Optimize the Database Index Structure

Presented by: XingChi Wang
Opportunity: Why Learned B-tree?

Challenges of Traditional Indexes

- An index is a structure that sorts the values of one or more columns in a database table.
- Most common database index: B-tree.
- B-trees know nothing about the distribution of the data.
- B-trees do not take advantage of more common patterns prevalent in datasets.

Challenges with Search Time

- Search time for B-Tree index structure is $O(\log n)$.
- For a very large $n$ this value can be grow extensively.
- Can we link machine learning with index structure?
Solution: Overall Structure

Stage 1

Stage 2

Model 1.1

Model 2.1

Model 2.2

Model 2.3

Stage 3

Model 3.1

Model 3.2

Model 3.3

Model 3.4

How can we create these models? CNN? Or.. Capsules?
Solution: Capsule B-Tree?

Capsule Networks: are Neural networks that try to do Inverse Graphics.
Many capsules, each capsule tries to predict the presence and instantiation parameters of an object at a given location.
Achieved state of the art performance and better results than CNN.
Solution: Capsule Implementation

How can capsule handle complex data distribution? Composed images?

Primary Capsules

Strong Agreement

Routing by Agreement

Inverse Rendering

Image
Conclusions and Next Steps

Routing by Agreements

- Can handle complex distributions.
- Tends to choose the solution that makes all the capsules achieve accurate predictions for the capsules in the next layers.
- Capsules can be applied to the index problem to outperform CNN.
- Requires less training data.

Next Steps...

- Start the implement based upon Capsule Networks approach.
- Collect real-world datasets to start the experiments
- Start experimental study to compare between B-tree and our learned index structure
THANK YOU

nashaata@ualberta.ca
aindrila@ualberta.ca
xingchi@ualberta.ca