The Big Data Paradigm Shift
Insight Through Automation
Agenda

• The Problem
• Emcien’s Solution:
  – Algorithms solve data related business problems
• How Does the Technology Work?
• Case Studies
The Problem

• Data is growing at an unprecedented rate
• **Less than 1%** of data is analyzed

Old Paradigm: Manually Intensive Analysis

Collect

Analyze

Report

Unpredictable

Slow

Expensive
New Paradigm: Automation of Analysis

Collect  Solve  Review & Act

PREDICTABLE
FAST
ECONOMICAL
Emcien’s Unique Value Proposition

Emcien’s **automatic pattern-detection platform** delivers timely mission critical insights from data

- Automated analysis for fast, predictable, accurate insight
- Applicable across all data types:
  - Structured & Unstructured data, Text or Numeric
- Algorithms designed to **solve business problems**
Types of Data: Structured, Unstructured, Static, Streaming…

Social, Blogs, Newsfeeds

Email Data

Click stream data

Machine Data
Network log files

Marketing Data
Sales Data
Corporate Data
Limitations of Current Solutions

Manually Intensive
- Very slow and unreliable
- Search or query based
- Visualization as a means for discovery → High error

Only certain data types
- Numerical analysis only
- Text only, NLP methods, very high set up cost

Data staging
- Streaming data and recent analysis
- At-rest data and historic analysis

Lack of Scalability
Current approaches focus too much on storage methods
Another View of the Big Data Stack

Use Cases
(Industry specific or cross industry)

Sectors
Banking, Insurance, Manufacturing, Retail
Internet, Telco, Intel/Security, Healthcare

Analysis Layer
Algorithms and Analytics

Value Layer

Processor

Storage

Infrastructure

Data

Our Focus

Value Layer
Sectors
Analysis Layer
Infrastructure

Data

Click stream
Machine Data
Corporate Data
How Does Our Solution Work?

- Big Data problems need graph analysis
- Framework for analyzing relationships
- Highly scalable representation

Data values → Tokens → Graph
Tokens are linked if they occur together

Unstructured

Structured
Algorithms Solve to Extract Patterns

• Algorithms surface the highly relevant dependencies
  – Defocus the redundant/noise to surface the signal
Data Patterns That Reveal “The Insight”

Algorithms designed to reveal graph constructs that solve a business problem

<table>
<thead>
<tr>
<th>Solving a Graph Problem</th>
<th>Results in Solving a Business Problem</th>
</tr>
</thead>
</table>
| Loosely Federated Communities           | - Reveals *groups that behave similarly*  
|                                         | - Reveals dimensions that bind the group  
|                                         | - Impossible to detect in a typical querying system                                                 |
| Cliques                                 | - Highly correlated elements  
|                                         | - Optimal query that would lead to insight                                                          |
| People Network                          | - Reveals influence network of individual  
|                                         | - Highly predictive for adoption behaviors                                                          |
| Substitute Nodes                        | - Nodes that behave very similarly  
|                                         | - ID theft or product substitutes                                                                  |
Algorithms Are Highly Scalable For Big Data

Traditional Data Storage:
- Linear growth with transactions
- Very large storage requirements are
- Increases response time

Graph Data Storage:
- Size of total number of entities
- E.g. Store has 500,000 items → graph has 500,000 nodes
- Weights updated with transactions
- Delivers a global view of the data
Speed of Data → Answers
Access Time vs. Processing Speed

Traditional Data Storage:
• Limit is query speed
• In-memory, hadoop cluster approaches
• Highly dimensional data is a problem
• Unstructured content is a problem

Graph Data Storage:
• All results are Pre-computed (like Google)
• Pre-computing speed: 50K trans/sec compute on 1-core 8GB RAM system
• Speed of response is “access time”
The Business Problems We Solve Across Different Types of Data

Automatically Extract Dependencies
- Web click-stream - Reveal click patterns & market segments
- Sales data - Reveal consumption patterns and propensity
- Clinical trials - extract hypothesis for testing

Social Patterns & People Network
- Marketing - Reveal conversation patterns, people communities
- For Intel - Reveal bad actors based on conversation patterns

Surprising Streaming Content
- Machine Network traffic – Reveal network intrusion
- Sales transactions – Reveal fraud based on unusual patterns

Entity Resolution / Cleansing
- Patterns automatically clusters similar entities.
- Example – credit card transactions, insurance claims with varying merchant names
Intel Case Study (1/4)
Cyber Threat Monitoring with Open Source Data

Customer Overview And Current Situation
• Federal agency is failing to keep up with the activity and data in open source
• Open Source (social, IRC, blogs, etc.) are a key source of communication for underworld
• Link analysis leads to **people of interest network** – which is key for intelligence

Customer Objective
• Federal agency requires fast methods to process high volume open source data
• Need automated methods to highlight **conversations of interest**
• Need automated link analysis to focus on **people of interest**
• Fast and continuous data processing to **keep up with the speed of crime**
Influential people are ranked based on conversation relevance.

