Optimizing content delivery through machine learning

James Schneider
Anton DeFrancesco

Obligatory company slide

SECURBORATION

Our Research Areas

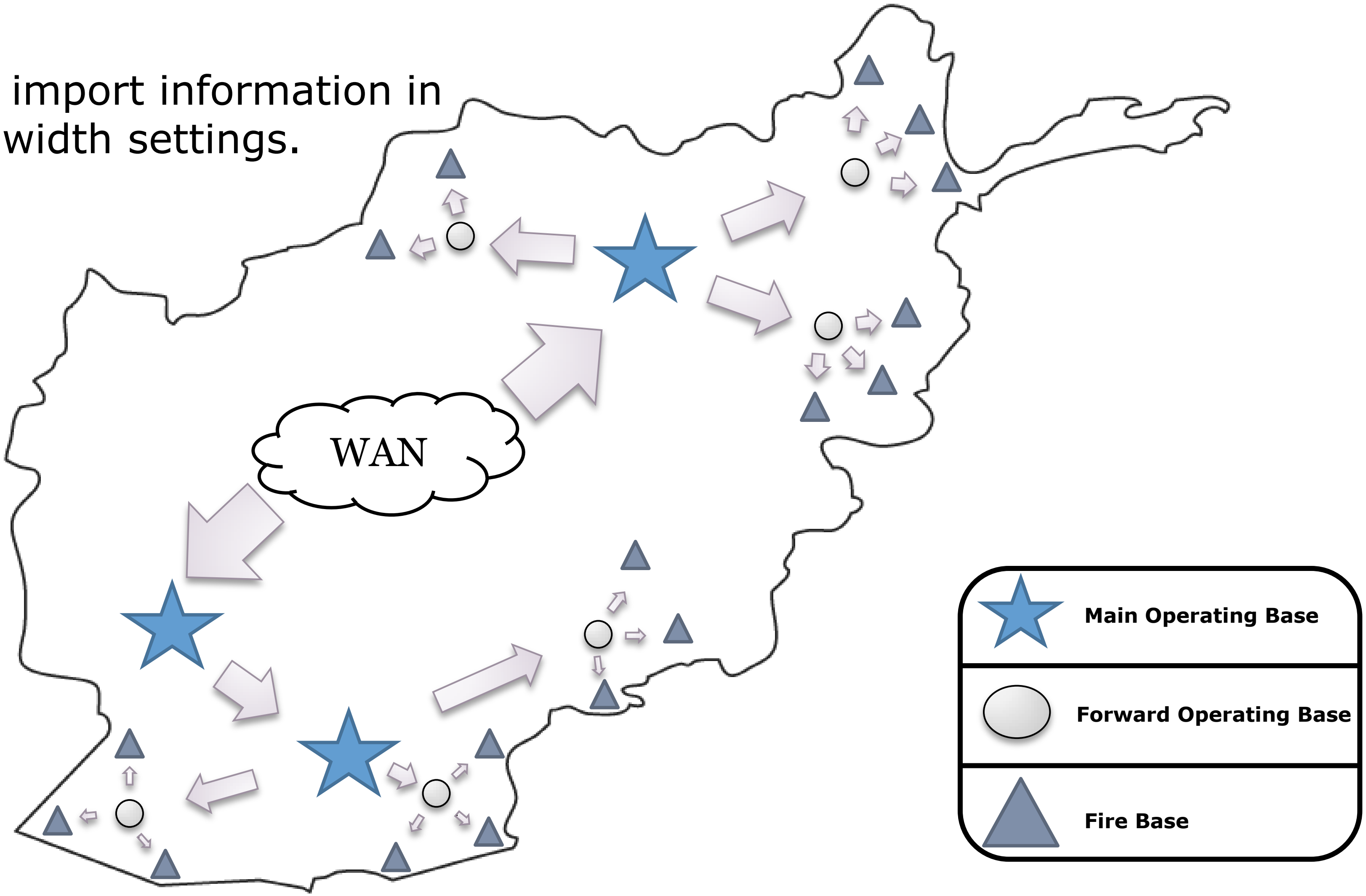
Massive parallel processing

