Technical Report

Which programming language should a company use?  
A Twitter-based analysis

Version 1.0

CRIM-14/10-23-MODL

Giuseppe Destefanis, Ph.D.  
Postdoctoral Researcher  
Advanced Software Modeling and Development  

October 23, 2014
# TABLE OF CONTENTS

1. INTRODUCTION ........................................................................................................ 3

2. METHODOLOGY ........................................................................................................ 4

3. RESULTS AND DISCUSSION .................................................................................. 6

4. THREATS TO VALIDITY ...................................................................................... 12

5. CONCLUSION ......................................................................................................... 12

6. ACKNOWLEDGEMENT ......................................................................................... 13

7. REFERENCES ........................................................................................................... 13

LIST OF TABLES ........................................................................................................ 15

LIST OF FIGURES .................................................................................................... 16
1. INTRODUCTION

The choice of which new programming language to learn is a question that many developers encounter during their career. While interesting, it is a difficult question to answer and there is no definitive answer to it. Many characteristics have to be taken into account when defining a good programming language. In one sense, the quality of a programming language is largely determined by the market demand for developers. We can reposition the original question to “What is the programming language most required by the job market?”. With the large diffusion of Internet, and widespread use of smartphones, tablets and laptops, digital media today is one of the most important recruitment channel used by companies for hiring new staff. In the last two years, social networks have also been used as a recruitment option. Twitter is a social networking service which enables its users to exchange text-based messages of up to 140 characters known as “tweets”. Useful information for tracking or even forecasting behaviour when extracted in an appropriate manner lies embedded in the Twitter stream; Twitter currently has 200 million active registered users and rising as of March 2013.

This white paper aims to answer the following questions:

- How popular are various programming languages related to research for software developers on Twitter?
- Could it be possible to make predictive analysis using data collected on Twitter?

Several studies have analyzed diffusion [1], trends and popularity of programming languages. Some studies use the definitions of particular indexes that try to measure the popularity of programming languages, but to our knowledge no studies have tried to analyze the popularity of programming languages using social networks. The TIOBE programming community index [6] is a measure of the popularity of programming languages, calculated from a number of search engine results for queries containing the name of the language. The index covers searches in Google, Google Blogs, MSN, Yahoo!, Baidu, Wikipedia and YouTube. The index is updated once a month. TIOBE focuses on Turing complete languages, so does not provide information for instance about the popularity of SQL or HTML. According to the site, the TIOBE index is not about the best programming language or the language in which most lines of code have been written. However the site does claim that the number of web pages may reflect the number of skilled engineers, courses and jobs worldwide.

The PyPL [7], Popularity of Programming Language Index is created by analyzing how often language tutorials are searched on Google: the more a specific language tutorial is searched, the more popular the language is assumed to be. It is a leading indicator. The raw data comes from Google Trends, so that anyone can verify it, or make the analysis for his own country or other languages. Bissyande et al. [4] studied one hundred thousand open source software projects from GitHub to answer various research questions on the popularity, interoperability and impact of various languages measured in different ways (e.g., in terms of lines of code, development teams, issues, etc.). The study showed that earlier popular languages, such as C, are still used extensively, while the popularity of web development has made JavaScript and Ruby pervasive. Finally, Objective-C, a language tightly related to the products of a successful vendor, namely Apple, has also been gaining much attention.
Delorey et al. [11] used survey methodology to identify the factors that lead to language adoption. They analyzed large datasets, including over 200,000 SourceForge projects and multiple surveys of 1,000-13,000 programmers. Using this data, they established baseline quantitative models of the language adoption process. The authors report several findings: language adoption does not have a long tail; a small number of languages account for most language use; intrinsic features have only secondary importance in adoption. Open source library availability is the most influential factor in selecting a language for a project. Social extrinsic factors such as existing team experience also rate highly. Languages features such as performance and semantics do not. Developers will steadily learn and forget languages, but only of a limited variety. Developers select more varied languages if their education exposed them to different language families. When considering intrinsic aspects of languages, developers prioritize expressivity over correctness.

