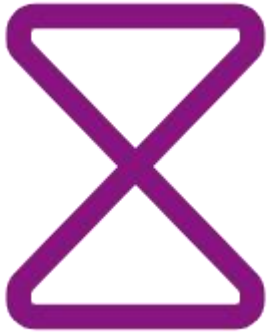
LabelEase: A Semi-Automatic Tool for Efficient and Accurate Trace Labeling in Microservices

Shenglin Zhang[1], Zeyu Che[1], Zhongjie Pan[1], Xiaohui Nie[2], Yongqian Sun[1], Lemeng Pan[3], Dan Pei[4]

Nankai University[1], CNIC[2], Huawei[3], Tsinghua University[4]

Content

南開大學 Nankai University



Background

Challenges

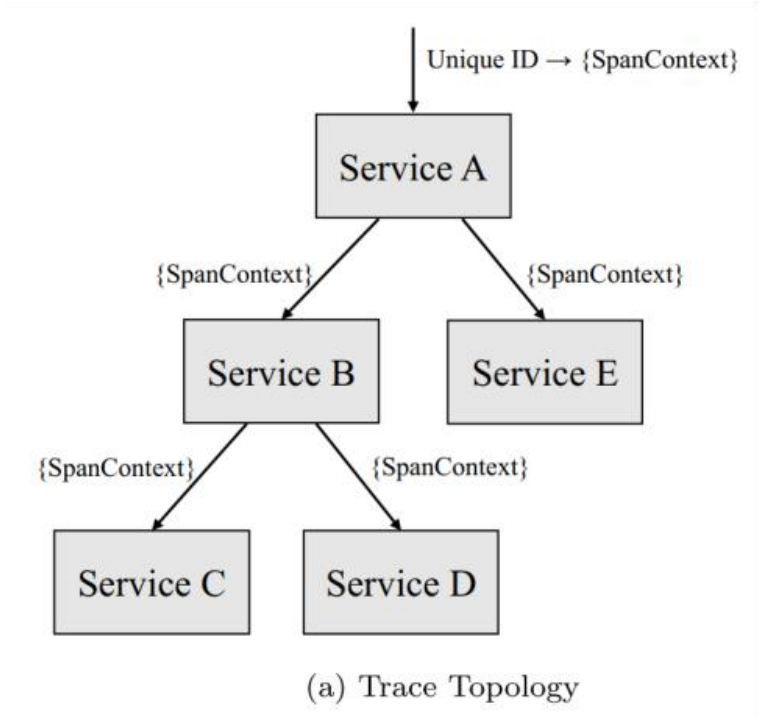
Approach

Implementation

Evaluation

Conclusion

Trace and Span



(a) Trace Topology

Traces and Spans in Distributed Tracing

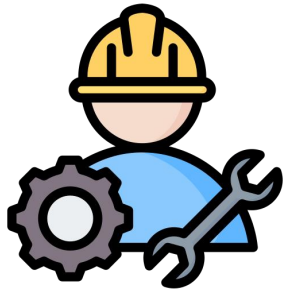
```
{
    "traceId": "5982fe77008310cc80f1da5e10147517",
    "spanId": "be2d01e33cc78d97",
    "parentSpanId": "ebf33e1a81dc6f71",
    "timestamp": 1458702548786000,
    "duration": 13000,
    "localEndpoint": {
      "serviceName": "client",
      "ipv4": "192.168.1.2",
      "port": 9411},
    "remoteEndpoint": {
      "serviceName": "server",
      "ipv4": "127.0.0.1",
      "port": 3306 },
    "annotations": [{
        "timestamp": 1458702548786000,
        "value": "cs"
      },{
        "timestamp": 1458702548799000,
        "value": "cr"
      },{
        "timestamp": 1458702548788000,
        "value": "sr"
      },{
        "timestamp": 1458702548796000,
        "value": "ss"}]
}
```

An Example of Span Log

- Large-scale microservice systems could contain tens of thousands of service instances
- Trace data is crucial for understanding system behavior, diagnosing performance issues, and ensuring reliability

Trace-Based Anomaly Detection and Root Cause Localization

When a microservice system fails, it will degrade user experience and bring loss to companies

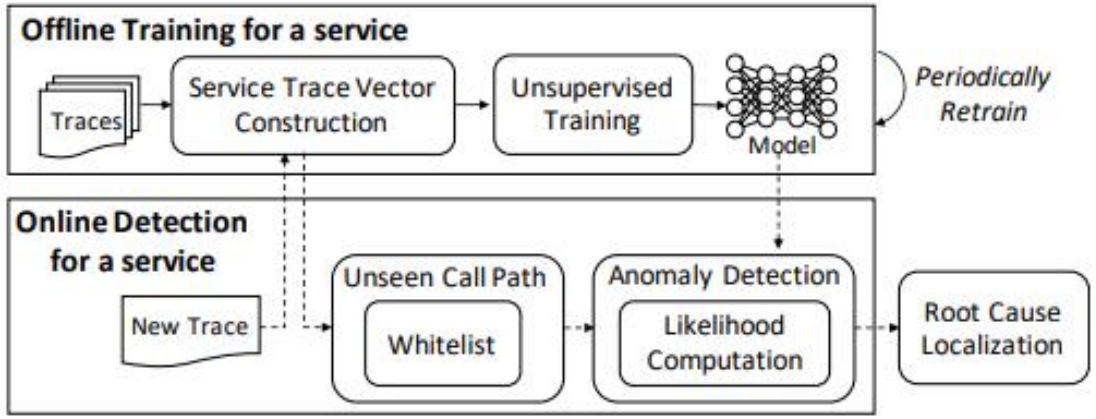


Trace-based anomaly detection and root cause localization can help operators timely mitigate failures and improve user experience

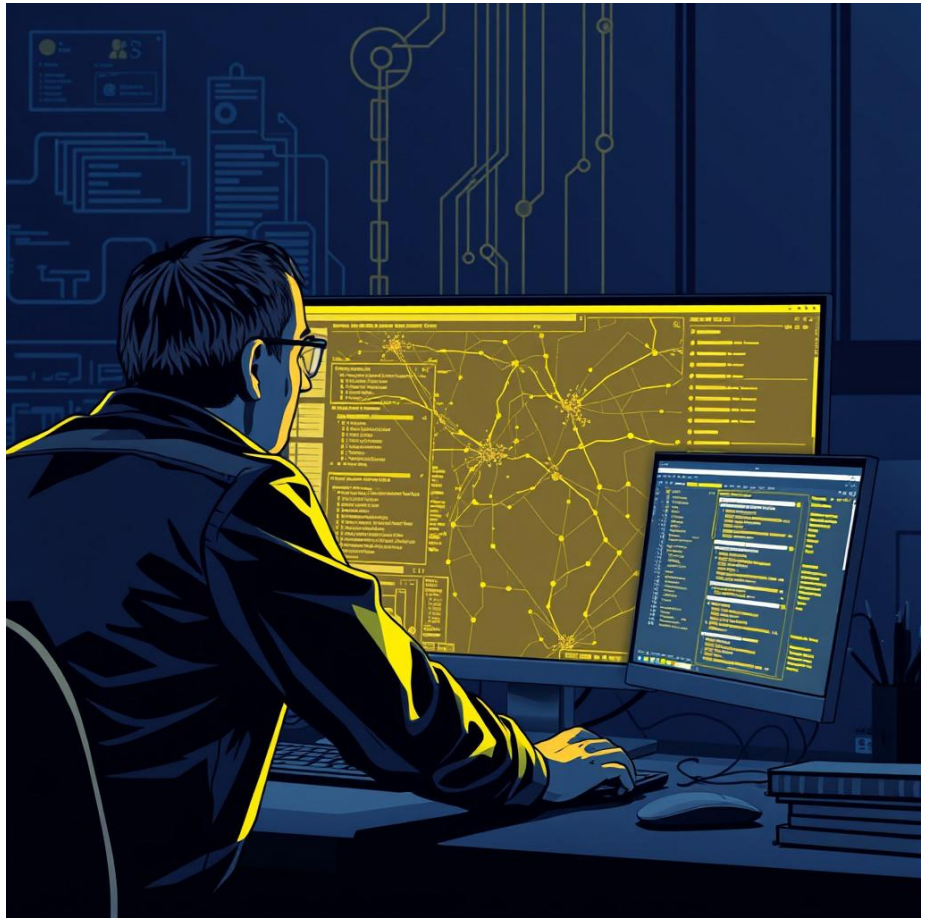
- Anomaly detection identifies anomalies in service execution and generates alerts
- Root cause localization identifies the specific component or subsystem responsible for each failure

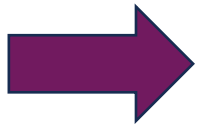
Why Traces Need to Be Labeled

- Trace-based anomaly detection methods usually apply deep learning methods to learn the normal patterns from "normal data", the high-quality data without anomalies are crucial
- High-quality data is important to evaluate and improve the performance of trace-based anomaly detection and root cause localization methods
- There are only a few public trace datasets
  - Limited number of traces
  - Limited types of failures
  - Not generalized



Manual Trace Labeling Is Time-Consuming and Labor-Intensive



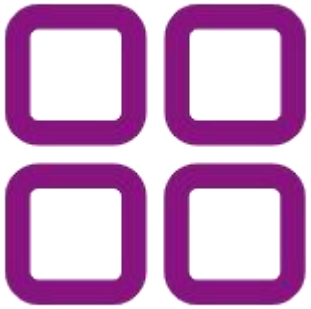


Our goal is to design a semi-automatic trace labeling tool for efficient and accurate trace labeling

