Perfect, Bhadra Rao 👌 — understood.

We’ll avoid mentioning SAP or Port-Out applications explicitly (we’ll keep it generic — like “decision event” or “fraud evaluation window”) and keep the focus purely on the data science and modeling journey.

Here’s your final 15-minute presentation script, re-written for clarity, confidence, and storytelling impact — ready for you to deliver at the Data Science Summit.

---

🎤 Title Slide: Detecting Fraud in Real Time — Sequence-Based Feature Management and Modeling in Telecom

> “Good morning everyone, I’m Bhadra Rao, and I’m excited to share our work on

‘Detecting Fraud in Real Time using Sequence-Based Feature Management and Modeling in Telecom.’

Telecom fraud isn’t random — it evolves through sequences of customer and system activities.

Our goal was to transform those raw event logs into meaningful behavioral signals that help us anticipate fraud in near real time.”

---

🧭 Slide 1: Agenda (0:30 sec)

> “Here’s the flow for today:

1️⃣ Business Motivation & Challenge

2️⃣ Data Selection & Preparation

3️⃣ Sequence-Based Feature Engineering

4️⃣ Model Integration & Results

5️⃣ Key Insights & Q&A.”

---

💡 Slide 2: Business Motivation & Challenge (1.5 min)

> “Fraud is dynamic — static attributes alone can’t capture behavior over time.

In telecom, even a fractional improvement in detection accuracy translates into massive financial impact and stronger customer trust.

The challenges we face are:

• Extremely high-volume, complex event logs

• A rare-event rate — only about 0.33 % fraud

• And noisy signals, where both legitimate and fraudulent actions mix together

So our key question was —

How can we convert noisy, high-volume sequential logs into explainable signals that detect fraud patterns in real time?”

---

📊 Slide 3: Data Source — Telegence Channel (1.5 min)

> “We used the Telegence Channel, which captures memo-type activities — essentially event logs that record every account action.

Each event is linked by two identifiers — the Billing Account Number (BAN) and an Application ID representing a customer interaction session.

We started with around 250 unique memo types.

Using a chi-square test against fraud labels, we selected the top 13 memo types that were statistically most predictive — for example ESCC, 1100, EQUIP, CPCS, and so on.

This reduced dimensionality while preserving the strongest behavioral signals.”

---

⚙️ Slide 4: Pre-Processing Pipeline (48-Hour Window) (2 min)

> “Before creating features, we built a structured pipeline with a 48-hour observation window for every BAN–Application pair.

The steps are straightforward:

1️⃣ Join the base table with Telegence logs using BAN and Application ID.

2️⃣ Select all events within 48 hours prior to the decision event.

3️⃣ Filter down to the 13 selected memo types.

4️⃣ Sort events from oldest → latest.

5️⃣ Remove repeated memo types at the same timestamp.

6️⃣ Compute the time gap = decision time – event time (in seconds).

7️⃣ Finally, build ordered arrays: memo\_type\_ordered and time\_gap\_array.

Example: For BAN\_101,

• Memo Types = [ESCC, 1100, CPCS]

• Time Gaps = [18000, 7200, 1800] seconds.

These arrays capture the temporal rhythm of account activity.”

---

⏱️ Slide 5: Statistical Time-Gap Features (Transformation Features) (1.5 min)

> “From those time-gap arrays, we extracted statistical features that describe temporal spread and intensity.

Metrics include:

min\_time\_gap – shortest delay between actions

max\_time\_gap – longest idle period

median\_time\_gap – central tendency

stddev\_time\_gap – variability

order\_length – number of distinct events

For example:

• BAN\_101 → gaps [18000, 7200, 1800] → median 7200 sec, stddev 6710.

• BAN\_202 → [36000, 10800, 1800] → stddev 15100 → more irregular pattern.

These features quantify how consistent or chaotic the user’s behavior is.”

---

🔣 Slide 6: Token Rarity (IDF-Style Feature) (1.5 min)

> “Next, we used an IDF-style rarity measure inspired by text analytics.

The idea is simple — rare events are more likely to be abnormal.

Formula:

\text{rarity} = \log\_2!\left(\frac{N}{1+df}\right)

where df is the number of BANs containing a memo type.

We mapped every memo type to its rarity and took the average per sequence.

For instance, if CPCS occurs in very few BANs, its rarity is high — contributing to a higher avg\_token\_rarity.

This helped surface sequences containing unusual activity combinations.”

---

🔁 Slide 7: Transition Probabilities & Sequence Likelihood (2 min)

> “We then modeled how events transition from one type to another.

Step 1: Count all pairwise transitions among the 13 memo types.

Step 2: Convert them into conditional probabilities, P(next | current).

Step 3: From these, compute three sequence-level metrics:

prob\_product – product of transition probabilities

geo\_mean – geometric mean

prob\_entropy – a measure of unpredictability

Example for BAN\_101 with [ESCC, 1100, CPCS]:

prob\_product = 0.044, geo\_mean = 0.21, prob\_entropy = 1.22

High entropy → more randomness → potentially anomalous behavior.”

---

🧠 Slide 8: Bi-LSTM Autoencoder (Anomaly Feature) (2.5 min)

> “To capture deeper structure, we used a Bi-LSTM Autoencoder.

It’s trained only on non-fraudulent sequences so it learns normal patterns.

During inference, if the model cannot reconstruct a sequence well — it means the pattern deviates from normal.

Example:

BAN Input Seq Reconstructed AE Error

101 ESCC\_1100\_CPCS ESCC\_1100\_CPCS 0.00

202 CNTA\_ELG\_EQUP CNTA\_ELG\_X 0.52

303 EQUIP\_CPCS EQUIP\_X 0.71

Higher error ⇒ stronger deviation from normal sequences, and potential fraud.

This unsupervised score became a powerful additional feature for our classifier.”

---

📈 Slide 9: Model Performance — Impact of Sequence Features (2 min)

> “Let’s see how these features impacted our models.

Baseline LightGBM with 15 tabular features achieved:

Precision = 0.019

Recall = 52.6 %

AUC = 0.896

AUC-PR = 0.048

After adding sequence-based features:

Precision improved by 24 % (to 0.0235)

Recall jumped by 37 % (to 72.3 %)

AUC rose to 0.935 (+4.4 %)

AUC-PR doubled to 0.096

These are strong gains for rare-event fraud detection.

The Bi-LSTM hybrid helped analyze patterns, but LightGBM remained the most efficient for deployment.”

---

🔚 Slide 10: Key Insights & Closing (1.5 min)

> “To wrap up:

We transformed raw telecom event logs into structured behavioral features.

We captured temporal spread, event rarity, and sequence likelihood — all critical behavioral signals.

The Bi-LSTM Autoencoder highlighted structural anomalies unseen by static models.

Combined, these features boosted precision, recall, and AUC for fraud detection.

Our next goal is to integrate this framework into a real-time streaming pipeline — detecting fraud as it happens.

Thank you — I’m Bhadra Rao, and I welcome your questions.”

---

🕐 Duration Summary (≈ 15 min Total)

Section Time

Opening & Agenda 1 min

Business Motivation 1.5 min

Data Source 1.5 min

Pre-Processing Pipeline 2 min

Time-Gap Features 1.5 min

Token Rarity 1.5 min

Transition Probabilities 2 min

Bi-LSTM Autoencoder 2.5 min

Model Performance & Closing 1.5 min

Total ≈ 15 min

---

Would you like me to now prepare a speaker-note version (with on-screen cues like “hover over table,” “pause here,” “point to BAN\_202,” etc.) so you can practice exactly how to speak and gesture during the Summit?