Q1 2021

Ten Mistakes to Avoid

In the Machine Learning Life Cycle

By Fern Halper

Sponsored by

CLOUDERA

tdwi
Transforming Data With Intelligence™

tdwi.org

Ten Mistakes to Avoid

In the Machine Learning Life Cycle

By Fern Halper

FOREWORD

Machine learning—where systems examine data to identify patterns with minimal human intervention—is becoming part of the analytics fabric of many organizations as its competitive value becomes understood. Organizations are making use of machine learning (ML) technologies in numerous ways. Some may sound familiar, such as using ML to build churn models or predict fraud. Others seem more revolutionary, such as using ML to diagnose cancer or improve crop yield. ML is being used across the enterprise and across industries, and those organizations that are already using ML technologies are gaining value from them.

It is no surprise, then, that TDWI research also indicates that *demand* for machine learning is growing. In fact, it is currently in the early mainstream phase of adoption among TDWI survey respondents. If users stick to their plans, a majority of organizations will be using the technology in the next few years. Once organizations get comfortable with machine learning, they may also use techniques such as deep learning for use cases that often involve image recognition (e.g., identifying objects such as cars for automated auction sites or for medical use cases such as disease testing) or venture into other AI-based technologies such as natural language processing.

Although many companies are excited about machine learning, they often overlook some key success factors. To succeed in ML, enterprises must embrace the full ML life cycle in a unified way—from data management and governance to data engineering to building the model and putting it into production while ensuring that the organizational culture embraces predictive applications. Some of the biggest challenges with ML have to do with everything around the actual ML workflows, including preparing and automating data pipelines, creating explainable predictions, managing models, and building trust, so it is important to not make the interrelated ten mistakes described here.

1 MISTAKE ONE:
FAILURE TO IDENTIFY THE RIGHT USE CASE FOR ML

Analytics succeeds when it provides business value, yet at TDWI we often see that organizations do not think through which business problems will be best to address with machine learning. It sounds obvious, but it is critical to understand the business problem your organization is trying to solve with machine learning. Doing so will help to articulate and frame the machine learning model. This is often harder than it sounds. Many organizations are not used to the thought process involved in being proactive rather than reactive. It requires a shift in mindset and someone to guide the process.

It is also a good idea to pick an initial problem that is visible and offers easy access to the data. It can make sense to choose a problem where you have past results and, ideally, metrics. For example, in the case of customer churn, you might have past churn figures for a certain class of customer, what steps you implemented to reduce churn, and a record of how well your actions worked. Additionally, you can show how your models might have done better at predicting churn using past data.

Because data science should be linked to business strategy and goals, it also makes sense to get the business involved early on. Collaboration can happen at different levels to identify the right model and formulate how to structure the model. For instance, the data scientist may collaborate with the business. Analysts might collaborate with data scientists. The mistake some companies make is that they create their ML model in a vacuum and then are surprised when the business says it isn't useful.

2 MISTAKE TWO:
FAILURE TO CONSIDER THE DATA INFRASTRUCTURE AND ARCHITECTURE

As organizations make the move to ML, they are often dealing with more data sources and more data types than they typically would find in the data warehouse that supports their reports and dashboards. In response to user demands, the software vendor and open source communities are supplying many new data platforms, tools, and capabilities, purpose-built for modern data and its use cases.

TDWI research respondents often recognize the need to expand their data infrastructure to support advanced analytics such as machine learning. If demand is increasing, organizations should not make the mistake of failing to plan for what their data infrastructure and architecture should look like to support the ML life cycle. Satisfying ML's diverse data requirements involves a modern data management infrastructure, typically including cloud-based platforms for data warehousing, data lakes, and data integration, to handle the compute-intensive, iterative nature of ML model building. However, the infrastructure should meet data scientists where they need to be, whether that is on premises or in the cloud.

The key is that the architecture should be flexible and scalable with access to elastic compute. If an architect with experience in big data, the cloud, and new software such as containers isn't part of the team, the organization should consider hiring one. Additionally, the data infrastructure must be secure and governed so it can be trusted. That means putting a governance plan in place that spans a hybrid environment (see Mistake #8).