Content

Nankai University

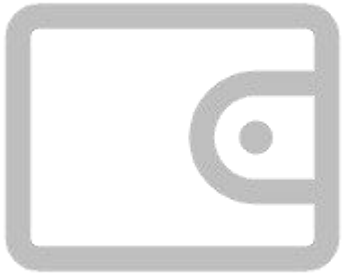
Background



Challenges

Approach

Implementation

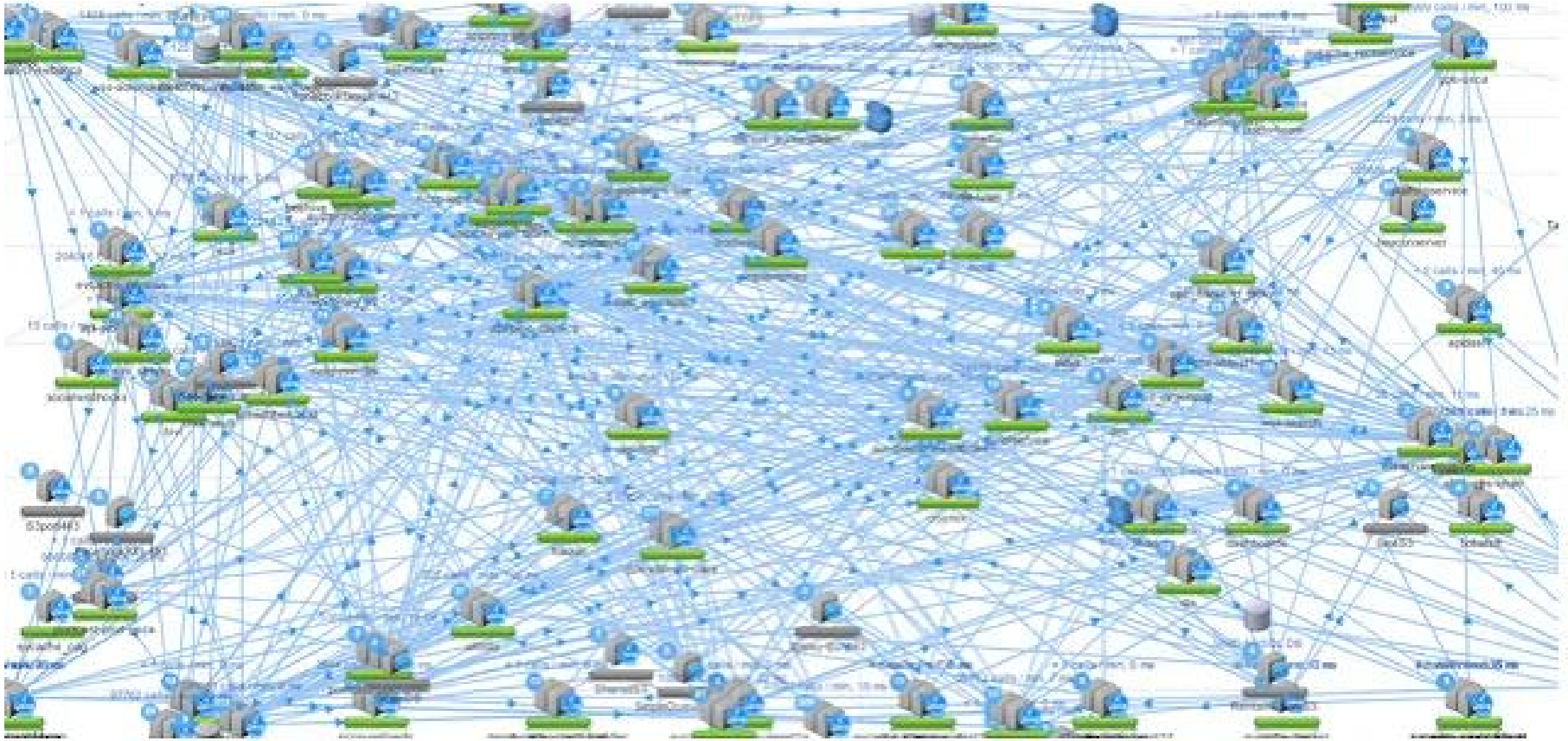


Evaluation

Conclusion

Massive Trace Data

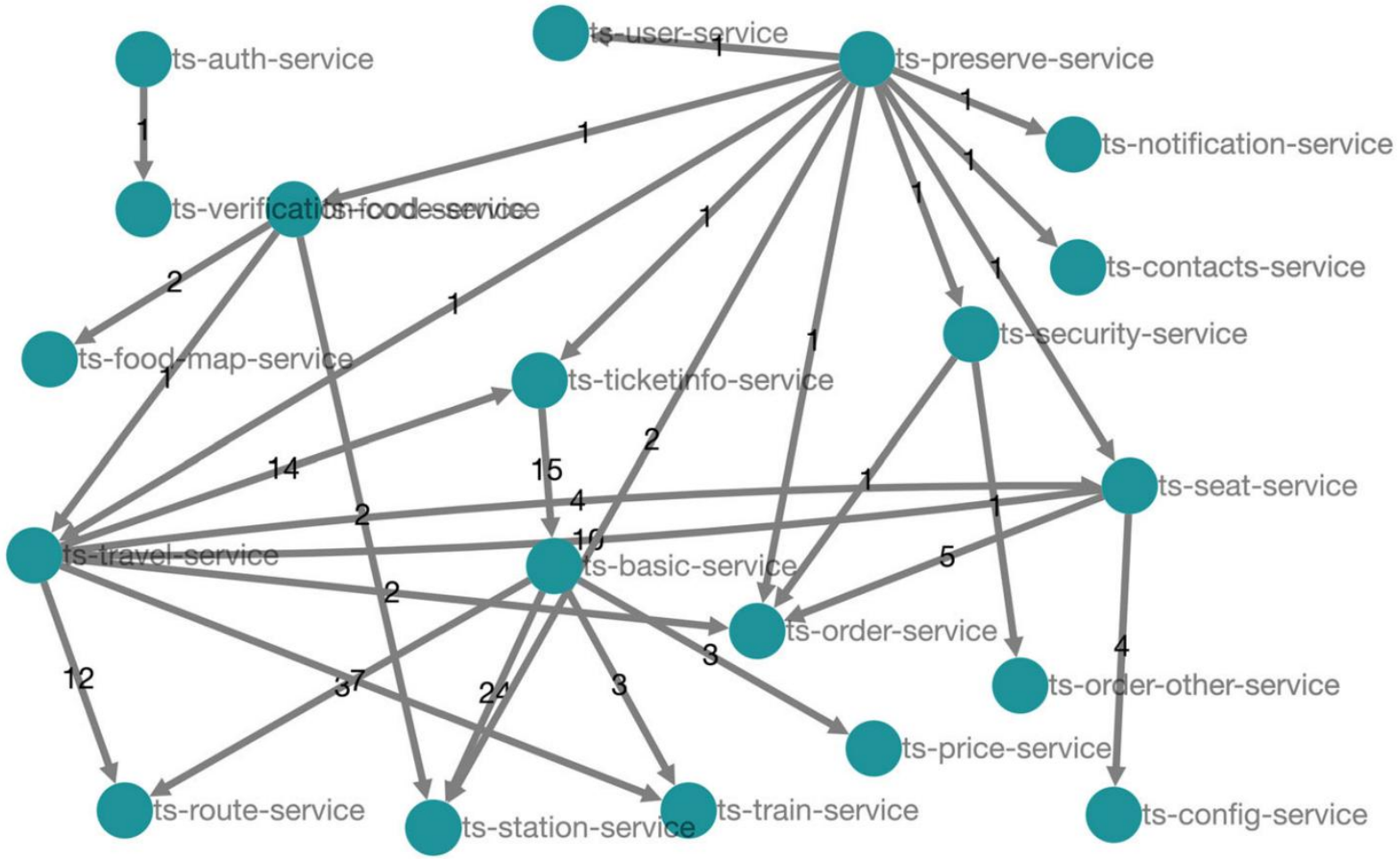
南開大學 Nankai University



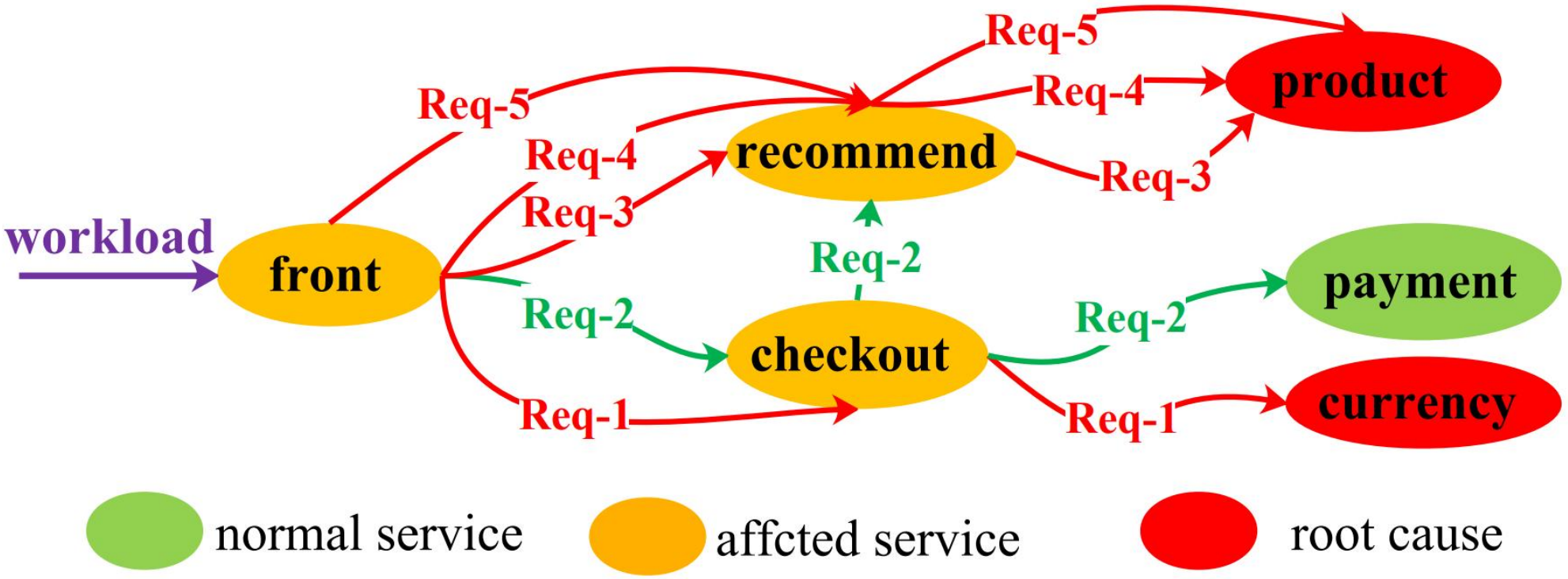
- Microservice systems could generate a massive amount of trace data every day for labeling
- E.g. , the microservice systems of eBay generate about 150 billion traces everyday!

Complex Trace Structure

- A trace is a complex structure arising from the hierarchy of parallel service invocations
- It can contain dozens to hundreds of services



Hard-To-Determine Root Cause



- Root cause labeling needs to consider multiple traces in a period
- It is challenging to check all the anomalous trace data and conclude the root cause information

LabelEase

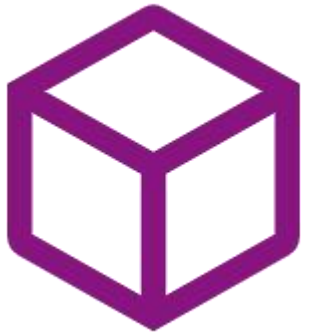
A semi-automatic trace labeling tool for efficient and accurate trace labeling

