Fraud detection without feature engineering

Pamela Vagata

stripe

Stripe

A global technology company that builds economic infrastructure for the internet

Help more companies get started and thrive, and ultimately grow the GDP of the internet

Fraud Prevention

Classical Machine Learning

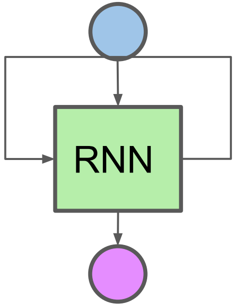
- Easy to understand modelling techniques
- Models-as-Data
  - ML Production Systems can be runtime independent from training/offline experimentation systems
- Feature Engineering
  - increasingly complicated features
  - Manual, arduous process
  - Complex infrastructure: aggregates, joins
    - ~consistency across online/offline

Learning Behavioral Patterns

- Fraudulent intent is latent in the observable behavioral patterns
- Model Engineering rather than feature engineering:
  - Simplify the feature complexity while improving prediction accuracy
  - Moving the toil of feature engineering from the ML engineer to the model itself
- Can we engineer a model that learns to predict fraudulent intent from raw behavioral sequences?

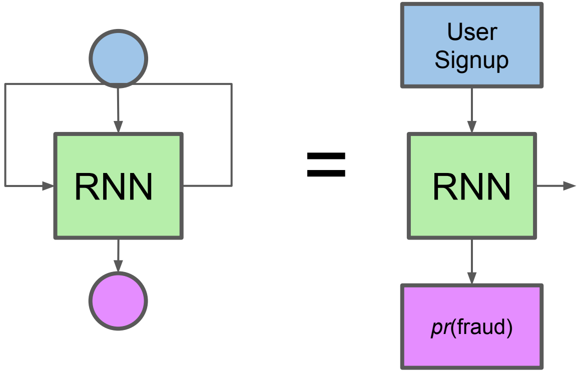
Recurrent Neural Networks

- Family of Neural Networks
  - Recurrent Loop
  - Models Sequential Data



Recurrent Neural Networks

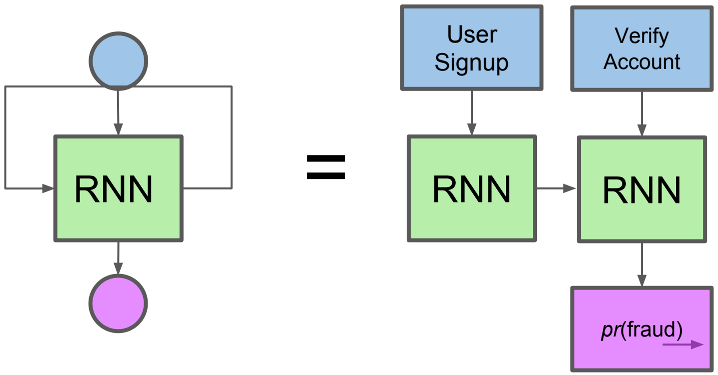
- Family of Neural Networks
  - Recurrent Loop
  - Models Sequential Data



