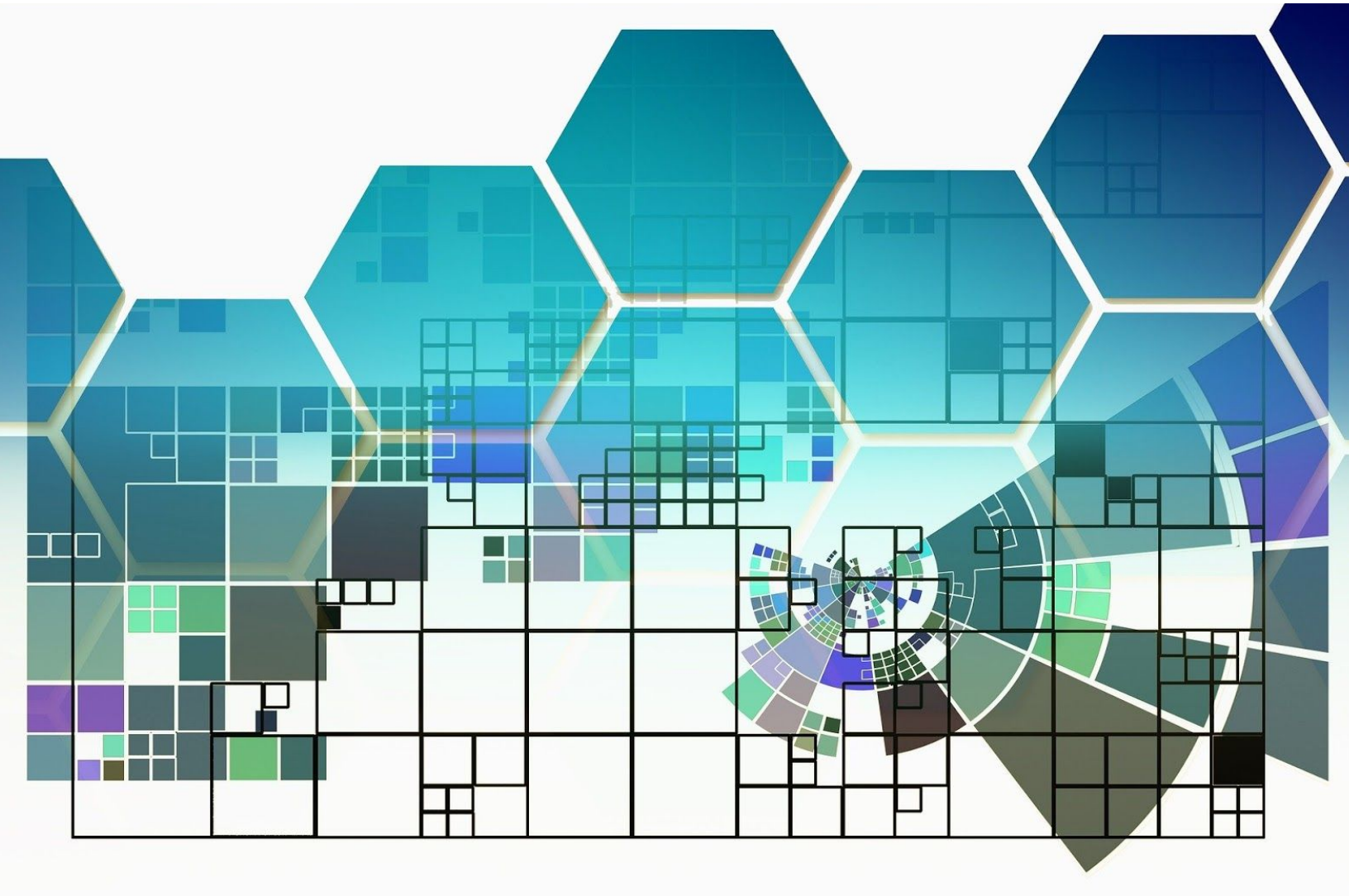
Cognilytica White Paper

AI Data Engineering Lifecycle Checklist

Following Steps for AI Project Success



Prepared for:

CLOUDERA

The Data Lifecycle for AI

Machine learning is powering most of the recent advancements in artificial intelligence including autonomous systems, computer vision, natural language processing, predictive analytics, and a wide range of applications among the seven patterns of AI. However, in order for these systems to be able to create accurate generalizations, these machine learning systems must be trained on data. The more advanced forms of machine learning, especially deep learning neural networks, require significant amounts of data to be able to create models with acceptable levels of accuracy. If machine learning systems are going to learn from this data, then this data needs to be clean, accurate, complete, and well-labeled so the resulting machine learning models are accurate. Whereas it has always been the case that garbage in is garbage out in computing, it is especially the case with regards to machine learning data.

