Programming Languages Contrasted and Compared

Emmanuel Ameh

Table of Contents

Introduction to Programming Languages and the Assignment………………………………….3

Recommendations for Calculator Program……………………………………………………...4

Recommendations for Chatbot Application………………………………….………………….5

Employee Analytics Software………………………………………………………………….6

Recommendations for Employee Churn/Retention Determination System…………………….8

Recommendations for Recommender Systems………………………………………………….10

References……………………………………………………………………………………….11

Programming Languages Contrasted and Compared

Software companies may be called upon to design, provide, and install software products for business clients in the office environment. In this scenario, my software company has been tasked to create a calculator capable of running in many different environments, keeping in mind low developmental costs. In addition, a corporate-communications application, similar to a Chatbot application to operate on computers running the Windows Operating System has been requested. Three other applications and software were considered useful for this multinational client.

The software applications considered are, namely Recommender Systems, Employee Churn/Retention Determination, and Employee Analytics software. To compare and contrast are the following programming languages; C/C++, Python, and Java and how suitable they are to the development of the software stated above. Important considerations will be given to finding the most time and resource-efficient as well as technically effective solutions. Other considerations will include how far into the future the implemented software technology can survive in addition to how secure the software is against hacking both internal and external.

**Programming Calculator Functionality**

Calculators are used widely in almost all aspects of life, and sometimes a simple one comes handy when lots of numbers and or lots of mathematical operations are involved. It reduces mathematical errors and saves the brain some effort which can be channeled into other decision-making efforts which cannot be done using a calculator. Calculators have assumed all shapes and forms to include solar powered, scientific calculators, battery-powered, and phone calculators. A simple calculator can be made using a C++ program which is able to add, subtract, multiply and divide, two operands entered by the user, and other scientific and logarithmic operations (Kozacik, Chao, Paolini, Bonnett, & Kelmelis, 2017). The break and switch statements are used creating calculators and C++ as a programming language adds speed to the calculations. Complex calculations which might take half a second with other languages may take microseconds to compute using C++ (Kozacik et al., 2017). The reason I would opt for C++ to build such a calculator is due to the speed and also because there is no need for web functionality. If web functionality were to be built into the design, then I would have opted for Java and JavaScript. If the application was for data intensity, then I would have opted for the Python programming language.

**Chatbot Development**

In today’s fast-paced world, about a half of every business organization’s customers would prefer messaging communication to email communication. It is therefore important for business to adapt to new communications technologies such as Chatbots in order to improve on customer retention rates (Fontseca & Lobo, 2017). Chatbots are ideal for communication in the office environment. Selecting the appropriate chat bot is important for business. Chat-botting is a more personalized and intimate form of communicating with customers than formal emails and highly effective tools for organizations wanting to up-scale their marketing and customer services.

**Python for Chatbots**

The programming language in which that chatbot is built is just as important as the language the chatbot understands. Due to the programming language’s versatility, Python would be my first choice for coding the chatbot (Raj, 2019). With its consistent language and syntax, it is also ideal for new programmers entering the field of programming. Python also has expanded into scientific computing whilst encouraging open-source libraries paving the way for Research and Development (R&D) benefits.

Python also has its fair share of issues. For instance, it’s documentation is not as great as in C++ or Java. The language also lacks in simple and useful examples. When constructing a Chatbot, clarity is very important which is not so great in Python. Speed may also not be on the side of Python in comparison to C++ and Java, when it comes to Chatbot building. However, the end-user experience is not compromised which is more important than selecting a faster but less limited programming language such as C++ for chatbot-building. A few-more-millimeters functionality should not be made to compromise robustness of scope, hence would not be economical to select C++.

The Python language is the bedrock of Natural Language Processing (NLP) and Machine Learning. C++ is a very fast language with lots of library support such as Torch and TensorFlow however it lacks the magnitude of resources which Python has. Java Scripts and Java both have some capabilities with reference to Machine Learning (ML). Java programmers may rely on Machine Learning packages and on a plethora of libraries for JavaScripting. This may be great when applied to smaller datasets and simpler analysis but not suitable for larger datasets which Python’s libraries can readily handle.