stripe

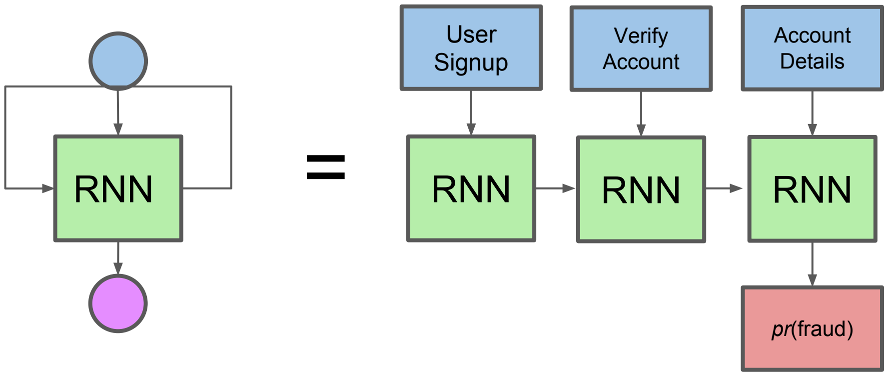
Recurrent Neural Networks

- Family of Neural Networks
    - Recurrent Loop
    - Models Sequential Data



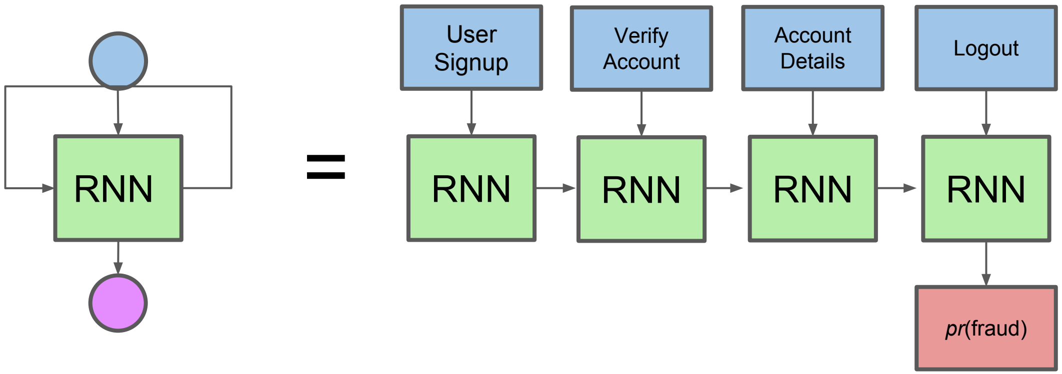
Recurrent Neural Networks

- Family of Neural Networks
  - Recurrent Loop
  - Models Sequential Data



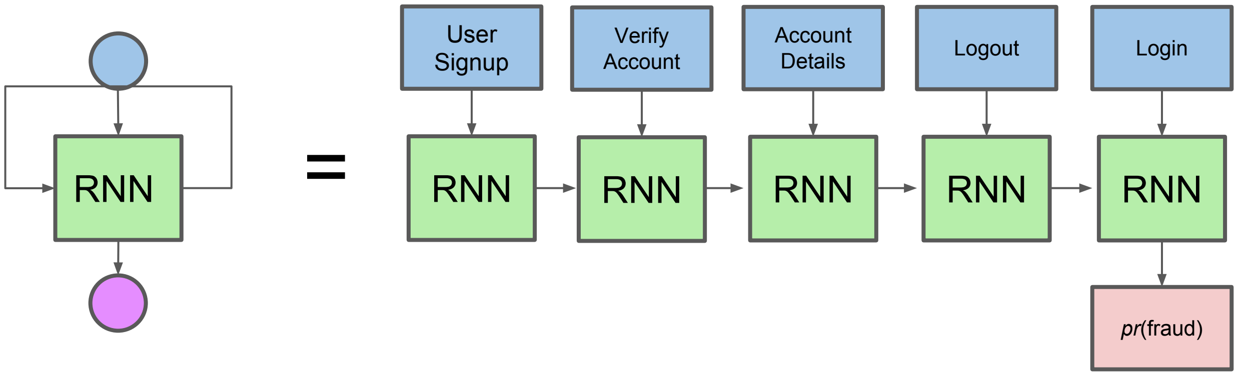
Recurrent Neural Networks

- Family of Neural Networks
  - Recurrent Loop
  - Models Sequential Data



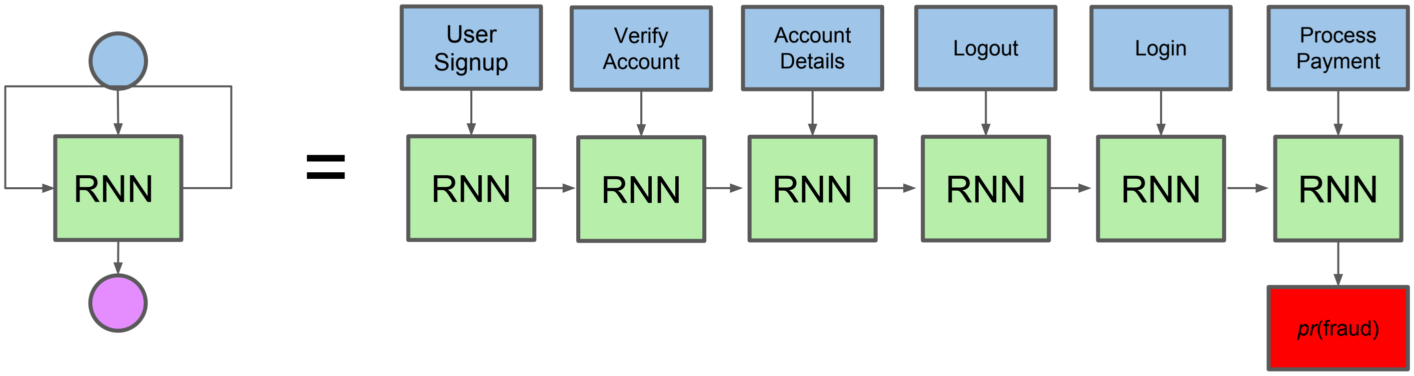
Recurrent Neural Networks

- Family of Neural Networks
  - Recurrent Loop
  - Models Sequential Data

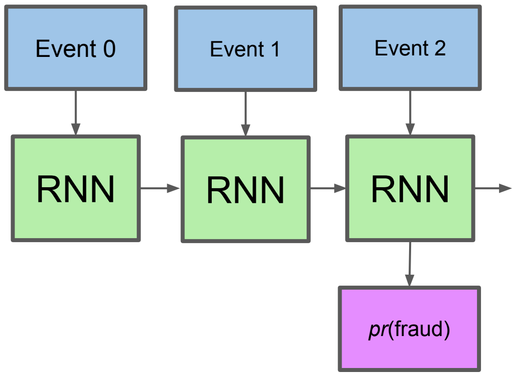


Recurrent Neural Networks

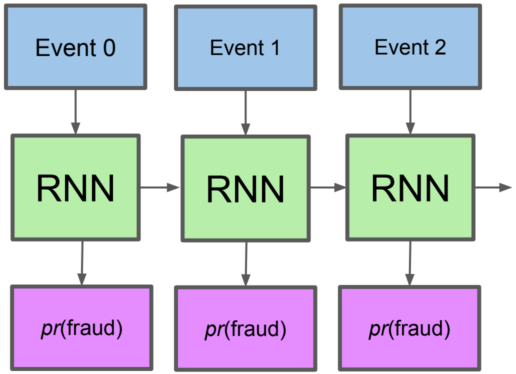
- Family of Neural Networks
    - Recurrent Loop
    - Models Sequential Data



Different Labels, Different Flavors



Many-to-one



Many-to-many

