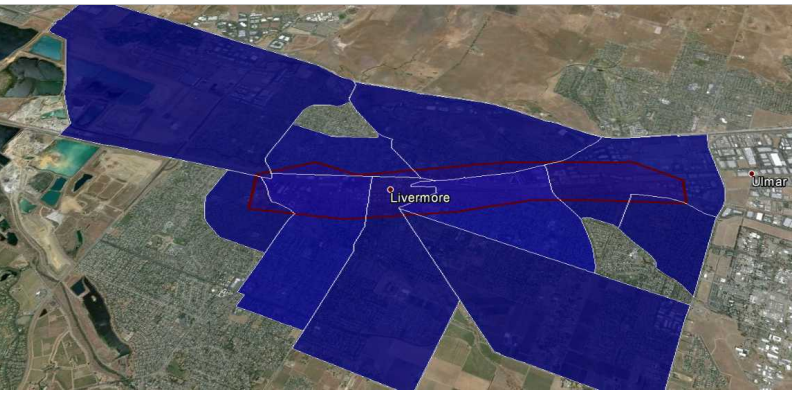
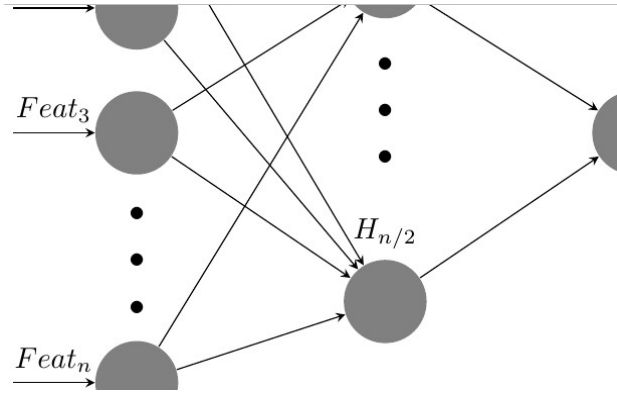
Exceptional service in the national interest

SAND2015-5768PE

Sandia National Laboratories



Score	Name
0	Add Two Numbers GPB Temp
0	Census Data Lookup
0	Integer Pass-through
0	Top Populated US Cities
0	Add Two Numbers XML Temp
0	XSLT Transform
0	Cascading Add Two Numbers



Census Modeling & Ranking Models

Max Smith, Rising Junior BSE Computer Science

Nerayo Teclemariam, 8112

Homeland Security Science and Technology

SHERPA

MICHIGAN ENGINEERING UNIVERSITY OF MICHIGAN

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2015-XXXXP

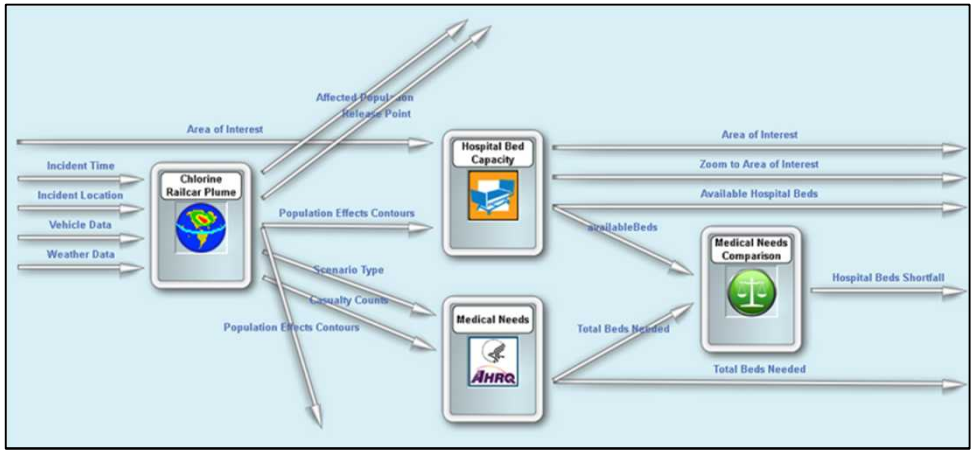
Overview

- SHERPA Introduction
- Census Model
- Template Ranking Scheme
  - Learning to Rank

