WHITE PAPER

Efficiency, Productivity and Speed to Deployment

Using Teradata Enterprise Feature Store to Improve Machine Learning and AI Implementation

By Dr. Chris Hillman, Principal Data Scientist, Teradata

09.20 / DATA ANALYTICS / WHITE PAPER

teradata.

Table of Contents

Leaders across industry sectors increasingly recognize the significant value of integrating predictive analytics into their strategies The value of these projects can be huge; consultancy group McKinsey estimates the annual value of AI's contribution across all industries at between $9.5 and $15.4 trillion![1] Boston Consulting Group found that 85% of companies surveyed think AI will offer a competitive advantage.[2] But to realize this value, enterprises must plan now and architect for the deployments of millions of predictive models in the near future. McKinsey has also pointed out that many current AI projects are more 'science-fair'[3] experiments than value producing enterprise-wide implementations. It is only at enterprise scale with tens of-millions of predictive analytics queries supporting hyper-personalization and automation across the business every day that the true transformative value of AI can be realized.

As outlined in Analytics 123: Enabling Enterprise AI at Scale, existing approaches to the preparation and development of machine learning and AI projects are flawed and cannot scale to meet the demand for predictive analytics at the heart of businesses of the future. That white paper provides a blueprint for a modern, more effective approach that can successfully support these rapidly emerging requirements. It highlights that the hardest part of machine learning is deploying and maintaining accurate models in production, and that the majority of effort in building predictive models is the work needed to gather, cleanse, integrate and manipulate the data needed for model training and scoring. This paper focuses specifically on how Enterprise Feature Stores dramatically reduce the time and costs of this crucial stage.

1 https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/the-executives-ai-playbook?page=industries/

2 https://www.bcg.com/publications/2017/strategy-technology-digital-is-your-business-ready-artificial-intelligence

3 https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/straight-talk-about-big-data

Enterprise Feature Stores are emerging in digital native businesses as a solution to the challenges of creating, maintaining and using machine learning and AI models at scale. Those that can successfully build and manage these systems for curating and reusing features as the critical input data for predictive models will secure significant advantages, and Teradata believes that any business of scale should be considering an Enterprise Feature Store now. Teradata provides best practices and a perfect platform for building an Enterprise Feature Store (EFS) that can greatly condense the processing cycles, vastly reduce the time to create, update, and implement models and provide a solid foundation for successful models in production at large scale.

From Science-fair To Value Delivery

To be effective, predictive analytic models must produce ongoing value to the business which exceeds the cost incurred in creating them. This might sound obvious, but with less than 20% of analytics projects seen as successful[4] many organizations are struggling to deliver. Many are stuck in a 'hobbyist' stage in which small teams of highly experienced data scientists work in silos to create specific machine learning and AI models. Data preparation and feature engineering are typically carried out on a project-by-project basis by discrete teams. Separate pipelines are often built on individual computers using small sub-sets of data. This method leads to what Google refers to as "pipeline jungles" (Sculley et al, 2014), which are effectively silos of valuable intellectual property that are difficult to catalogue, share and reuse.

To be effective, predictive analytic models must produce ongoing value to the business which exceeds the cost incurred in creating them

When models are created as one-off science projects, or used infrequently within smaller organizations, the "pipeline per project" may be an acceptable way to operate.

However, if large scale development and deployment of models is required, the repeated manipulation of large amounts of data becomes very inefficient and a barrier to success. Duplication of effort, extended project timeline and slow time to production are just the most obvious issues. Many businesses have found that the impact of machine learning and AI on their business is effectively constrained by what can be built by small teams of data scientists, often working in silos, and by the limited data they can manipulate on desktop computers. Models are developed without clear and planned collaboration and visibility of the business, potentially limiting their usefulness and acceptance. Coding errors can creep in and be hard to spot from the outside. If original authors of models move on, the business can be left with unexplainable code, and undocumented decisions on feature engineering and model development.

Yet many also find that machine learning and AI projects end up creating and using virtually identical features. If these can be curated, documented, shared, and reused, it creates the opportunity to greatly reduce the time and resource spent on feature engineering. Trust, transparency, and application of AI across the business can also be enhanced.