Content

南開大學 Nankai University

Background

Challenges



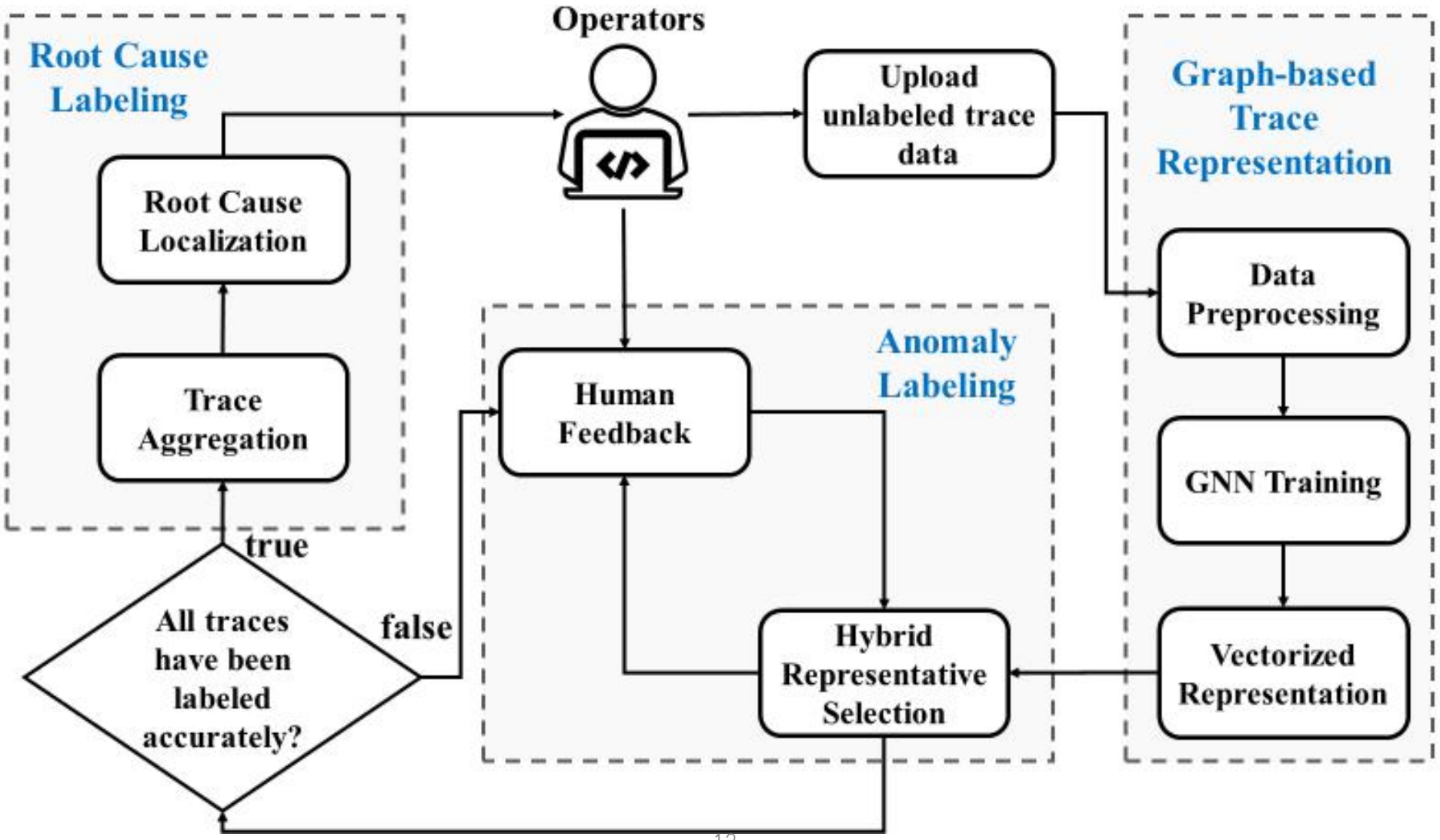
Approach

Implementation

Evaluation

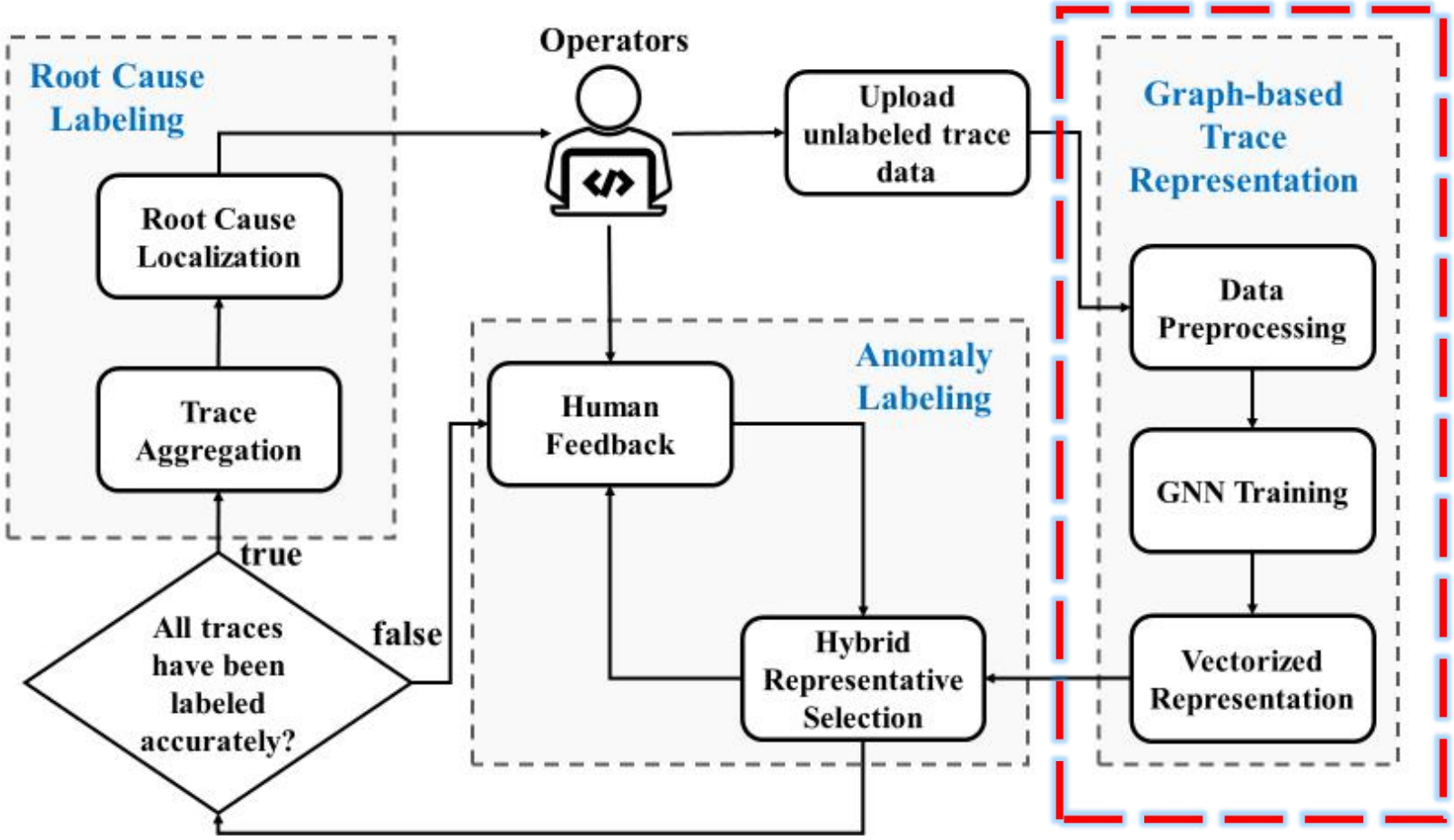
Conclusion

Overview


Root Cause Labeling
Root Cause Localization
Trace Aggregation
Operators
Upload unlabeled trace data
Graph-based Trace Representation
Data Preprocessing
GNN Training
Vectorized Representation
Anomaly Labeling
Human Feedback
Hybrid Representative Selection
All traces have been labeled accurately?
true
false

