Risk Analytics Using Knowledge Graphs / FIBO with Deep Learning

A conversation with:

Thomas Cook
Director of Sales, AnzoGraph DB
Cambridge Semantics

Greg Steck
Director, Data Engineering
FI Consulting

Sponsor:

Guest Speaker from:
Moderated by **Mike Meriton**
Co-Founder & COO, EDM Council

- Joined EDM Council full-time 2015 to lead Industry Engagement
- EDM Council Co-Founder & First Chairman (2005-2007)
- Former CEO GoldenSource (2002-2015)
- Former Executive for D&B Software and Oracle
- FinTech Innovation Lab – Executive Mentor (2011 – Present)
Thomas Cook is Director of Sales at Cambridge Semantics. He has a Masters in Computer Science from Texas State University and brings 20+ years experience in Software Engineering, ETL, Data Warehousing and Big Data. Prior to joining Cambridge Semantics, Thomas worked at SAS, IBM, Netezza and Talend.
Guest Speaker **Greg Steck**
Director of Data Engineering, FI Consulting

Greg Steck is the Director of FI Consulting’s Data Engineering practice that helps financial institutions understand their complex data problems and develop solutions to optimize performance and cost. He leads technical teams to develop and implement cloud solutions in the areas of portfolio credit risk management, model risk management, stress testing and risk analytics. Greg holds an M.B.A. from Thunderbird School of Global Management.
Risk Analytics using FIBO / Knowledge Graph with Deep Learning

October 21, 2020
Motivation

Gartner reported that close to 50% of AI and machine learning projects fail in part due to challenges with "data complexity and systems integration"
3 key data challenges for financial institutions

1. Integrating data silos
2. Staging data for frequently changing business requirements
3. Capturing data provenance
Addressing data complexity with a knowledge graph

Key Benefits:

• Support regulatory compliance by natively capturing detailed data lineage
• Reducing time spent on data engineering tasks by over 50%
• Overcome limitations of machine learning
• Enable self-service by connecting business terms to data elements
• Turn tacit knowledge into explicit knowledge
Knowledge graph can add value to many different use cases

• Risk Analytics
• Model Risk Management
• Customer 360
• Anti-Money Laundering
• Fraud Detection
Case study: Using Fannie Mae public data to build a deep learning model

Use loan-level data, macroeconomic data including house price index and unemployment rate from the FRED database, to build a neural network model to predict the loan outcomes.

Layer in COVID daily active cases for reporting / analytics

Description of datasets

• Single family mortgage loans
• 40 million loans
• 20 years of monthly snapshot history = 2bn monthly observations
• Unemployment and HPI sourced from Federal Reserve Bank of St. Louis database (FRED)
• COVID daily cases by FIPS
Business Problems for Risk Analytics
1. Integrating internal and external disparate datasets

Data scientists and analysts spend most of their time on data preparation rather than on generating insights.
2. Tracking how data moves and transforms from source systems to end data product

The people challenged by lack of data provenance:

• The modeling team manually tracking data lineage for BCBS239 compliance
• The developer releasing an application change without understanding downstream data dependencies
• The business analyst spending hours tracking files for internal audit request
3. Staging analytics-ready data for frequently changing business requirements

- Loan Performance
- Acquisitions
- Macro Variables

Complex ETL process with lots of business logic

Denormalized table that meets specific business requirements
More business problems

• Lack of common data fields and schemas that make sharing data difficult across departments and teams
• Producing and maintaining quality model risk management data at low cost
• Long throughput time to generate analytics
• Complex IT infrastructure to produce and execute models
Contrasting relational database and graph approaches across the risk analytics pipeline

Each step in the pipeline presents unique data challenges that graph can either solve or significantly improve. We’ll discuss each step and compare relational database and graph approaches.
A. Designing the data model

A. Design Data Model
B. Create script to ingest and store data
C. Stage data for analytics and modeling
D. Train Neural Network Model
E. Forecast Scenarios and Reporting
Challenges designing a relational data model
Benefits of using FIBO to model graph data
Graphs better reflect real-world relationships
FRED Data Deep Dive - Categories

Category API Response
{
  "id": 125,
  "name": "Trade Balance",
  "parent_id": 13
}
FRED Data

