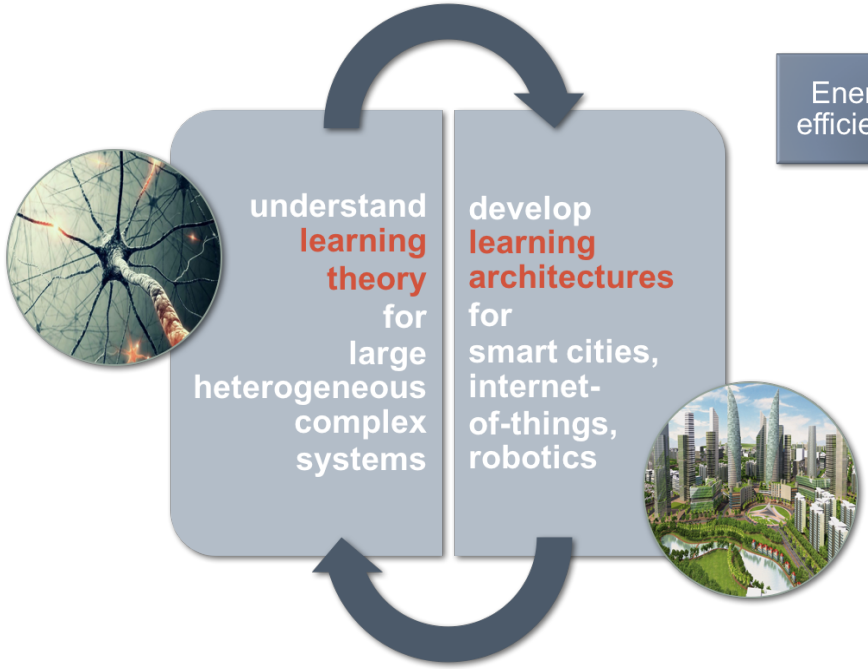
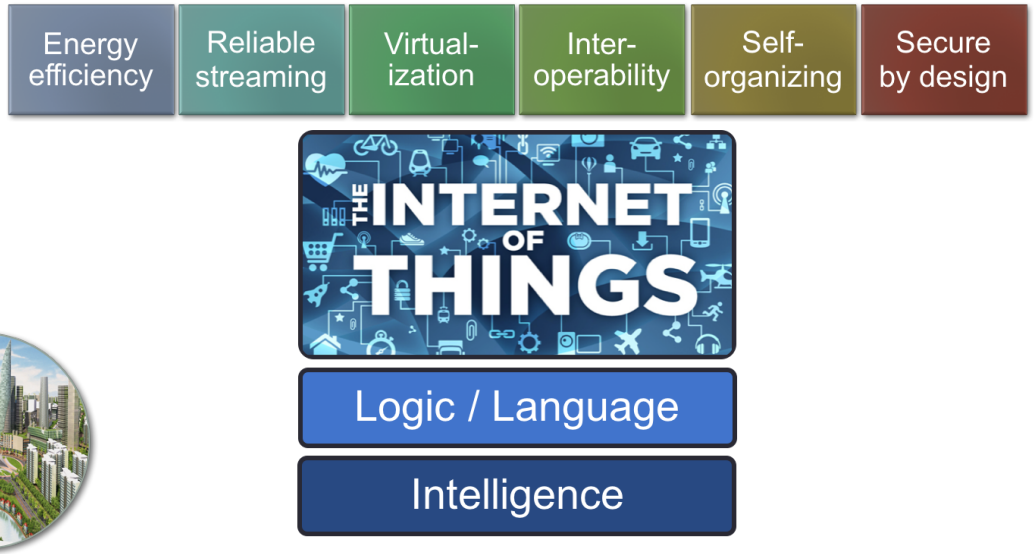
Shaowei Lin (SUTD)

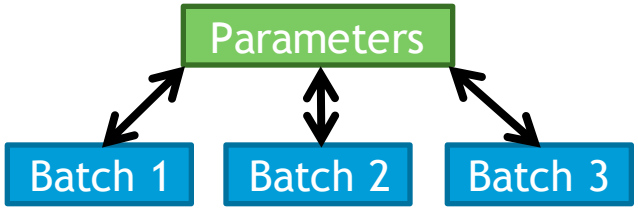
2006-2011	Ph.D. Mathematics, Berkeley	algebraic singularities in Bayesian statistics
2011-2012	Postdoc, Berkeley-Stanford	mathematical challenges in deep learning
2012-2015	Group Leader, S&S, I2R	deep learning for wireless sensor networks
2016-now	Asst. Professor, ESD, SUTD	distributed functional machine intelligence