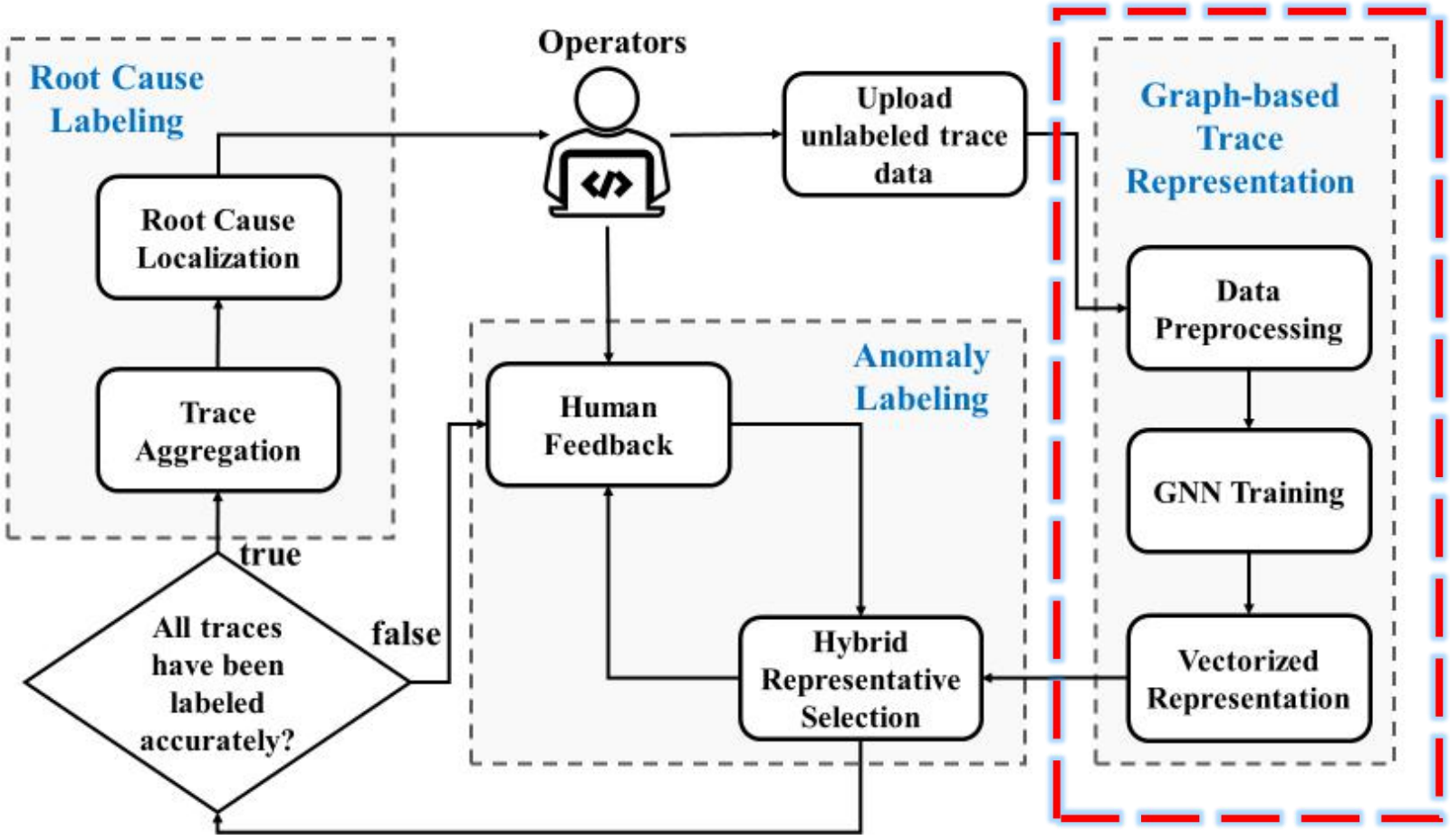
Graph-Based Trace Representation

- Data Preprocessing
- GNN Training
- Vectorized Representation



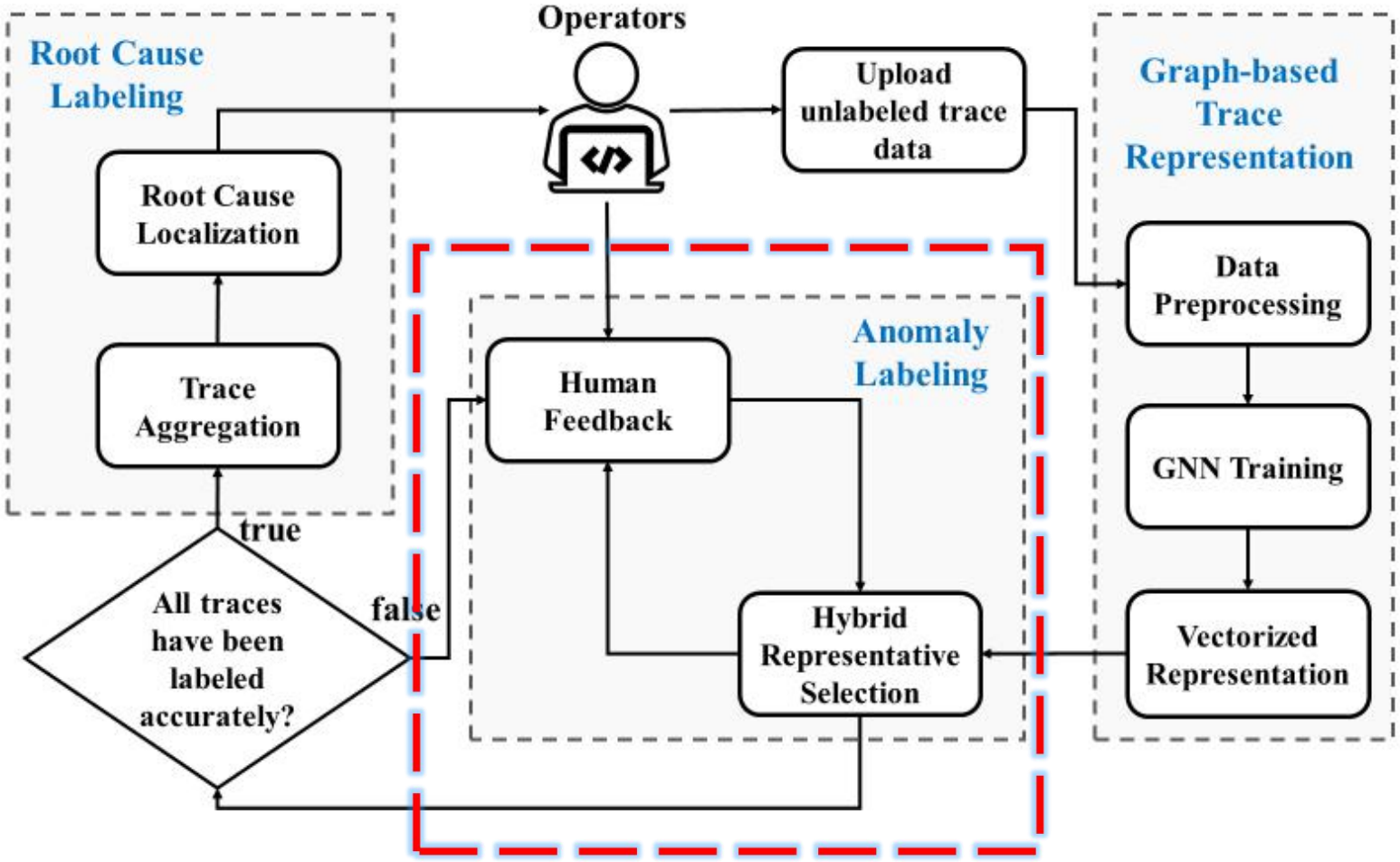
Graph-Based Trace Representation

- Data Preprocessing
  - Convert the raw trace data into a graph
  - Focus on semantic information, temporal features, status codes within the span, and the calling relationships between different services
- GNN Training
  - Graph contrastive learning
  - Differentiate between various traces based on their representations
- Representation vectorization



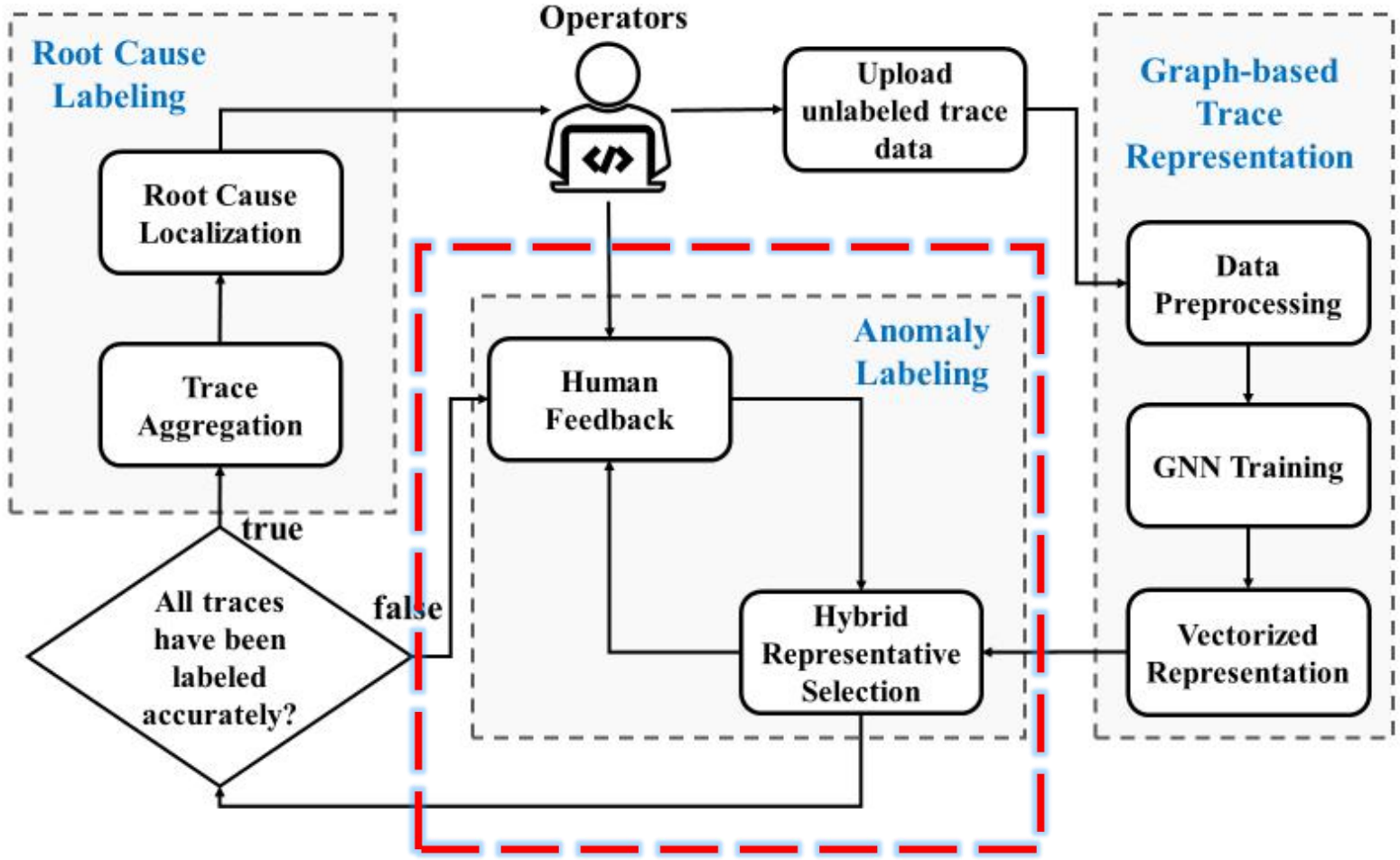
Anomaly Labeling

- Hybrid Representative Selection
  - Partition traces into different classes
  - Select the cluster center point as the most representative trace for operators to label
- Human Feedback
  - Operators label the most representative trace in each cluster
  - LabelEase automatically labels the remaining traces



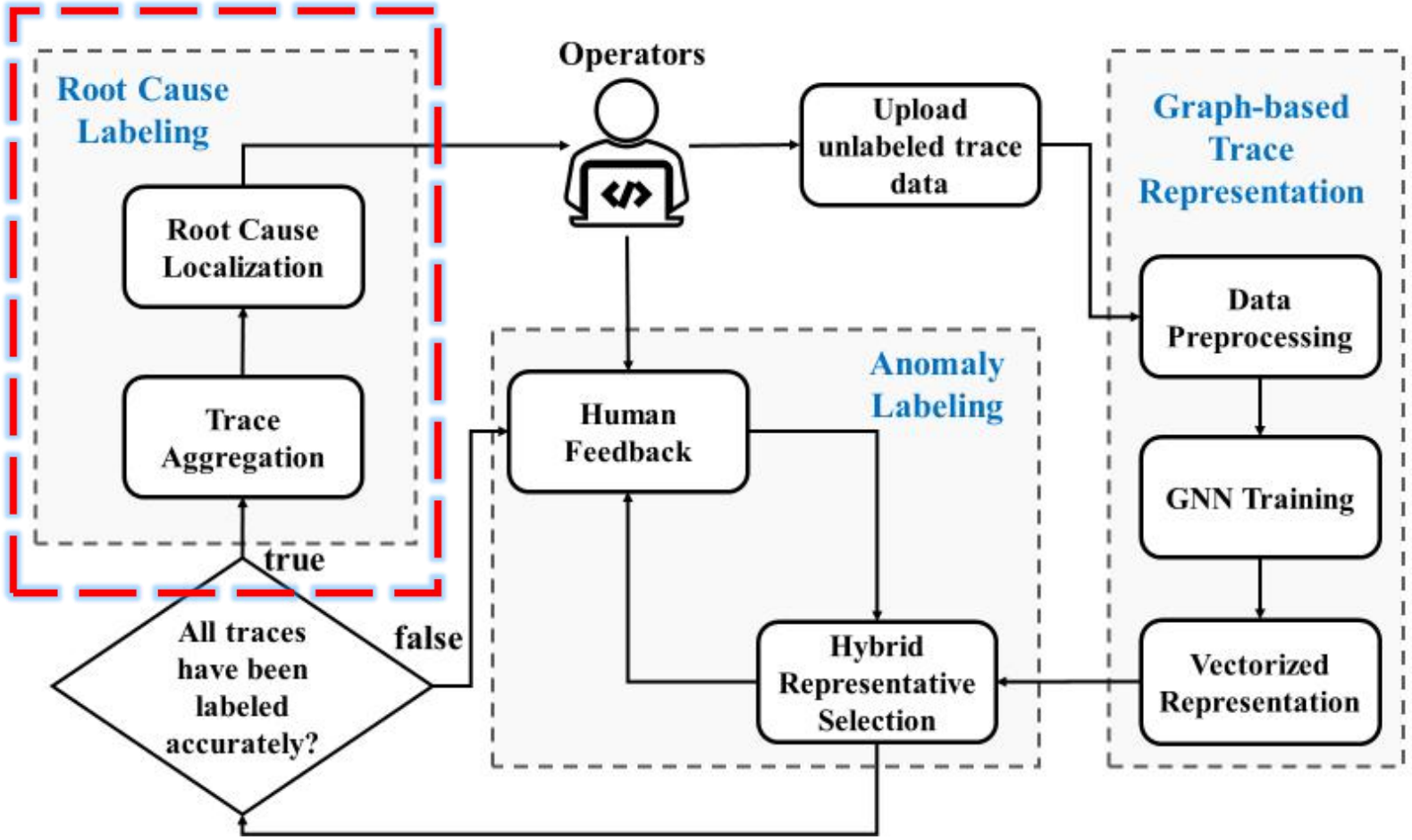
Anomaly Labeling

- Use active learning to select the most informative data points from an unlabeled dataset for labeling through human feedback
- Minimize the labeling effort while ensuring high-quality data labels