Encoding Event Sequences

EVENT 0

event_type: “signup”

timestamp: “2017-12-30 00:00:00”

event_metadata_categorical

event_metadata_numeric

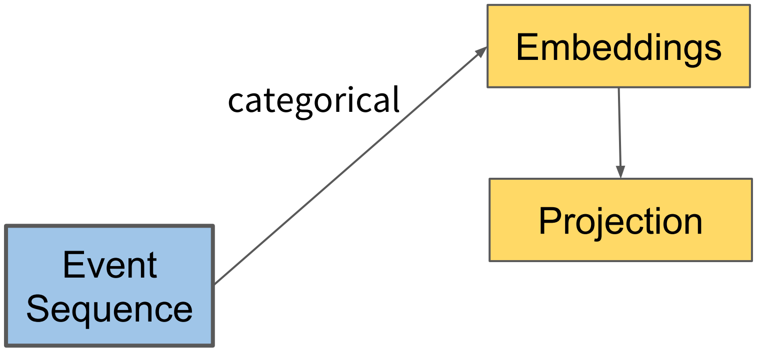
- Encode event types as categorical values
- Timestamps: delta-encoded
- Categorical metadata:
  - Map distinct categorical values to a vocabulary
  - Jointly trained embeddings for categorical values
- Numeric metadata:
  - Scaled between 0 and 1

stripe

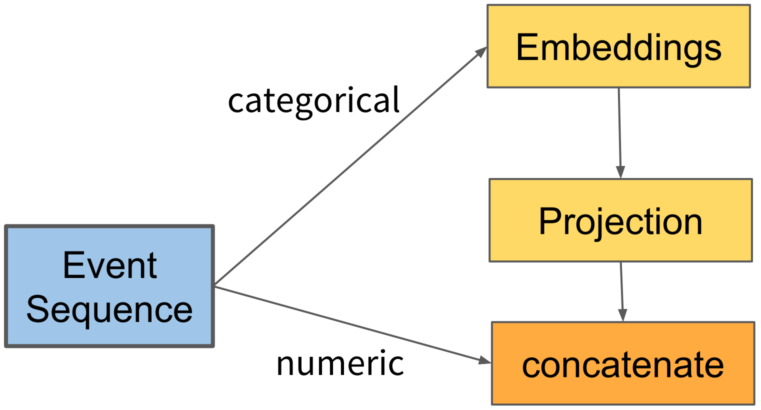
Training with Events

- Data Pipeline: leverage Spark to produce training data
- Serialized to parquet
- Sequence of events where each event is
  - Array of categorical_vocabulary_index
  - Array of numeric values
- Labels
  - For each event (if available)
  - For each sequence
- Deserialized with pyarrow library
  - Handles complex datatypes such as arrays

