**Pembelajaran mendalam terdistribusi dengan Horovod dan Azure Databricks**

Di Azure Databricks, Anda melakukan pembelajaran mendalam terdistribusi dengan menggunakan kerangka kerja sumber terbuka Horovod di atas TensorFlow, Keras, dan PyTorch.

**Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan mampu:

* Pahami apa itu Horovod dan bagaimana Horovod dapat membantu mendistribusikan model pembelajaran mendalam Anda.
* Gunakan HorovodRunner di Azure Databricks untuk pembelajaran mendalam terdistribusi.

1. **Pengantar**

Azure Databricks mendukung kerangka kerja Horovod Uber untuk menjalankan pekerjaan pelatihan pembelajaran mendalam yang terdistribusi di Spark.

Dalam modul ini, Anda akan belajar cara mendistribusikan pembelajaran mendalam di Azure Databricks dengan HorovodRunner.

**Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan mampu:

* Pahami apa itu Horovod dan bagaimana Horovod dapat membantu mendistribusikan model pembelajaran mendalam Anda.
* Gunakan HorovodRunner di Azure Databricks untuk pembelajaran mendalam terdistribusi.

1. **Memahami Horovod**

Horovod dapat membantu ilmuwan data saat melatih model pembelajaran mendalam. Sebelum kita dapat menjelajahi Horovod, mari kita tinjau apa itu pembelajaran mendalam dan apa masalahnya saat melatih model pembelajaran mendalam.

1. **Ulasan singkat tentang pembelajaran mendalam**

Pembelajaran mendalam adalah subbidang pembelajaran mesin dan mengacu pada model yang terdiri dari beberapa lapisan, juga dikenal sebagai jaringan saraf dalam. Proses pelatihan dimulai dengan pengiriman data ke lapisan input dalam **batch**.

Data dianalisis oleh lapisan input dan diteruskan ke lapisan berikutnya sampai mencapai lapisan output dan menghasilkan prediksi. Prediksi dibandingkan dengan nilai aktual yang diketahui dan berdasarkan hasil ini, bobot, dan nilai bias direvisi untuk memperbaiki model.

Batch diproses oleh jaringan melalui beberapa perulangan atau **periode**. Setiap periode, model mencoba untuk lebih meningkatkan prediksi dengan memperbarui nilai bobot dan bias.

1. **Pembelajaran mendalam dengan Azure Databricks**

Dalam Azure Databricks, kita dapat melatih model pembelajaran mendalam menggunakan kerangka kerja sumber terbuka populer untuk Python: TensorFlow, PyTorch, dan Keras. Saat kami menggunakan salah satu kerangka kerja simpul tunggal ini untuk melatih model pembelajaran mendalam, kita harus menggunakan kluster simpul tunggal di Azure Databricks.

Model pembelajaran mendalam mendapat keuntungan dari memiliki lebih banyak data: semakin banyak data, semakin besar kemungkinan kita akan mendapatkan model yang lebih baik atau lebih akurat. Meskipun disarankan untuk melatih model pembelajaran mendalam pada kluster simpul tunggal, model atau data Anda mungkin terlalu besar untuk disimpan dalam memori satu mesin. Sebuah kluster simpul tunggal yang tidak cukup adalah masalah yang mungkin dihadapi para ilmuwan data ketika melatih model pembelajaran mendalam. Untungnya, Horovod dapat membantu dalam skenario ini.

1. **Memahami Horovod**

Horovod adalah kerangka pelatihan terdistribusi sumber terbuka dan merupakan alternatif untuk melatih model pada kluster simpul tunggal. Horovod memungkinkan ilmuwan data untuk mendistribusikan proses pelatihan dan memanfaatkan pemrosesan paralel Spark.

Karena model pembelajaran mendalam berisi lapisan yang perlu diproses secara berurutan, dan menggunakan hasil perantara untuk meningkatkan model di akhir periode, pemrosesan paralel model pembelajaran mendalam dapat dengan cepat menjadi rumit. Horovod dirancang untuk mengurus manajemen infrastruktur sehingga para ilmuwan data dapat fokus pada model pelatihan.

Horovod dinamai tarian tradisional di mana mitra berpegangan tangan sambil menari dalam lingkaran. Horovod berutang nama ini karena memungkinkan simpul pekerja untuk berkomunikasi dengan simpul pekerja lain, untuk menghindari kemacetan di simpul driver.

Saat Horovod digunakan di atas salah satu kerangka kerja pembelajaran mendalam (TensorFlow, PyTorch, atau Keras), Horovod melatih beberapa model pada batch himpunan data input yang berbeda pada pekerja yang terpisah. Dengan kata lain, beberapa model dilatih secara paralel pada pekerja terpisah menggunakan subset data yang berbeda.

Pada akhir periode, bobot dikomunikasikan antara pekerja dan bobot rata-rata semua pekerja dihitung. Selanjutnya, periode baru dapat mulai menggunakan bobot rata-rata baru dan sekali lagi, beberapa model dilatih secara paralel.