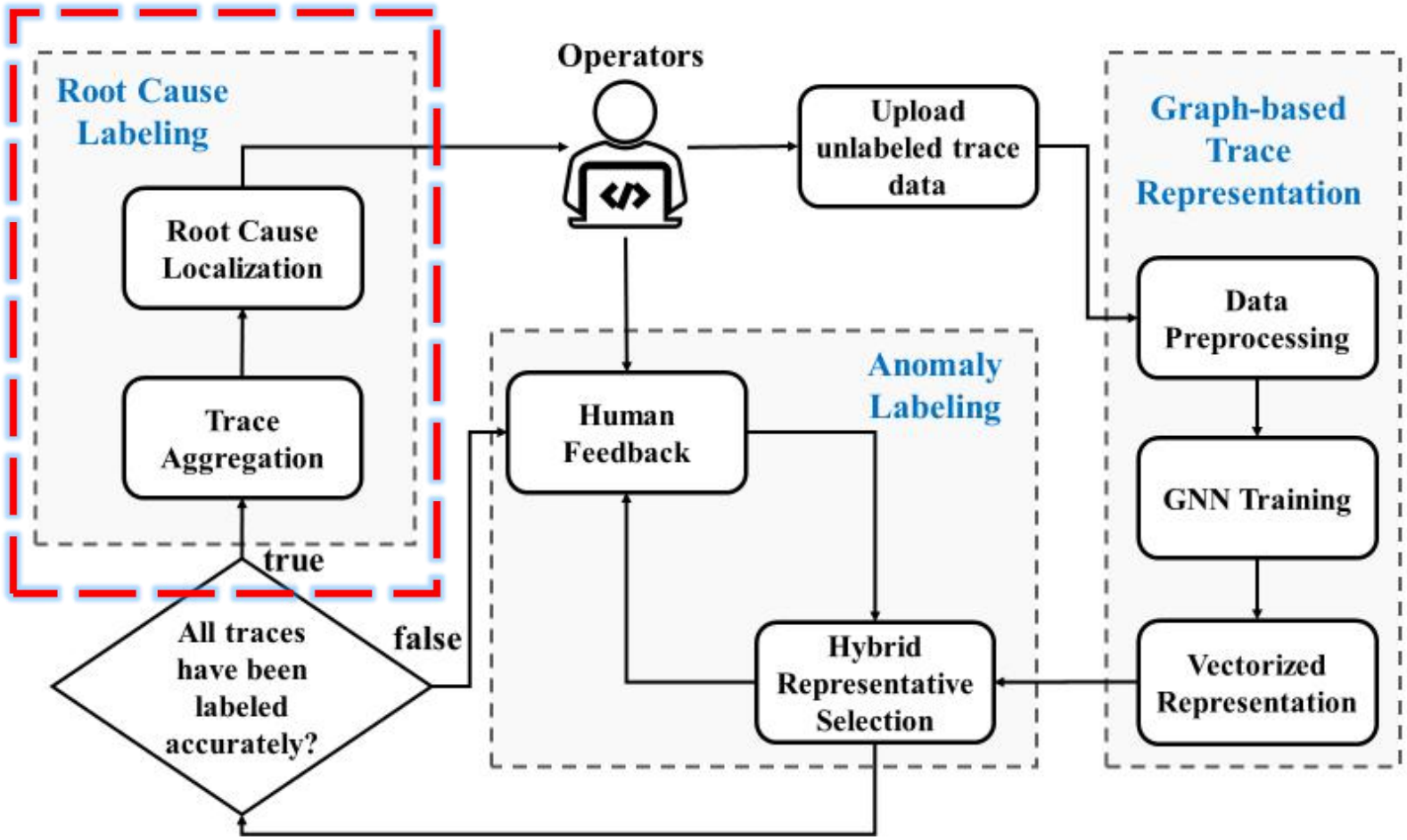
Root Cause Labeling

- Trace Aggregation
- Root Cause Localization



Root Cause Labeling

- Trace Aggregation
  - Partition the trace data into abnormal periods
- Root Cause Localization
  - The spectrum-based fault localization (SBFL) technique
  - Service dependency topology
  - Service latency information



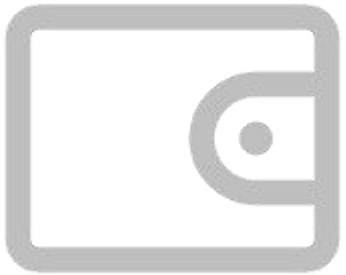
Content

Background

Challenges

Approach

Implementation



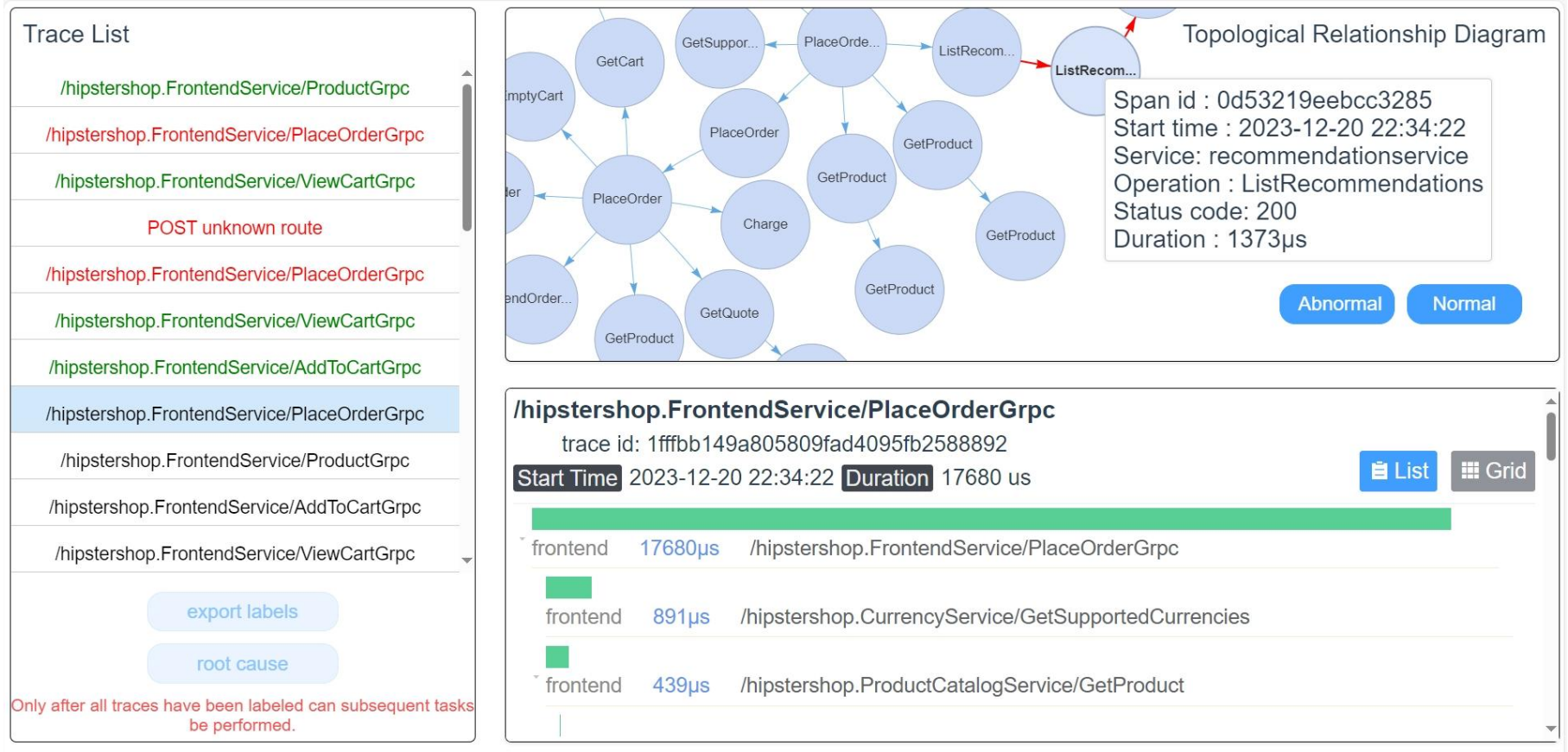
Evaluation

Conclusion

Trace Anomalies Labeling

Nankai University

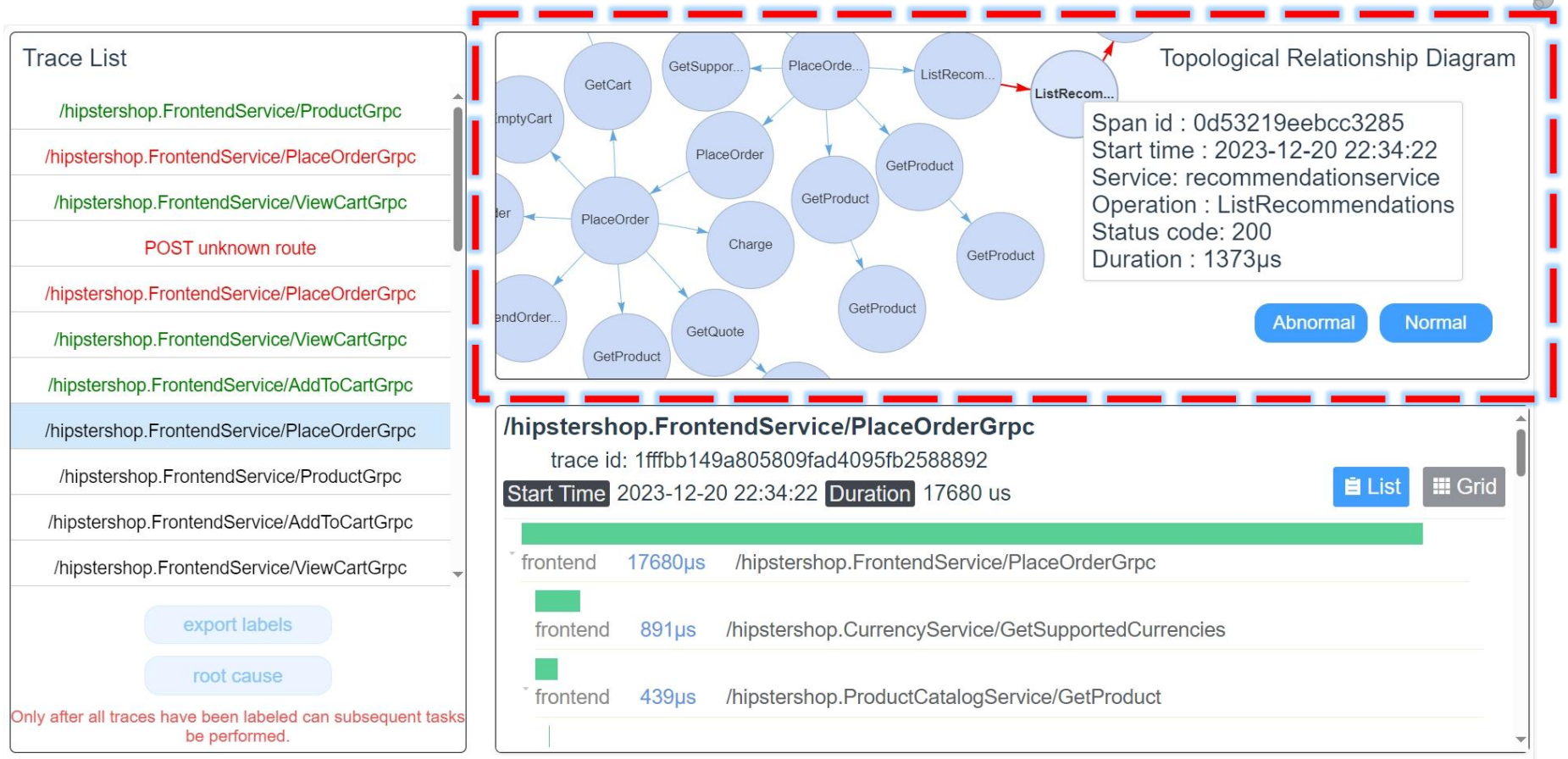
After LabelEase selects representative traces for labeling, operators will label the given traces on this page



