NetClinic: Interactive Visualization to Enhance Network Fault Diagnosis

Zhicheng Liu, Bongshin Lee, Srikanth Kandula and Ratul Mahajan
The Problem: Diagnosing Enterprise Networks

- **Faults:** Anomalies in application behavior
  - Cannot send email, browser extremely slow, network connectivity down …

- **Difficulty in identifying culprits / root causes**
  - Network components interact in complex ways
  - Information overloading: too many variables
Modeling Complex Interaction as a Dependency Graph

Not diagnosing at the level of network path and individual switches
Modeling Complex Interaction as a Dependency Graph

Not diagnosing at the level of network path and individual switches
Motivation for Visual Analytics

- Automated diagnosis tools are not always accurate
  - Rely on minimal application specific semantic knowledge
  - Mostly statistical

- Even when true culprit is identified
  - Need for exploration and verification
    - Ground truth is not known before-hand

An ideal visual analytics problem
Automated Diagnosis

NetMedic [Kandula et al., SIGCOMM 2009]

- Variable Level: Performance Counters

- Component Level: Statistical Abnormality

- Edge Level: Potentiality of Impact
  - Statistical analysis of joint behavior of neighbors

- Network Level: Given a faulty component, Identifying Culprits
  - Rank edge weights to order likely causes
The Reciprocal Nature of Human Sensemaking

Dynamic mixture of top-down and bottom-up processes

DATA
recognize/construct a frame

FRAME
define, connect and filter the data

[Klein et al. 2006]
Data-Frame Interaction in Network Diagnosis

Problem: a SQL client cannot talk to the server

Network Level: other clients are overloading the server

Edge Level: check server machine application state

Component Level: firewall rules of the server machine changed

Variable Level: i/o operations normal CPU / memory normal

the fault is due to the change in firewall rule

client using a port number blocked by the firewall change

Frame

Data
Design Considerations

- Output of automated engines can be used as useful frames

- Show outputs at all levels of abstraction
  - Minimal constraints on navigation across levels of abstraction

- Flexible exploration
  - Top-down exploration: verify the output of automated analysis
  - Bottom-up exploration: form and evaluate own hypothesis
Main Design Challenge: Graph Layout

Using a force-directed layout
Machine-based clustering

![Diagram of machine-based clustering](image)

- A networking application
- A local application
- Neighbor set
- Firewall rule
- Application config
- Machine config
- Machine
### Performance Counter View

#### app186 emailmonkey.exe 9000 9000 2 1 :srikanth_test1

<table>
<thead>
<tr>
<th>Metric</th>
<th>Historical avg</th>
<th>Current avg</th>
<th>% change</th>
<th>Historical “training” values</th>
<th>Current values</th>
<th>Statistical abnormality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process :: page faults/sec</td>
<td>51.38</td>
<td>24.02</td>
<td>-53.25%</td>
<td>00.00</td>
<td>91.74</td>
<td></td>
</tr>
<tr>
<td>Process :: % processor time</td>
<td>0.0064</td>
<td>0.072</td>
<td>13.42%</td>
<td>00.00</td>
<td>01.08</td>
<td></td>
</tr>
<tr>
<td>Process :: % privileged time</td>
<td>0.0013</td>
<td>0.0013</td>
<td>3.07%</td>
<td>00.00</td>
<td>00.30</td>
<td></td>
</tr>
<tr>
<td>err: emailMonkey :: numberofsmtpmessagessentok</td>
<td>0.0521</td>
<td>0.0480</td>
<td>-7.95%</td>
<td>00.00</td>
<td>07.00</td>
<td></td>
</tr>
<tr>
<td>Process :: % user time</td>
<td>0.0051</td>
<td>0.0059</td>
<td>15.96%</td>
<td>00.00</td>
<td>00.91</td>
<td></td>
</tr>
<tr>
<td>APP_PKTS_IN</td>
<td>284.39</td>
<td>258.00</td>
<td>-9.28%</td>
<td>04.00</td>
<td>408.00</td>
<td></td>
</tr>
<tr>
<td>APP_BYTES_IN</td>
<td>790,610.04</td>
<td>668,292.20</td>
<td>-15.47%</td>
<td>117.00</td>
<td>1,149,035.00</td>
<td></td>
</tr>
</tbody>
</table>
Qualitative User Study

• Participants: 10 graduate students + 1 system engineer working on computer networks or operating systems
• Data: real environment with faults injected
  • Ground truths known
• NetClinic: suggest top 5 most likely causes
  • True culprit inside these five 50% of the time
• Training: 4 machines, 243 nodes, 683 links
• Test: 7 machines, 682 nodes, 2045 edges
• Video-taped, think-aloud protocols, semi-structured interviews
Tasks

- Given a reported problem, use NetClinic to find out the network component that most likely caused the problem.

<table>
<thead>
<tr>
<th>Symptom of Fault</th>
<th>Causes</th>
</tr>
</thead>
<tbody>
<tr>
<td>The email client on a machine is experiencing some errors</td>
<td>The client’s configuration is broken</td>
</tr>
<tr>
<td>Some SQL clients are experiencing poor performance</td>
<td>Another client is overloading the server</td>
</tr>
<tr>
<td>An email client can’t get up-to-date data from server</td>
<td>The remote drive is dismounted</td>
</tr>
<tr>
<td>Some users were unable to access a specific feature of a Web-based application</td>
<td>The firewall along the path was blocking https traffic</td>
</tr>
<tr>
<td>Some clients cannot connect to the database serve</td>
<td>A port used by the problematic clients had been blocked by a change in firewall rules on the server machine</td>
</tr>
</tbody>
</table>
Results

- True culprits correctly identified in 29 out of 33 tasks (88%)
  - Culprits in top five suggestions 50% of the time

- Completed all 3 tasks within 1 hour
Survey

- subjective opinions on graph layout, visual design, usefulness, ease of learning

19 questions

most favorable

least favorable

ease of learning
recover from mistakes

focus+context highlighting
graph layout

questions
Flexibility in Exploration Strategies

- Most did not adopt a “least-effort” strategy
  - Verify all five suggestions before start self-exploration

- Using one diagnosis as entry point to learn about the problem

- Generate and verify frames, use automated diagnoses to make sure nothing was overlooked

- Not using network level diagnosis at all
Related Work

- Security monitoring / intrusion detection in computer networks
  - [Erbacher et al. 2002, Mansmann et al. 2007]
  - Tasks are different from fault diagnosis
- Visualization-based network diagnosis
  - SCUBA, nCompass, and MTreeDX
  - Mostly visualizing raw data
- Visual analytics in relationship networks
  - E.g., social networks [Social Action, 2006]
Contributions

- Coupling visualizations with a sophisticated reasoning engine
  - Integrated automated analyses across multiple levels
  - Explicit design consideration of sensemaking processes
  - A novel semantic graph layout design
Future Directions

- Scalability
  - Integrating machine-level diagnosis
- More evaluation
  - Long term study with professional administrators
Thank you

Questions?