Team ID: BD2_343_387_429_910

Project title Chosen :

- Spark Streaming for Machine Learning
    - Spam Detection

Design details:

- Continuous input data stream received from TCP socket as Discretized Stream - **DStream** which stores it in Spark's memory for processing. Internally it is represented as continuous series of RDDs. Each RDD (resilient distributed dataset) in a DStream contains data from a certain interval. An RDD is a collection of elements partitioned across the nodes of the cluster that can be operated on in parallel.
- **Spark NLP** is a Natural Language Processing (NLP) library built on top of Apache Spark ML. It provides simple, performant & accurate NLP annotations for machine learning pipelines that can scale easily in a distributed environment. Spark NLP comes with 1100+ pretrained pipelines and models in more than 192+ languages. It supports nearly all the NLP tasks and modules that can be used seamlessly in a cluster.
- Document Assembler is the entry point for every Spark NLP pipeline as it creates the first annotation of type Document. Tokenizer separates a piece of text into smaller units called tokens which can be words, characters or subwords. Normalizer removes all dirty characters from text following a regex pattern and transforms words based on a provided dictionary.
- Lemmatizer finds lemmas out of words with the objective of returning a base dictionary word. StopWordsCleaner takes a sequence of strings as input and drops all the stop words from it. The Finisher outputs annotation values into a string.
- Term frequency-inverse document frequency (TF-IDF) is a feature vectorization method used in text mining to reflect the importance of a term to a document in the corpus.**Naive bayes** is a classification technique based on Bayes' Theorem with an assumption of independence among predictors.

Surface level implementation details about each unit:

- When running a Spark Streaming program locally, always use "local[*n*]" as the master URL, where *n* > number of receivers to run. Streaming context is initialised and socketTextStream is used to read streaming data from socket into DStream.
- We then process each RDD parallelly wherein the first step is to convert the json object into a dataframe. Each row in a batch is read as a separate column, and the features (Subject, Message, Spam/Ham) are extracted. We then present the data in required dataframe format using union.
- The Spam/ham column is further encoded to 0 or 1 to make processing easier. Special symbols are removed from the columns.Using Spark NLP's pipeline we then perform tokenization, normalization, lemmatization and stop words removal.

- TF and IDF are implemented in HashingTF and IDF. HashingTF takes an iterable as the input. Each record could be an iterable of strings or other types. Naive Bayes is used as the incremental model from scikit learn. Partial fit is used to achieve this.

Reason behind design decisions:

- Spark NLP is used because it delivers scalable, high-performance and high-accuracy NLP-powered software for real production use and provides a unified solution for all our NLP needs. It takes advantage of transfer learning and implements the latest and greatest algorithms and models in NLP research.
- Tf-idf is one of the best metrics to determine how significant a term is to a text in a series or a corpus. tf-idf is a weighting system that assigns a weight to each word in a document based on its term frequency (tf) and the reciprocal document frequency (tf) (idf). The words with higher scores of weight are deemed to be more significant.
- Naive Bayes classifier has a very important role in this process of filtering e-mail spam. The quality of performance Naive Bayes classifier is based on dataset used. Datasets that have fewer instances of e-mails and attributes can give good performance for Naive Bayes classifier. Naive Bayes classifier also can get highest precision.

Key Takeaways:

We went in depth into spark streaming and realised just how interesting and well structured of a technology it was.
Apache Spark is **RDD — Resilient Distributed Dataset**. RDD contains an arbitrary collection of objects. There are two ways to create RDDs: parallelizing an existing collection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem.
Natural Language processing is an expansive field filled with innovation and we’ve just barely scratched the surface with techniques such as stemming, lemmatization and when to use them.
Naive Bayes classifier is used as an incremental learner for the project and TFIFD is used to pick out important words from the stream
The training dataset should not overfit the model and we assume predictors to be independent in order to use naive bayes.