

CS6784

Advanced Topics in Machine Learning

Spring 2014

Thorsten Joachims
Cornell University

Outline of Today

- Introduction
 - Thorsten Joachims + Joshua Moore
- Overview of Class Topics
 - Structured Prediction
 - Machine Learning with Humans in the Loop
 - Learning Representations
- Administrivia
 - Goals for the Class
 - Pre-Requisites
 - Credit Options and Format
 - Project and Assignments
 - Course Material
 - Warm-up Assignment
 - Contact Info

Topic 1

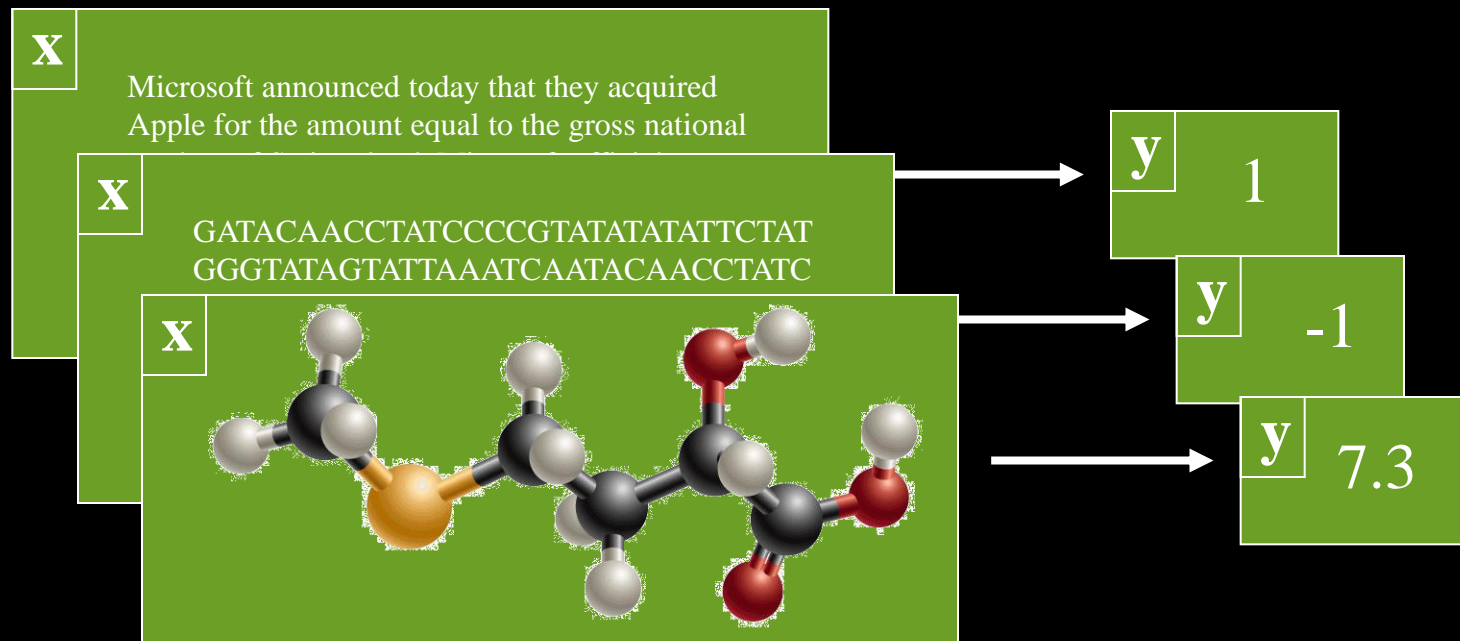
Structured Output Prediction

Conventional Supervised Learning

- Find function from input space X to output space Y