Thus, the big challenge for organizations looking to make use of advanced machine learning models is getting access to large volumes of clean, accurate, complete, and well-labeled data to train their own internal models. Alternatively, organizations need to have access to relevant, high-quality third-party models. Regardless of whether organizations build their own models or get them from third-parties, the work must be done to make sure that the data behind the models is at the required levels of quality.

The Quality of Data Models Depends on the Quality of Training Data

- ★ Clean, curated data is needed to build, train and validate models
- ★ Clean, curated data is needed to provide accurate results
- ★ A model trained on bad data will result in bad results even if the input data fed into that model in production is good quality
- ★ Bad data fed into a model built with a subset of clean data will also result in garbage results
- ★ Paying attention to data science at the expense of data engineering will result in garbage
- ★ You can't build the DIKUW (Data, Information, Knowledge, Understanding, Wisdom) pyramid on a foundation of trash

It's important for companies to focus on the complete data lifecycle. While this may not always be the most fun part of enterprise machine learning, the data lifecycle is incredibly important to get right. The data lifecycle starts well before any machine learning models can be built, and includes things like data collection and ETL or ELT processes, as well as data pipeline curation, orchestration, and automation.

In fact, according to the *Data Engineering, Prep, and Labeling for AI 2020* report produced by Cognilytica, Data preparation and engineering tasks represent over 80% of the time consumed in most AI and Machine Learning projects. Based on interactions with a large number of end-user enterprises, agencies, and organizations, the vast amount of time spent in a typical machine

learning AI project is spent on identifying, aggregating, cleaning, shaping, and labeling data to be used in machine learning models.

The key way to resolve the race for quality data is by focusing on short, quick data engineering tasks aimed at iteratively and quickly getting prepared data, called the "sprint to usable data". The challenge is that just about every machine learning and AI project is different and the requirement for data preparation and labeling tasks depends on the complexity of the task and availability of information. This requires an iterative approach that leverages solutions and best practices across the domains of data engineering, data preparation, and data labeling.

And once the model is in use, the need to continually train the model does not go away. Just like how good, clean data was needed for model training, so too is this data needed for model re-training. As models are put into production the need to continually monitor, adjust, and fine tune these models is important. Real world data changes — often times faster than expected — resulting in the need for quality data management and engineering even after models are deployed. To remain accurate, models in use or at the inference phase need retraining, continuous model iteration, and more. This is where a well-structured and rigorous approach to data engineering enables faster workflows and higher fidelity in your AI projects.

Key Components of the Data Lifecycle

For companies looking to successfully manage the full data lifecycle there are a number of key steps they should take into consideration in order to achieve success. Key steps are listed below:

Data Engineering

Data Engineering is a core part of the data lifecycle. As mentioned above, Data preparation and engineering tasks represent over 80% of the time consumed in most AI and Machine Learning projects. But what exactly does Data Engineering entail? Data Engineering comprises all engineering and operational tasks required to make data available for analytics.

It is important to note that Data Engineering is not the data lifecycle itself, but rather comprises a set of activities required to move, manipulate, and manage data as necessary for different parts of the data lifecycle. For the purposes of this discussion, the aspects of data engineering that we refer to here are those tasks that are required to get data into the right format for AI and ML projects. These tasks include:

Data Ingestion

- Getting data out of source systems and ingesting it into a data lake.

Data Preparation & Cleansing

- Removing duplicates, cleaning data, assuring accuracy of data, augmenting and enhancing data