Machine learning

Natural Language Processing

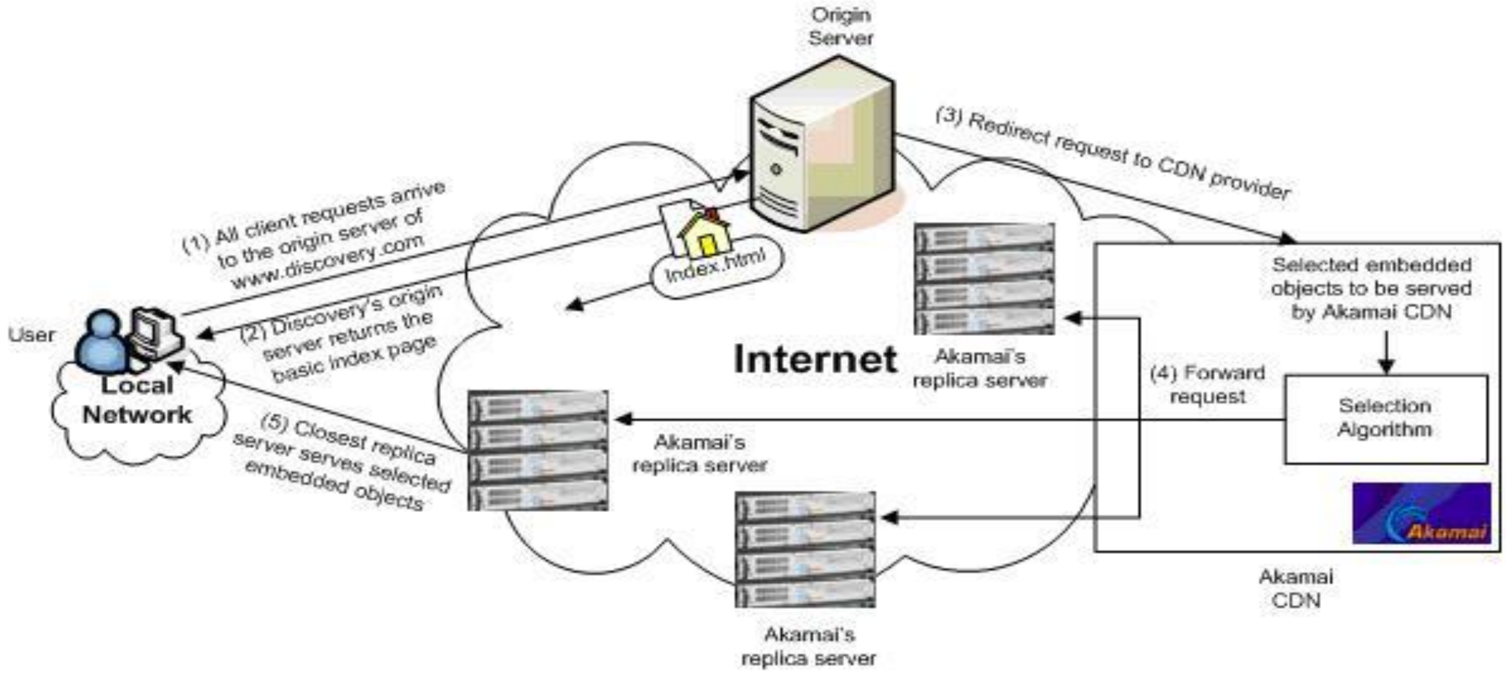
The problem

Prioritize import information in low bandwidth settings.



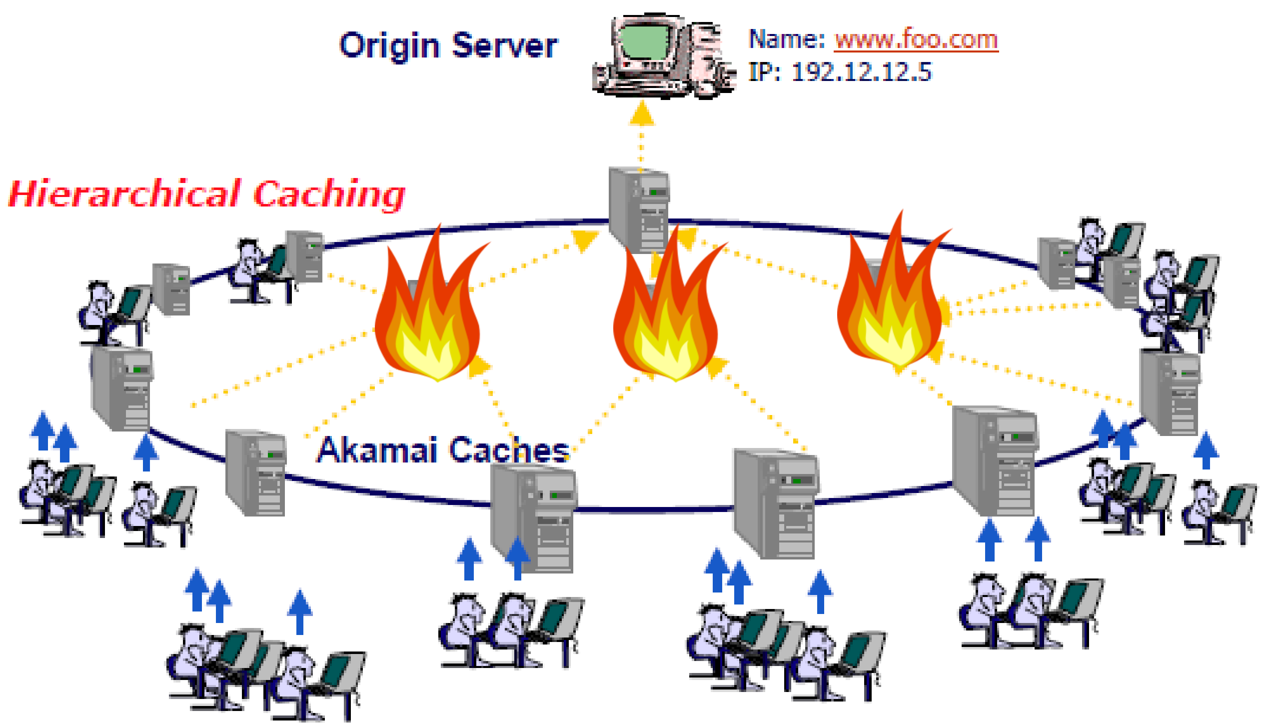
Solved Problem?