$$h: X \rightarrow Y$$

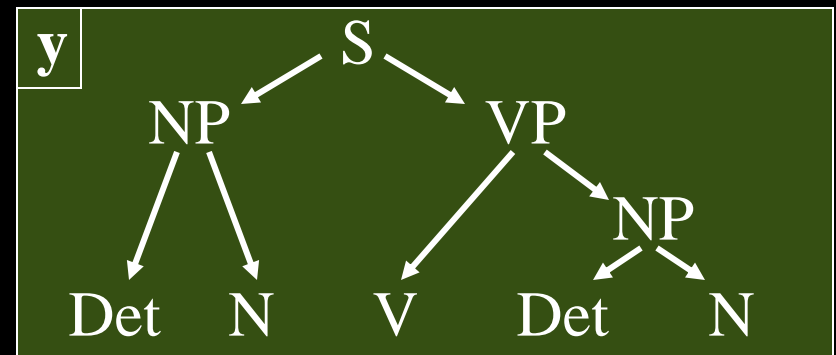
such that the prediction error is low.



Examples of Complex Output Spaces

- Natural Language Parsing
 - Given a sequence of words x , predict the parse tree y .
 - Dependencies from structural constraints, since y has to be a tree.

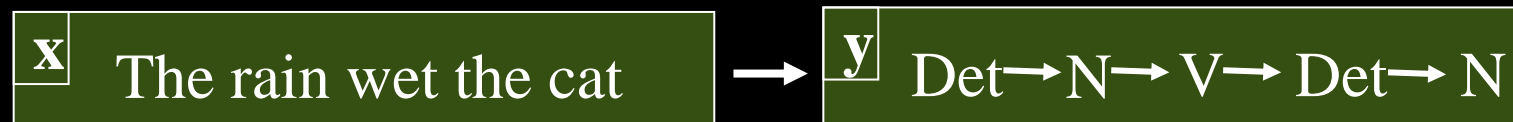
x The dog chased the cat



Examples of Complex Output Spaces

- **Part-of-Speech Tagging**

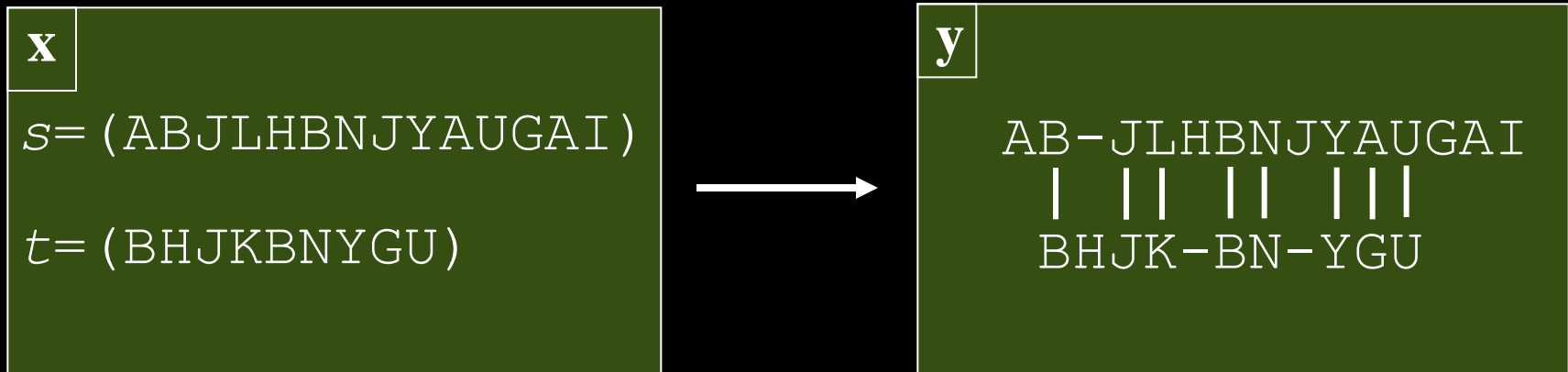
- Given a sequence of words x , predict sequence of tags y .
- Dependencies from tag-tag transitions in Markov model.