3 MISTAKE THREE:
FAILURE TO CONSIDER THE NEED FOR DATA ENGINEERS AND TOOLING

As organizations start to ramp up their ML efforts, one of the first issues they may encounter is a lack of talent. They are typically focused on hiring one or several data scientists to build out some of their models. These data scientists are often trained in computer programing and know open source languages such as R and Python, as well as statistics. If the organization is lucky, some of their data scientists have strong communication skills and are business savvy.

Yet, in the rush to hire data scientists to build models, organizations often forget about other skills needed for machine learning, such as data engineering. Among their other job responsibilities, data engineers may set up the processes for data acquisition, transformation, and validation. This includes assembling the data, including features (e.g., engineered attributes such as ratios or metrics) that will be needed to train a model. It will also include assembling pipelines of fresh data to help score a model once it is in production. Although data scientists are often able to build out pipelines, this isn't a scalable, long-term solution.

Data engineers are technical—they may know Java, R, or Python. They are also SQL experts. Data scientists we speak with at TDWI often say that they need more data engineers to help them build the data pipelines for machine learning models. Additionally, it is not just about hiring data engineers; it is important to give them the tools they need to be productive. This includes tools for building pipelines, orchestrating them, managing multiple pipelines, and automating them. Vendors are starting to provide tools to help build complex pipelines and automate them. Organizations should consider these tools as part of their plans.

4 MISTAKE FOUR:
FAILURE TO THINK ABOUT DATA PREPARATION AND FEATURE ENGINEERING

As organizations start to utilize ML, they sometimes don't think enough about the features they'll use to build the models. Feature engineering involves constructing new derivative attributes that can represent the problem to be solved. This involves transforming raw data into something more meaningful, i.e., something clearer than what is available in the raw data. Features can be relatively straightforward (such as a ratio) or they may be more complex (such as a customer loyalty score). Data scientists will often spend a great deal of time developing features, and they would argue that feature engineering is part art and part science.

The data used to construct the features must be trusted and of high quality. The old adage "garbage in garbage out" applies here. Although data scientists might operate in sandboxes that are part of their data platform, that doesn't mean they want to use low-quality data or spend most of their time cleaning up the data. Yet, TDWI research indicates that a majority of organizations are not satisfied with the quality of their data. Many organizations don't do a good job profiling newer forms of data that might be of interest to data scientists (see Mistake #8).

Additionally, as organizations train business analysts to use ML and automated ML tools to build models, they should pay particular attention to what is involved in developing features for use with ML algorithms. Some tools on the market claim to automate the feature-building process. However, model builders will still need to validate that the feature selected makes sense and will most likely still need to build additional features.

MISTAKE FIVE:
FAILURE TO CONSIDER THE NEED FOR MLOPS

In addition to data engineers and data scientists, another skill set needed for machine learning is MLOps (also known in some organizations as DataOps or ModelOps). Among other responsibilities, the MLOps team puts the models into production. Similar to DevOps, which focuses on application development, MLOps focuses on all of the steps for operationalizing models in an agile way while ensuring high-quality results. These steps include model validation, certification, deployment, and monitoring. Similar to data engineers, MLOps is a technical team that often has programming skills in languages such as Java or Python. They are well versed in APIs, modeling frameworks (such as PMML), and newer techniques for putting models into production, such as containers. They are the team that typically handles more "back room" model issues such as management and monitoring (see Mistake #6).

The MLOps team may also be charged with validating models against technical factors such as alignment of data in the target business process: Do the models work in the system they'll be embedded in as opposed to just with the training data? Will the model work with existing data endpoints? Is there easy access to the data? Sometimes organizations end up having to rewrite the entire model because it doesn't work with the target system. That takes time and introduces errors.

Some organizations manage to put models into production using the skills of their data scientists. That can work when an organization only has a handful of models in production; however, the mistake these organizations are bound to make is that this approach isn't scalable, especially for models that need to be updated frequently (see Mistake #6). Moreover, putting models into production isn't work that data scientists typically like to do. An MLOps team frees up data scientists to do the work they are good at and enjoy doing.