To be successful, businesses increasingly have to be able to identify, leverage, and monetize their data assets effectively—and do so at scale and in near-real time. They must also be able to transparently account for the outcomes of AI and machine learning decisions by pointing directly to the data and models that delivered them. Doing all of this in timescales that meet the demands of business users, and in a cost-effective manner that delivers rapid return on investment, is the challenge. Creating an Enterprise Feature Store is the solution.

4 White A (2019), Gartner Predicts 2019: Data and Analytics Strategy; https://blogs.gartner.com/andrew_white/2019

What is Feature Engineering?

Features are the input data for predictive models, they are comprised of data representing a measurable property which is used to predict a certain value or outcome relevant to the business, known as a class or classes. For example, customer churn predictions are used in many businesses. The **Class** in this case is churn, and it has a value of "churn" or "no churn." The **Features** used as input data for a model to predict churn could include business information like "length of contract remaining," personal information such as "age," customer experience information such as "number of dropped calls in latest month," purchase information such as "percentage drop in data usage," and marketing information such as "upgrade offer sent." Some of this data is a simple retrieval or aggregation of data from the data warehouse but some has a more complex derivation, combining data from several different data sets.

"Feature engineering" is the term given to the steps taken to transform data from its basic state into something that is useful in training a model to predict the class that is of interest. A feature is rarely found as a single existing column in a Data Warehouse, it is usually the result of taking data and applying various transformation steps that produce a new entity. This usually involves large numbers of data joins often on extremely large datasets where domain specific information is stored in separate tables and schemas. But it also requires more than simple aggregation of data from different sources and is a complex, time-consuming and highly skilled task.

Features are the valuable intellectual property of an enterprise that will differentiate it from the competition

Data scientists use statistical and mathematical approaches such as Principal Component Analysis (PCA), Fast Fourier Transformation (FFT), and Class Dependent Binning as well as recoding techniques such as Embeddings to create useful features.

The way in which features are "engineered" greatly affects the accuracy of machine learning and the AI model built using them, and features are highly valuable to an enterprise. They are the product of business expertise, technical expertise and a great deal of experimentation and iteration over model training and evaluation. They are the valuable intellectual property of an enterprise that will differentiate it from the competition and should be treated as first-class entities in a data management eco-system.

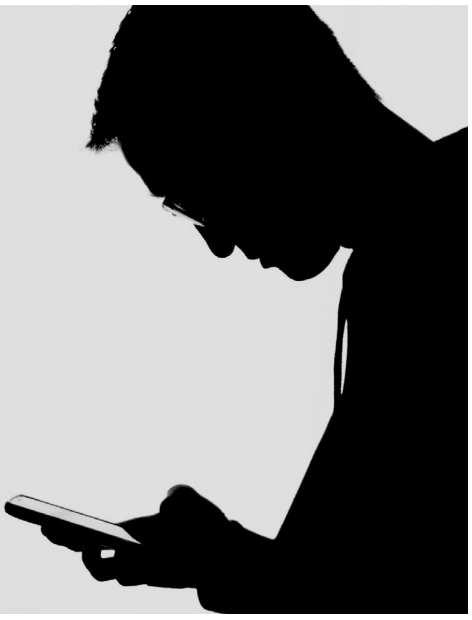
A Thought experiment – which Scenario results in a better model?

Scenario 1

An extremely talented data scientist builds a data set that can be used to train an accurate predictive model. They complete all of the feature engineering tasks, and leave a novice to use a tool of choice to build a model but the novice is not allowed to change the input data.

Scenario 2

A novice builds a data set that can be used to train an accurate predictive model. They complete all of the feature enginfeering tasks and an extremely talented data scientist can use the tool of their choice to build a model but the data scientist is not allowed to change the input idea.



teradata.

Organizations that can better engineer and use features as inputs to their machine learning and AI projects will see better, more reliable and more predictive results that deliver significant competitive advantage.