- Content Delivery Networks or CDN's Provide easy dissemination of data
  - Akamai, Azure, CloudFront, CloudFlare



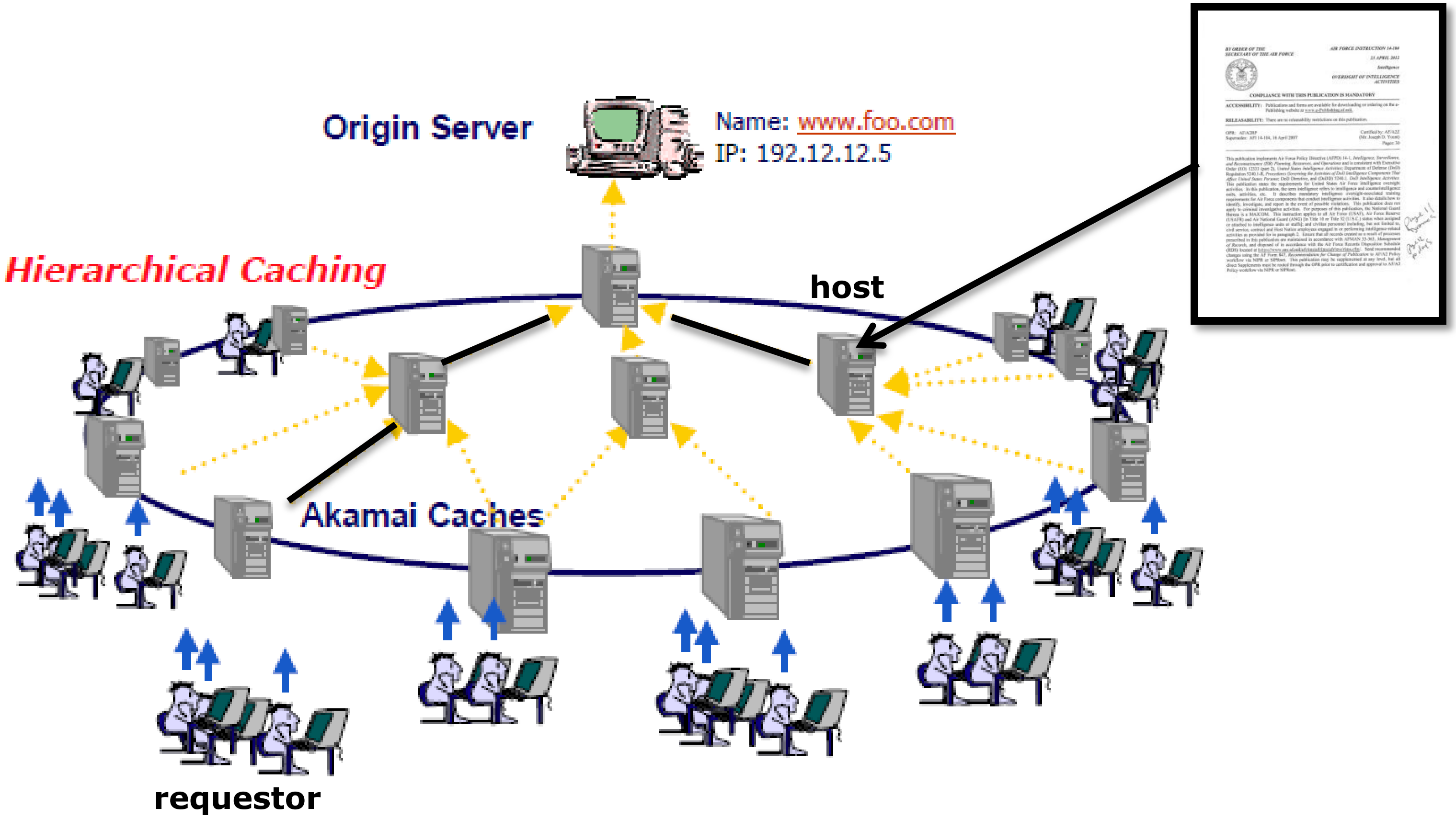
- There are still issues with CDNs

Everyone requests at once



The average transmission rate decreases *(aka the buffering problem)*

Local host doesn't have the file



The local transmission rate is now lower for the requestor.

Military environment

- Military environment offers unique challenges they bring to Content Delivery Networks
  - Such as …

- In normal conditions, replication servers are not subjected to daily motor rounds.