6 MISTAKE SIX:
FAILURE TO REGISTER AND MANAGE/MONITOR MODELS

Just like any piece of software, a model needs to be managed and versioned. It is important to version the model in the building stages as well as when you've made a change to a model that has been deployed (much like you do with traditional software applications). Yet, many organizations don't capture information about the models they build, such as who created it, what data was used for it, and other important metadata. That doesn't provide much control over machine learning and can cause problems once that model is put into production. It is one thing if an organization only has a few models deployed; that can be managed manually. However, some organizations have hundreds or even thousands of models in production. How would they track those in a file system? It is important to organize and track both trained and deployed models.

Successful organizations use some kind of model repository or registry to keep track of their machine learning models and information about these models. Features will vary by vendor. Look for features to help govern models, track history, and manage who can approve changes. Some products have prebuilt model life cycle templates. Others include tools to document models as well as collaboration tools to help data scientists, data engineers, and MLOps work on models and move them from training to deployment. Vendors may also offer software to help manage the model workflow.

Likewise, model management includes monitoring model performance over time. It is not simply a matter of building a model and letting it run for years; models can become outdated as environmental factors change. For instance, some organizations building ML models for e-commerce recommendation engines update the models daily using new data to keep up with changes in user behavior patterns. Without such updates, that model will degrade. Of course, not all models need to be updated daily; the update frequency depends on the use case.

The key point is that without monitoring, you will not know when your model has degraded. Some organizations will monitor using a champion/challenger approach by creating their own algorithms that they put into production. The idea is to be able to track that the champion model (the one in production) continues to perform efficiently against a challenger model. Some organizations use tooling to collect performance data from the models and use that data set to assess model accuracy. If a model starts to degrade, it will need to be retrained.

7

MISTAKE SEVEN:
FAILURE TO CONSIDER DATA ISSUES IN PRODUCTION

Once a model is deployed, it will need to be fed fresh data for scoring. For instance, in the recommendation engine example described above, the model will be scored using data attributes for each new customer browsing the site. That customer will be scored against the model. Model scoring will require gathering data, performing any preprocessing, and then calculating the features that need to be input into the model.

Organizations think about the data pipeline for model development but often neglect it for deployment—especially for engineered features. For instance, if an ML model utilizes some sort of customer metric as an input, that same metric will need to be recalculated with new data as the model is run. There will be batch versus real-time considerations as well. This should be automated in the pipeline.

Additionally, data isn't static. That same customer data or product information may change because the information in a source system changed. A product may be discontinued and removed from the source system, for instance. That will impact the model itself and will also impact the data pipeline that may feed an updated model. This is why both managing and versioning models (including what data goes into the model and what features are engineered) and having someone in charge of the pipelines are so important.

MISTAKE EIGHT:
FAILURE TO GOVERN DATA AND MODELS

Governance—rules and regulations to maintain data and model integrity—sometimes seems to be one of the last things organizations think about when moving to more advanced forms of analytics. Respondents to TDWI surveys cite that data governance is one of their top challenges in any data and analytics endeavor. In reality, you don't want to think you were using a set of data with high integrity only to find out it was of poor quality.

Data scientists like to build models with clean data. Cleaning data might involve deduplication, identity resolution, name and address cleansing, and dealing with missing data or outliers. It should also include profiling data to ensure data accuracy, timeliness, consistency, and completeness. When it comes to modeling, a machine learning model is only as good as the data used to build it. The surefire way to lose the trust of the enterprise in advanced analytics is to taint model integrity with poor-quality data.

Of course, with machine learning, in addition to data governance, model governance also becomes a priority. Model governance is about the policies, accountabilities, processes, and controls put in place by an organization to make sure that the models it builds and puts into production can be trusted. This includes data governance, but it also includes the processes described in Mistake #6. Automation can help. Organizations should look for tools that enable automatic lineage tracking, metadata management, model cataloging, and feature cataloging. All of these can help with model governance.

Model governance will necessitate building a team that includes IT and data engineers as well as MLOps, data scientists, and business analysts.

MISTAKE NINE:
FAILURE TO TAKE INTERPRETABILITY AND EXPLAINABILITY INTO ACCOUNT

Although this mistake isn't necessarily happening yet, it will happen more often as organizations build more complex models or utilize augmented intelligence. To many, ML may feel like a "black box" that ingests data and produces results and what goes on in the middle is unknown. Yet, model builders as well as business teams need to trust and use the resulting predictions.