Chen et al. [9] reviewed the history of a group of programming languages and showed that two kinds of factors, intrinsic factors and extrinsic factors, could affect the evolution of a programming language. Intrinsic factors are the factors that can be used to describe the general design criteria of programming languages. Extrinsic factors are the factors that are not directly related to the general attributes of programming languages, but still can affect their evolution. By collecting historical data, a data warehouse has been established, which stores the value of each factor for every programming language. The programming language trends are described and evaluated by using these data. Empirical research attempts to capture observed behaviours by empirical laws, statistical methods are used to describe historical programming language trends and predict the evolution of the future trends. Multivariate multiple regression method has been used to construct the statistics models for programming language trends. After statistics models are constructed to describe the historical programming language trends, they are extended to do tentative prediction for future trends. The models are validated by comparing the predictive data and the actual data.

Delorey et al. [11] analyzed data collected from the CVS repositories of 9,997 open source projects hosted on SourceForge in an effort to understand trends in programming language usage in the open source community between 2000 and 2005. The trends considered include: the relative popularity of the ten most popular programming languages over time, the use of multiple programming languages by individual programmers and by individual projects, and the programming languages most often used in combination. Karus et al. [10] investigated the revision data of 22 OSS projects and tracked the evolution of multiple programming language usage. The study confirmed that the most popular language in OSS software projects is Java followed by C. Over the last decade, C has lost its high share to various other languages of which Java has been among the more popular ones. Despite becoming popular in just a few years, Java has not been able to grow its share significantly during the last years.

2. METHODOLOGY

In order to analyze information related to the popularity of programming languages, tweets regarding job recruiting activity, (focusing on software engineering jobs) were collected from Twitter between January 2011 and October 2013 using Twitter4J, an unofficial Java library for the Twitter application programmers interface (API) from
We developed a Java-based Twitter timeline collector. We have analyzed all collected tweets in order to delete those not connected to job recruitment. We chose precise and indicative keywords for collecting tweets using the following pattern:

```
name_of_programing_language+developer+job
```

and we used the same pattern for all the programming languages considered in our study. After we collected all tweets related to job recruitment, we performed an analysis using regular expressions in order to remove tweet as the following:

We performed the analysis considering the following programming languages: Java, JavaScript, C-family, Objective-C, Php, Python and we also checked the request for Flash developers to demonstrate that it is also possible to highlight decreasing trends on Twitter. Considering the research of C, C++, C\# developers, it was difficult to divide tweets related to request for C developers and C++ developers because tweets like the following often occur:

Considering the high numbers of tweets similar to those showed in the above examples, we decided, in order to simplify the study without affecting the generality of the discussion, to consider a single set of tweets that contains C, C++, C\# job requests and this we called the C-family set.
Table 1 shows the total number of tweets collected for each singular programming language considered. Java dominates the list as the most popular language tweeted about. This is an expected result due to network effect and network externality.

<table>
<thead>
<tr>
<th>Language</th>
<th># of collected tweets</th>
<th>Period</th>
</tr>
</thead>
</table>

Table 1. Collected/selected job-related tweets per language

A network exists when a product’s value to the user increases as the number of users of the product grows.

3. RESULTS AND DISCUSSION

In order to analyze tweets collected in the period from January 2011 to October 2013, we performed a statistical analysis using the R tool based on time series analysis. A seasonal time series consists of a trend component, a seasonal component and an irregular component. Decomposing the time series means separating the time series into these three components.

To estimate the trend component and seasonal component of a seasonal time series that can be described using an additive model, we used the decompose() function in R. This function estimates the trend, the seasonality, and the irregular components of a time series that can be described using an additive model. The time series of the number of tweets per month related to a recruitment research staff with knowledge of particular programming language could be identified as a seasonal series.

By analyzing figures 1 to 6, it is possible to evaluate different trends for each programming language and this fact helps us to answer the first research question. The request for JavaScript, Objective-C, and Python developers is increasing. A small decreasing trend appears for Java developers, but the demand (the observed curve) is quite stable and Java continues to be the most required programming language (Java is the most required programming language on Twitter, but with decreasing request for Java developers in the last nine months considered in this analysis - figure1).
Which programming language should a company use? A Twitter-based analysis.

Figure 2: Java time series

Figure 3: C-family time series

Figure 4: Objective-C time series

Figure 5: Php time series

Figure 6: Python time series
Also requests for PHP developers (figure 5) follow a decreasing trend while the request for Python developers is increasing (figure 6). The situation related to C family (figure 3) is disconnected from the others, it is notable that a decreasing trend exists, but considering the correlation study between all the requests for programming languages developers, it is possible to see that there are no significant correlations between C-family and the others programming languages considered. This could be attributed to the fact that C and C++ are used for critical software applications, financial applications and embedded systems, and also to the difficulties found during the first phase of this study where tweets related to C-family were collected. C# is widely used as programming language for web applications, and for future studies it would be better to try to consider this programming language alone. The correlation matrix between the different numbers of tweets related to programming languages analyzed is shown in the Table 2.

