*, http://spectrum.ieee.org/static/interactive-the-top-programming-languages-2016.

[6] ISO/IEC 2001. *International Standard ISO/IEC 9126-1. Software engineering - Product quality - Part 1: Quality model*.

[7] KDnuggets 2016. *R, Python Duel As Top Analytics, Data Science software*, http://www.kdnuggets.com/2016/06/r-python-top-analytics-data-mining-data-science-software.html.

[8] Lecun, Y., Bengio, Y. and Hinton, G. 2015. Deep Learning. *Nature*, 521 (May. 2015), 436-444. DOI=http://dx.doi.org/10.1038/nature14539.

[9] Lin, H. X., Choong, Y. and Salvendy, G. A. 1997. Proposed index of usability: A method for comparing the relative usability of different software systems. *Journal of Economic Behavior & Organization*, 16, 4-5, 267-277. DOI=http://dx.doi.org/10.1080/014492997119833.

[10] Mayhew, D. J. 1999. The Usability Engineering Lifecycle. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (The Pittsburgh, The Pennsylvania, May 15 - 20, 1999). CHI '99. ACM, New York, NY, 147-148. DOI=http://doi.org/10.1145/632716.632805.

[11] Nickerson, R. and Landauer, T., 1997. Human-computer interaction: Background and issues. In *Handbook of Human-Computer Interaction*, 2nd edition. M. G. Helander, T. K. Landauer, and P. V. Prabhu, Eds. Elsevier Science. Elsevier Science, Amsterdam, NL, 3-31.

[12] Nielsen, J. 1993. *Usability Engineering*. Academic Press Limited, 23-70.

[13] Nielsen, J. 1994. Usability Inspection Methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (The Boston, The Massachusetts, April 24 - 28, 1994). CHI '94. ACM, New York, NY, 413-414. DOI=http://doi.org/10.1145/259963.260531.

[14] Nielsen, J. 2012. Usability 101: Introduction to Usability. https://www.nngroup.com/articles/usability-101-introduction-to-usability/.

[15] Rampasek, L. and Goldenberg, A. 2016. TensorFlow: Biology's Gateway to Deep Learning?. *Cell Systems*, 1, 2 (Jan. 2016), 12-14. DOI= http://dx.doi.org/10.1016/j.cels.2016.01.009.

[16] Winckler, M. and Pimenta, M. S. 2002. Avaliação de usabilidade de sites web. In *Proceedings of Escola de Informática da SBC Sul* (The Porto Alegre, The Rio Grande do Sul, May 13 - 17, 2002). ERI-2002. SBC, Caxias do Sul, RS, 87-13