\rightarrow Similarly Named-Entity Recognition, Protein Intron Tagging, etc.

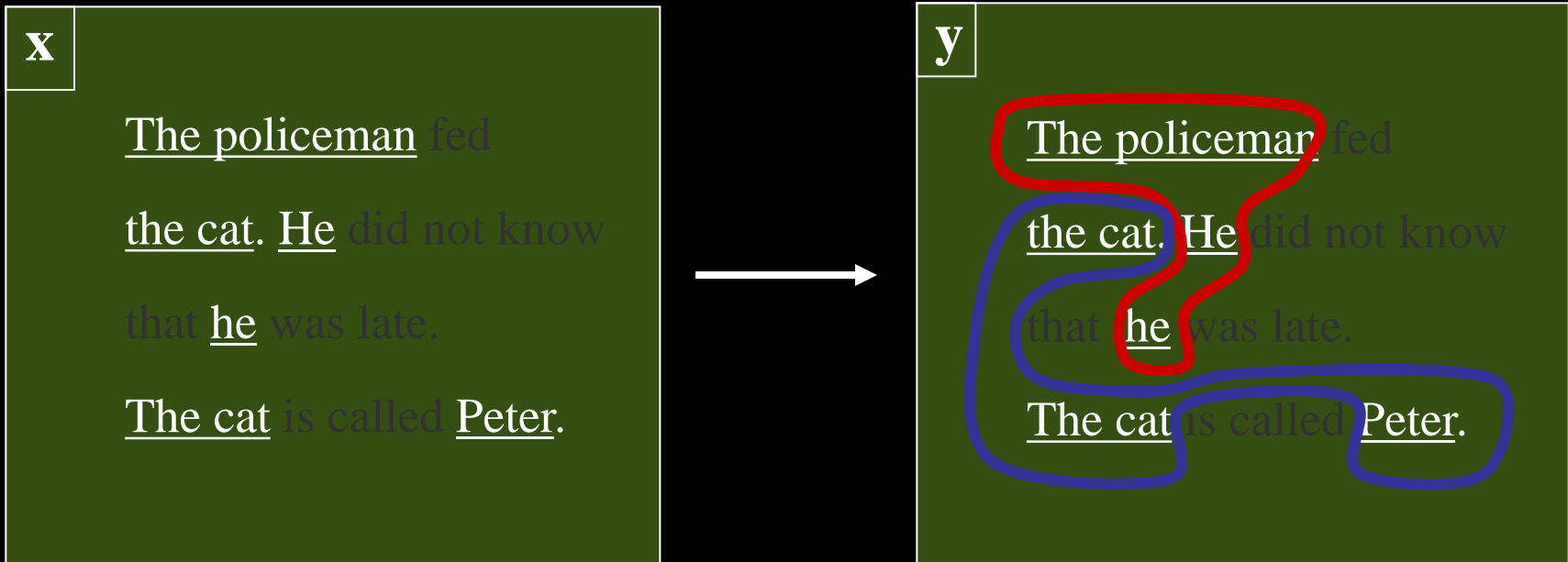
Examples of Complex Output Spaces

- Protein Sequence Alignment
 - Given two sequences $x=(s,t)$, predict an alignment y .
 - Structural dependencies, since prediction has to be a valid global/local alignment.



Examples of Complex Output Spaces

- Noun-Phrase Co-reference
 - Given a set of noun phrases x , predict a clustering y .
 - Structural dependencies, since prediction has to be an equivalence relation.
 - Correlation dependencies from interactions.



Examples of Complex Output Spaces

- Multi-Label Classification

- Given a (bag-of-words) document x , predict a set of labels y .
- Dependencies between labels from correlations between labels (“iraq” and “oil” in newswire corpus)

x Due to the continued violence in Baghdad, the oil price is expected to further increase. OPEC officials met with ...