With a well-engineered set of features even a novice data scientist can produce an accurate, useable, and predictable model in a short period of time using a modern model building tool. Conversely, it would be very difficult for a novice to understand and take all the necessary steps to create a dataset that even an experienced data scientist could use to build an effective, performant model. This is the rationale behind the emphasis on the feature engineering process. Although less 'sexy' than model building, it is where differentiation and real value is distilled from data and one of the main drivers of producing robust, accurate models. It can take time, experimentation, deep domain knowledge and sometimes luck to find the stable, valuable features needed for prediction, so it makes sense to find ways to store, protect and reuse the outputs of this intensive work

What is an Enterprise Feature Store?

Teradata has long argued[5] that these valuable company assets should be collected into an Enterprise Feature Store (EFS); a curated, managed repository of features that have been used in successful predictive models.

An EFS provides the foundation for not only reusability and efficiency, but also increased consistency, robustness, and ROI of machine learning and AI projects. By collecting tried and tested features, an EFS creates a single source of data inputs to feed into machine learning and AI projects, dramatically reducing time to production and increasing efficiency of data science teams. Cataloguing and documenting features makes them accessible and available for reuse and research, significantly increasing model-building output as data scientists no longer spend 80% of their time repeating data preparation tasks. Making better use of data scientists reduces time to production.

Reusing proven precomputed features also reduces overall processing cycles as computationally intensive work can be done once and periodically refreshed as batch processes.

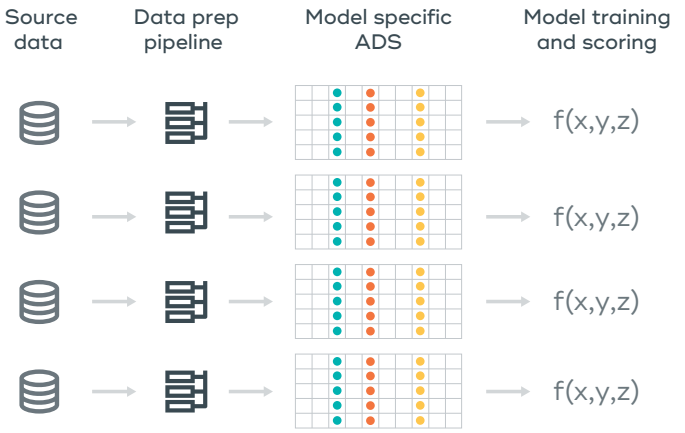
A leading financial services company currently utilizes a Teradata EFS with 1,400 variables. This enabled them to complete a specific analysis within hours, while their competition took weeks to complete the same analysis.

Those models are also likely to be more robust as the chance of errors from incorrectly coded logic are greatly reduced through reusing features that have already been created, documented, and tested with live data. Using the same metrics across processes and models not only minimizes data replication and redundancy but the fidelity of models can all be better assured as the precise conditions when a prediction was made can be robustly recreated as needed to ratify outputs. Consistent definitions used across the business also helps with governance and regulatory requirements. The EFS stores all the data needed to demonstrate exactly how and why a decision was made.

An EFS can also be a valuable tool in breaking down silos and increasing collaboration with data science teams and between data science and the business. As businesses look to generate returns from data assets, they'll also want the ability not only to analyze readily available information, but to build capacity to invest in further analysis and uncover new opportunities that monetize data or can transform business operations. The large number and variety of features that will accumulate over time can be continuously updated, documented and used to support new machine learning and AI projects across the business

5 Bill Franks, Former Chief Analytics Officer, Teradata outlined this methodology and advocated many of the ideas built upon in this document in his white paper of 2006 (Franks,2006)

The one pipeline per model approach



One pipeline per model

Redundant infrastructure, processing and effort

Limited re-use of pipeline or features

DSs functioning mostly as data janitors

Long data prep cycles and poor time to market

High TCO

Poor productivity and data silos

Inefficient allocation of resources

Consistency

- Same metrics used across processes and models
- Data replication, redundancy and errors all minimized
- Consistent enterprise-wide definitions aid regulatory and governance compliance

Robustness

- Reuse of existing documented and tested features reduces errors from incorrectly coded logics
- Recreation of precise conditions when a prediction was made support robust accountability

Reusability

- Reusable assets from analytic projects save time and increase productivity
- Catalogued to ensure availability for reuse and research

Efficiency

- Computationally intensive features are already materialized reducing overall processing cycles
- Data scientists and analysts can utilize their time on analyzing data and solving business problems, rather than repeating data preparation steps.