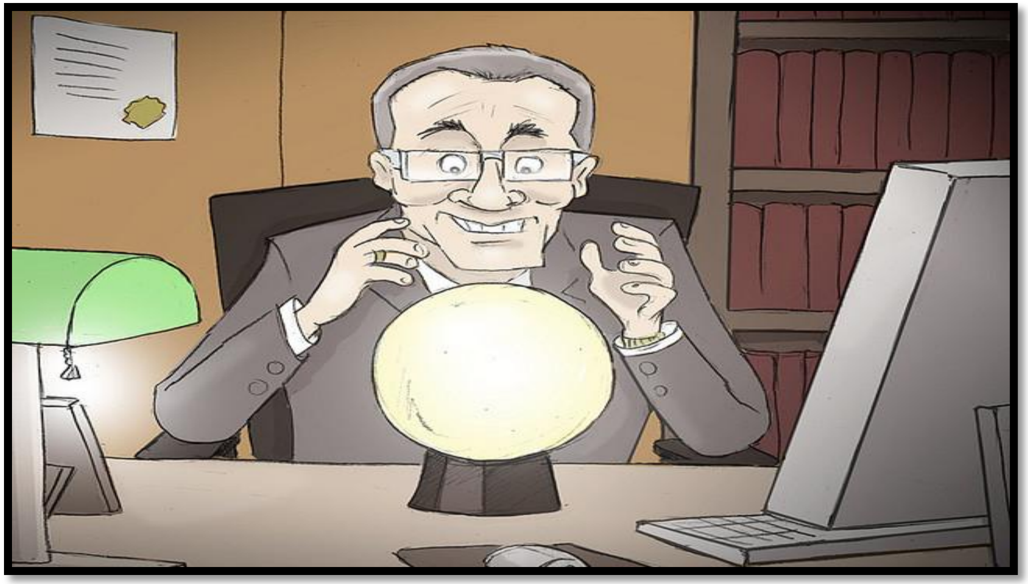
Military environment additional issues

- Extreme bandwidth restricted
  - Multiple layers of security
- Environmental disruption
  - Microwave and satellite equipment, rain or dust easily disrupt
- Enemy congestion
  - Ddos attacks and flooding of receiving towers with noise are common attacks
- Huge file sizes
  - MQ-9 requires 2 Mb/s data link
- Extremely large burst moments
  - Everyone needs the data for their mission now.

Solutions

- We can optimize information flow through our network increasing throughput
  - This is the goal of scheduling algorithms

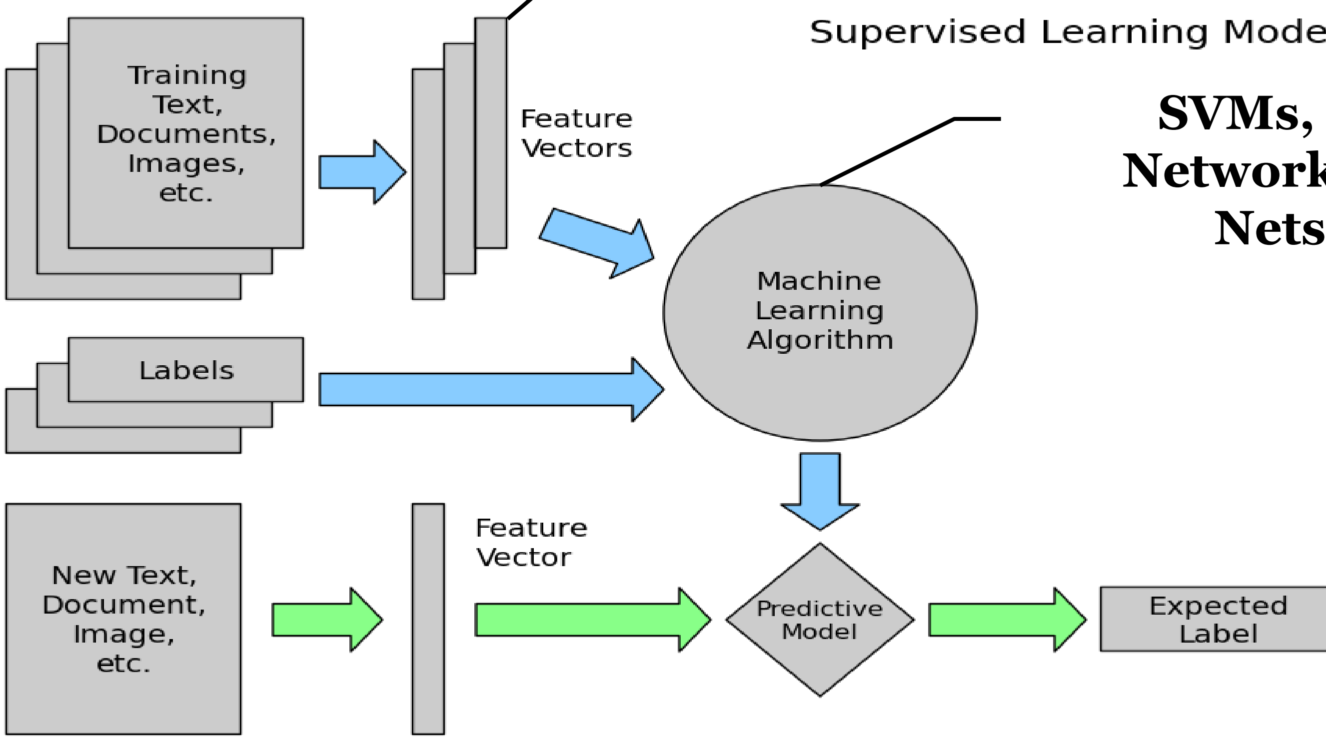
- But can we send information before its requested instead?