SHERPA

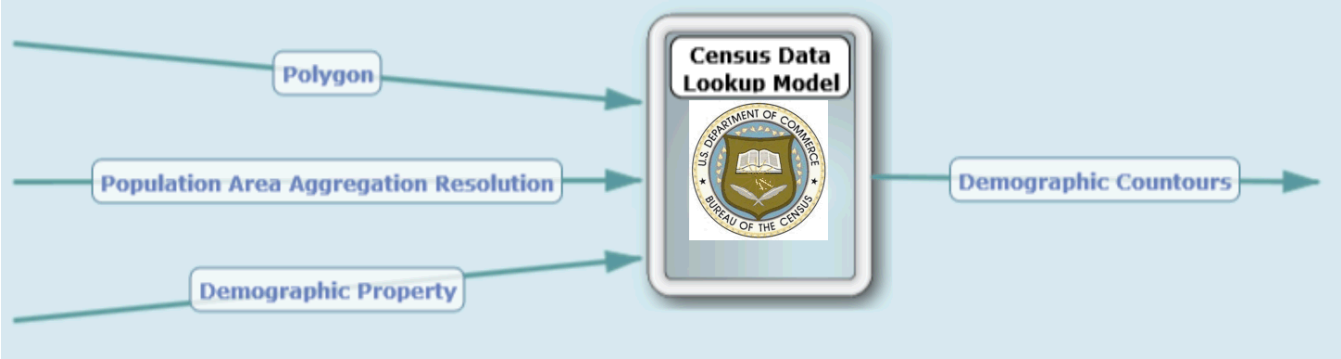
- Modeling & simulation for exercises, responses, and planning
- Focuses on homeland security emergencies