Trace Anomalies Labeling

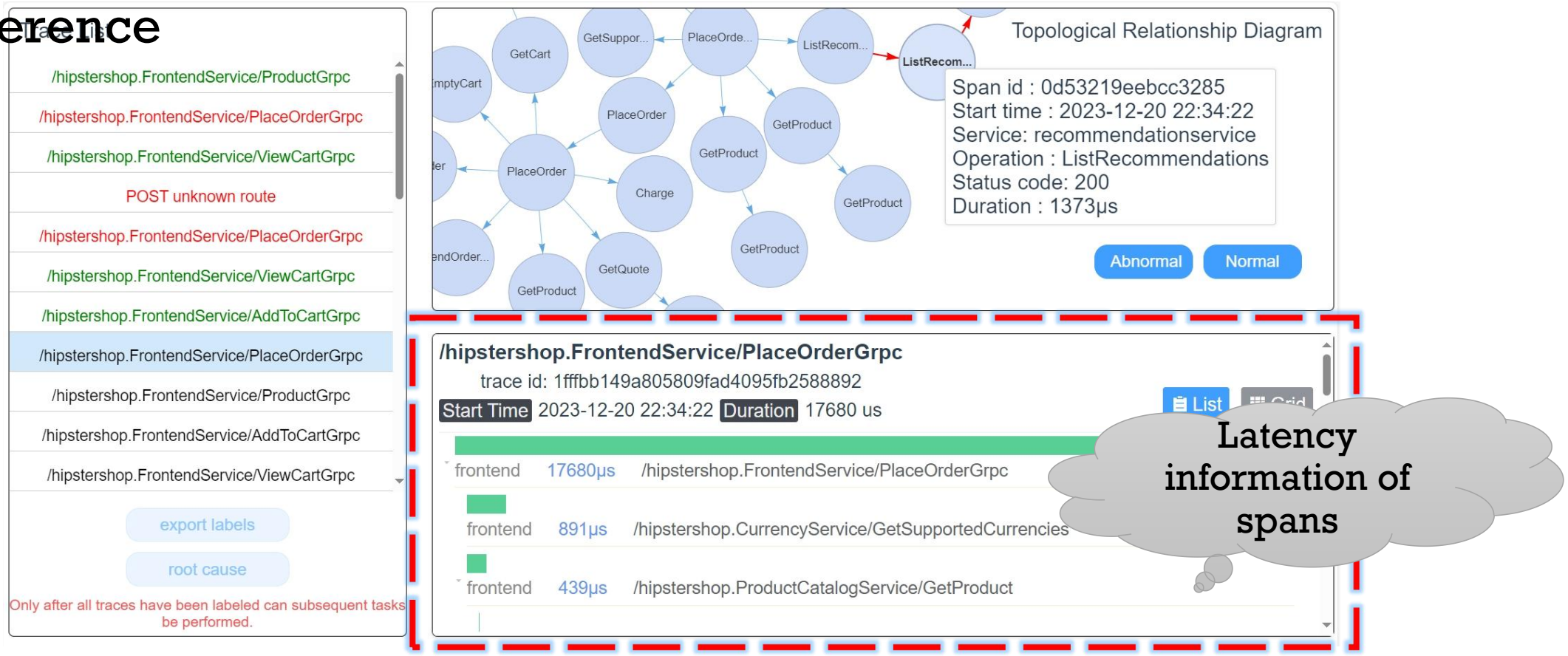
- Span ID, start time, serice name, operation name, status code, and duration

Trace and span-related information



Trace Anomalies Labeling

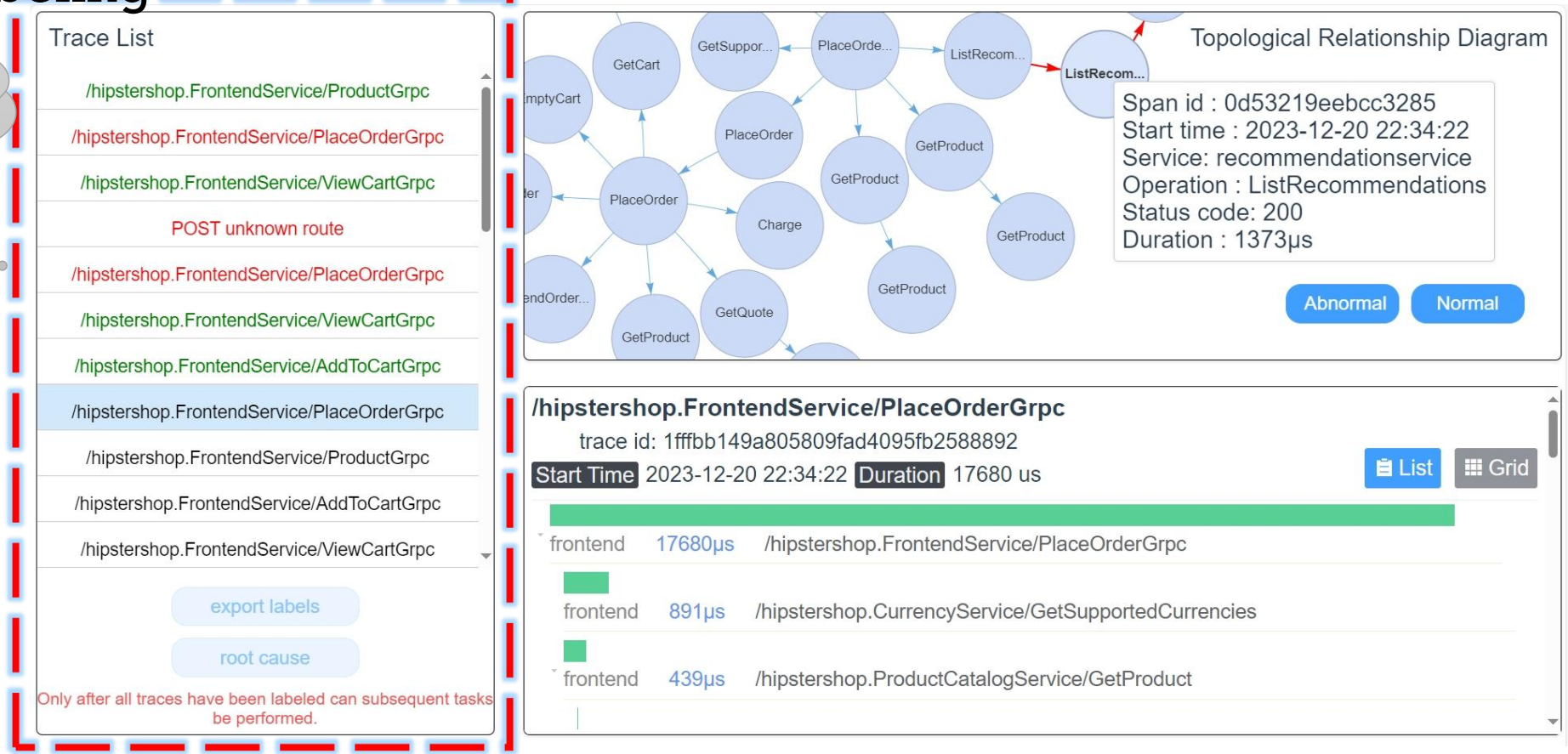
- The average value, standard deviation, and other information about each similar call path in the entire dataset as a labeling reference



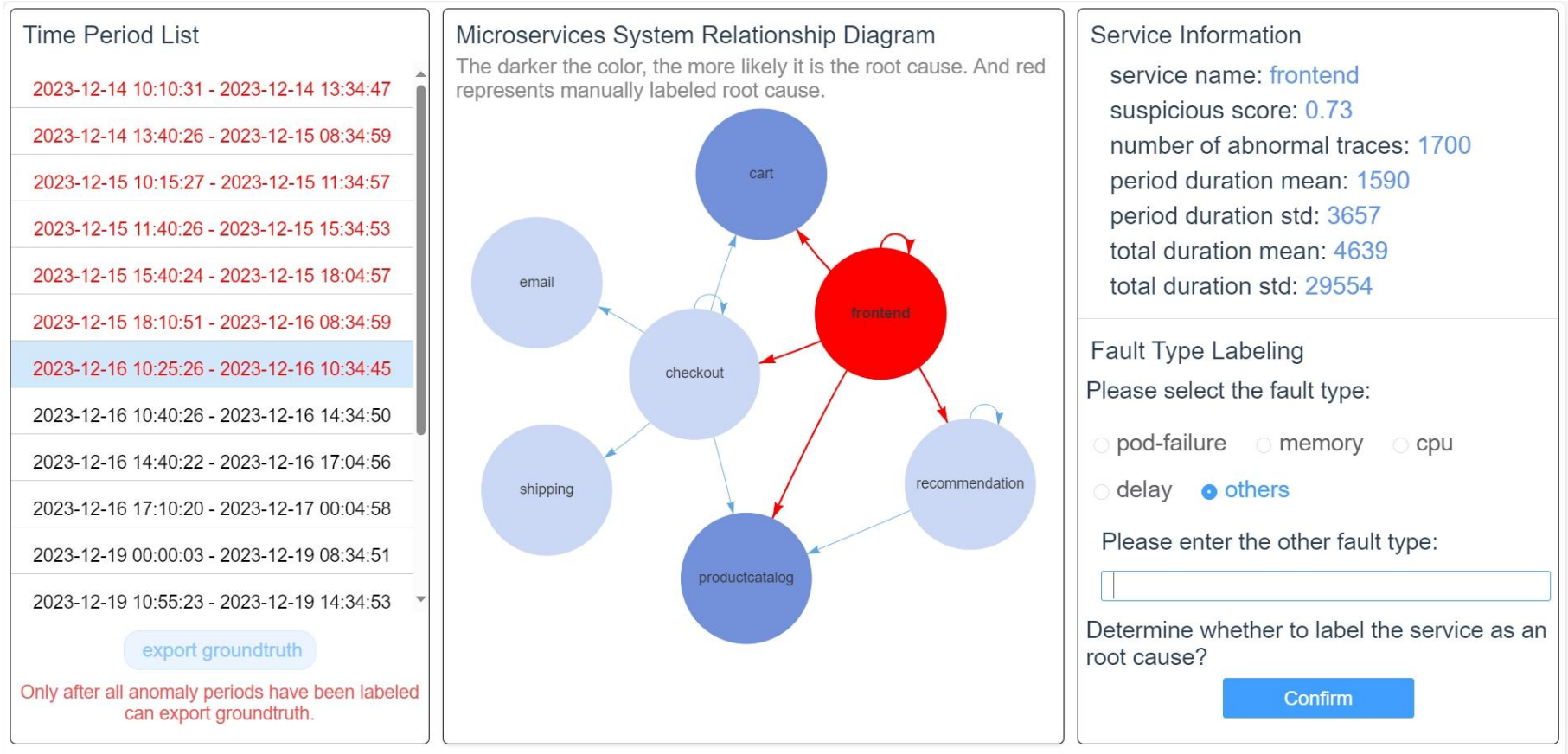
Trace Anomalies Labeling

- Each trace is visually distinguished: green signifies a label of “normal”, red indicates a label of “abnormal”, and black denotes traces awaiting labeling