ROI

- An Enterprise Feature store allows business to easily drive value from data by analyzing readily available information.
- Production processes are simpler as the data is already in the correct format for scoring with live data.

What should go into an EFS

Every organization is different and will find themselves conducting different analyses on different data sets. But, in any company as the analytics team matures, it will find as Uber did, "many modelling problems use identical or similar features, and there is substantial value in enabling teams to share features between their own projects and for teams in different organizations to share features with each other"[6] These reoccurring data and processing steps are what should populate the EFS where they can be standardized and stabilized. It may not make sense to generate an EFS if minimal analysis is expected on an ongoing basis, but it does not take long for the amount of analysis to quickly rise to the point where it is valuable. For example, an organization implementing new CRM initiatives will certainly expect to execute many sophisticated analyses on their customers—and this is a case where an EFS makes great sense. Using the retail industry as an example, features to precompute and include in an EFS could include:

- Basic RFM metrics—total spend, number of transactions, and time since last transaction
- Average transaction size, revenue, and profitability (can be computed from above)
- Average distinct products, categories, or departments per transaction
- Discount, markdown, coupon, and other similar information
- Lifestyle metrics that can be identified from the transaction data, such as Low-Carb purchasing, allergy sensitive, or baby present
- Demographic, lifestyle, mail responsiveness, or other data purchased from third parties or acquired directly from customers
- Survey data from customer surveys
- Scores from any number of statistical models or deep dive analyses
- The EFS may also store computed values for these metrics against a number of common and useful dimensions:

6 https://eng.uber.com/michelangelo-machine-learning-platform/>

teradata.

- Time: Metrics might be stored for the current period, plus several past periods.
- Store or Location: Monitor customer behavior by store, store format, or region.
- Channel: See how customer behavior varies based on what channel is being used.
- Product: Compare patterns across products, categories, or departments.

Some of these metrics will be recognized and relevant in other industries, some will not. The specific metrics are not important—instead, think about how having the most frequent features to hand could accelerate the production and use of key insights across your business

Think about how having the most frequent features to hand could accelerate the production and use of key insights across your business

An Enterprise Feature Store might have several physical tables at different levels of aggregation. For example, demographics will be stored just at the customer level. However, spend related metrics could be available by customer, by store or by time-period. A series of views can provide access to different combinations of data. In some cases, it might be desirable to combine demographics with quarterly spend information. In other cases, a year of information might be desired for a specific demographic. With the appropriate combination of tables and views, the end user can be given whatever is required.

When designing an EFS, also remember that many metrics can be computed directly from other metrics, and so do not need to be explicitly stored. For example, given total spend and total number of transactions, you can compute average spend per transaction. Additional processing that can be done directly against the entity-level EFS does not need to be physically stored. Rather, views can be utilized to give access to the information. Another example would be storing results for each customer for each quarter while using a view to combine the quarterly data into an annual figure for the end user.

It is also possible to have different parts of the EFS updated on different schedules, as dictated by their usage. For example, a customer cluster classification may only be updated monthly or even quarterly, but basic customer metrics might require weekly refreshment. Other infrequently changing metrics can be updated on a trigger basis as and when needed. The unique circumstances surrounding each individual organization will determine what analysis is planned, what system resources are available, and how to define the refresh methodology.

Obviously, it is not possible to compute every possible variable in an EFS and models will invariably have features that are specific to their requirements or features that need to be tweaked. The goal is to standardize the most common and widely used variables. In some cases, the standard EFS will be all that is required for a new analysis to be completed from start to finish. In others it will be necessary to enhance the EFS with additional variables. As additional variables become quite common and are used in successful models, they should be added to the EFS. Ideally, obscure metrics or metrics computed across unimportant or uncommon dimensions should not be included.