Machine learning

- Sadly, we don't have crystal balls, but we can use machine learning.

Issue, You need to know which features to extract!



SVMs, Neural Networks, Bayes Nets, etc.

Feature selection

- What is a feature?
  - Question: "What do children love about SpongeBob?"

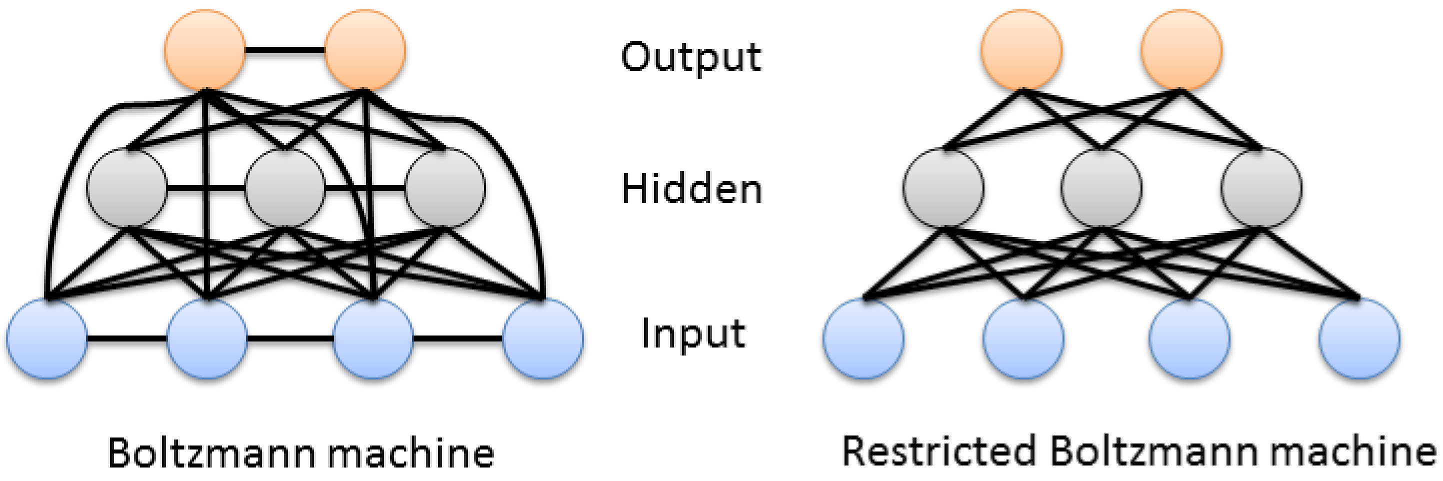
- *His personality* – **ok feature**
- *Color scheme* – **bad feature**
- *The number of holes in body* – **bad feature**

  - These are all **features** of SpongeBob, and **feature selection** is just selecting those that help us predict what makes him SpongeBob.

- Really SpongeBob is the combination of his features we call this a higher level feature. SpongeBob exists in some higher dimensional space as a linear combination of features.

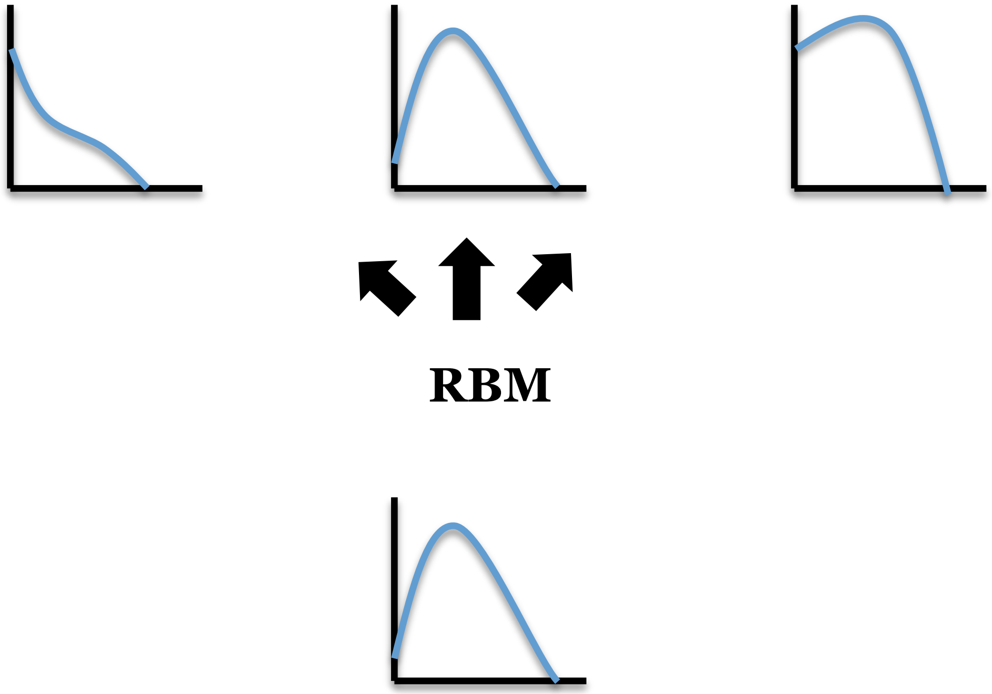
How to do this in automated fashion?