Selected trace list



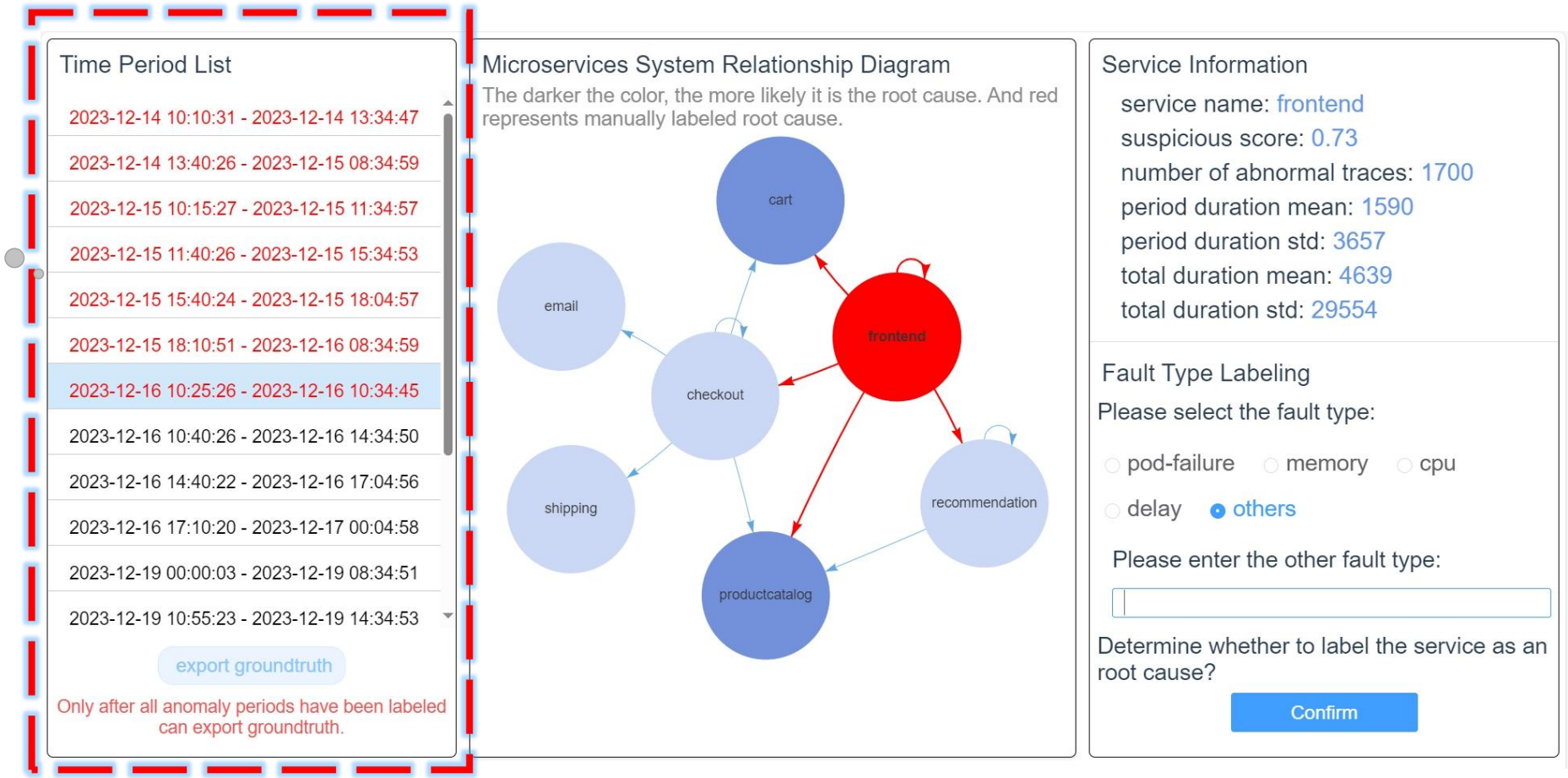
Root Cause Labeling



After all traces are labeled, operators will go to the page of labeling root cause.

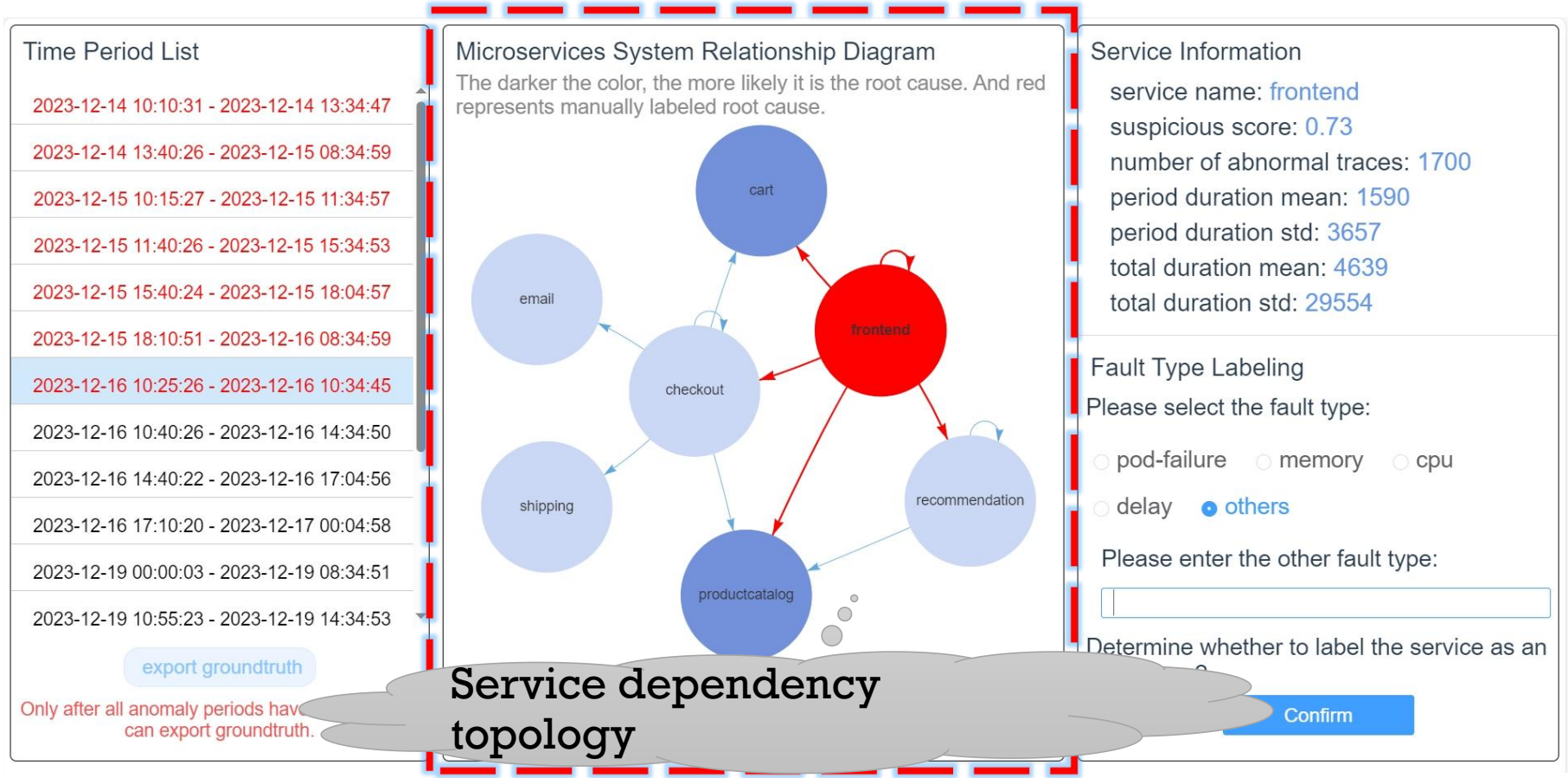
Root Cause Labeling

Fault period list



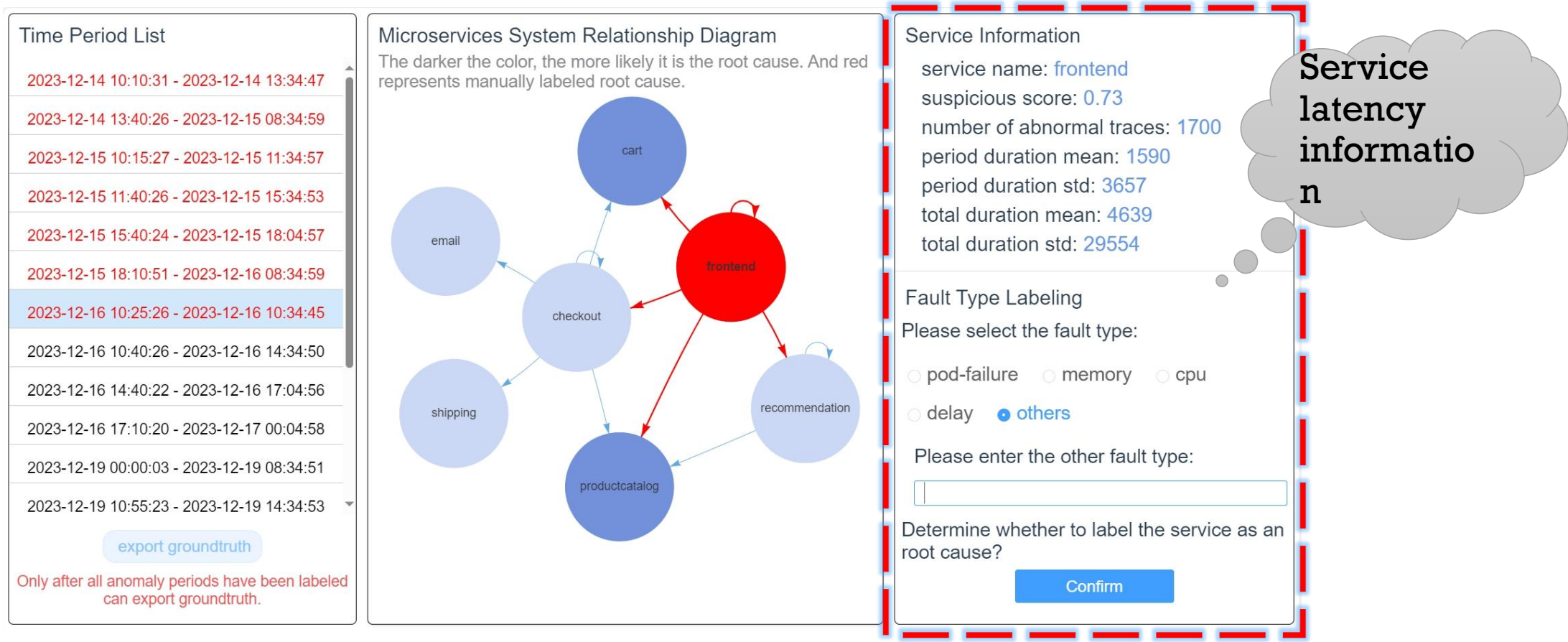
- LabelEase identifies the impact of each fault by merging the anomalous traces occurred closely
- Red means it has been labeled and black denotes period awaiting labeling

Root Cause Labeling



- For each period, a microservice system relationship diagram is featured in the center of the page
- The higher the suspicious score, the more likely the root cause is, and the darker the service will be displayed on the interface

Root Cause Labeling



- Upon selecting a service, detailed information about it is displayed
- Operators can set the type of fault to label the root cause
- The service labeled as the root cause is highlighted in red

Content

南開大學 Nankai University

Background

Challenges

Approach

Implementation

Evaluation

Conclusion

Datasets and Metrics

南開大學 Nankai University

- We use two datasets, denoted as Dataset 1 and Dataset 2.