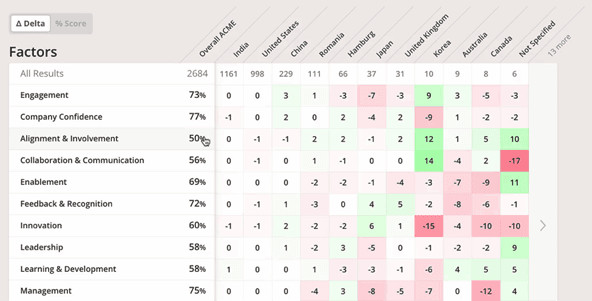
C++ programming language is a highly performance-savvy language. C++ is the go-to option for fast completion of projects and also offers higher-levels of abstraction. Python has Artificial Intelligence Markup Language (AIML) functionality making it easier for the composition of syntax for Chatbot features that are complex. Let’s have a look at the key positive aspects of Python for AI-powered Chatbot development. Quite apart from its extensive library, Python presents a highly interactive and modular code with the highest compatibility possible across several multiple platforms. Although Java is excellent for coding algorithms with advanced features, and also highly portable, it is not suitable for sentiment analysis, which is possible with Python.

For smart chat-botting, sentiment analysis must be in-built as a major feature of the chatbot. Sentiment analysis is the ability of the chatbot to determine the emotions and sentiment of the customer from the type of words the customer is using. When the language is becoming not pleasant the chatbot may its service for a live-person to take over the conversation. The Natural Language Toolkit (NLTK) within Python caters to sentiment analysis. By examining manually annotated data, bots can be trained for sentiment-recognition using NLTK. The training will consist of three different lists of positive comments, negative comments, and a mixture of comments. The sentiment analysis improves with more data infused into the training.

**Employee Analytics Software**

Many a pundit have expressed reservations on automating the office environment for reasons that workers would be replaced. Machines and automation may help alleviate redundant and time-consuming work from employees, freeing time for core business initiatives to be tackled (Marshall, Mueck, & Shockley, 2015). Smart-automation systems do well at going through large volumes of data to reveal important insights. Solutions software such as Luminoso, Gusto, and Culture Amp, assist Human Resource personnel understand the large amounts of the employees’ data for improving employee happiness’ all-year-round.

There are definitely pros and cons to using these new technologies just as in all past technologies. Just as there may be errors with automated recruitment, Artificial Intelligence powered tools may harbor errors from employee data analysis (Marshall, Mueck, & Shockley, 2015). For an instance, should the AI tool be fed with insufficient data, it may result in the bias of its insights as it concentrates on one faction of the organization. My Employee Analytics Software will be based on case studies, examples, and references in order to understand how the tool will function in the real workplace-environment.



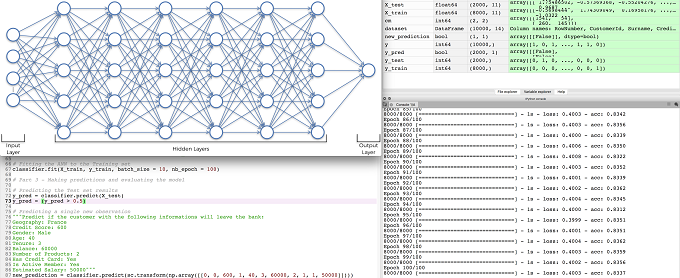
*Figure 1.* New Tools for Measuring and Improving Employee Engagement. Retrieved from www.ssir.org.

My Analytics software will make use of heatmaps to assist Human Resource personnel have a visual outlook of its data on employees. For instance, in the picture below global employee-survey-scores by tenure, age, gender, team , department and country for a multinational organization is demonstrated. This gives the organizations the tools necessary for further R&D on how to add value to the organization. . This instrument is important for continuous development, goal tracking, and performance reviews. Employees’ performance must be assessed holistically across the business organization. Reporting must be tailored to the exact specifications through the filteration of evluation-data which is based on levels, departments, groups, and such criteria beffiting for efficient and effective calibrations meetings. Skills inventory can then be managed one employee at a time. Knowing top performers’ skills will help to drive success company-wide. A good understanding of the team skills and at the organizational level will help strategic-talent-planning as well as Learning and Development (L&D) programs. This software will also help to manage and set new agile goals, track and set goal progress whist associating the goals with department objectives.

**Employee Churn Detection**

Employee churn detection or employee retention determination is a data analytics headache for most organizations since there is no art to determine the minds construction in the face. Employees vamoose from organizations for different reasons. Some of the reasons include chasing for more money, lack of cordiality, skills-misfit, and several other reasons. Given a voluminous dataset with a large sample of this multinational organizations’ employees of about 80,000, I will make use of Deep Learning and Python programming (SSIR.org, 2019), to predict the employees that most likely to vacate post within the six months. With these predictions made, the Human Resource Department and other stakeholders will perform a cost-benefit-analysis to see if the employee’s exit will in anyway be detrimental to the organization. Chances are, they will find solutions using the recommender system previously designed to resolve the employee’s concerns.