Data Transformation

- Formatting, transforming, and otherwise manipulating data to a desired state

Data Governance

- Systems necessary for data access control and data lineage

Performance Optimization

- Optimize the performance of data pipeline operations

Data and Production Orchestration

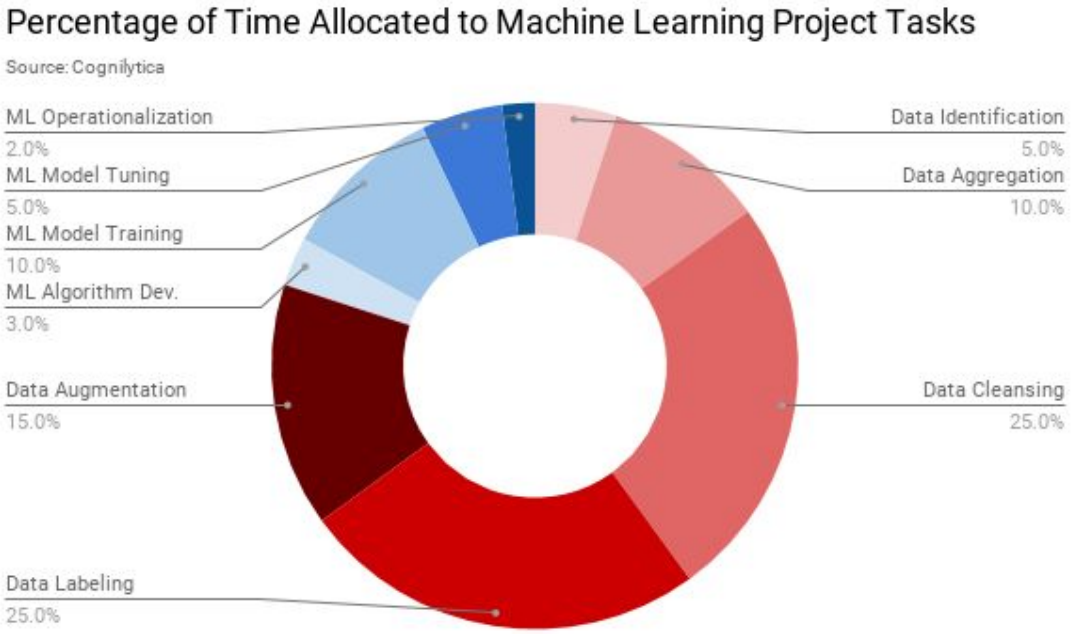
- One key aspect to modern data engineering is the complex data orchestration and automation of data pipelines. This includes the use of tools like Apache Airflow and APIs for delivering pipelines in multiple projects whether or not they are ML projects.
- The deployment of models into actual production systems, at scale

Data Engineering Frameworks

- Data engineering frameworks are the current state of the art for complicated and modern projects as AI and ML projects are. Spark, a cloud-native, containerized, and managed framework is being key to modern data engineering.

Data Engineering is all the piping and plumbing needed to get data from its original state, and into an end state. Data comes in from multiple sources including object store, file systems, EPR, CRMs, and databases. Data engineering cleans the data, performs feature engineering, transforms the data, and other necessary tasks to make it usable.

With all these tasks and functions that data engineering encompases, it should come as no surprise then that roughly 80% of all AI Projects are Data Engineering tasks. The graph below identifies the average proportion of time spent at various phases of an AI project:



Overview of the Data Engineering Lifecycle

Data Preparation and curation is also a very necessary step in the data lifecycle. In the context of machine learning, data preparation solutions are most concerned with making sure that the data being fed into and used to train machine learning models are clean, accurate, complete, and relevant for machine learning purposes. Specifically, AI-relevant data preparation steps include the following:

Select Data

➔ You want to make sure that appropriate data is selected. Make sure to have a rationale for inclusion or exclusion of certain data.

Clean Data

➔ Data that is collected from various sources usually doesn't arrive in a clean format. Cleaning data includes standardizing formats across different data sources, replacing or deleting incorrect data, anonymizing data, reducing all forms of data noise

Construct Data

➔ Derived Attributes Generated Records

Integrate Data

➔ You want to merge data that has been collected for various sources.

Format Data

➔ In order to have machine learning systems properly use your data, you'll want to standardize formats across different data sources (data types, fields, matched formats, currency or metric conversions, etc.).

Dataset

➔ You want to make sure that your datasets reflect current, accurate information, not old, obsolete, or out of date information that can taint resulting models. If using a sub-section of your data make sure you have a description of the dataset used.

Data Collection & Acquisition Steps

Data, and in particular big data, is fueling AI. In fact, the explosion of big data, and the know-how and infrastructure to deal with it, is what's making the AI resurgence happen. In the past, AI growth slowed because of limited data sets, lack of representative sample data vs. real-time actual data, and the inability to deal with tons of data. However, today companies have real-time, limitless, always available data with power to handle machine learning.