Ultimately, the decision about what to include in an EFS will come down to some very basic trade-offs. Each additional variable in the EFS equates to more processing time, more complex scripts, and more storage space and that needs to be weighed against the value of insights it could deliver to the business. Note that when generating an EFS in real time through views, space is still a consideration alongside processing time. Practicality should determine which variables make the final cut

What's the ROI on an Enterprise Feature Store?

It is important to understand the costs involved and the effort required to build an Enterprise Feature Store. These include the resources and investment required to define and implement the EFS initially. Once established, scheduled processes to refresh the features need to be considered and these may involve more processing than any single run. Finally, additional disk space will be required to host results.

However, for reasons outlined above, these costs must be weighed against not only the time and cost savings delivered through making existing processes more effective, but also the competitive advantage, business agility and increased opportunity that better predictive analytics can deliver. Specifically, an EFS will quickly generate ROI by reducing the huge amount of time analysts are currently spending creating data sets.

Our analysis suggests that a Teradata Enterprise Feature Store can deliver 75% savings at the critical data preparation and manipulation stage of an AI project and nearly 40% savings overall.

A major cellular company has created a 450-variable customer EFS in Teradata Vantage. By leveraging the standard data source, development of new models was cut from weeks to days.

The table below represents the average time spent in each step of an analysis based on Teradata data scientists' implementations and split by the steps documented by CRISP-DM (Wirth 2000), a long-standing methodology originally published by a consortium that includes Teradata. In addition to these time savings, although the EFS may have more fields than any single

Average time spent on each step of the predictive analytics process	project based pipelines		Teradata Enterprise Feature Store
	Percentage of effort	Number of days to implement	Number of days to implement
Business Understanding	5 - 10%	3	3
Data Understanding	10 - 15%	5	1
Data Preparation	30 - 60%	15	4
Modelling	20-30%	6	6
Evaluation of Results	20-30%	6	6
Deployment	5 - 10%	3	3
Total Time		38	23

teradata.

process requires, it is more efficient to run one single large process over many smaller ones. Plus, analysts can experiment with additional data elements that may not have been worth computing for just a single project.

Having common metrics to hand, precomputed within the database, will enable many other applications or processes to leverage the information and extract value. Many of these other uses would not warrant a special process just by themselves and the availability of this data may encourage experimentation that unearths unexpected but valuable insights.

Finally, the additional disk space required to host the EFS is relatively inexpensive. When the benefits of the analytics are considered, the additional storage space should be easily justified.

Practical steps and Best Practice guidelines for creating an EFS

The rationale behind creating an enterprise feature store is to support predictive analytic models that produce ongoing value to the business which exceeds the cost incurred in creating them. Therefore, the business community must play a large role in defining the objectives and focus of the EFS. The specific contents, logic, and physical storage aspects of an EFS will require a team effort to outline and without active business involvement and support, there may be an extended period of trial and error to finalize these.

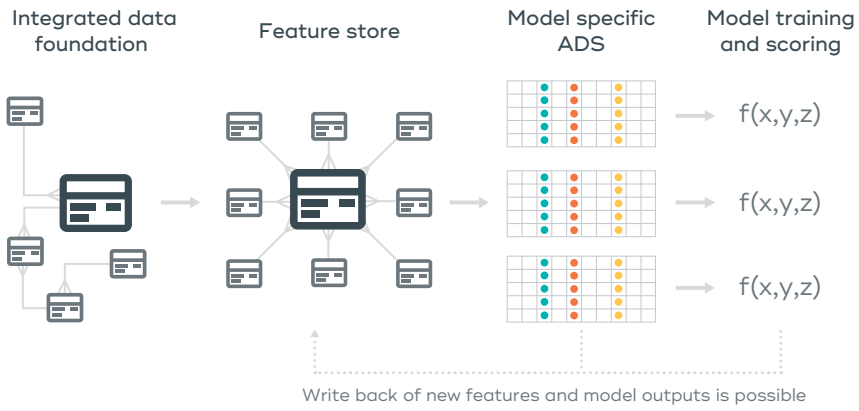
Analysts will play a large role in creating and defining the metrics and logic. Architects and data modelers will play a key role in designing and optimizing the processes. Database administrators will play a key role in automation, scheduling, and resource allocation for the ongoing processing.