**Tip**

Temukan informasi selengkapnya tentang **Horovod di sini**. : <https://github.com/horovod/horovod>

Baca tentang **Horovod di Azure Databricks** dalam dokumentasi di sini. : <https://learn.microsoft.com/id-id/azure/databricks/machine-learning/train-model/distributed-training/>

1. **HorovodRunner untuk pembelajaran mendalam terdistribusi**

Anda ingin melatih model pembelajaran mendalam menggunakan kerangka kerja sumber terbuka TensorFlow, Keras, atau PyTorch di Azure Databricks. Anda mencoba menggunakan kluster simpul tunggal tetapi model atau data Anda terlalu besar untuk kluster. Dalam hal ini, Anda dapat memilih untuk menggunakan Horovod di atas pekerjaan Anda sejauh ini untuk mendistribusikan pelatihan. Mari kita jelajahi cara melakukannya menggunakan **HorovodRunner**.

1. **Memicu Horovod di Azure Databricks dengan HorovodRunner**

HorovodRunner adalah API umum, yang memicu pekerjaan Horovod. Keuntungan menggunakan HorovodRunner daripada kerangka kerja Horovod secara langsung yaitu HorovodRunner telah dirancang untuk mendistribusikan pekerjaan pembelajaran mendalam di seluruh pekerja Spark. Karenanya, HorovodRunner lebih stabil untuk pekerjaan pelatihan pembelajaran mendalam yang sudah berjalan lama di Azure Databricks.

1. **Proses Horovod**

Untuk mendistribusikan pelatihan model pembelajaran mendalam menggunakan HorovodRunner, Anda harus melakukan hal berikut:

* Siapkan dan uji kode simpul tunggal dengan TensorFlow, Keras, atau PyTorch.
* Migrasikan kode ke Horovod.
* Gunakan HorovodRunner untuk menjalankan kode dan mendistribusikan pekerjaan Anda.

1. **Menjalankan pelatihan simpul tunggal**

Sebelum bekerja dengan Horovod dan HorovodRunner, kode yang digunakan untuk melatih model pembelajaran mendalam harus diuji pada kluster simpul tunggal. Setelah berfungsi, pastikan untuk menyelesaikan prosedur pelatihan utama menjadi satu fungsi Python. Fungsi ini akan digunakan nanti untuk memulai eksekusi terdistribusi atas kode Anda.

1. **Bermigrasi ke Horovod**

Setelah Anda menguji kode simpul tunggal untuk melatih model pembelajaran mendalam, Anda harus memigrasikannya ke Horovod sebelum Anda dapat memicu pekerjaan dengan HorovodRunner.

1. Impor kerangka kerja Horovod sebagai hvd.
2. Inisialisasi pustaka Horovod dengan hvd.init().
3. Sematkan satu GPU per proses. Penyematan diperlukan untuk menonaktifkan pemetaan acak pekerja dan menghindari bentrokan. Penyematan dilompati saat menggunakan CPU.
4. Tentukan bagaimana Anda ingin mempartisi atau mengambil sampel data sehingga setiap pekerja menggunakan subset data yang unik untuk melatih model. Sebagai praktik terbaik, pastikan subset memiliki ukuran yang sama. Tergantung pada himpunan data input, ada beberapa teknik untuk melakukan pengambilan sampel. Misalnya, Anda dapat menggunakan Petastorm untuk bekerja dengan himpunan data dalam format Apache Parquet. Pelajari lebih lanjut tentang pustaka sumber terbuka Petastorm di sini.
5. Skala tingkat pembelajaran dengan jumlah pekerja untuk memastikan bobot disesuaikan dengan benar setelah setiap periode.
6. Gunakan pengoptimal terdistribusi Horovod untuk menangani komunikasi antar pekerja.
7. Siarkan parameter awal sehingga semua pekerja memulai dengan parameter yang sama.
8. Simpan titik pemeriksaan hanya pada pekerja 0 untuk mencegah konflik antar pekerja.
9. **Menggunakan HorovodRunner**

Untuk menjalankan HovorodRunner, Anda harus membuat instans **HorovodRunner** di mana Anda menentukan berapa banyak simpul (didefinisikan oleh argumen **np**) yang Anda inginkan untuk mendistribusikan pekerjaan Anda. Anda dapat menentukan untuk menggunakan satu simpul jika Anda ingin menguji pada kluster simpul tunggal dengan **np=-1.** Terakhir, Anda dapat memicu pekerjaan pelatihan Horovod dengan HorovodRunner dengan menjalankan fungsi Python yang Anda buat untuk kode pelatihan Anda.

**4. Latihan**

Sekarang, ini adalah kesempatan Anda untuk mendistribusikan pembelajaran mendalam di Azure Databricks.

Dalam latihan ini, Anda akan:

* Jalankan pelatihan simpul tunggal dengan PyTorch dan migrasikan ke Horovod.
* Distribusikan pelatihan model dengan HorovodRunner.

1. **Instruksi**

Ikuti petunjuk berikut untuk menyelesaikan latihan:

* Buka petunjuk latihan di <https://aka.ms/mslearn-dp090.> \ <https://microsoftlearning.github.io/dp-090-databricks-ml/>
* Selesaikan latihan Pembelajaran mendalam terdistribusi dengan Azure Databricks.

**5. Ringkasan**

Dalam modul ini, Anda belajar cara mendistribusikan pembelajaran mendalam di Azure Databricks.

Sekarang setelah Anda menyelesaikan modul ini, Anda dapat:

* Pahami apa itu Horovod dan bagaimana Horovod dapat membantu mendistribusikan model pembelajaran mendalam Anda.
* Gunakan HorovodRunner di Azure Databricks untuk pembelajaran mendalam terdistribusi.