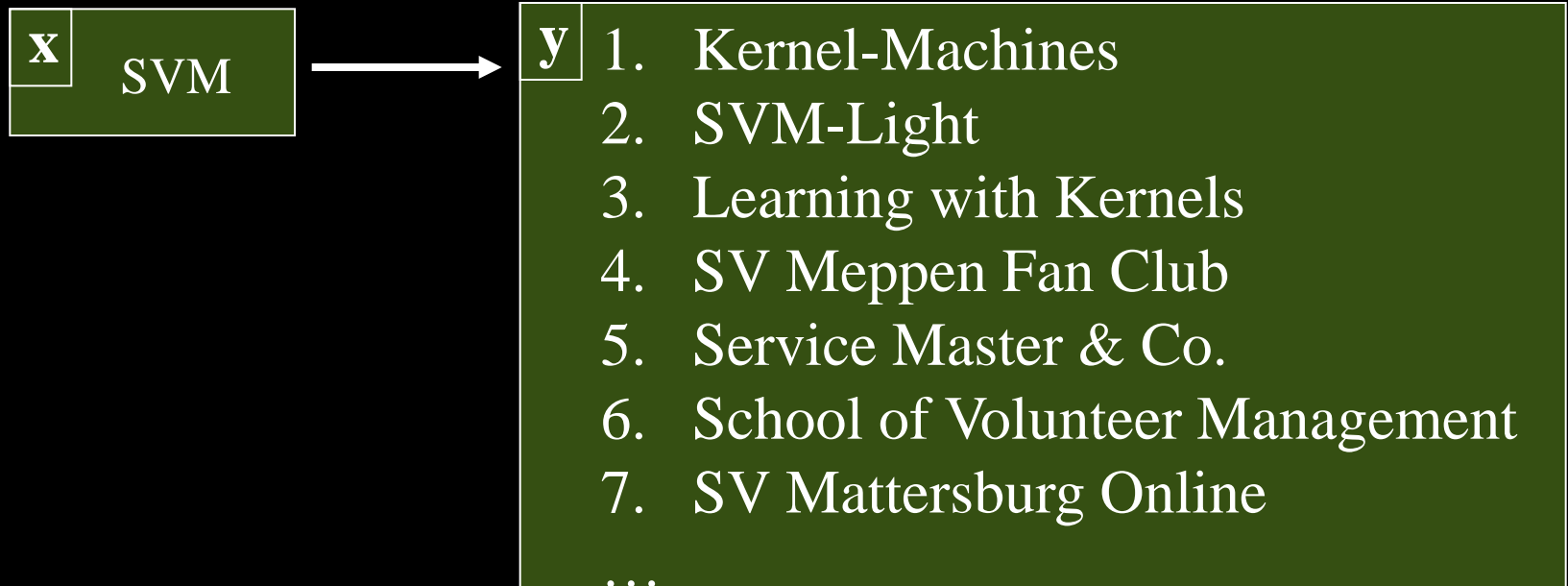
y

-1	antarctica
-1	benelux
-1	germany
+1	iraq
+1	oil
-1	coal
-1	trade
-1	acquisitions

Examples of Complex Output Spaces

- Information Retrieval

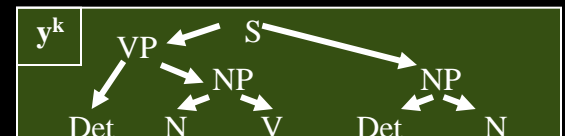
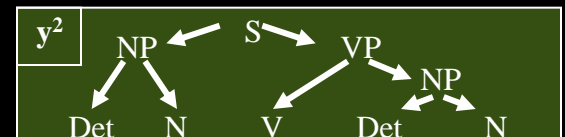
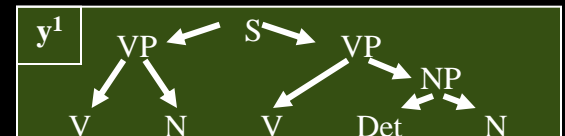
- Given a query x , predict a ranking y .
- Dependencies between results (e.g. avoid redundant hits)
- Loss function over rankings (e.g. AvgPrec)



Why is Structured Output Prediction Interesting?

