Business Analyst Implementation

Michael Wacey

For Professor Adewale Akinfaderin

SEAS 6414: Python Applications in Data Analytics

March 9, 2024

George Washington University

Business Analyst Implementation

The judicious use of AI and Machine Learning capabilities can support and augment the process of business analysis. Large organizations expend considerable resources on the process of understanding their current business, defining the future state of their business, developing a plan to get from current to future, and identifying technology to support the future state. For example, the author worked on a project for Verizon to redefine their Quote to Cash business process. This involved over 40 business analysts working for more than ten months.

AI can be used to support the conversion of inputs to reports and the validation of completeness and consistency. Business analyst consultants will still conduct interviews and workshops to collect the inputs and will verify the correctness of the results. This will result in higher quality results that can be delivered sooner.

Initially, it is anticipated that Business Analyst will be used by individual business analyst consultants. Over time, larger teams will adopt it for collaboration. Ultimately, consulting firms and larger organizations will adopt Business Analyst as a core tool.

For more details about the architecture and approach used in Business Analyst, see my companion paper submitted to SEAS 8500. This paper focuses on implementation issues.

# Language Selection

The Business Analyst platform will rely heavily on Machine Learning. In addition, it is a modern platform that will need to stand the test of time. The ability to find developers to build and maintain the platform is an important consideration.

| Language | Benefits | Concerns | Notes |
| --- | --- | --- | --- |
| C | Long use, many developers, high performance, many libraries. | Many risks with memory access. Some aspects can be tricky to use. |  |
| C++ | Same as C but includes object-oriented approaches and type safety. | Difficult syntax in many cases due to backwards compatibility with C. |  |
| C# | Modern version of C/C++ that address many of the shortfalls of both. |  |  |
| COBOL | Heavily used in business software. | Primarily used for existing COBOL environments, not for new implementations. |  |
| GO | A new language with some very nice features. | Few libraries, too new/ |  |
| Java | Powerful language with good library support. | Limited by the Java runtime environment. |  |
| JavaScript | Widely used and supported in browsers for browser side execution. | Very limited syntax. |  |
| Python |  |  | See additional benefits in Jenkins, T. (2004). |
| Rust | A new language with some very nice features. | Few libraries, too new. |  |
| SQL | Good for set based access, primarily a language for databases. | It is difficult to control execution path. |  |
| Visual Basic | A key component of the Microsoft ecosystem. Integrates well with their tools. | Has limited support for modern software engineering. |  |

Based on the information collected above and a review by the key stakeholders, a hybrid approach to the Business Analyst Platform was selected. In this approach, Java will be used for server work, JavaScript will be used for User Interface work, SQL will be used for database work, and Python will be used for Machine Learning work. There will be interfaces between all these levels. Unless there is a compelling reason not to, all interfaces will be through RESTFul APIs.

# Finding Data for Training

The platform will be used to convert notes from meetings and workshops into an intermediate formal language. Once complete and consistent, the formal language will be converted into Business Analysis documents. Therefore, the model training will require notes from meetings and workshops as the features to be trained on. The decision has been made to conduct supervised training, so for example Business Analysis documents will also be needed.

Given the proprietary and trade secret nature of the notes and documents, it is hard to envision a source of these. Therefore, the team will use experts to create the needed information for training. At first this may seem to be a daunting task. However, the team will break the problem down and solve small pieces of the problem. For example, they will create several versions of statements that define an organization’s goal. All of these will be converted into a formal language. This will be used for training.

Since the focus of Business Analyst is on a limited scope, the creation of sample data will be a manageable task. However, it is well understood that this created data will be of limited variety and will probably be very clean. Therefore, once the initial model is working, it will be important to find some friendly initial clients. These can provide data and the system can be trained on it. This will make the system more robust while providing some value to the initial clients.