- Fully connected neural network
  - Activations are probabilistic in nature don’t depend on energy function



Restricted Boltzmann Machines

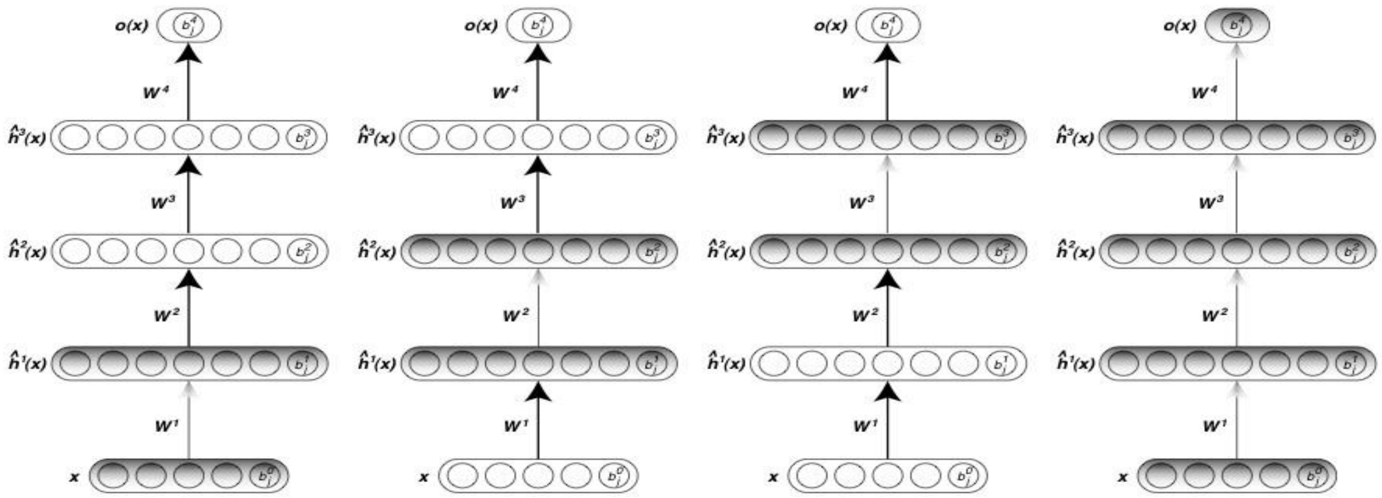
- What is this thing trying to do?
  - Guess the distribution from the limited data points it has access to!



