Microsoft

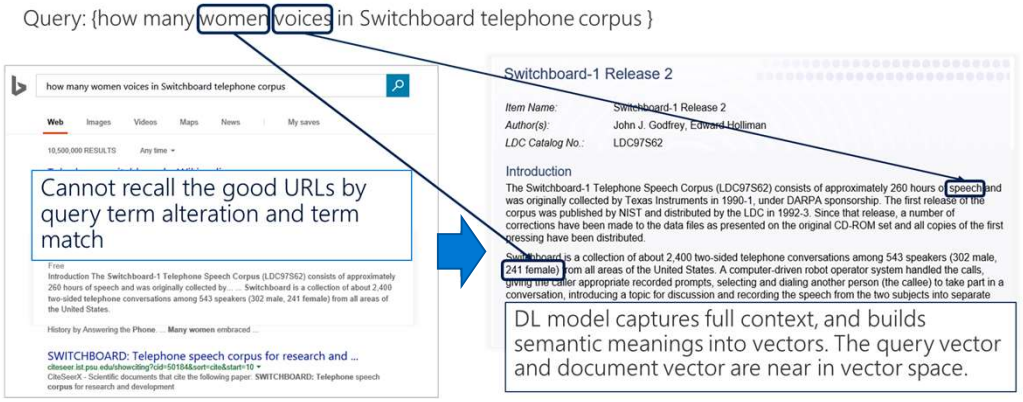
Deep Learning Inference Service at Microsoft