House Price Index
Appleton, WI

Unemployment Rate
Appleton, WI

Appleton, WI

Loan Data

Loan-12345

Property-6789

MSA-11540

Appleton, WI
Create links between disparate data sets

House Price Index
Appleton, WI

Unemployment Rate
Appleton, WI

Appleton, WI

San Antonio, TX

San Antonio-New Braunfels, WI

Loan-12345

Property-6789

MSA-11540

MSA-41700

FRED Data Deep Dive
Relational approach needs to revisit join logic each time

<table>
<thead>
<tr>
<th>Series Name</th>
<th>Category Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unemployment Rate in Appleton, WI</td>
<td>Appleton, WI</td>
</tr>
<tr>
<td>House Price Index in Appleton, WI</td>
<td>Appleton, WI</td>
</tr>
<tr>
<td>Per Capital Personal Income in Appleton, WI</td>
<td>Appleton, WI</td>
</tr>
<tr>
<td>Unemployment Rate in San Antonio, TX</td>
<td>San Antonio, TX</td>
</tr>
<tr>
<td>House Price Index in San Antonio, TX</td>
<td>San Antonio, TX</td>
</tr>
<tr>
<td>Per Capital Personal Income in San Antonio, TX</td>
<td>San Antonio, TX</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MSA Label</th>
<th>MSA ID</th>
<th>Loan ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appleton, WI</td>
<td>11540</td>
<td>A</td>
</tr>
<tr>
<td>Appleton, WI</td>
<td>11540</td>
<td>B</td>
</tr>
<tr>
<td>San Antonio-New Braunfels, TX</td>
<td>41700</td>
<td>C</td>
</tr>
<tr>
<td>San Antonio-New Braunfels, TX</td>
<td>41700</td>
<td>D</td>
</tr>
</tbody>
</table>

FRED Data Deep Dive
Consistently link disparate datasets for reuse

Relationship created on 8 Oct 20 to correctly link the loan MSA to FRED MSA categories

FRED Data Deep Dive
Query disparate data in few lines with graph patterns

FRED Data Deep Dive

Query disparate data in few lines with graph patterns

FRED Data Deep Dive

FRED Data Deep Dive
B. Create script to ingest and store data
Comparison between graph and relational approaches for data ingestion vary depending on tooling

Depends on largely on what type of data is being integrated, what frequency it is available, etc.

Our focus will be on challenges associated with integrating data from the FRED API into our graph database
Overhead associated with creating script to ingest data
Graph model can better accommodate different types of data

```
{
    "seriesID": UNRATE,
    "categoryID": 4563,
    "date": "2019-01-01",
    "value": "50",
    "revisedDate": "2020-02-01"
}
```
C. Staging for analytics and modeling

A. Design Data Model
B. Create script to ingest and store data
C. Stage data for analytics and modeling
D. Train Neural Network Model
E. Forecast Scenarios and Reporting
Data needs to be denormalized to improve query speed and combine disparate datasets.
Capture data provenance in another silo
Creates multiple views for similar business requirements

Table for Business Requirements X

Table for Business Requirements Y

Table for Business Requirements Z

Creating a model for business requirements can lead to multiple views for similar needs. This diagram illustrates how different tables and relationships can be used to represent various business requirements, emphasizing the role of relational databases in managing and querying data efficiently.
One graph for multiple use cases
Graph captures metadata in the same database.
Capturing provenance about calculated fields in SPARQL

```
insert {

<< ?snap fi:hasLoanAge ?loan_age >> prov:wasGeneratedBy fi:DNN-Microservice-74593 .
}

where {

# mortgage contact -> loan and execution date
?mc fibo-ln:setsOutTermsFor ?loan ;
  fibo-c:hasExecutionDate ?exec_date .
# loan -> snapshot
?loan fibo-ks:hasSnapshot ?snap .
# snapshot -> snapshot date
?snapshot fibo-dt:hasAsOfDate/time:inXSDDate ?date .

# calculation: snapshot date – execution date = loan age
bind((?year(?date) - ?year(?exec_date)) as ?loan_age)
}
```

- Graph model provides a native way to capture data provenance
- For data steps outside of the graph, we can still capture that information automatically and store it in the graph using Python decorators
D. Train neural network model
Graph contains x and y vars for a predictive model