Table 2 shows a strong correlation between requests for Python developers and JavaScript developers, the same thing happens for Objective-C and JavaScript, and also a good relation exists between PHP and Java and PHP and JavaScript. These are expected and well-known results because PHP, Java and JavaScript are related to development of mobile applications and web applications due to popularity reached by smartphones, tablets, and mobile devices always connected to Internet.

<table>
<thead>
<tr>
<th></th>
<th>Java</th>
<th>JavaScript</th>
<th>C-Family</th>
<th>Objective-C</th>
<th>PHP</th>
<th>Python</th>
<th>Flash</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java</td>
<td>1</td>
<td>0.3</td>
<td>0.2</td>
<td>0.1</td>
<td>0.5</td>
<td>0.3</td>
<td>-0.3</td>
</tr>
<tr>
<td>JavaScript</td>
<td>0.3</td>
<td>1</td>
<td>-0.1</td>
<td>0.7</td>
<td>0.5</td>
<td>0.9</td>
<td>-0.7</td>
</tr>
<tr>
<td>C-family</td>
<td>0.2</td>
<td>-0.1</td>
<td>1</td>
<td>-0.1</td>
<td>0.3</td>
<td>-0.1</td>
<td>0</td>
</tr>
<tr>
<td>Objective-C</td>
<td>0.1</td>
<td>0.7</td>
<td>-0.1</td>
<td>1</td>
<td>0.2</td>
<td>0.7</td>
<td>-0.6</td>
</tr>
<tr>
<td>PHP</td>
<td>0.5</td>
<td>0.5</td>
<td>0.3</td>
<td>0.2</td>
<td>1</td>
<td>0.4</td>
<td>-0.3</td>
</tr>
<tr>
<td>Python</td>
<td>0.3</td>
<td>0.9</td>
<td>-0.1</td>
<td>0.7</td>
<td>0.4</td>
<td>1</td>
<td>-0.7</td>
</tr>
<tr>
<td>Flash</td>
<td>-0.3</td>
<td>-0.7</td>
<td>0</td>
<td>-0.6</td>
<td>-0.3</td>
<td>-0.7</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2. Correlation Matrix

In order to better evaluate the data series generated by Twitter, we also analyzed the number of tweets related to request for Flash developers (fig. 7). Apple devices products like iPad and iPhone do not support Flash technologies and this is one of the factors that have led to a reduction in request for Flash developers.
The trend component indicates a strong decrease, and the correlation between the number of requests for Flash developers and the number of requests for JavaScript developers is -0.7: we observe a growth in requests for JavaScript developers and a decrease in requests for Flash developers, and this fact is also shown between Flash and Objective-C; the growth of Objective-C, the decrease of Flash.

With a time series that can be described using an additive model with increasing or decreasing trend and seasonality, it is possible to use Holt-Winters exponential smoothing to make short-term forecasts [16] [17] [18]. Holt-Winters exponential smoothing estimates the level, slope and seasonal component at the current time point. Smoothing is controlled by three parameters: alpha, beta, and gamma, for the estimates of the level, slope of the trend component, and the seasonal component, respectively, at the current time point. The alpha, beta and gamma parameters all have values between 0 and 1, and values that are close to 0 mean that relatively little weight is placed on the most recent observations when making forecasts of future values. Using Holt Winters exponential smoothing, considering a forecast for the next 5 months (from December 2013 to April 2014), it is possible to provide an explanation for the third question.
Technical Report
Which programming language should a company use? A Twitter-based analysis.

Figure 8: Java request forecast

Figure 9: JavaScript request forecast

Figure 10: C-family request forecast

Figure 11: Objective-C request forecast

Figure 12: Php request forecast

Figure 13: Python request forecast
Analyzing figures 8 to 13, it is possible to see that the demand for Java developers is expected to be stable for the next 5 months, while a growth is predicted for JavaScript, Objective C, Php, Python and a short decrease for C-family.