Big data has taught us how to handle the 4 V's of data including:

Volume

★ This is how to handle terabytes of existing data at rest

Velocity

★ This is how to quickly handle data in motion including streaming data

Variety

★ This is how to handle data in many forms including structured data, unstructured data, text data, multimedia, and more

Veracity

★ This is how to deal with data in doubt including data inconsistency, incomplete data, ambiguous data, data latency, and model approximations

Data Acquisition

As a collection of technologies, data engineering represents infrastructural, data-centric systems and solutions that are primarily oriented towards the movement, manipulation, and operation on big data sets. Data engineering technologies primarily emerge from their roots in

Extract-Transform-Load (ETL) solutions that have existed for many decades. In particular, the core features of ETL that are relevant to AI include:

- ➢ Extracting relevant data from existing data stores, data lakes, data warehouses, and other repositories of structured and unstructured information to be used in machine learning model training data sets or in support of those machine learning models.
- ➢ Transforming that data using rules or other combination logic to conform to the requirements of machine learning model creation, support, and maintenance.
- ➢ Loading the transformed data into the required data store and format to support machine learning model generation, training, and maintenance.

Additionally, big data processing for many companies has moved to the cloud. This means that rather than housing and storing data in-house, ETL data as well as real-time streaming data is all pushed in whatever form to the cloud. This allows companies to have flexibility, agility, simplification of operation, better reliability, and security.

Data Aggregation

One of the key steps in data acquisition is the combination of information from multiple sources in order to provide adequate data as needed to train machine learning models. This involves aggregating information from multiple sources of structured and unstructured data as well as the merging of data that might exist in different states of validity.

Key data aggregation steps include:

- ➔ Identification of sources of structured and unstructured data to support machine learning needs
- ➔ Determination of missing information that can be merged from multiple sources
- ➔ Determination of varying data quality levels from multiple data sources and rules for merging of data
- ➔ Creation of data pipelines to facilitate data aggregation and merging

Data Cleansing Steps

When using data to train machine learning models, data cleansing is a very necessary step. In particular its important to follow these steps when cleaning your dats:

Formatting

- Standardizing formats across different data sources (data types, fields, matched formats, currency or metric conversions, etc.)

Replacing incorrect data

- ➔ Any obviously incorrect information should be replaced during the cleansing phase

Enhancing / Augmenting Data

- ➔ Add additional dimensions with pre-calculated amounts. Aggregate information as needed.
- ➔ Enhance with third-party data
- ➔ "Multiply" image-based data sets if there aren't sufficient for training

Removing extraneous information and de-duplication

- ➔ Remove irrelevant data from training to improve results. Including irrelevant pixels from images

Noise reduction and Disambiguation

- ➔ Reduce all forms of data noise: information noise, visual noise, audio noise.

Data anonymization

- ➔ If personally identifiable information (PII) is not needed, remove it before feeding to models

Normalizing Data

- ➔ Standardize data values over regions that make training more effective and efficient

Data sampling

- ➔ For very large data sets, look at extracting a sizable representative subsample for ML training

Data Selection & Sampling Steps

When selecting data for machine learning, there are two sorts of data filtering needed before you can use this information for ML purposes. The first is Data Sampling, and in particular for very large data sets. The second is Data Attribute pruning to reduce overall size and data complexity.

Data Sampling

For very large data sets, you'll want to extract a sizable representative subsample for ML training. If you have a very large dataset, such as Terabytes, Petabytes, or more, then you don't need all of that data for training purposes. You'll want to select a representative, significant sized sample that is well-balanced in terms of data and devoid of unintentional or intentional informational biases.

Data Attribute Pruning

For the records and data you're planning on using, you don't necessarily need every data field, attribute, or metadata. You'll want to remove unnecessary data that bloats the data set and complicates training that adds no additional value to the ML model

Data Labeling & Annotation Steps

In order for machine learning approaches to work, and in particular Supervised Learning, they must be fed clean, well-labeled data that the system can use to learn from example. However, most data collected from companies in unstructured data. In fact, up to 80%-90% of the content generated is in the form of unstructured data which is any data that's not organized in a way that computers can easily process the information. Examples of unstructured data include emails, documents, images, videos, social media posts, and a wide range of documents in paper and converted forms such as invoices, purchase orders, communications, contracts, applications, IDs, meeting notes, and contracts.