Jonathan Soifer, Software Engineer

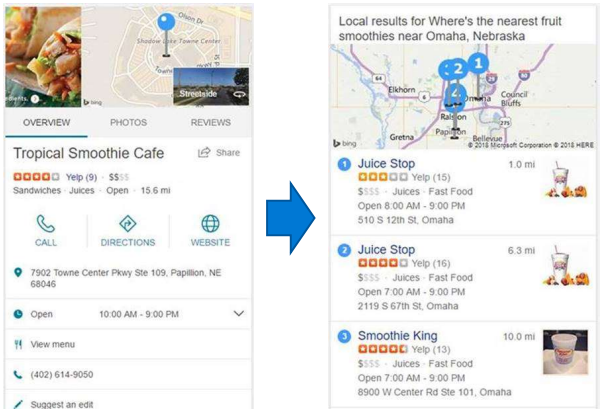
OpML ‘19

Deep Learning at Microsoft

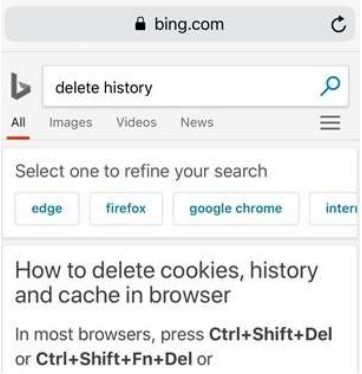
Web Search



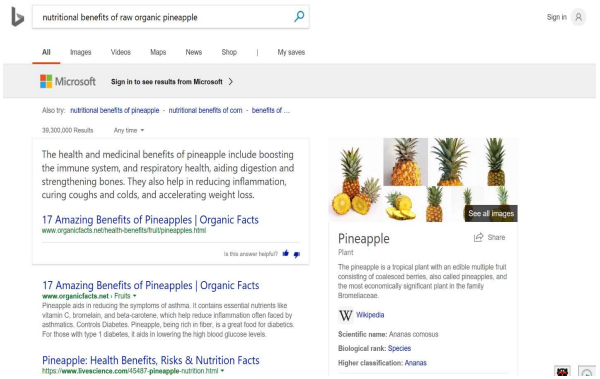
Entity Search



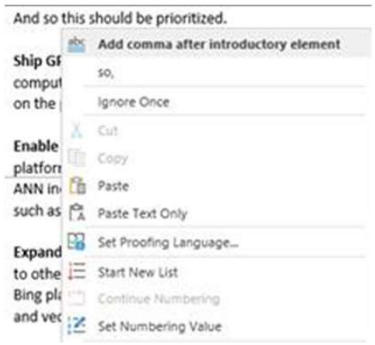
Conversational Search



QnA at Web Scale



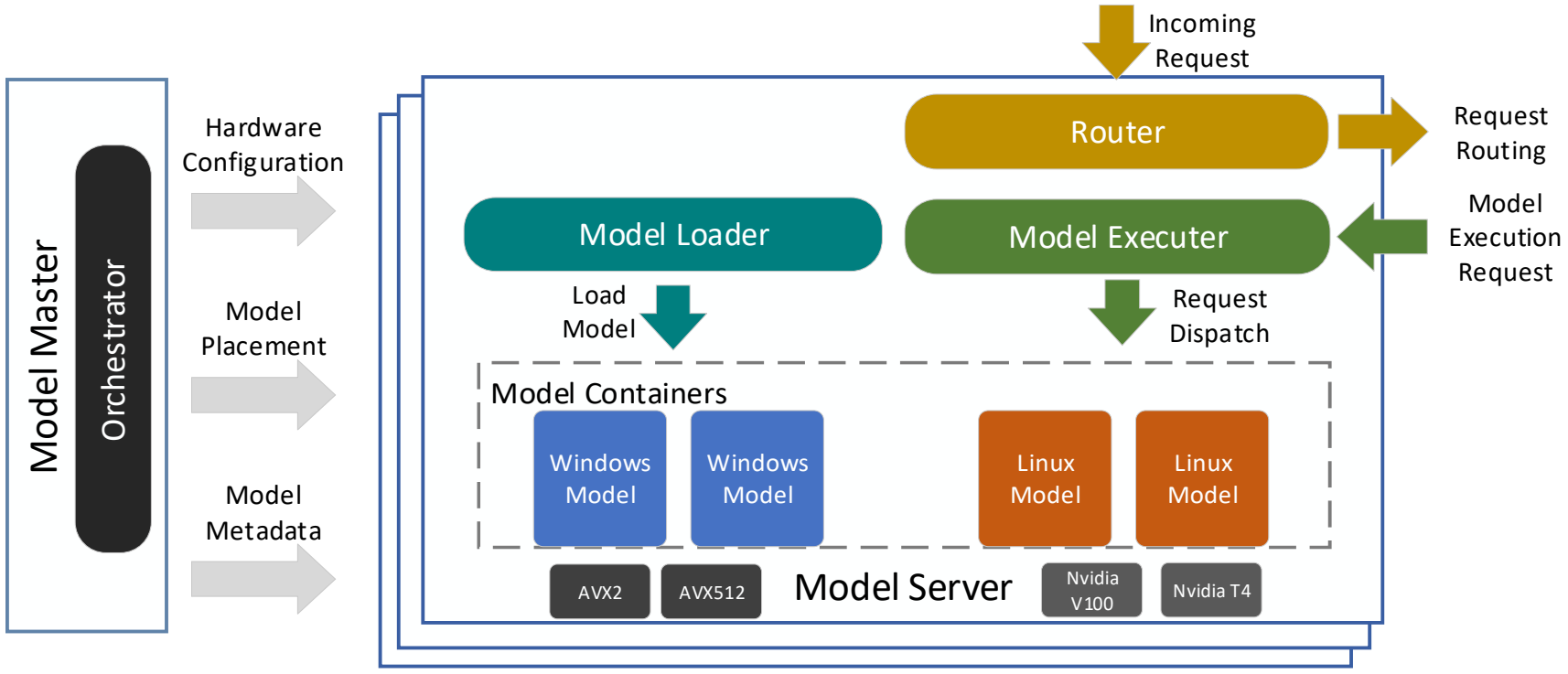
Grammar Checking



Deep Learning Inference Service

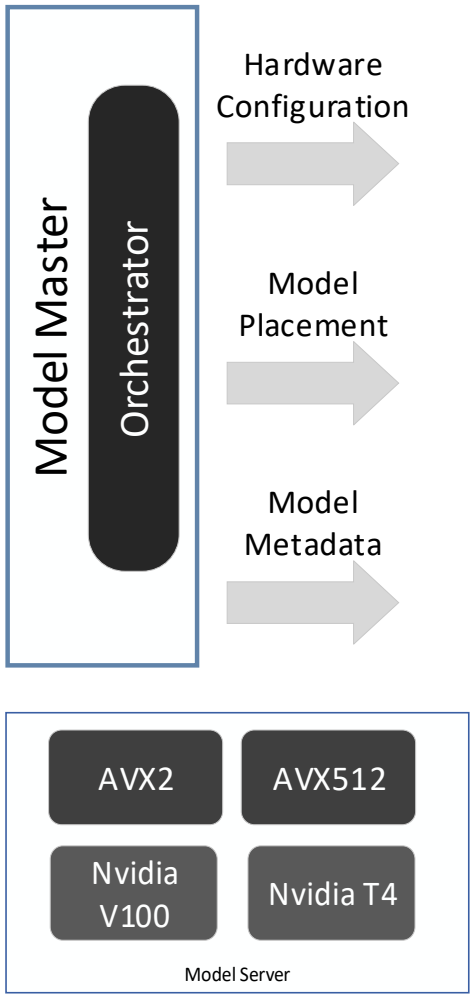
Performance	• Single digit millisecond inference
Scalability	• Millions of DNN model inferences per second • Tens of thousands of servers • Forty data centers worldwide
Agility	• New model deployment in minutes
Flexibility	• Variety of deep learning frameworks (ONNX, TF, PyTorch, etc) • Variety of hardware (CPU, GPU, etc) • Support Linux and Windows

System Overview



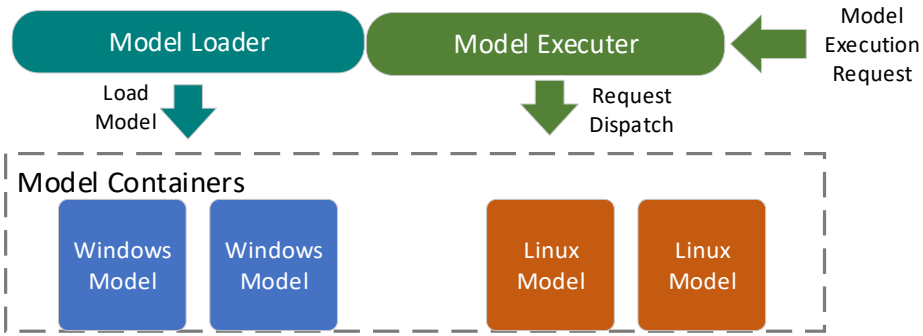
Intelligent Model Placement

- Model master has overview of all server resources
  - Available CPU cores and memory, CPU instruction set, etc
- Model master has overview of all model resource requirements
  - Determined via cloud validation pipeline
- Multi-tenant model hosting
  - Reduces resource fragmentation
- Dynamic model scaling
- Custom hardware management



Resource Isolation

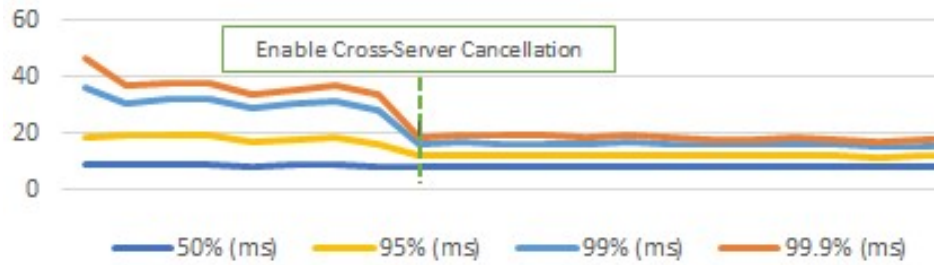
- For low-latency inference, data access should be localized
- Model instances are isolated in containers
  - CPU affinity -> data locality
  - NUMA affinity -> fast memory access
  - Memory restrictions -> model won't go to disk
- Reduces interference in multi-tenant server
- Efficient inter-process communication between server and model



Efficient Routing

- Unique challenges
  - Burst traffic
  - Large batch requests
  - Tail model latency is near performance SLA
- Support configurable routing algorithm per model
- Support backup requests with cross-server cancellation
- In practice, less than 1ms of system overhead





Other Notable Features

- Model optimization through DeepCPU
  - Presented in an earlier talk
- Service runs on both Linux and Windows
- Pluggable orchestrator
  - Supports Model Master, Yarn, and Kubernetes
- No-downtime model update
- Full per-model monitoring and metrics

Takeaways

- Deep learning at scale presents unique challenges for distributed serving platforms
- Intelligent model placement, resource isolation, and efficient routing are critical components for Microsoft-scale inference needs
- These techniques provide the foundation for the future as models become larger and more compute-intensive

Microsoft

Questions?