- Collections of models are called templates



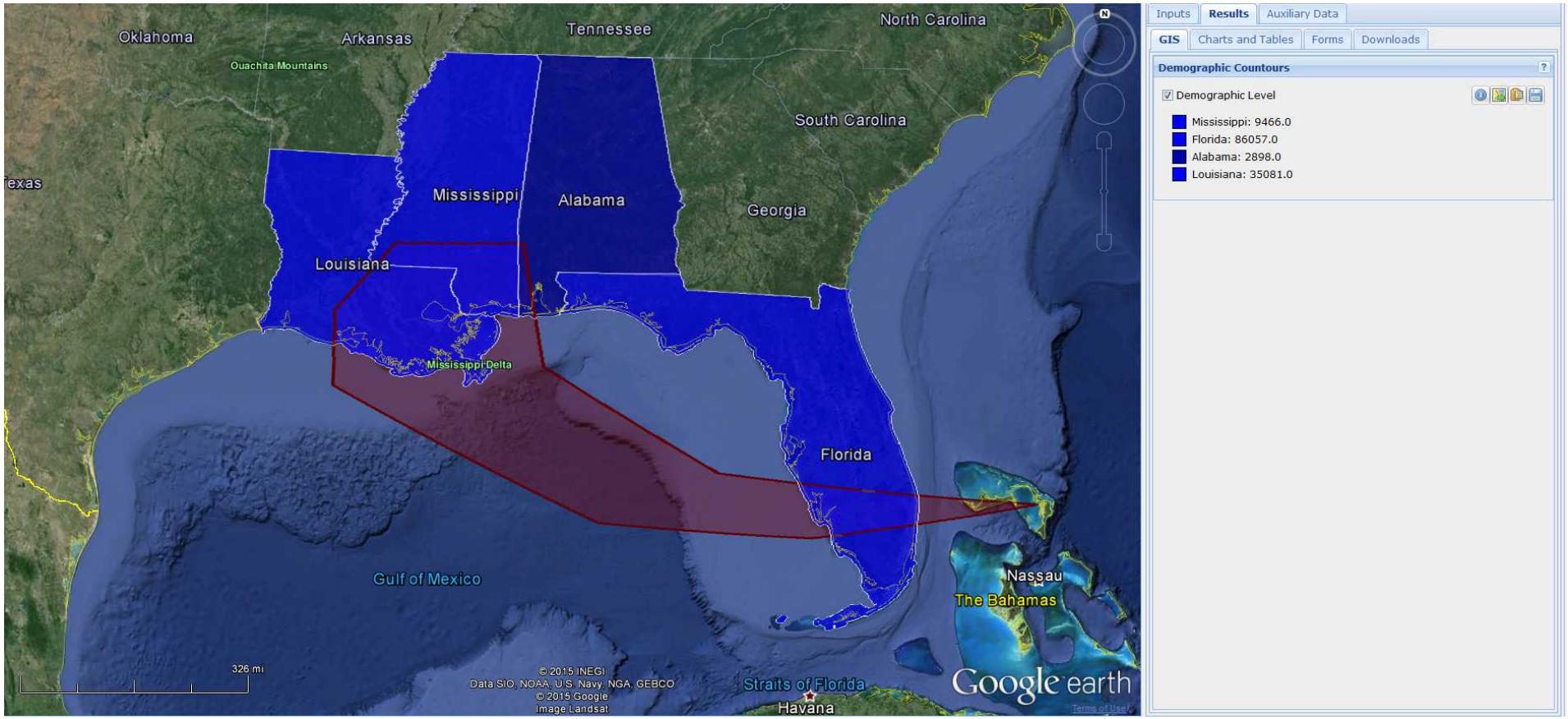
Census Data Lookup Model

- Given an Area of Interest:
  - Return Demographic counts
    - Age, Sex, Race, etc.
  - Delimited by Geographic resolution
    - State, County, Tract
- Recursive approximation



Census Data Lookup Model

- Population >85 years old by state



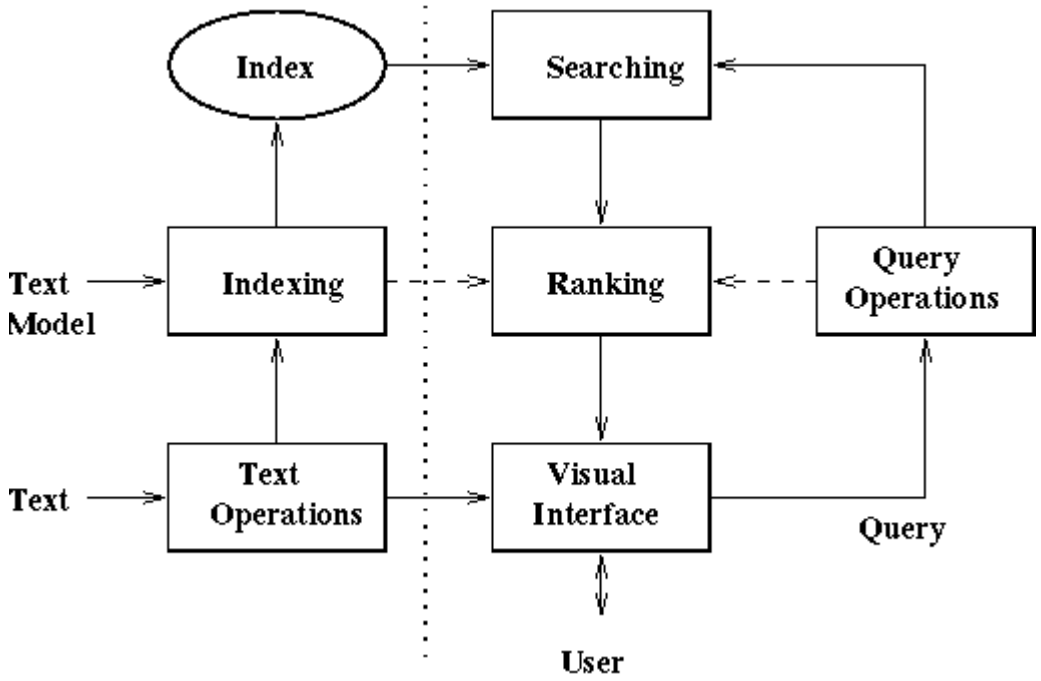
How to Organize Templates

- Hundreds of models (and growing!)
- Can permutate into billions of templates

- How do I find the templates I want to use?

