CHAPTER 40

# Google Cloud Dataflow

Google Cloud Dataflow provides a serverless, parallel, and distributed infrastructure for running jobs for batch and stream data processing. One of the core strengths of Dataflow is its ability to almost seamlessly handle the switch from processing of batch historical data to streaming datasets while elegantly taking into consideration the perks of streaming processing such as windowing. Dataflow is a major component of the data/ML pipeline on GCP. Typically, Dataflow is used to transform humongous datasets from a variety of sources such as Cloud Pub/Sub or Apache Kafka to a sink such as BigQuery or Google Cloud Storage.

Critical to Dataflow is the use of the Apache Beam programming model for building the parallel data processing pipelines for batch and stream operations. The data processing pipelines built with the Beam SDKs can be executed on various processing backends such as Apache Apex, Apache Spark, Apache Flink, and of course Google Cloud Dataflow. In this section, we will build data transformation pipelines using the Beam Python SDK. As of this time of writing, Beam also supports building data pipelines using Java, Go, and Scala languages.

## Beam Programming

Apache Beam provides a set of broad concepts to simplify the process of building a transformation pipeline for distributed batch and stream jobs. We'll go through these concepts providing simple code samples:

- A Pipeline: A Pipeline object wraps the entire operation and prescribes the transformation process by defining the input data source to the pipeline, how that data will be transformed, and where the data will be written. Also, the Pipeline object indicates the distributed processing backend to execute on. Indeed, a Pipeline




E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_40