X vars:
- FICO Scores
- Origination Loan Amount
- Origination Interest Rate
- Debt-to-Income Ratio
- Loan-to-Value Ratio
- Loan Type
- Origination Channel

Y vars:
- Loan Balance
- Days Delinquent
- Prepayment
- REO
- Default

X vars:
- Property Type
- Zip
- MSA

Observations:
- House Price Index
- Median Income
- Unemployment Rate
- COVID Active Cases

Macroeconomic Variables
Training neural network using TensorFlow + AnzoGraph

Faster executions will enable:

- More iterations of our stress scenarios
- Increased speed-to-delivery of insights

All while maintaining a flexible, reusable data model and tracking data lineage

- 1TB data ingested
- 49bn triples
- 4 node r5.24xlarge K8 cluster
- 25 min to load data
- < 2 min to query 33 loan-level fields and export to parquet for model training
- 20 min to train one epoch of 1.6bn observations
- 15 min for loan-level calculations

Faster executions will enable:

- More iterations of our stress scenarios
- Increased speed-to-delivery of insights

All while maintaining a flexible, reusable data model and tracking data lineage
KG embeddings and graph neural networks
E. Forecast Scenarios

A. Design Data Model
B. Create script to ingest and store data
C. Stage data for analytics and modeling
D. Train Neural Network Model
E. Forecast Scenarios and Reporting
Adding more tables and joins to accommodate new data
Extend the current model to accommodate new data
TensorFlow prediction in AnzoGraph

SPARQL
TensorFlow Prediction Function

# run model | ?latest replaces ?prev because

LoanSnapshot

12/31/19

:L-789

hasLoanBalance

$100k

hasPredictedState

Prepay

:wasGeneratedBy :hasHDFSJSON

12/31/19

:M-123

hasCreatedDate

{}
Visualizing data lineage in Solidatus
Getting started with a successful POC

1. Find a good use case:
   • Combine disparate datasets
   • Complex relationships between data elements
   • Frequently changing business requirements
   • Need to capture data lineage

2. Iterate on building the ontology

3. Show business value by:
   • Demonstrating ability to incorporate business requirement changes quickly
   • Improved data management capabilities for machine learning
   • Data lineage saving time and money
Recap – Key Benefits from Knowledge Graph

• Reducing time spent on data engineering tasks by over 50%
• Support regulatory compliance by natively capturing detailed data lineage
• Overcome limitations of machine learning
• Enable self-service by connecting business terms to data elements
• Turn tacit knowledge into explicit knowledge
What’s makes AnzoGraph DB powerful

**DATA CONNECTIVITY**
- Remote access 200+ data sources
- Data Virtualization
- ELT, ETL, Streaming

**FASTEST DATA LOADING**
- Parallel data loading
- 250 GB/hr/32vCPU server

**HORIZONTAL SCALABILITY**
- Linear scaling to handle billions or trillions of triples

**FASTEST QUERY & RICH ANALYTICS**
- Graph Algorithms
- Data Science Algorithms
- BI/DW Analytics
- Inferencing
- Geospatial Algorithms
- Build-Your-Own

**BUILT ON STANDARDS**
- SPARQL/RDF
- SPARQL*/RDF*
- Cypher/BOLT
- RDFS+

---

**Analytical Benchmarks**

<table>
<thead>
<tr>
<th></th>
<th>Multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnzoGraph DB when compared to Neo4j on and industry standard TPC-H benchmark</td>
<td>217x</td>
</tr>
<tr>
<td>AnzoGraph DB LUBM Benchmark over previous fastest results</td>
<td>113x</td>
</tr>
<tr>
<td>AnzoGraph DB vs SPARK SQL and SPARK GraphFrames</td>
<td>10-300x</td>
</tr>
</tbody>
</table>
Catalog and map your existing data assets – structured or unstructured.

Translate dataset into graph models. Add business definitions, object types, and relationships with semantics.

Create blended analytic ready datasets. Connect graph models. Transform data. Harmonize into canonical models.

Analyze data using semantic and graph models. Export data and provide services for use with BI, analytics, and machine learning tools.
Questions?
FOR MORE INFORMATION:

Thomas Cook
Director of Sales, AnzoGraph DB
Cambridge Semantics
thomas.cook@cambridgesemantics.com