Deep learning

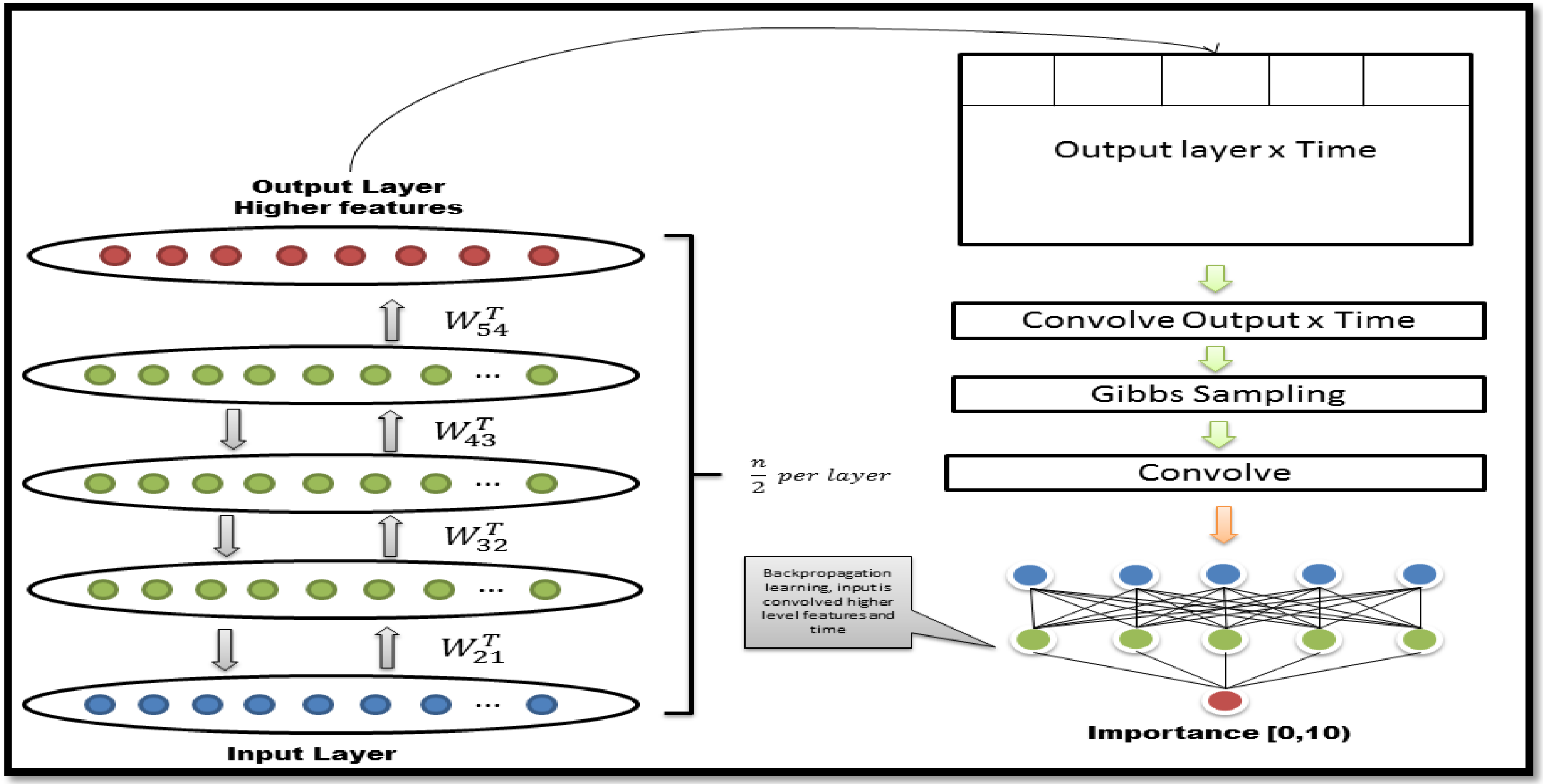
- The basic features are not enough they are like the colors on SpongeBob, we want higher level features.
  - Higher level features are things such as he has a smile on his face

- To accomplish this we use deep learning which is just stacking of the Restricted Boltzmann Machines



Deep Learning prediction architecture

- Our scheduling algorithm is designed to determine the importance of a document with respect to time of any specific user site



Experiments

- Atmospherics dataset of 5,512 documents
  - Provided by Army core of engineers
  - Extended with random information from globalsecurity.org and news articles
  - Information pertaining to Civilian environment, military installations and civilian structures

- Trained against Support Vector Machine (Radial basis function kernel) and Naïve Bayes algorithms
  - Both algorithms had to use features picked prior to running while the deep learning had to run in an unsupervised fashion
  - These features were principal component analysis of the major terms in each document

Results - Accuracy

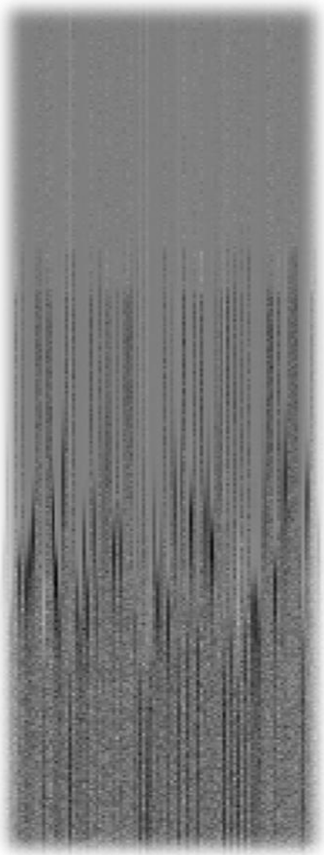
- Naïve Bayes – 69.7149%
  - Runtime of 5 ½ hours when feature selection taken into account
- SVM (RBF) – 72.4973%
  - Runtime of 7 hours when feature selection taken into account
- RBM (10,000 hidden) – 73.2558%
  - Runtime of 1 hour
- RBM (100 hidden) – 74.0311%
  - Runtime of 5 minutes
- Sequence Predict (100x100 hidden) – 74.5381%
  - Runtime of 2 hours

Results – Memory Complexity

- The cost of the more a more efficient runtime
  - Everything dimensional sample must be stored in memory.