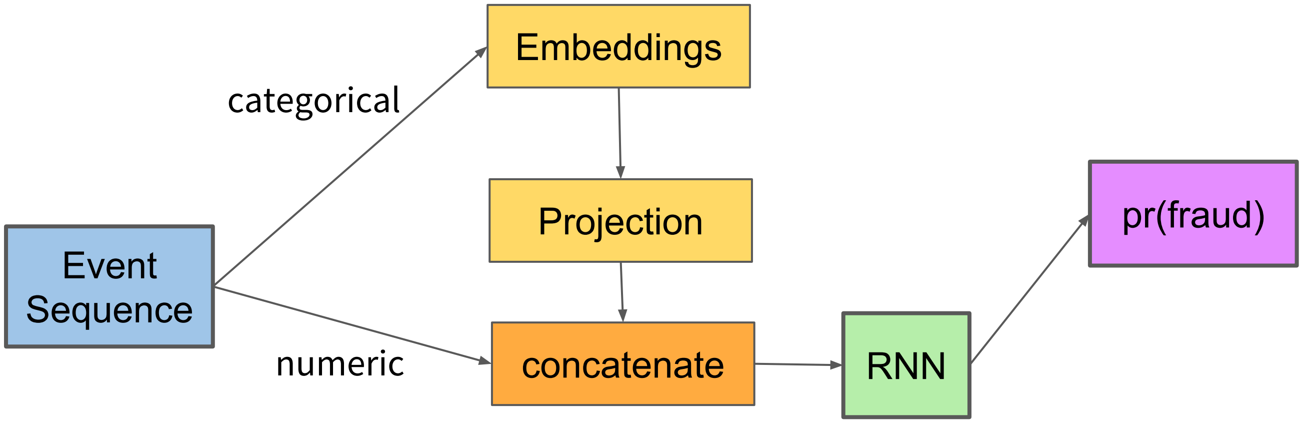
Model Architecture



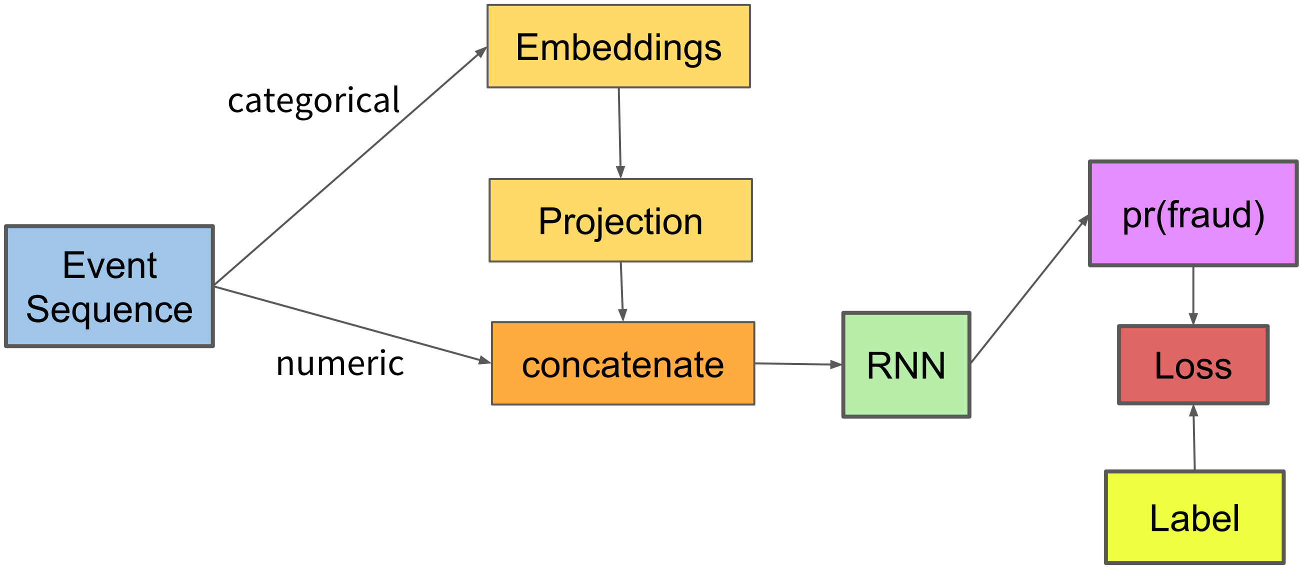
Model Architecture



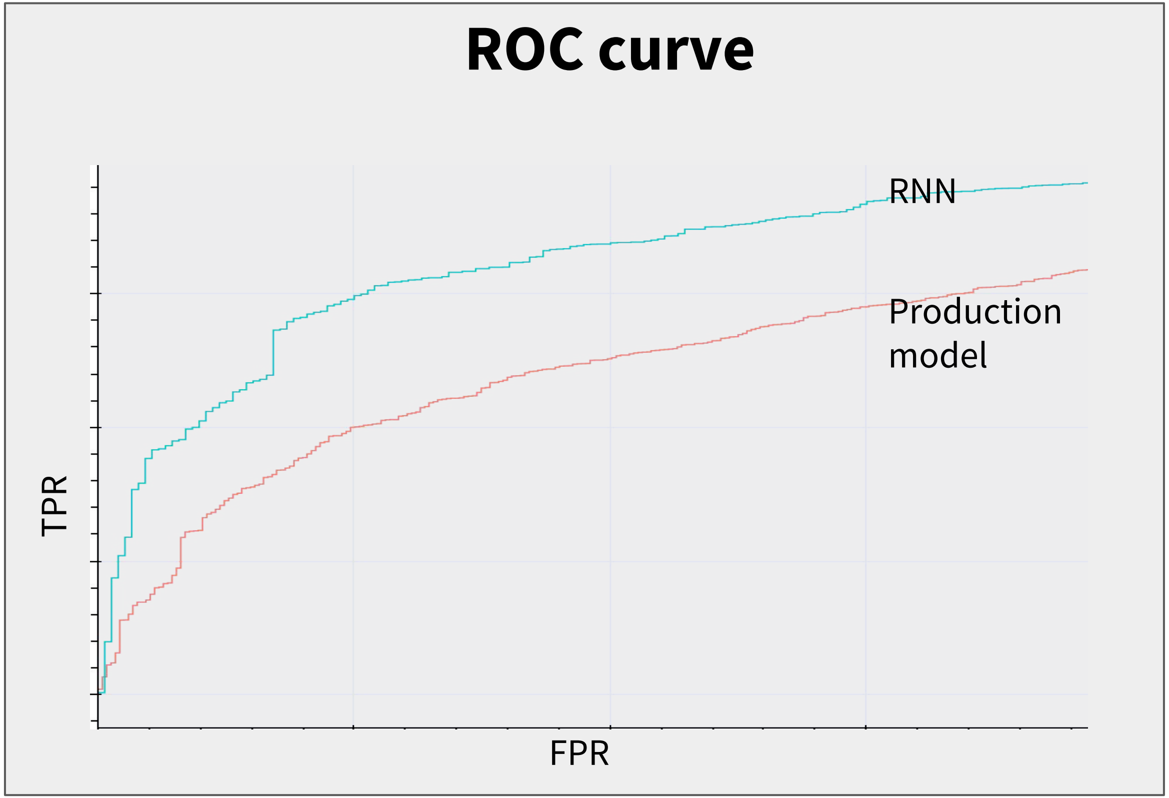
Model Architecture



Model Architecture



Results



Some Future Work

- Explanation models
- Applying Attention
- Applications to other modeling challenges at Stripe
  - f(event_sequence) => Pr(label)
- Data/Models that don’t fit in memory
  - Distributed/parallel training/serving
- Future challenges span:
  - Systems/infrastructure work
  - ML/modelling work

Key takeaways

- Combine deep learning and minimally transformed metadata from real-world events
    - Leverage computationally powerful hardware
    - Shifting compute from data-engines to modeling hardware
    - Reduce the complexity of feature-transformation data pipelines
- Mindset shift
    - From: combining features into increasingly complex features
    - To: raw signals that capture highest information related to the ML task
- Model Engineering
    - Design a model architecture for the ML task and available data

Thank you!

stripe