Information Retrieval

- Obtain information relevant to a query



http://people.ischool.berkeley.edu/~hearst/irbook/1/node5.html

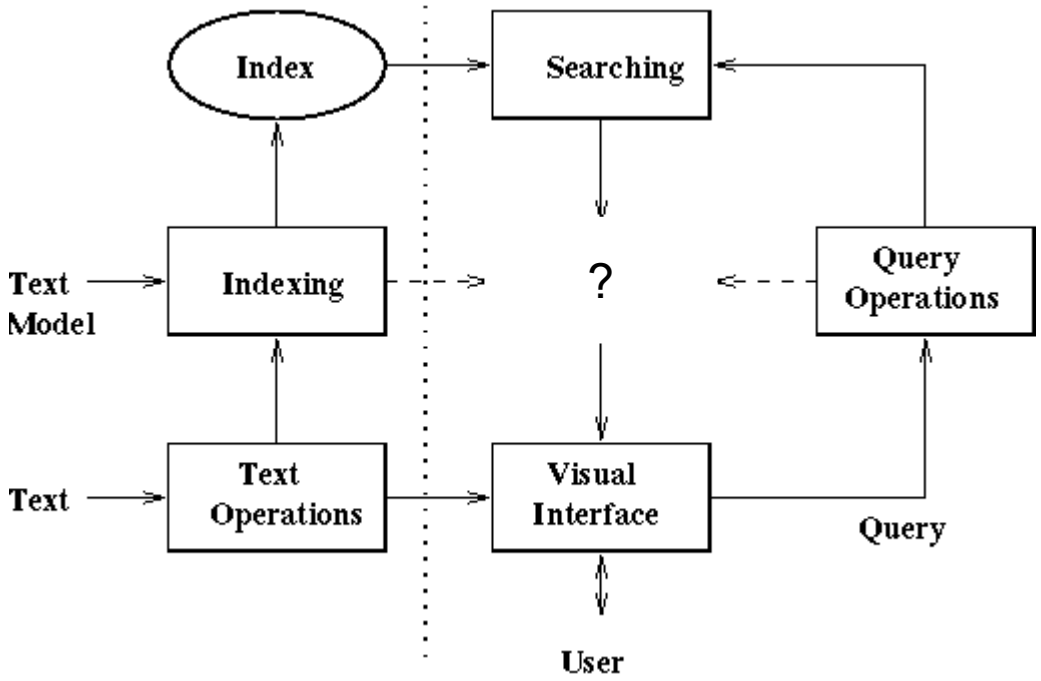
Information Retrieval

- Search via text matching implemented
- No scoring system (all assigned zero)

Score	Name	Runnable	Interactive	# Runs
0	Add Two Numbers GPB Template	✓		0
0	Census Data Lookup	✓		0
0	Integer Pass-through	✓		0
0	Top Populated US Cities	✓		0
0	Add Two Numbers XML Template	✓		0
0	XSLT Transform	✓		0
0	Cascading Add Two Numbers and get City List Template	✓		0
0	Cascading Add Two Numbers XML Template	✓		0

Information Retrieval

- Missing “Ranking”



http://people.ischool.berkeley.edu/~hearst/irbook/1/node5.html

Ranking

- An initial framework was first built
- Assigned ranks to models based on an average of 3 properties
  - Agility
  - Quality
  - Access
- Ranked templates based on the “weakest link” model in use

- Need more intelligent way to evaluate properties relevance

Learning to Rank

Problem Statement:

Suppose $\mathcal{Q}$ is the set of queries q, and $\mathcal{D}$ is the set of all documents d (with features $\vec{x}$). We aim to train a ranking model

$$f(q, d) \equiv f(\vec{x})$$

to assign ranks given document-query pair, or feature vector.

Learning to Rank

- Using machine learning to model rankings in information retrieval
- Ranking systems:
  - Pointwise
  - Pairwise
  - Listwise

Learning to Rank

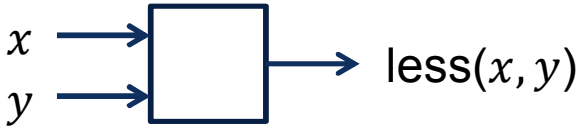
- Using machine learning to model rankings in information retrieval
- Ranking systems ($x \in \mathcal{D}$):
  - Pointwise

  - Pairwise
  - Listwise

Learning to Rank

- Using machine learning to model rankings in information retrieval
- Ranking systems ($x, y \in \mathcal{D} | x \neq y$):
  - Pointwise

  - Pairwise

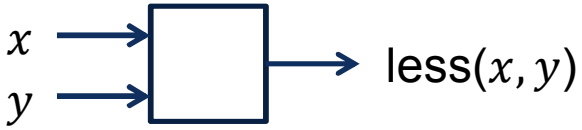


  - Listwise

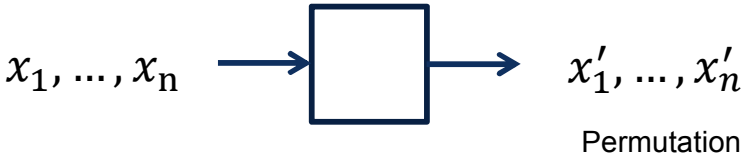
Learning to Rank

- Using machine learning to model rankings in information retrieval
- Ranking systems ($x_i, x_i', y \in \mathcal{D} | x \neq y$):
  - Pointwise