As with database design in general, there will be a logical view of an EFS, as well as a physical implementation of that view. Logically, end users can think of an EFS as containing one row for each entity and a range of columns containing various metrics for that entity.

For example, there is one row per product, which looks like a traditional flat file. Each row contains a variety of columns with metrics related to that product. Another example might be a table with one row per individual customer that contains a wide range of sales, margin, and product purchasing information for each customer.

Physically, the data may or may not be stored in exactly that format. There may be several data tables containing only certain columns that are related to each other in some fashion and computed together. For example, one table may contain customer demographics, while another may contain customer behavioral data. Those two tables may be updated on a very different schedule. A view is used to join these two different types of customer information together so that the physical storage difference is transparent to the end user.

The feature store approach



- "Off the peg" features dramatically improve analytic cycle time and time-to-market
- Extensive re-use reduces TCO and improves analytic data quality and predictive model accuracy
- ADS layer enables model-specific customization, whilst eliminating analytic data silos
- Separation of duties & improved productivity

teradata.

Many aggregations within an EFS use data spanning a long period of time. The lack of up-to-the-minute information will have minimal, if any, impact on results in most cases. However, it is also possible to have the EFS generated on demand so it's fully current, although this has additional performance issues to consider. Even if an organization strongly feels that only up-to-the-second data should ever be used and that precomputing key metrics is not an option, the concepts of an EFS still apply.

In these cases, the EFS may be comprised of a series of views, macros, or stored procedures in addition to materialized tables that are set up to physically create the complete set of features as required at run time. Most of the advantages of a fully materialized EFS, such as standardized methodologies and reduced effort spent on data preparation, will still be realized. The only advantage lost is the compute once, use many times aspect

The final decision about how to best store the data is determined by a skilled data modeler and DBA. There are plenty of tricks that can be used to minimize storage requirements. Regardless of how data are stored, end users should be able to quickly approximate the logical flat file style view of the data outlined above from the physical tables via a series of views or other techniques.

To maintain its utility, an Enterprise Feature Store must be actively managed. Due consideration must be given not only to data freshness and the frequency of updates, but also to the removal of redundant features

To maintain its utility, an Enterprise Feature Store must be actively managed. Due consideration must be given not only to data freshness and the frequency of updates, but also to the removal of redundant features that are no longer being used. Precise cataloguing of all features is critical to prevent a feature store becoming a feature swamp. The dependencies between and within features (including data, processes, models and functions) must be clearly documented and understood. Users must be able to discover and understand the features in order to effectively utilize them. Finally, data in an EFS will often be a snapshot of conditions at a certain point in time, so care will need to be taken to ensure that end users fully understand precisely what is represented in the EFS.

Teradata Vantage

Many systems simply cannot handle the degree of data manipulation, complex joins, and full table scan processing that are required for generating and updating an Enterprise Feature Store. Instead they pull data off their systems on a regular basis and process it with an external tool, such as SAS. This common approach introduces new challenges including the need to develop and enforce procedures for pulling data off the host system and to invest in and allocate sufficient network bandwidth to execute these transfers.

Crucially, creating duplicate copies of core data that can quickly become obsolete creates data quality and integrity issues. Frequently relying on samples as opposed to using all the data because an inability to scale architecture undermines the value of many AI and machine learning projects. Any results found on

teradata.

the extracts in the original environment must eventually be replicated with live data at enterprise-scale if the organization is to get full benefit. Too many projects fail because something outstanding was found on a sample, but there was no feasible way to apply the findings back to the entire database

One major internet player maintains a large Enterprise EFS in their Teradata solution that allows them to access the data needed for new models within hours. They are now able to develop response models for new campaigns within days of execution.