Data Labeling Methods and Approaches

If most of the data at a company is unstructured, then how do you get labeled data? There are a few ways to approach this:

Internal, Self-Managed Human Labor

- You can use your own internal workforce to do labeling

Outsourced, Self-Managed Human Labor

- You can use third-party providers to provide contract labor but you manage the work pool and quality of output.

Third Party Managed Labeling Providers

- You can use third parties that specialize in labeling data that provide a labeling workforce as well as manage workers and quality of work output.

User-Driven Labeling

- For companies that have a very large user base such as Google, Facebook, Amazon, Netflix, and other similarly large companies they are having their users do image and other labeling work for them. Think for example, CAPTCHA, and the user may not always be aware they are providing data labeling efforts.

Pre-Trained Models and Existing Labeled Data Sets

- Access already trained machine learning models that can be extended via Transfer Learning and other approaches, or access the underlying labeled data set to provide a

starting point for machine learning projects. Existing labeled data sets include ImageNet, MNIST, Visual Genome, many government and non-profit sources. Many vendors offer pre-trained models for specific application domains.

Labeled data is simply data where a label of the required classification or the final data output that you're looking to train on has been applied. For instance, if the image recognition algorithm must classify types of vehicles, these types should be clearly and accurately defined and labeled in a dataset. For obvious reasons, there are no ways to automate labeling unlabeled training data. If you could, then you wouldn't need Supervised Learning to classify these images (they'd be automatically classified otherwise!).

Data labeling takes much time and effort as datasets sufficient for machine learning may require thousands, millions, billions, or more data records to be labeled.

Alternate Methods for Data Labeling

In addition to the ways outlined above for different options on how to get your data labeled, companies are also turning to different ways to find usable data. Some of these tricks include:

- **Repurposing existing data**
  - Companies are looking to see if they already have an explicit or implicit categorization for training data. If so, that data may potentially be repurposed.
- **Harvest from online sources**
  - If you are able to find data sources that you can trust you can apply that data for your machine learning model
- **Existing labeled data set**
  - Depending on what you're trying to have your machine learning model learn, there are already existing labeled data sets that are available. Examples of these include ImageNet, MNIST, Visual Genome, as well as many government and non-profit sources.
- **Use Third-party Models**
  - There are companies out there that specialize in offering pre-trained models for specific domains.

Data Enhancing Steps

Since many companies practice the approach of storing data in the state it was created or collected, sometimes they need to add additional dimensions to the data. Or, if they are using third party datasets, sometimes additional enhancement is needed. Also, for certain situations companies may "multiply" image-based data sets if there aren't sufficient for training. This is done by taking their existing image-based data sets and flipping it, gray scaling it, zooming in, and zooming out to generate additional images and angles for data they know is good, clean, and labeled.

- **General Enhancing**

- Sometimes, adding necessary data is important for models to be accurately trained.

➔ **Filtering**

- Sometimes, it's important to filter data to remove outlier data and help reduce bias in the data.

➔ **Feature Engineering**

- This can be done to help assist with overall enhancement of the data set. Companies can use their domain knowledge to extract features from data to improve the performance of machine learning algorithms.

Data Splitting Steps

Once you have gone through the steps to get Clean, well-labeled, datasets for your machine learning model, you need to now split this data.The three subsets for Splitting Data Sets include: Training, Validation, and Test sets. When splitting the datasets, you want to make sure you're using random subsampling. The breakdown is usually as follows:

➔ **Training Datasets**

- This is the primary data that's used to train the model and find the optimal parameters to accomplish the machine learning goal
- Usually around 70% of the dataset is allocated for training purposes

➔ **Validation Dataset**

- Validation sets are used to tweak a model's hyperparameters, which are fundamental operational settings that can't be directly learned from data. The number and type of hyperparameters depend on the specific algorithm used.
- Usually around 15% of the dataset is allocated for validation

➔ **Test Dataset**

- Test datasets are needed to evaluate the model to see if it generalizes well for future, unseen data. Usually around 15% of the dataset is allocated for testing.

It's important to use different subsets for training and testing to avoid model overfitting, which is the incapacity for generalization In general, the more training data, the better the potential model will perform. However, without validation, the model might overfit or underfit data, which will be reflected in test data. Without test data, you don't know if the model will work!

Data Training and Retraining Pipelines

Once you've prepped and cleaned your data, and split the data to use for training, you're ready to actually use your model. Once your model is in the inference phase, it doesn't mean that training and retraining is done. In fact, it's very important to monitor models in the inference phase. In this phase you need to:

➔ **Optimize models**

- Prune decision-trees and neural networks

- Look at parts of the neural network that don't get activated after it's trained. These sections just aren't needed and can be "pruned" away.

