5 Steps to Maximize Business Impact with Machine Learning

BY INTEL® TIBER™ AI STUDIO TEAM(HTTPS://CNVRG.IO/AUTHOR/CONTRIBUTOR/)

One of the most enjoyable feelings I experience as a Data Scientist is watching the error rates fall as I work on my modeling projects. The iterative process of increasing the accuracy of a classification algorithm through feature engineering or driving down the mean squared error of a regression model through residual analysis is intensely satisfying. One can get lost, as I often do, as the hours and days pass in pursuit of the "perfect" model.

As a team leader and entrepreneur, though, I'm acutely aware that maximizing business value is of supreme importance. When I step back from my keyboard and close the Jupyter notebook, I force myself to think about whether my instincts to try a new modeling approach or run a new experiment are intended to improve business outcomes or to satisfy my ego. But simply asking yourself whether you should run that new experiment when you're in the middle of a project is not enough. In order to maximize the business impact of machine learning efforts, you need a process that ensures technical work is aligned with business objectives in each of the project ideation, planning, and implementation stages.

Here are 5 steps you can take to ensure that your machine learning efforts are properly aligned with the business outcomes you wish to achieve.

Using a Machine Learning Platform for Business Growth

1. Determine the desired business outcomes

Before beginning any technical work, it's imperative that you have a clear understanding of what the business wishes to achieve. One way to think about this is to imagine how the business would be different after the successful implementation of a project. Does the business wish to reduce customer churn? Is a department seeking to eliminate a manual task that keeps employees tied up? Is the Product team seeking to increase engagement in the mobile app to drive purchases? These are all examples of business outcomes that can be achieved through the application of machine learning.

Achieving these results begins with clearly stating the outcomes. Neglecting this step, say by rushing into the model building process, can lead to a great deal of time, effort, and resources put into building solutions that address the wrong questions. Clearly defining the desired business goals at the outset of a machine learning project provides a north star to follow.

2. Define the success criteria

Once the desired business outcomes are clearly stated, you need to define the measures of success that will track your progress towards those goals. These metrics will guarantee that your team's work is moving the business towards achieving its goals. The best metrics are specific and measurable. Specific metrics reduce ambiguity and increase focus. And, the easier it is to measure your success, the more certain you can be that you're achieving the results you set out for in the first place.

Suppose the business intends to reduce customer churn. Specific and measurable success criteria include a clear definition of churn and the desired threshold. For instance, a subscription service may define churn as the percentage of customers that cancel their subscription service within 3 months of signing up. A specific and measurable success criteria would be to reduce this number to 5% of all new customers within the next year.

As another example, consider a company whose operations team performs a significant amount of manual data entry. Individuals on this team are able to process approximately 20 documents per day with the current system. The business is determined to cut costs and would like to increase the number of documents processed per employee to 30 documents per day. The specific business metric is the average number of documents processed per day by employee.

While specific and measurable criteria are ideal, it may sometimes be necessary to settle for more subjective criteria. For instance, a business may desire to improve employee or customer satisfaction but it might be difficult to pin this to a specific set of metrics. In this case, you should explicitly state who is going to make this judgment call.

3. Translate the business success criteria into a machine learning metric

After defining the business objectives and success criteria, it's time to define the successful outcome in technical terms. For machine learning projects, this means selecting and optimizing an appropriate metric that directly relates to the business metrics chosen in the previous step. Machine learning models are trained by optimizing towards metrics such as accuracy, area under the ROC curve (AUROC), or root mean squared error (RMSE). By selecting metrics that map to business outcomes, data scientists can optimize directly towards the goals the business wishes to achieve. Choosing inappropriate metrics that aren't tied to business outcomes leads to project failure and business losses.

It's important to note that selecting an out-of-the-box metric such as accuracy or AUROC may not be good enough. For instance, optimizing towards these metrics when facing a class imbalanced dataset (https://towardsdatascience.com/dealing-with-imbalanced-classes-in-machine-learning-d43d6fa19d2) can lead to suboptimal models. This situation often occurs in areas such as fraud detection where the number of examples of fraudulent activity is significantly smaller than the number of non-fraudulent instances. Selecting the wrong metric in this instance can lead to a model that looks great from the perspective of accuracy but still misses the most costly transactions. The importance of selecting machine learning metrics that directly relate to business value cannot be understated.

4. Establish a baseline

Regardless of the metric you decide to optimize, it's necessary to establish a baseline measure of performance. This baseline provides a point of comparison that enables you to track your progress. It also allows you to judge the rate of return you'll get by increasing the complexity of your modeling solution. Suppose you work for a real estate firm and are asked to build a model to predict the price of a house. You decide to optimize for RMSE and build a linear regression model with features including the square footage of the house, the number of bedrooms and bathrooms, and other information. While this linear model may not lead to the lowest RMSE, it does have some attractive qualities. The coefficients of the model can be used to explain the importance of the input features and the functional form is very easy to explain. If building more complex models leads to only marginal improvements in RMSE, you may even decide to deploy the simpler linear model because of its explainability and simplicity.

