Learning to partition unbounded graph streams

Vasiliki Kalavri, Boston University
vkalavri@bu.edu

Collaboration with Michal Zwolak, Zainab Abbas, Sonia Horchidan, Paris Carbone
(KTH Royal Institute of Technology)
Graph streams

• Possibly unbounded sequences of timestamped relationships (edges)

• User interactions, financial transactions, driver-client locations in ridesharing services, etc.

• Continuously ingested from external, often distributed sources
Graph streams

- Possibly unbounded sequences of timestamped relationships (edges)
- User interactions, financial transactions, driver-client locations in ridesharing services, etc.
- Continuously ingested from external, often distributed sources
Streaming edge partitioning

Edge stream

Partitioner

Assignment scores

On-the-fly decisions

Balance number of edges per partition
Minimize number of replicated vertices
Stateful edge partitioning has better performance
But state can grow indefinitely for unbounded streams

- Current assignment of vertices to partitions needs to be stored
- The state needs to be queried and updated for every edge in the stream
- Difficult to support high-throughput streams with **global mutable state**

Can we partition unbounded graph streams with high quality and bounded state?
1. Encode the non-Euclidean input graph stream into vectors of defined size in latent space.

2. Predict assignment probabilities for all partitions.

3. Apply assignment heuristics.

ML-added graph partitioning
With graph representation learning
Overview of GCNSplit

Serving Edge Streams

New Edge

Unlabelled Data

Unsupervised Training (Min-Loss)

GCN (GraphSage)

Plug-in Partitioner (HDRF)

Labeled Data

Supervised Training (Cross-Entropy)

Assignments

Live Graph Partitions

*Min-Cut Model Optimization

*Load Constraints

*Bounded-Size Model
GCNSplit offers partitioning quality on par with stateful partitioning.

![Graph](image)
GCNSplit requires considerably smaller state

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Edges</th>
<th>GCNSplit state</th>
<th>HDRF state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twitch</td>
<td>153K</td>
<td>1.6MB</td>
<td>4.1MB</td>
</tr>
<tr>
<td>Deezer</td>
<td>125K</td>
<td>126KB</td>
<td>5.4MB</td>
</tr>
<tr>
<td>Bitcoin</td>
<td>234K</td>
<td>166KB</td>
<td>19MB</td>
</tr>
<tr>
<td>Reddit</td>
<td>5.9M</td>
<td>385KB</td>
<td>47MB</td>
</tr>
<tr>
<td>Synthetic</td>
<td>1.3B</td>
<td>115KB</td>
<td>&gt;116GB</td>
</tr>
<tr>
<td>Papers</td>
<td>1.6B</td>
<td>147KB</td>
<td>&gt;116GB</td>
</tr>
</tbody>
</table>
GCNSplit can leverage parallelism to improve throughput
GCNSplit can generalize to unseen graph streams

- Twitch user-to-user networks speaking various languages
- Training on 10K edges sampled from the DE and RO networks
Limitations and future work

• Performance is highly dependent on the **quality of training data**
  • Rich feature sets lead to lower replication factor
  • High partitioning quality as long as the graph stream’s characteristics do not change drastically
• In case of major concept drift GCNSplit behaves like **hash partitioning**
  • Constraints guarantee good load balance
  • Partitioning decisions equivalent to random assignment
• **Continual learning** methods can be used to update the model *incrementally*
  • Detect drift and use graph sampling to incorporate new knowledge while maintaining old one
Learning to partition unbounded graph streams

Vasiliki Kalavri, Boston University
vkalavri@bu.edu

Collaboration with Michal Zwolak, Zainab Abbas, Sonia Horchidan, Paris Carbone
(KTH Royal Institute of Technology)

LDBC TUC, Aug 16