  - Pairwise

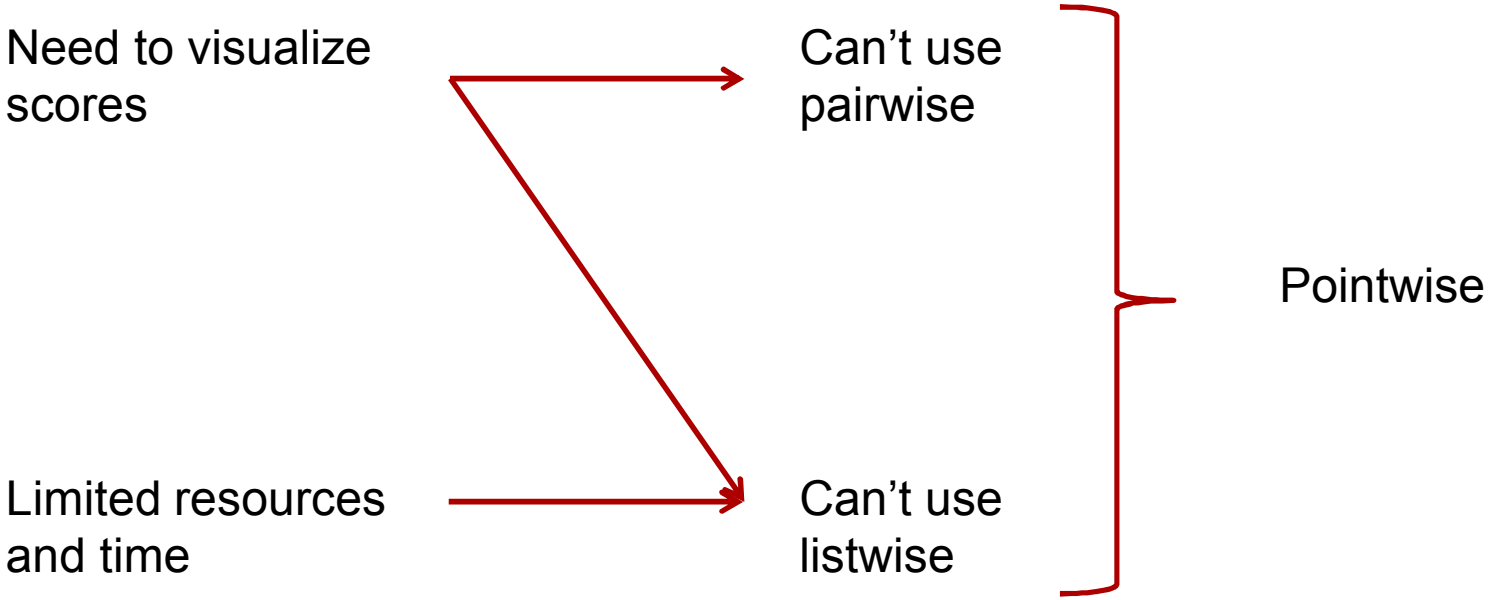


  - Listwise



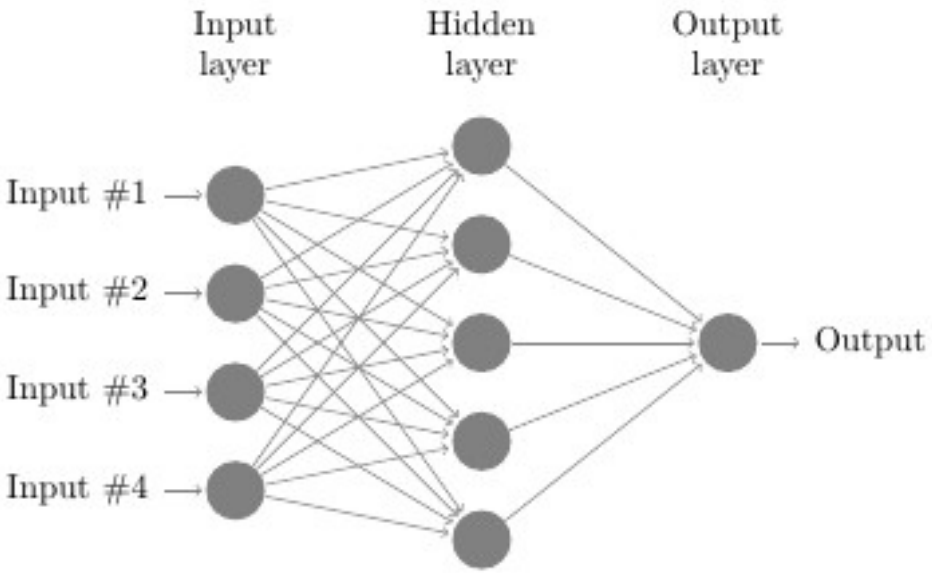
Learning to Rank

- Which system do we implement?