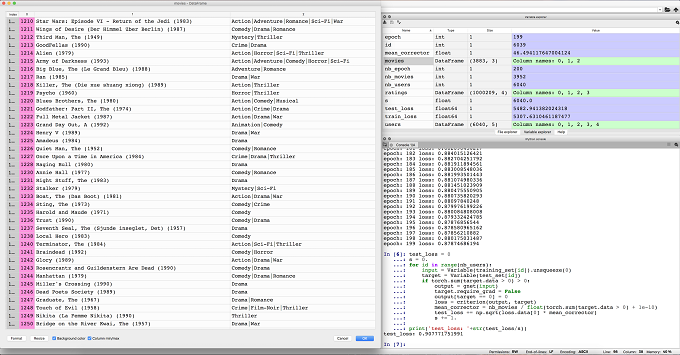
To come up with this dataset, Human Resource Department gathered-information such as employee\_id, employee\_appraisal, gender, age, tenure, employee\_\_status, employee\_skillset, employee\_conduct, etc. will be collected from the Global Employee database. During the last two years, data (including the variables) on employees who left the organization will be studied. The variables which turned to be determinants in the employee’s decision to leave the organization will now be used to design and efficient algorithm together with the remaining employees data will be fed as input data to deep neural networks for output of potential-employee-exits.

*Figure 2*. Sample Deep Neural Network showing input, hidden and output layers with Python code directly underneath. Output is on the right-hand side of image.

Because of the data intensity Python is the only suitable programming language for programming this all-important functionality (Dolatabadi & Keynia, 2017). Java and C++ cannot handle the massive data input. C++ might try to but it might take days to execute if not more than twenty hours. By succeeding in this project, I will be creating significant added-value for the organization. With the application of the Deep Learning model and Python encoding, the organization will significantly improve on employee retention.

**Recommending a Recommender System**

Taking cues from Netflix movie recommendations to Amazon product-suggestions, a good recommender system is very valuable in the office and organizational environment, not only for Human Resource personnel, but also for executive decision-making any other decisions in the organization (Dolatabadi & Keynia, 2017). Given a dataset on employees, information from employee coaches, information from peer workers, information from teams, industry information, information on competitors, information from the organizations executives etc., a recommender system should be able to use Deep Learning and Neural Networks programming in Python to recommend courses for new employees, in-service-courses for current employees, and recommend employees for promotion.



*Figure 3*. Sample Recommender System Programming with Netflix movies dataset to the left and Python programming to the right.

Recommendations from human supervisors may be tainted with bias, human element, and error and therefore a recommender system, not to totally replace human recommendations but to work side-by-side to present the most formidable recommendation in the best interest of the organization (Cheng et al., 2016) is recommendable. This recommendation system software will be akin to the recommendations given after video data is replayed at a basketball or soccer game where the referee’s decision is in disputable. With the above five implementations, this multinational company will be using state-of-the-art technology which will help it improve on its profitability.

References

Cheng, H. T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., ... & Anil, R. (2016, September). Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems* (pp. 7-10). ACM.

Dolatabadi, S. H., & Keynia, F. (2017, July). Designing of customer and employee churn prediction model based on data mining method and neural predictor. In *2017 2nd International Conference on Computer and Communication Systems (ICCCS)* (pp. 74-77). IEEE.

Fontseca, J., & Lobo, J. (2017). Choosing the best language to build your AI chatbot. Retrieved from https://techcrunch.com/2017/12/20/choosing-the-best-language-to-build-your-ai-chatbot

Kozacik, S., Chao, E., Paolini, A., Bonnett, J., & Kelmelis, E. (2017, May). Improving developer productivity with C++ embedded domain specific languages. In *Disruptive Technologies in Sensors and Sensor Systems* (Vol. 10206, p. 102060R). International Society for Optics and Photonics.

Marshall, A., Mueck, S., & Shockley, R. (2015). How leading organizations use big data and analytics to innovate. *Strategy & Leadership*, *43*(5), 32-39.

Raj, S. (2019). *Building chatbots with Python: using natural language processing and machine learning*. Apress.

SSIR.org (2019). New Tools for Measuring and Improving Employee Engagement. Retrieved from https://ssir.org/articles/entry/new\_tools\_for\_measuring\_and\_improving\_employee\_engagement