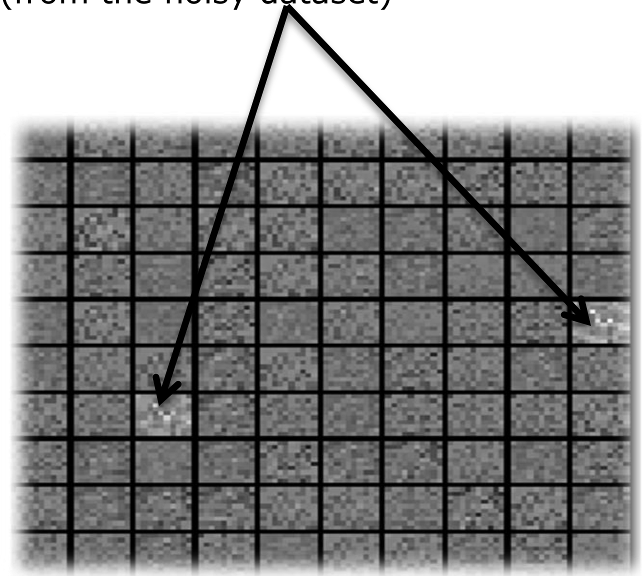
- Naïve Bayes - 352 MB
- SVM (RBF) – 254 MB (libSVM)
- RBM (10,000 hidden) – Unmeasured maxed out 192 GB server, had to shutdown services in order to run.
- RBM (100 hidden) – 3.22 GB
- Sequence Predict (100x100 hidden) – 7.8 GB

Results – learned weights



Level 1 weights

Discovered these extremely low weights corresponded to topic groups [terrorism, terrorist] and [bomb, bombing] (from the noisy dataset)



Level 2 weights

Black spots are higher weight score
Lighter spots are lower weight score

Conclusion

- We found that our sequential prediction engine works great for discrete ranked data
  - Runs into issues with continuous problems

- More accurate method than pervious methods
  - Including human handled feature selection

- More research is needed into the effect of the size of the deep network on accuracy
  - Current theory is the curse of dimensionality is at work, will need to prove this is correct

[1]	ICT, "ICT Facts and Figures," 2011. [Online]. Available: http://www.itu.int/ITU-D/ict/facts/2011/material/ICTFactsFigures2011.pdf. [Accessed 6 2 2013].
[2]	Internet World Stats, "Internet Users in the World," 30 June 2012. [Online]. Available: http://www.internetworldstats.com/stats.htm. [Accessed 6 February 2006].
[3]	D. Webb, "On the Definition of a Space Weapon," 2005. [Online]. Available: http://praxis.leedsmet.ac.uk/praxis/documents/space_weapons.pdf. [Accessed 6 2 2013].
[4]	S. J. Russel and P. Norvig, Artificial Intelligence: A Modern Approach, Upper Saddle River: Prentice Hall, 2003.
[5]	C. Rhode, "Intro Neural Networks," 1 Janurary 2010. [Online]. Available: http://lowercolumbia.edu/students/academics/facultypages/rhode-cary/intro-neural-net.htm. [Accessed 13 Feburary 2013].
[6]	D. Hebb, The Organization of Behavior, New York, 1949.
[7]	W. McCulloch and P. Walter, "A Logical Calculus of Ideas Immanent in Nervous Activity," Bulletin of Mathematical Biophyscis, vol. 5, no. 4, pp. 115-133, 1943.
[8]	K. Fukushima, "Cognitron: A self organizing multilayered Neural Network," Biological Cybernetics, vol. 20, no. 3-4, pp. 121-136, 1975.
[9]	Statistics 4 u, "www.statistics4u.com," Statistics 4 u, 1 January 2008. [Online]. Available: http://www.statistics4u.com/fundstat_eng/img/hl_backprop.png. [Accessed 4 Feburary 2013].
[10]	P. Webos, Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences, Harvard University, 1974.

[11]	**K. Shachar, S. Rosset and C. Perlich, "Leakage in data mining: formulation, detection, and avoidance," Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, vol. 1, no. 1, pp. 556-563, 2011.**
[12]	A. Waibel, T. Hanazawa, G. Hinton, K. Shinkano and K. Lang, "Phoneme recognition using time-delay neural networks," IEEE transcactions of acoustic, speech and signal processing, vol. 37, no. 1, pp. 328-339, 1989.
[13]	USDA, "ars.usda.gov," United States Department of Agriculture, 18 August 2010. [Online]. Available: http://www.ars.usda.gov/Research/docs.htm?docid=9124&page=4. [Accessed 4 Feburary 2013].
[14]	H. O. Simon, Neural Networks and Learning Machines, Pearson Education, 2008.
[15]	Vietdungiitb, "www.codeproject.com," Code Project, 31 May 2012. [Online]. Available: http://www.codeproject.com/Articles/376798/Large-pattern-recognition-system-using-multi-neura. [Accessed 4 Feburary 2013].
[16]	G. Hinton, S. Osindero and Y. Teh, "A fast learning algorithm for deep belief nets," Neural Computation, vol. 18, no. 1, pp. 1527-1554, 2006.
[17]	M. A. Carreira-Perpinan and G. Hinton, "On contrastive divergence learning," in Artifical Intelligence and Statistics, 2005.
[18]	H. Larochelle, "http://www.dmi.usherb.ca/~larocheh/index_en.html," Hugo Larochelle, 12 March 2012. [Online]. Available: http://www.dmi.usherb.ca/~larocheh/images/deep_learning.jpg. [Accessed 2013 6 Feburary].
[19]	G. Hinton, S. Osindero and Y.-W. Teh, "A fast learning algorithm for deep belief nets," Neural computation, vol. 18, no. 7, pp. 1527-1554, 2006.
[20]	Y. Bengio, Learning Deep Architectures for AI, 2009.