Pattern Detection to Isolate Conversations of Interest

Open Source
Chat rooms
Social Streams
emails

Photos – meta data
Twitter
Maps/ geo-location

Bomb attack next week
Helicopter ready for take-off
Tanks turn on crowd

People Graph: Seeds to Reveal network of People of Interest
Intel Case Study (3/4)
Cyber Threat Monitoring with Open Source Data

Influential people ranked based on conversations

Overview
250,000 Accounts
Over 1,000,000 Connections

Highly relevant” People of Interest” network
Intel Case Study (4/4)
Cyber Threat Monitoring with Open Source Data

The silent signal – Automatically detecting a sleeper cell

Overview
250,000 Accounts Analyzed
Over 1,000,000 Connections
1 Account of Interest
Network Traffic Log Files (1/6)
Revealing Patterns In Machine-to-Machine Data

Customer Overview
• Research Institute has thousands of users on their network
• Must provide controlled safe access for the internal working labs and the outside network
• Control illegal intrusions, malicious malware and illegal data transmissions

Customer Objective – Automate Process of Intrusion Detection
• Scan streaming machine-to-machine log file output
• Detect surprising/interesting anomalies/beacons
• Automatically send short list of top ranked “questions” to ask of the data into existing tools (such as CA, Sumologic, Splunk, etc.)
Example Use Cases

1. Summarize and Rank Log File data based on “Surprising flow patterns”

   • Rank Machines based on their “influence” in the network

3. Detect “communities of machines” based on how they “talk to each other”
Network Traffic Log Files (3/6)
Reveal Surprising Patterns In Network flow Data

Surprising network activity
within the data flow
Network Traffic Log Files (4/6)
Ranked Summary of “Surprising” Events

Full Log File Details

Big Data
Example data size:
1,000 transactions/sec

Emcien Scout

CA, Sumologic, Splunk

Top “Surprising” Clusters

Example data size:
1,000 transactions/sec
Network Traffic Log Files (5/6)
Most “Influential” Nodes on network

![Emcien Scout](image)

<table>
<thead>
<tr>
<th>Score</th>
<th>Account</th>
<th>Migs</th>
<th>Conver</th>
<th>Audience Size</th>
<th>sister</th>
<th>followers</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10.80.13.43</td>
<td>336</td>
<td>31</td>
<td>23</td>
<td>28</td>
<td>-</td>
</tr>
<tr>
<td>32</td>
<td>10.15.40.139</td>
<td>376</td>
<td>14</td>
<td>12</td>
<td>21</td>
<td>-</td>
</tr>
<tr>
<td>28</td>
<td>10.43.4.19</td>
<td>273</td>
<td>19</td>
<td>11</td>
<td>15</td>
<td>-</td>
</tr>
<tr>
<td>24</td>
<td>10.80.13.143</td>
<td>254</td>
<td>24</td>
<td>8</td>
<td>15</td>
<td>-</td>
</tr>
<tr>
<td>21</td>
<td>10.104.21.165</td>
<td>71</td>
<td>15</td>
<td>15</td>
<td>4</td>
<td>-</td>
</tr>
<tr>
<td>18</td>
<td>10.80.14.185</td>
<td>10</td>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>13</td>
<td>10.90.22.70</td>
<td></td>
<td></td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>10.40.50.123</td>
<td></td>
<td></td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Audience size for each machine

Most influential machines on the network based on communication patterns
Network Traffic Log Files (6/6)
Machine Communities based on “how they talk”
Customer Overview And Current Situation
• Federal agency is failing to keep up with the activity and data in email
• Too much data and current tools are manually intensive

Customer Objective
• Federal agency requires fast methods to process high volume of email data
• Need automated methods to highlight conversations of interest
• Need automated link analysis to focus on people of interest based on emails
• Fast and continuous data processing to keep up with the speed of crime
Intel Case Study (2/6)  
Automatic Data Collectors

- Content extracted from emails
- Addresses extracted and linked

**Diagram:**
- **Email Data**
- **EmcienScout Extraction Program**
  - Email Content
  - Email Addresses And Phone nums
  - People Network Analysis
  - Text Summary
Intel Case Study (3/6)
Automatic Email Extraction

Extracts all Addresses in Header AND Body

From: daniel.brown@enron.com
To: dan.leff@enron.com, david.delaney@enron.com
Subject: FW: EES Employee Issues
Cc: kalen.pieper@enron.com, judy.gray@enron.com
Bcc: kalen.pieper@enron.com, judy.gray@enron.com

Messages extracted, each word tokenized and connected into graph.

We are working to gather as much information as possible....
Intel Case Study (4/6)
Automatic Email Summarization

Summarize content from emails to better understand group conversations.
Program extracts To/From email addresses and phone numbers from suspects email account

Newly created contacts file is loaded into Scout People Graph

Large complex graph is created using emails and phone number connections
Initial “bad actor” seed accounts (emails addresses or phone numbers) are selected or entered.

Ranked List of other accounts that are likely involved with seed accounts based on their connections.
How Emcien Fits Into Your Ecosystem

Feed downstream systems with data output

Production Servers

API

UI

UI for Analyst who wants to review results

Structured Data

- Sales Trans
- Bank Trans
- Insurance Claims

Unstructured Data

- Social Media
- News / Blogs
- Emails / Chat

Unstructured Data (Machine Data)

- Server Logs
- Web logs
- Security Logs

Pattern Detection Platform

Compressed Graph Data Representation

Feed downstream systems with data output

Pattern Detection Platform

Compressed Graph Data Representation

Structured Data

- Sales Trans
- Bank Trans
- Insurance Claims

Unstructured Data

- Social Media
- News / Blogs
- Emails / Chat

Unstructured Data (Machine Data)

- Server Logs
- Web logs
- Security Logs
Questions?
Types of Data

- Many types of Data
  - Structured, Unstructured
  - Text, Numeric, Machine
- In many states
  - Static (slow batch)
  - Streaming or fast batch
Limitations of Current Solutions

• **Manually Intensive**: Very slow and unreliable
  – Search or query based
  – Visualization as a means for discovery → High error

• **Limitation based on data types**
  – Numerical analysis only
  – Text only, NLP methods, very high set up cost

• **Limitation based on data staging**
  – Streaming data and recent analysis
  – At-rest data and Historic analysis

• **Scalability**
  – Current approaches focus too much on storage methods