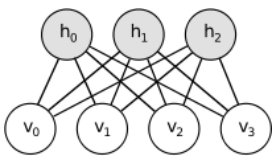
Distributed Algorithms

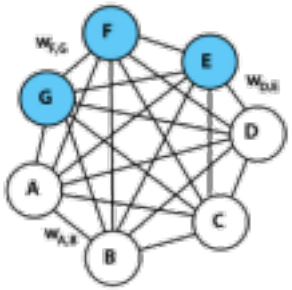
Model- vs Data-parallelism



"What we're missing is the ability to parallelize the training of the network, mostly because the **communication is the killer**... Eventually a lot of the deep learning task will be done on the device, which will keep pushing the need for **on-board neural network accelerators**" – LeCun, 2015

Minimum Probability Flow

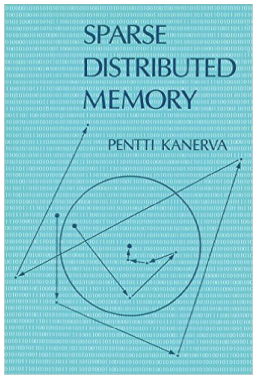


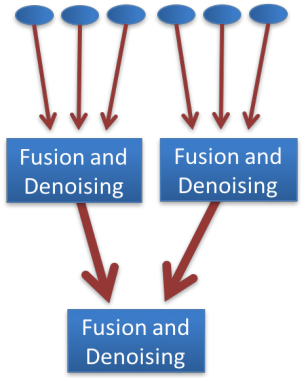


- min **KL[X_0||$X_\infty(\theta)$]** ~ Maximum likelihood
- min **KL[X_0||$X_\infty(\theta)$] – KL[X_1||$X_\infty(\theta)$]** ~ Contrastive divergence
- min **KL[X_0||$X_\varepsilon(\theta)$]** ~ Minimum probability flow **(Hebbian learning!)**

Hyperdimensional Computing

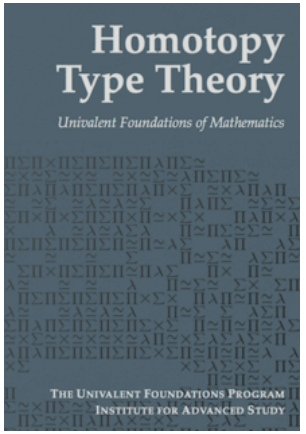
Representing words in 10,000-dim binary space (random indexing). With $2^{10,000}$ distinct vectors, code words are sparsely located. Distributed representation. Nearby vectors represent similar words. Combine words using addition and matrix multiplication.

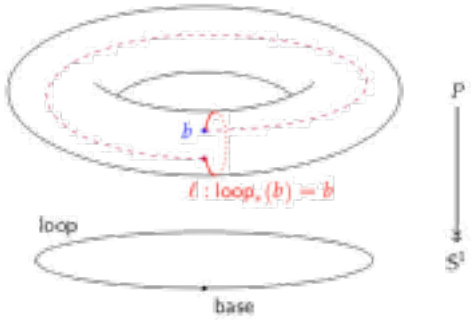




Functional Language

Language: how do neural networks talk to each other?
Reproduction: how do programs write better programs?





Return of Type Theory

- Functional programming in data analytics
- Linked data in semantic web
- Category theory in algebraic geometry
- Homotopy type theory (HoTT) in mathematical foundations

HoTT – Topological Interpretation of Logic

- (Term, Type) = (Proof, Proposition) = (Program, Type) = (Point, Space)
- Proof that terms a, b are equal = Paths between points a, b
- Proof of P(a) can be transported along path to get proof of P(b)

Proof-checking and Proof-assistants

- LaTeX::Papers ↔ Coq::Proofs
- Writing a Program ↔ Proving a Theorem
- Pattern-matching, tactics – easy for neural nets

Human vs Machine Language

- Sentences are concatenation of words
- Programs are compositions of functions