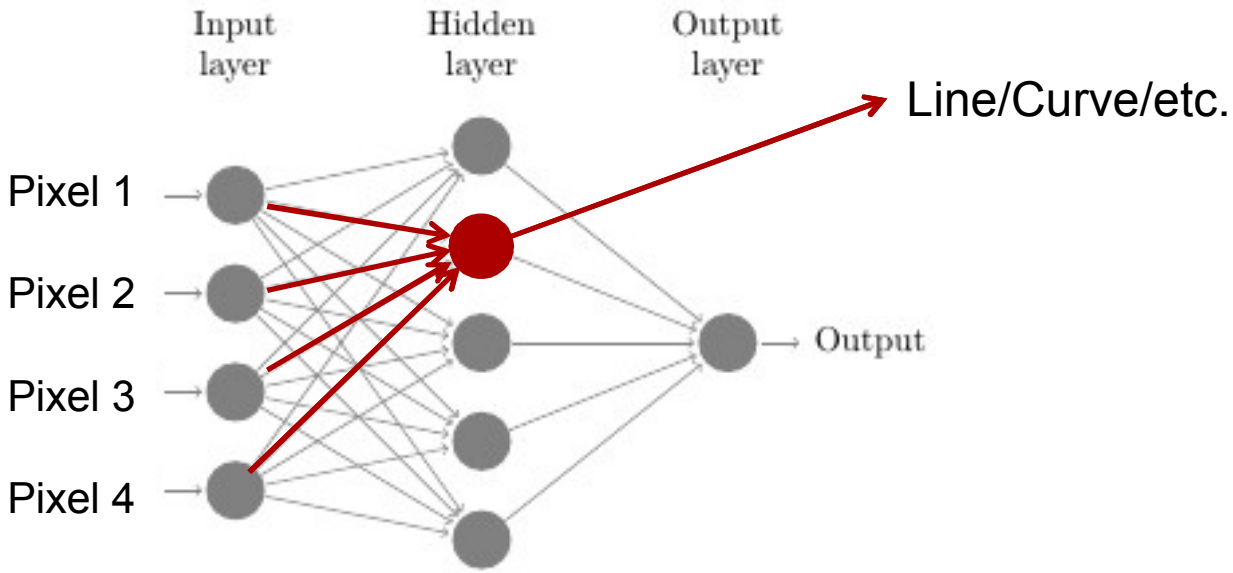
Multilayer Perceptron

- More complex features are learned through error minimization



Multilayer Perceptron

- More complex features are learned through error minimization



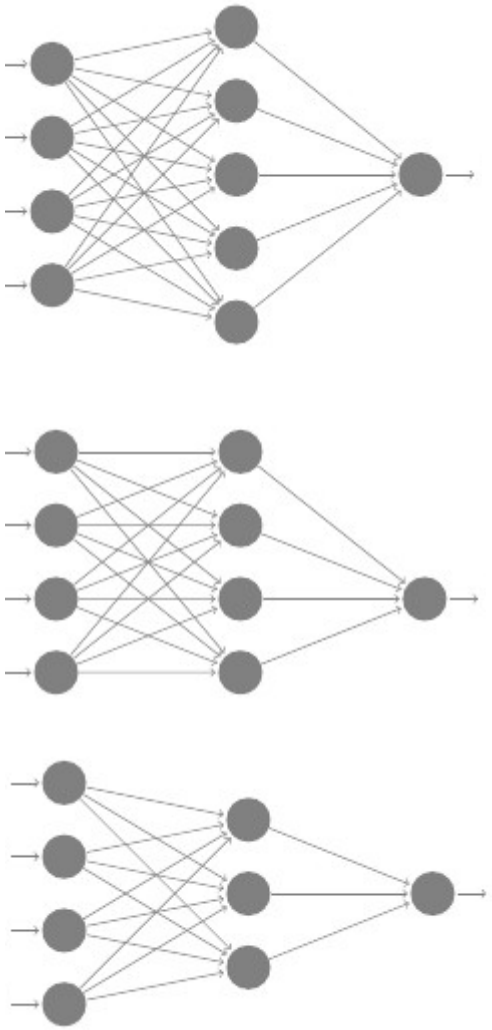
Multilayer Perceptron

- Data broken into 3 groups:
  - Training (n=15)
  - Cross-validation (n=5)
  - Testing (n=5)

Multilayer Perceptron

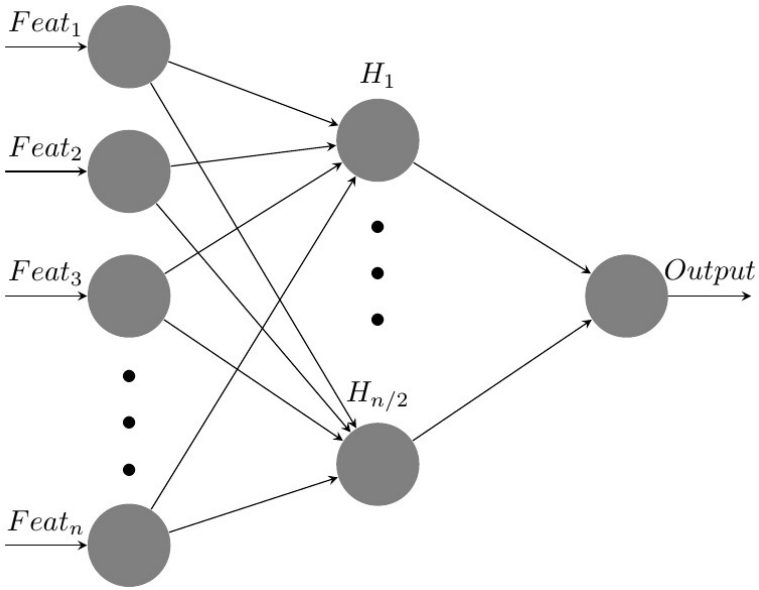
- Data broken into 3 groups:
  - Training (n=15)
  - Cross-validation (n=5)
  - Testing (n=5)

- Several architectures were trained using training set



Multilayer Perceptron

- Data broken into 3 groups:
  - Training (n=15)
  - Cross-validation (n=5)
  - Testing (n=5)
- Several architectures were trained using training set
- Best performing model is selected from cross-validation set
  - Ensures more robust model as it's not fitted to the test error



Example Output

- Test set error: ~6%
- Example outputs:

Correct	Estimate
9.0	9.0423
9.0	9.0423
9.0	9.0668
9.0	9.0668
8.0	9.2893
6.0	5.9320
6.0	6.2783
6.0	5.9320
10.0	9.4719

Improvements/FutureWork

- Extremely small amount of data (n=25), but very hard to amass
- Explore listwise approach (and other algorithms)
- Explore additional more complicated features
  - Click-through-rate
  - Number of runs
  - Word2Vec analysis
  - NLP

Value Add

- Users can now easily find the most relevant templates for their needs. This allows for the best data and planning to occur, since the most robust and relevant simulations are used.
  - Greater Accuracy
  - Better User Experience

Conclusion

- Many thanks to whom none of this would've been possible without: Nerayo Teclemariam, Russell Gayle, Trisha Miller, Zach Heath, Stephen Mueller, Joshua Bauer, Lynn Yang, Lynne Burks, Andrew Rothfuss, and Colleen Meyerkorth

- DHS S&T Resilient Systems Division: Chase Garwood

- Questions?

References

- Li; “A Short Introduction to Learning to Rank.”
- Cao, et al.; “Learning to Rank: From Pairwise Approach to Listwise Approach.”
- Burges et al.; “Learning to Rank using Gradient Descent.”
- Chapelle, Chang; “Yahoo! Learning to Rank Challenge Overview.”
- Sculley; “Large Scale Learning to Rank.”
- http://people.ischool.berkeley.edu/~hearst/irbook/1/node5.html