	Source	#Normal traces	#Anomalous traces
Dataset 1	A benchmark microservice system developed by us	103078	27285
Dataset 2	A top-tier commercial bank	93579	19207

- Evaluation metrics

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$F_1\ score = 2 \times \frac{Precision \times Rcall}{Precision + Recall}$$

LabelEase vs. Baseline Methods

- LabelEase shows superior performance over all baseline approaches across both datasets, achieving F1-scores of 0.99 and 0.98, respectively
- It shows superior efficiency compared to the baseline methods, boasting the shortest time overhead and highest labeling efficiency

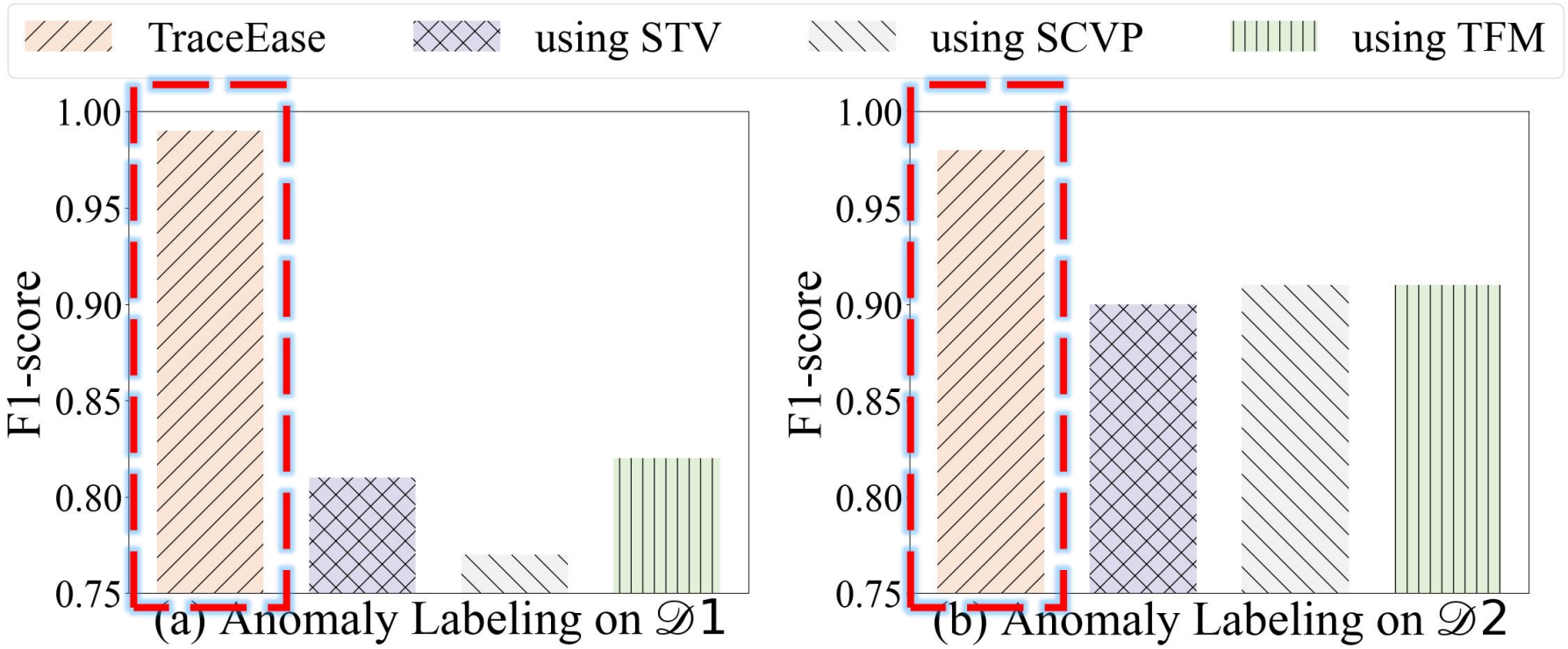
TABLE I

THE EFFECTS OF *LabelEase* IN COMPARISON WITH DIFFERENT APPROACHES ON TWO DATASETS

Approach	$\mathscr{D}1$				$\mathscr{D}2$			
	Precision	Recall	F_1-score	Time	Precision	Recall	F_1-score	Time
LabelEase	1	0.98	0.99	6.53s	0.96	0.99	0.98	21.68s
MultimodalTrace [37]	0.2	0.15	0.17	1.9min	0.17	0.15	0.16	1.6min
TraceAnomaly [8]	0.94	0.67	0.78	27.2min	0.21	0.2	0.2	22.2min
CRISP [9]	0.8	0.57	0.67	26.1min	0.24	0.21	0.23	17.3min
TraceCRL [10]	0.39	0.28	0.33	8.9h	0.48	0.43	0.45	7.2h
TraceVAE [11]	0.4	0.3	0.35	1.7h	0.14	0.77	0.23	51.3min
TraceSieve [3]	1	0.74	0.85	7.6min	0.17	0.15	0.15	9.8min

Effect of Graph-based Trace Representation

- LabelEase with the graph-based representation outperforms other trace vectorization methods



Effect of Clustering Strategies

南開大學 Nankai University

- The K-means algorithm shows superior effects and the shortest computational time compared with baseline methods

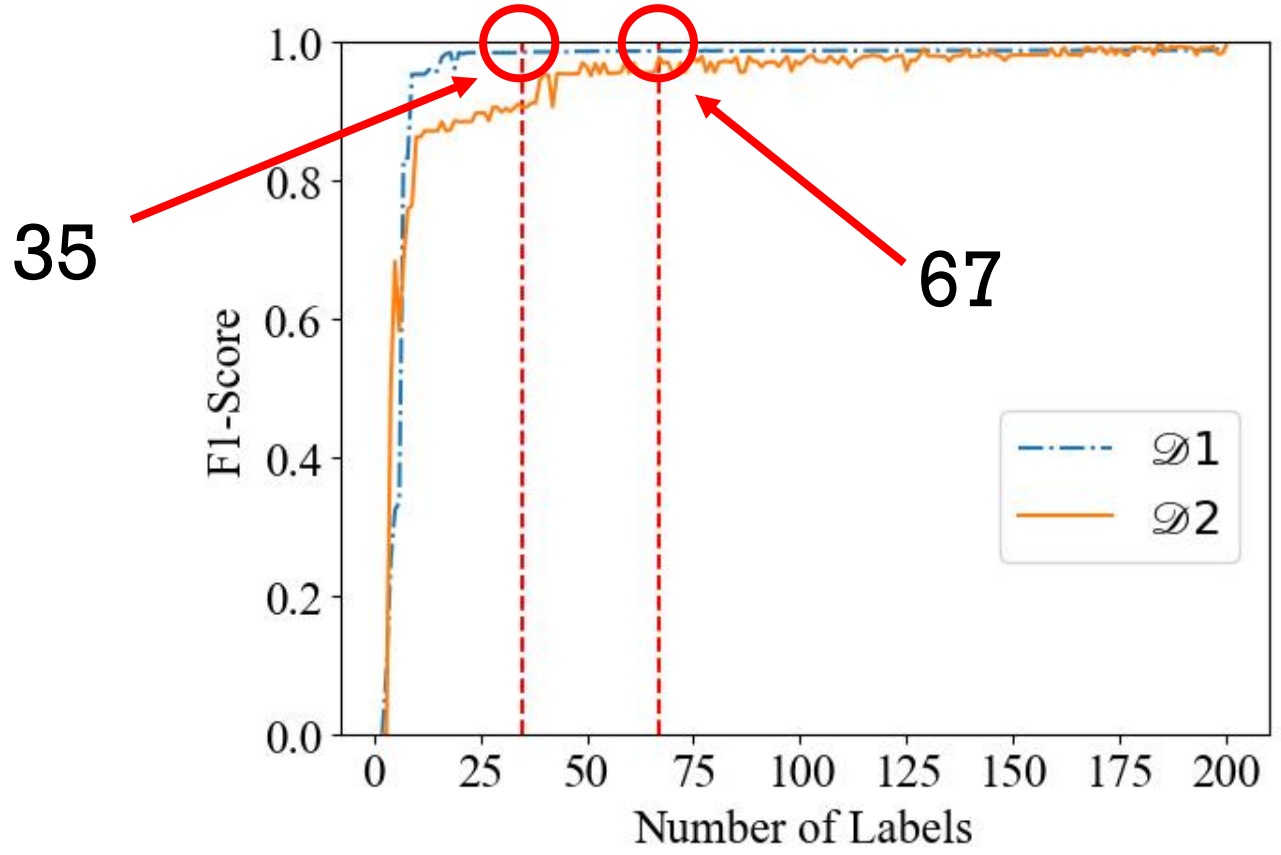
TABLE II

THE EFFECTS OF THE DIFFERENT CLUSTER METHODS ON TWO DATASETS

Approach	$\mathscr{D}1$				$\mathscr{D}2$			
	Precision	Recall	F_1-score	Time(s)	Precision	Recall	F_1-score	Time(s)
using K-means	1	0.98	0.99	6.53	0.96	0.99	0.98	21.68
using hierarchical clustering	0.99	0.98	0.99	96.9	0.96	0.95	0.96	51.98
using DBSCAN	0.99	0.92	0.95	116.07	0.83	1	0.91	97.55
using Spectral clustering	0.99	0.98	0.98	200.84	0.96	0.93	0.94	264.24
using random selection	0.8	0.2	0.32	-	0.95	0.33	0.49	-

Sensitivity of the Number of Traces to be Labeled

- How the F1-score of LabelEase changes with different trace numbers of labels



Effect of Labeling Root Cause

- The precision of fault period detection is 0.89.
  - As long as an anomalous trace is labeled, it is possible to accurately identify the period in which the corresponding fault occurred, thus enabling precise root cause labeling
- The recall is 0.44 when localizing the fault period
  - Anomalies in the unreported fault periods are not reflected in the traces but primarily in metrics and logs

Content

南開大學 Nankai University

Background

Challenges

Approach

Implementation

Evaluation

Conclusion

Conclusion

- A novel tool LabelEase, *a semi-automatic trace labeling tool for efficient and accurate trace labeling*
- A series of experimental studies to evaluate LabelEase's effectiveness and efficiency using two datasets
- Publish a high-quality dataset https://doi.org/10.5281/zenodo.13338156
- LabelEase will surely contribute to the rapid development of AIOps

Thank you!