5. Deploy an MVP quickly and begin iterating immediately

In his Rules of Machine Learning (http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf), Google Research Scientist Martin Zinkevich describes best practices for building machine learning products. Early on in a product's development, Zinkevich stresses that developers should "Keep the first model simple and get the infrastructure right" (Rule #4). Deprioritizing modeling gains allows machine learning engineers to sort out the numerous infrastructure issues that need to be solved before the benefits of machine learning predictive models can be fully realized. The simple model provides baseline metrics and behavior to compare against and a reliable pipeline gives you the ability to test more complex models.

Zinkevich isn't the only heavy hitter to emphasize this iterative approach to building machine learning products. Peter Skomoroch, CEO of SkipFlag and former lead data scientist at LinkedIn and AOL, recently stated that developing machine learning

algorithms without user feedback is risky and can lead to unintended consequences. Instead, teams should "ship a complete MVP in production ASAP, benchmark, and iterate" (https://cdn.oreillystatic.com/en/assets/1/event/292/Executive%20Briefing_%20Why%20managing%20machines Andrej Karpathy's advice for picking a first model is "Don't be a hero." (https://karpathy.github.io/2019/04/25/recipe/) The Director of AI at Tesla says that in the early stages of a product, developers should copy and paste the simplest architecture they think will solve the problem. "You're allowed to do something more custom later and beat this", he continues.

Conclusion

In order for machine learning projects to have a massive business impact, practitioners need to be relentlessly focused on asking the right questions. Rather than dive directly into technical details and implementation, data scientists must ensure that they understand the business problem as specifically as possible. This involves speaking with stakeholders and translating their needs into technical requirements. Once that's done, data scientists should work to integrate their work into products as soon as possible and get feedback from customers. This feedback loop will guarantee that data scientists are optimizing towards the success of their business rather than meaningless vanity metrics.

Clearly defining business goals, choosing appropriate measures of success, and deploying solutions iteratively are each important steps towards building great machine learning products. But machine learning solutions have a host of other requirements, many of which are different from the requirements of traditional software application development. For instance, whereas the marginal cost of software (https://ckluis.com/the-marginal-cost-of-software-approaches-zero-7fda166f219f) approaches zero over time, machine learning models require additional investments in orchestration to ensure model performance doesn't degrade below acceptable thresholds.

About the author:

Luigi Patruno is an expert data scientist and entrepreneur focused on helping companies utilize machine learning to create competitive advantages for their business. He has built and deployed thousands of machine learning models at Fortune 500 companies, startups, and academia and has taught graduate level courses in machine learning and data engineering. As the founder of MLinProduction.com (http://MLinProduction.com), Luigi creates and curates content with the goal of educating data scientists, data analysts, and machine learning engineers about best practices for running resilient machine learning systems in production. If you're interested in learning about how to build, deploy, and run production machine learning systems, sign up to follow his blog at MLinProduction.com (https://mlinproduction.com/?utm_source=cnvrg&utm_medium=blog&utm_campaign=5_steps_maximize_business_impact).

References

- What is the CRISP-DM methodology? (https://www.sv-europe.com/crisp-dm-methodology/)
- Dealing with Imbalanced Classes in Machine Learning (https://towardsdatascience.com/dealing-with-imbalanced-classes-in-machine-learning-d43d6fa19d2)

- Rules of Machine Learning: Best Practices for ML Engineering (http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf)
- Why managing machines is harder than you think (https://cdn.oreillystatic.com/en/assets/1/event/292/Executive%20Briefing_%20Why%20managing%20machir
- A Recipe for Training Neural Networks (https://karpathy.github.io/2019/04/25/recipe/)
- Three Ways to Identify the Right Metrics in a Deep Learning Strategy (https://insidebigdata.com/2018/04/27/three-ways-identify-right-metrics-deep-learning-strategy/)
- Machine Learning: Balancing model performance with business goals (https://towardsdatascience.com/machine-learning-balancing-model-performance-with-business-goals-57eaa870ff66)
- Metrics Gone Wrong – How Companies Are Optimizing The Wrong Way (https://www.kdnuggets.com/2016/04/metrics-gone-wrong-optimizing-wrong-way.html)

