Agenda

1) Deep Learning and Real Time World
2) Deep Learning and the Apache Kafka Ecosystem
3) Building Neural Networks with TensorFlow and H2O
4) Deployment of Neural Networks with Kafka Streams
Agenda

1) Deep Learning and Real World Applications
2) Deep Learning and the Apache Kafka Ecosystem
3) Building Neural Networks with TensorFlow and H2O
4) Deployment of Neural Networks with Kafka Streams
... allows computers to find hidden insights without being explicitly programmed where to look.
Real World Examples of Deep Learning

Language Translation

Picture Analysis (Face, Location, Items, ...)

Speech and Video Generation

Intelligent Decision Making

http://wellroundedfashion.com/2015/01/13/trendy-tech-camouflage-from-facial-detection/
https://www.technologyreview.com/s/602796/starcraft-will-become-the-next-big-playground-for-ai/
Deep Learning to Analyze and Act on Critical Business Moments

- Fraud Detection
- Price Optimization
- Cross Selling
- Customer Service
- Transportation Rerouting
- Predictive Maintenance
- Inventory Management

Windows of Opportunity

- Seconds
- Minutes
- Hours
Live Demo

Deep Learning in Action

http://playground.tensorflow.org/
Agenda

1) Deep Learning and Real World Applications
2) Deep Learning and the Apache Kafka Ecosystem
3) Building Neural Networks with TensorFlow and H2O
4) Deployment of Neural Networks with Kafka Streams
Apache Kafka Ecosystem and Deep Learning

- Kafka Connect
- Go / .NET / Python
- Kafka Producer
- Rest Proxy
- Schema Registry
- Kafka Streams

Production ML App → Kafka → Model Building

- Model Params
- Training Data

Output

Feature Data Input

DB → App → DB → App → DB

Kafka Streams

KSLQ
Agenda

1) Deep Learning and Real World Applications
2) Deep Learning and the Apache Kafka Ecosystem
3) Building Neural Networks with TensorFlow and H2O
4) Deployment of Neural Networks with Kafka Streams
Languages, Frameworks and Tools for Deep Learning

There is no Allrounder!
Deep Learning with H2O.ai

H2O Engine

R / Python / Scala / Flow UI

Java Code
H2O Deep Water (TensorFlow, MXNet, ...)

Deep Water
(H2O + TensorFlow)

Pre-Defined Networks +
User-Defined Networks

Use Case: Airline Flight Delay Prediction

Machine Learning Algorithm: Deep Learning using Neural Networks

Technology: H2O.ai, TensorFlow
1) Deep Learning and Real World Applications
2) Deep Learning and the Apache Kafka Ecosystem
3) Building Neural Networks with TensorFlow and H2O
4) Deployment of Neural Networks with Kafka Streams
Stream Processing

REQUEST/RESPONSE

DATA AT REST

DATA IN MOTION

Apache Kafka and Deep Learning – Kai Waehner
Stream Processing Pipeline

- **Stream Ingest**
  - Messaging
  - APIs
  - Integration
  - Adapters / Channels

- **Stream Preprocessing**
  - Normalization
  - Filtering
  - Transformation
  - Enrichment
  - Aggregation

- **Stream Analytics**
  - Contextual Rules
  - Windowing
  - Patterns
  - Analytics
  - Machine Learning
  - ...

- **Stream Outcomes**
  - Index / Search
  - Analytics / DW Reporting
  - Process Management
  - Applications & APIs
  - Analytics (Real Time)

**Applying an Analytic Model is just a piece of the puzzle!**
Kafka Streams (shipped with Apache Kafka)

Map, filter, aggregate, apply analytic model, “any business logic”

Input Stream (Kafka Topic) → Stream Processing Microservice (Kafka Streams) → Output Stream (Kafka Topic)