➔ **Continue to measure prediction errors**
- Perhaps real-world data isn't matching test data well

➔ **Optimize performance and efficiency**

Retraining, Troubleshooting, and Debugging Pipelines

Machine learning data pipelines are not only used during initial training -- they are also important during model iteration. Over time, models that might perform with adequate accuracy, performance, precision, and other metrics can decay in those measures over time. This is known as "model drift", which is the problem of the model providing decreasing accuracy and performance over time due to changes in real-world data.

Sometimes model drift is caused by changes in underlying data or the quality of that data over time. ML engineers and data scientists know this problem well and as such include in their approaches methods by which models can be continuously iterated and modified. As such, the data engineering lifecycle needs not only include the pipeline for training the initial model but continuous, iterative pipelines for ongoing model versioning and iteration.

In addition, another reason why models that might make incorrect predictions is because an automated data pipeline has issues with data collection, manipulation, or faulty data. Rather than retraining the model, the input data needs to be changed in some manner. Proactive monitoring of the data pipeline is key to fixing issues of model performance.

➔ **Development of Retraining Pipelines**
- Iterative, continuous pipelines for retraining existing models and factoring in new data sets

➔ **Debugging and Troubleshooting Pipelines**
- Keeping information flows efficient and effective is key to successful iterative model development
- Successful AI projects will provide the ability to monitor data pipelines proactively, and the ability to then easily troubleshoot data pipeline issues including visual troubleshooting.

➔ **Implementation of "ML Ops" approaches for continuous model monitoring and management**
- Tools, processes, and methodologies for handling the continuous management and monitoring of ML models in the operational environment

➔ **Governance frameworks for handling models that fall out of organizational compliance**
- Tools, processes, and methodologies for dealing with models that might no longer meet organizational needs, run into compliance or regulatory considerations, or other factors that require organizations to re-evaluate the use of a given model.

Taking the Next Step

To learn how to unlock the power of data engineering at scale for analytics success, read Cloudera's Taking Your Data Lifecycle to the Next Level eBook

Related Research

- Data Engineering, Prep, and Labeling for AI 2019 Report (CGR-DE100)
- Data Engineering, Prep, and Labeling for AI 2020 Report (CGR-DLP20)
- Data prep and Labeling Infographic (CGIG053)
- AI Today Podcast, Data Prep & Labeling (Podcast #086)

About Cognilytica

Artificial Intelligence (AI) and related technologies will impact all industries and all corners of the world. Without insight into how AI will impact you and your business, you risk being left behind. Cognilytica is an analyst firm that provides real-world, industry and adoption focused market research, intelligence, advisory on Artificial Intelligence (AI) and related areas.

- Cutting through the Hype by Focusing on Adoption — Cognilytica cuts through the noise to identify what is really happening with adoption and implementation of AI in public, private, and academic settings. We focus on the usage of AI in the real world, not the buzzword hype.
- Industry-Leading Market Research — Market-level research on application, use cases, and comparative research on the state of AI adoption in the industry. Focusing on real-world adoption of AI technology and cutting-edge application.
- Advisory with Knowledgeable Experts — Get access to knowledgeable research analysts that spend their time immersed in the world of AI implementation and adoption.
- Research through Conversation — Cognilytica generates its research through direct conversation with industry thought-leaders, technology practitioners, and business decision-makers. We ignore the press releases and skip the hype to produce unique, original research through direct engagement.
- Bootcamp and Training Opportunities — A three day "fire hose" of information that prepares you to succeed with your AI & ML Project Management efforts, whether you're just beginning them or are well down the road with implementation. Cognilytica's training is the only public course that Cognilytica offers, reflecting the best thinking and research that Cognilytica produces.

Cognilytica analysts publish research reports, white papers, and briefing notes at regular intervals that are available to our annual subscribers as well as for one-off purchase. Cognilytica offers advisory time with analysts virtually or on-site Analysts are also available for commissioned research projects, white papers for internal or external consumption, and speaking engagements at client events or public conferences. If you have an AI market intelligence or research need that can be fulfilled with our industry knowledge, body of research, methodology, and expertise, Cognilytica analysts are available to assist. Visit www.cognilytica.com for more information on opportunities you can take advantage of.