Top MLOps guides and news in your inbox every month

AI & Machine Learning   Data Science   Announcements

Recent Posts

- This Month in MLOps and LLMOps (https://cnvrg.io/mlops-llmops-monthly-03-2024/)
- Beyond ChatGPT: Building a LangChain-Powered App with Intel® Tiber™ AI Studio, Part 2 (https://cnvrg.io/building-langchain-powered-app-pt2/)
- Beyond ChatGPT: Building a LangChain-Powered App with Intel® Tiber™ AI Studio, Part 1 (https://cnvrg.io/building-langchain-powered-app-pt1/)
- Unlocking Efficient Deep Learning: Intel® Tiber™ AI Studio MetaGPU Sharing (https://cnvrg.io/introducing-metagpu-sharing/)
- Enhance Large Language Models Leveraging RAG and MinIO on Intel® Tiber™ AI Studio (https://cnvrg.io/enhance-large-language-models-leveraging-rag-and-minio-and-cnvrg/)
- Easily build an automated video transcription AI pipeline with Whisper in Intel® Tiber™ AI Studio (https://cnvrg.io/easily-build-an-automated-video-transcription-ai-pipeline-with-whisper-in-cnvrg-io/)
- Unlocking AI's Potential: Supercharging Enterprises with LLMs, cnvrg, Intel Developer Cloud, and Redis (https://cnvrg.io/ai-transformation-cnvrg-intel-redis/)
- The Anatomy of an AI Blueprint: A step by step guide (https://cnvrg.io/the-anatomy-of-an-ai-blueprint-a-step-by-step-guide/)
- How to create your own low code AI solution with Intel® Tiber™ AI Blueprints (https://cnvrg.io/how-to-create-your-own-low-code-ai-solution-with-cnvrg-io-ai-blueprints/)

You might also like

- The Ultimate Guide to Building a Scalable Machine Learning Infrastructure (https://cnvrg.io/building-scalable-machine-learning-infrastructure/)
- How to apply LSTM using PyTorch (https://cnvrg.io/pytorch-lstm/)
- Deep Learning Guide: How to Accelerate Training using PyTorch with CUDA (https://cnvrg.io/pytorch-cuda/)
- Getting Started with Sentiment Analysis using Python (https://cnvrg.io/sentiment-analysis-python/)
- How to Apply Hyperparameter Tuning to any AI Project (https://cnvrg.io/hyperparameter-tuning/)
- How to use random forest for regression: notebook, examples and documentation (https://cnvrg.io/random-forest-regression/)
- The Definitive Guide to Semantic Segmentation for Deep Learning in Python (https://cnvrg.io/semantic-segmentation/)
- The essential guide to resource optimization with bin packing (https://cnvrg.io/bin-packing/)
- How to build CNN in TensorFlow: examples, code and notebooks (https://cnvrg.io/cnn-tensorflow/)

(https://cnvrg.io/bin-   (https://cnvrg.io/gaudi-ai-   (https://cnvrg.io/lenovo-   (https://cnvrg.io/dell-

0.5
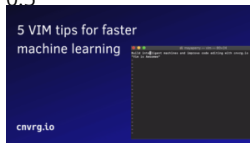
(https://cnvrg.io/5-vim-tips-for-faster-machine-learning/)   (https://cnvrg.io/how-to-compile-tensorflow-1-12-on-ubuntu-16-04-using-docker/)   (https://cnvrg.io/5-things-to-consider-before-building-an-in-house-data-science-platform/)   (https://cnvrg.io/how-to-setup-docker-and-nvidia-docker-2-0-on-ubuntu-18-04/)
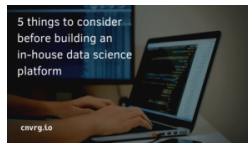
ALSO ON CNVRG.IO.DEV

The Anatomy of an AI Blueprint: A step by …

a year ago · 3 comments

For an overview on AI blueprints please refer to part 1 of this blog here

How to create your own low code AI solution …

a year ago · 1 comment

AI Blueprints are pre-assembled Machine Learning (ML) pipelines …

How to setup Docker and Nvidia-Docker 2.0 …

6 years ago · 3 comments

A step-by-step guide for data scientists working on Deep Learning …

Deploy your mach learning models w

6 years ago · 1 comment

Deploy machine learn models with kuberne just a single click, a d

0 Comments   Login ▼

AI:OS for Machine Learning

Schedule Demo (/demo/)

Platform	Solutions	Resources	Company
Overview (/Platform/)	Data Scientists (/Solutions/Data-Scientists/)	Blog (/Blog/)	Careers (/Careers/)
Enterprise (/Enterprise/)	Executives (/Solutions/Executives/)	Mlcon 3 (/Mlcon-3/)	News (/News/)
Core (/Platform/Core/)	IT Experts (/Solutions/It-Experts/)	Docs (Https://App.cnvrg.io/Docs/)	Intel Software (Https://Www.intel.com/Cor Overview.html?
Features (/Features/)	Software Developers (/Solutions/Software-Developers/)	Wiki (/Wiki/)	_hstc=62913267.B8e8baa8
Integrations (/Integrations/)	MLOps (/Platform/Mlops/)	Webinars (/Webinars-And-Workshops/)	Privacy (/Privacy-Policy/)
		Case Studies (/Case-Studies/)	Terms Of Use (/Terms-Of-Use/)