Interpretability is about understanding an algorithm and how changes to it can change the output. *Explainability* involves understanding the *why* behind an ML prediction in a way a human can understand. If a model is built and put into production, the output of this model—and what went into determining the output—need to be understandable.

For instance, a customer should be able to understand why his loan or credit card application was rejected. That means that the person building the model needs to be able to explain its output. This is important for fairness and transparency; it also comes into play in certain regulatory requirements such as Article 22 of the GDPR, which states that users have the right to review automated decisions.

The good news is that work is going on in this area. An organization should be aware of how a model provides output and how it will use that output. Ask what features exist in a product to help provide explanations for users that include items such as feature importance, derived feature importance, and how variables interact with each other. This information is often presented in a dashboard or in a visualization so nontechnical users can understand a model without having to write code.

10 MISTAKE TEN:
FAILURE TO CONSIDER CULTURE AND PROCESS ISSUES

As we've noted, if you are going to take action on your analytics, you will also need to consider getting buy-in from the people who are involved day to day in the analytics. Often, advanced analytics projects fail because of people, politics, and processes. Cultural buy-in—aligning the belief system of an organization—is critical. Trusting the model is a piece of this, but it goes further. If you're going to put a model into a call center process, for instance, you must make sure that the call center agents know what to do with the output and see the value in using this output as part of their job.

This cultural buy-in is crucial. Successful organizations make sure to communicate the value of the analytics projects before, during, and after implementation. Executive leadership can make a big impact because leaders can articulate the vision and set out common goals. Contributions are recognized. These organizations also provide training in a collaborative atmosphere. Some organizations might also use formal change management practices, such as ADKAR (awareness, desire, knowledge, ability, reinforcement).

There are other creative ways organizations build an analytics culture. Some organizations advertise analytics project results throughout the company (e.g., in newsletters, in portals, on C-suite TV monitors) to market their wins. Still others host lunch-and-learn sessions to talk about analytics and share success stories.

Building the analytics culture can take time, but it is worth it. TDWI research shows that organizations that move forward with more advanced analytics (such as ML) are more likely to measure top- and bottom-line impact from their analytics efforts.

ABOUT THE AUTHOR

Fern Halper, Ph.D., is VP of TDWI Research. She has co-authored several books and published hundreds of articles, reports, and webinars on data mining and information technology. Halper focuses on advanced analytics, including predictive analytics, machine learning, text analytics, cloud computing, and big data analytics approaches. She has been a partner at Hurwitz & Associates and a lead analyst for Bell Labs. Her Ph.D. is from Texas A&M University.

ABOUT CLOUDERA

CLOUDERA

At Cloudera, we believe that data can make what is impossible today possible tomorrow. We empower people to transform complex data into clear and actionable insights. Cloudera delivers an enterprise data cloud for any data, anywhere, from the edge to AI. Powered by the relentless innovation of the open source community, Cloudera advances digital transformation for the world's largest enterprises. Learn more at Cloudera.com.

ABOUT **TDWI**

TDWI is your source for in-depth education and research on all things data. For 20 years, TDWI has been helping data professionals get smarter so the companies they work for can innovate and grow faster. TDWI provides individuals and teams with a comprehensive portfolio of business and technical education and research to acquire the knowledge and skills they need, when and where they need them. The in-depth, best-practices-based information TDWI offers can be quickly applied to develop world-class talent across your organization's business and IT functions to enhance analytical, data-driven decision making and performance. TDWI advances the art and science of realizing business value from data by providing an objective forum where industry experts, solution providers, and practitioners can explore and enhance data competencies, practices, and technologies. TDWI offers five major conferences, topical seminars, onsite education, a worldwide membership program, business intelligence certification, live webinars, resourceful publications, industry news, an in-depth research program, and a comprehensive website: tdwi.org.

Transforming Data With Intelligence™

555 S. Renton Village Place, Ste. 700
Renton, WA 98057-3295

T 425.277.9126
F 425.687.2842
E info@tdwi.org