Deployed Anywhere
Java App, Docker, Kubernetes, Mesos, “you-name-it”
When to use Kafka Streams for Stream Processing?
A complete streaming microservices, ready for production at large scale

```java
public static void main(String[] args) throws Exception {
    Properties config = new Properties();
    config.put(StreamsConfig.APPLICATION_ID_CONFIG, "wordcount-example");
    config.put(StreamsConfig.BOOTSTRAP_SERVERS_CONFIG, "kafka-broker:9092");
    config.put(StreamsConfig.KEY_SERDE_CLASS_CONFIG, Serdes.String().getClass().getName());
    config.put(StreamsConfig.VALUE_SERDE_CLASS_CONFIG, Serdes.String().getClass().getName());

    KStreamBuilder builder = new KStreamBuilder();
    KStream<String, String> textlines = builder.stream("TextlinesTopic");
    KStream<String, Long> wordCounts = textlines
                           .flatMapValues(value -> Arrays.asList(value.toLowerCase().split("\W+")))
                           .groupBy(key -> word)
                           .count("Counts")
                           .toStream();
    wordCounts.to(Serdes.String(), Serdes.Long(), "WordsWithCountsTopic");

    KafkaStreams streams = new KafkaStreams(builder, config);
    streams.start();
}
```
Use Case: Airline Flight Delay Prediction

Machine Learning Algorithm: Neural Network built with H2O and TensorFlow

Streaming Platform: Apache Kafka and Kafka Streams
H2O.ai Model + Kafka Streams

1) Create H2O DL model

```
// Create H2O object (see deeplearning_fe7c1f02_08ec_4070_b784_c2531147e451.java)
hex.genmodel.GenModel rawModel;
rawModel = (hex.genmodel.GenModel) Class.forName(modelClassName).newInstance();
EasyPredictModelWrapper model = new EasyPredictModelWrapper(rawModel);
```

2) Configure Kafka Streams Application

```
// Configure Kafka Streams Application
final String bootstrapServers = args.length > 0 ? args[0] : "localhost:9092";
final Properties streamsConfiguration = new Properties();
// Give the Streams application a unique name. The name must be unique
// in the Kafka cluster
// against which the application is run.
streamsConfiguration.put(StreamsConfig.APPLICATION_ID_CONFIG, "machine-learning-example");
// Where to find Kafka broker(s).
streamsConfiguration.put(StreamsConfig.BOOTSTRAP_SERVERS_CONFIG, bootstrapServers);
```

3) Apply H2O DL model to Streaming Data

```
@RequiredArgsConstructor
final class StreamingData {
    // Private fields
    private final String label;
    private final String flightInput;
    private final String flightDepTime;
    private final String origin;
    private final String dest;
    // Getters and setters
}

public void apply(StreamingData data) {
    // Transform data
    long delay = calculateDelay(data);
    // Write transformed data to Kafka
    kafkaProducer.send(new ProducerRecord<>("delayed-flights", offset, data));
}
```

4) Start Kafka Streams App

```
// Start Kafka Streams Application to process new incoming messages from Input Topic
final KafkaStreams streams = new KafkaStreams(builder, streamsConfiguration);
stream.cleanUp();
stream.start();
```
Github Examples: Kafka + Deep Learning

https://github.com/kaiwaehner/kafka-streams-machine-learning-examples

1) `git clone` ... 2) `mvn clean package` ... 3) look at implementations and unit tests
Online Model Training with Apache Kafka and Kafka Streams

How to improve models?
1. Manual Update
2. Automated Batch
3. Real Time

Your choice... All possible with Kafka!
Caveats for Online Model Training

- Processes and infrastructure not ready
- Validation needed before production
- Slows down the system
- Only a few ML implementations → Build your own!
- Many use cases do not need it

→ Do it only when really needed!
→ Data Scientist and Developers have to work together continuously (org + tech!)
→ Mission critical, scalable production deployment is key for success of Machine Learning projects
→ Apache Kafka Ecosystem for Batch and Real Time Machine Learning (Training, Inference, Monitoring)
Use SQL-like queries to write stream processing applications with Kafka Streams

```sql
SELECT STREAM
  CEIL(timestamp TO HOUR) AS timeWindow, productID, COUNT(*) AS hourlyOrders, SUM(units) AS units
FROM Orders
GROUP BY CEIL(timestamp TO HOUR), productID;
```

<table>
<thead>
<tr>
<th>timeWindow</th>
<th>productID</th>
<th>hourlyOrders</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>08:00:00</td>
<td>10</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>08:00:00</td>
<td>20</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>09:00:00</td>
<td>10</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>09:00:00</td>
<td>40</td>
<td>1</td>
<td>45</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Leverage Kafka and Kafka Streams from anywhere without coding

- Streaming ETL
- Anomaly Detection
- Real Time Monitoring
- etc.

→ ... maybe also use it to apply analytic models via user defined functions? 😊
Questions? Feedback? Please contact me!

Kai Waehner
Technology Evangelist
kontakt@kai-waehner.de
@KaiWaehner
www.kai-waehner.de
LinkedIn