When leveraging Teradata Vantage for your Enterprise Feature Store, these challenges are eliminated. Perhaps the single factor most affecting of an organization to maintain a successful Enterprise Feature Store is the ability of Teradata Vantage to process massive amounts of data in a scalable and timely fashion. Plus, using in-database functions and the advanced SQL Engine, it's possible to explore your data and generate all the logic required for the Teradata EFS.

The ability of Teradata Vantage to handle huge volumes of detailed data enables users to add variables and dimensions as needed without restructuring the underlying database—a step often required by other platforms. Given the capabilities of the Teradata Vantage analytics platform and the advanced processing features it possesses, it doesn't make sense to extract data and execute processing outside of the data warehouse environment, particularly when the processing involves joining large datasets and transforming them during the process of feature engineering.

Even when it is necessary to use an external tool, either to access a specific algorithm or because it is preferred by the team training the predictive model, that tool can now download only the data it requires from a ready-to-go source. Instead of downloading detailed data and aggregating them, the offline tool can simply access the Enterprise Feature Store.

This minimizes the impact of data transfer and eases the translation of any analysis into the production environment, since only the model training was run outside of Teradata Vantage. The scoring routine for the model is more easily translated into the production environment since the offline tool took data directly from the same EFS that the production scoring routine needs to use.

Using the Bring your own Model (BYOM) methodology and employing techniques such as PMML/Mleap import, in-database Python/R processing and SQL conversion, external models can be scored in-database directly against the Teradata EFS.

Get Ready To Be Data Driven

The importance of data and the role of machine learning and AI in organizational success in the coming years is well understood. However, deploying successful AI models that create value is hard. Inefficient processes, low productivity in data science teams, duplication of effort and mismatches between data and business teams consistently hinder the effective use of predictive analytics at scale. A piecemeal approach, point solutions and experiments constrained by systems incapable of handling the scale of data needed to create productive live systems, are depriving many of the opportunity to prosper with AI and machine learning.

teradata.

These pipeline approaches are putting organizations at risk today. Not only will they increasingly lose customers' business to those that can leverage analytics at scale to become truly data-driven and responsive, but they will not spot opportunities to transform and capitalize on new high growth markets. Without a well curated set of features many will find it too expensive, too slow and too complex to leverage their own data to make good business decisions. They risk losing market share, margin and revenue as a result.

Even those that have successfully used some AI or machine learning in their business without creating an enterprise feature store may be exposing themselves to unforeseen risk. As scrutiny of the use of algorithms increases from regulators, legislators and the public, a business must be able to transparently show how decisions were made. You don't want to be trying to unravel code written by a long-departed data scientist as regulators or the media ask you to justify your actions.

One well-known retailer has a Teradata EFS with 1,200 variables. The EFS was implemented as one component of an initiative that shortened model development from many weeks to days in most cases.

It provides immediate cost savings through more efficient processes and more productive use of data scientist's time. More fundamentally it provides the foundation to rapidly scale up the use of machine learning and AI across the organizations. More models can be trained and scored with live enterprise data to provide business users with new insights and faster automation of key processes.

Few analysts will complain about having to spend less time preparing data and more time doing value-added analysis. Few data scientists will complain about having another source of robust metrics available for inclusion in their models. Few data warehousing teams will complain about having their platform become more of a standard data source than it is today and minimizing the number of tools and applications that need to be supported for data management activities.

If your organization hasn't yet developed a Teradata Enterprise Feature Store architecture, you should consider doing so very soon. Market leaders in a wide range of industries have begun implementing these architectures. And as they begin executing more analyses as a result, they will further distance themselves from their competition. Why not become one of the organizations leading the pack instead of playing catch-up?

About Teradata

With all the investments made in analytics, it's time to stop buying into partial solutions that overpromise and underdeliver. It's time to invest in answers. Only Teradata leverages all of the data, all of the time, so you can analyze anything, deploy anywhere, and deliver analytics that matter most to your business. And we do it on-premises, in the cloud, or anywhere in between. We call this pervasive data intelligence. It's the answer to the complexity, cost and inadequacy of today's analytics. And how we transform how businesses work and people live through the power of data. Get the answer at **Teradata.com**.

17095 Via Del Campo, San Diego, CA 92127 Teradata.com

teradata.