The website www.tiobe.com reports the long term trends for the top 10 programming languages from 2010 to November 2013. Exploring this website, it is possible to check differences between the results produced by our Twitter-based analysis and the popularity of programming languages defined by TIOBE index. Making a comparison between our results and popularity indicated by TIOBE index we can say that: Java is the most popular programming language and also the Twitter-based analysis confirms this claim. The trend is decreasing also for TIOBE index. JavaScript is doing well also for TIOBE index. That might come as no surprise. JavaScript is the lingua franca of websites nowadays and our Twitter-based analysis confirms this fact. Objective-C is gaining popularity and also this fact is confirmed by our analysis results. Php is decreasing also according to the TIOBE index as confirmed by our results. Python is increasing according to our results and analyzing the tables present on TIOBE website, it is possible to confirm that there is a positive trend. Regarding the C-family, it is not possible to compare our data with TIOBE index related to C, C++ and C#, because in TIOBE index they are considered separately.

To check the validity of our forecast analysis, we have made a comparison among our Holt-Winters results and the TIOBE-index value from December 2013 to April 2014. Figure 15 shows the programming languages’ popularity from TIOBE from December 2013 to April 2014 of Java, Objective C, Php, Python and JavaScript.

Regarding the Java programming language, our forecast suggests a stable request of developers; the TIOBE index confirms our result. The Java popularity has been stable during these 5 months.

![Figure 14: TIOBE popularity](image-url)
The Objective-C popularity has increased from December 2013 to April 2014, and also in this case our results are aligned.

Regarding the Php popularity, our forecast highlights a small growth while the TIOBE graph shows a decrease.

Python popularity has been stable according to the TIOBE graph, while our forecast presents a growth.

JavaScript popularity is slightly growing according to the TIOBE graph, and also our results are aligned.

4. THREATS TO VALIDITY

This study bears some threats to validity mainly related to the origin of the dataset: the dataset is referred to a period of 34 months that starts from January 2011 to October 2013. The considered period is not very long; this could affect the forecast results and also the decomposition of additive time series. The other fact is due to the continuous increase of the number of Twitter users; this could affect the random component of the time series and could be a bias for the collected data. Size of datasets: we considered only English tweets, this may affect the time series and does not allow us to compute tweets related to research for programming languages developers generated in other languages (it is also true that English is the most popular language in the IT field). C-family set: we considered tweets related to C#, C++ and C as a unique set. Language of tweets: we have considered only tweets in English; this could affect the total number of tweets related to programming languages.

5. CONCLUSION

In this white paper, we described the findings of our empirical study analyzing a dataset of 415,976 tweets. These tweets are related to announcements which request developer experts in a particular programming language. We considered the following programming languages: Java, JavaScript, C-family, Objective-C, Php, Python and Flash. The main goal of this study was to find the most popular programming language on Twitter, a popular social network, analyzing specific tweets. Different assumptions made in the literature on the popularity of programming languages exist, but none of those considers social network analysis.

Our study shows that earlier popular languages, like C-family, are still strongly required by software companies; we confirm that Java is the most popular programming language, but JavaScript, Objective-C, and Python have also been gaining much attention. With this white paper, we have demonstrated that it is possible to obtain accurate results related to the programming languages popularity using Twitter. Further work needs to be done to establish whether it is possible to considerer a social networks factor in programming languages popularity index. Further research might explore in-depth Twitter georeferentiation to understand in which part of the world particular developers are required and a future study investigating tweet written in different languages (for example in Chinese language) would provide a more detailed analysis.
6. **ACKNOWLEDGEMENT**

The authors would like to thank Pierre André Ménard for his suggestions and comments.

7. **REFERENCES**


LIST OF TABLES

Table 1. Collected/selected job-related tweets per language .................................................. 6
Table 2. Correlation Matrix ....................................................................................................... 8
LIST OF FIGURES

Figure 1. JavaScript time series................................................................................ 7
Figure 2. Java time series........................................................................................ 7
Figure 3. C-family time series.................................................................................. 9
Figure 4. Objective-C time series............................................................................ 9
Figure 5. Php time series....................................................................................... 9
Figure 6. Python time series.................................................................................... 9
Figure 7. Flash time series...................................................................................... 9
Figure 8. Java request forecast ............................................................................... 12
Figure 9. JavaScript request forecast ................................................................. 12
Figure 10. C-family request forecast ...................................................................... 12
Figure 11. Objective-C request forecast ................................................................. 12
Figure 12. Php request forecast ............................................................................ 12
Figure 13. Python request forecast ........................................................................ 12
Figure 14. TIOBE popularity.................................................................................. 13