# Missing Data

Missing data will be a big problem for Business Analyst, but it will be able to be handled through the process. During interviews and workshops, there will be a lot of data presented but also a lot will be missing. It may be knowingly missing or just overlooked. The missing data will show up in multiple places. Interestingly, it will be hardest to detect in the original notes. Here, things like the date of the meeting, attendees, agenda and such can be identified as missing. For these items, there will be a simple imputation strategy. The meeting date can be imputed to be halfway between the meeting before and the meeting after. Attendees can be inputted from the invitees. The agenda will have to be left as missing. A more advanced approach would be to impute the agenda from the meeting notes.

Once the notes are converted to the formal language, data missing in the notes can be identified. For example, if the notes cover a discussion of an initiative but the length of time or priority of the initiative and never covered, this will be missing in the formal description. For the initially created training data, this issue can be rectified by simply adding the information to the training set. For the training data from the friendly clients, there will be two approaches. First, the team can go back to the friendly client and see if they have the missing data. If the data is not available, the team will need to develop an imputation strategy.

The trained model will need to be able to handle some missing data. For the most part in production use there will be little missing data and any that there is can be collected and rerun. However, there will be cases where something is just not known. In these cases, the model needs to produce a reasonable result.

# Model Interpretability

Generally, model interpretability is an important issue. While there is no one definition for what makes a model interpretable, there are some common understandings. These include simulatability, decomposability, and algorithmic transparency. While it is important to have these, it is not always possible. The use of Neural Networks both gives models great power but also limits this type of transparency.

In this case, we may be able to adopt a post hoc transparency. That is, the model can convert the meeting notes to the formal language. Then the formal language can be reviewed. If there are any issues, they will often be able to be addressed by adding to the notes. This will work since expert business analysts are collecting the data, passing it to the model, and reviewing the results. The model is primarily serving as a tool so they can do their job quicker and with higher quality. While this will not address all transparency needs, it will address many.

# Conclusion

The Business Analyst company has embarked on an interesting and challenging path. They have undertaken to apply AI and Machine Learning to an area that would not generally be considered ripe for AI and Machine Learning. However, by understanding the issues involved and judiciously using state-of-the-art techniques, Business Analyst has a reasonable chance of being successful. They have also chosen to create a useful tool that can make people more productive and able to produce higher quality deliverables. This approach makes the best use of Human and Machine abilities.

References

Adewale Akinfaderin, Matthew Chasse, Michele Donini, and Benjamin Fenker, Amazon Web Services (AWS) (2022). Machine learning model interpretability with AWS, <https://docs.aws.amazon.com/pdfs/prescriptive-guidance/latest/ml-model-interpretability/ml-model-interpretability.pdf>

Gómez-Caicedo, M. I., Gaitán-Angulo, M., Bacca-Acosta, J., Briñez Torres, C. Y., & Cubillos Díaz, J. (2022). Business analytics approach to artificial intelligence. *Frontiers in Artificial Intelligence*, 5, 974180. <https://www.frontiersin.org/articles/10.3389/frai.2022.974180/full>

Jenkins, T. (2004). The First Language - A Case for Python? *Innovation in Teaching and Learning in Information and Computer Sciences*, *3*(2), 1–9. doi:10.11120/ital.2004.03020004

Mui, Michael & Holler, Anne. (2019). Evolving Michelangelo Model Representation for Flexibility at Scale.

Stanley, J. (2017, June 13). Space, Time and Groceries. Grocery delivery visualized in python with datashader. tech-at-instacart. <https://tech-at-instacart.com/space-time-and-groceries-grocery-delivery-visualized-in-python-with-datashader-8b8e8712f26a>

Zachary C. Lipton. 2018. The Mythos of Model Interpretability: In machine learning, the concept of interpretability is both important and slippery. Queue 16, 3 (May-June 2018), 31–57. <https://doi.org/10.1145/3236386.3241340>

Zhang, Y., Sun, Y., & Liu, X. (2022, May 10). Rapid regression detection in software deployments through sequential testing [arXiv preprint arXiv:2205.14762]. <https://arxiv.org/abs/2205.14762>