- Application Perspective
 - Many interesting real-world problems have structure in outputs
- Research Perspective
 - Like a multi-class problem with exponentially many classes!
 - How to predict efficiently?
 - How to learn efficiently?
 - Potentially huge models!

X The dog chased the cat



Overview: Structured Output Prediction

- Existing methods and their properties / limitations
 - Generative models
 - Structural SVMs and other maximum margin methods
 - Conditional Random Fields
 - Search-based methods
 - Gaussian Processes
 - Kernel Dependency Estimation
- Applications
 - Search engines
 - Natural language processing
 - Reinforcement learning
 - Probabilistic reasoning
 - Computational biology

Topic 2

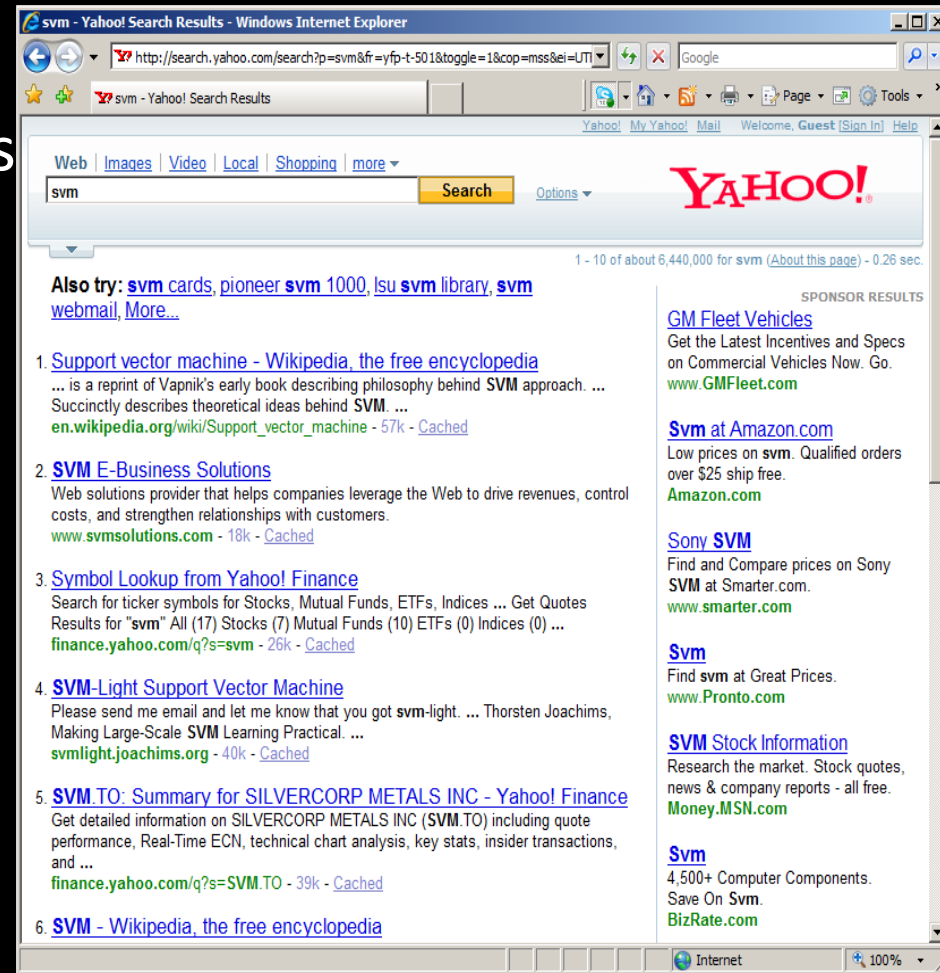
Machine Learning
with
Humans in the Loop

Interactive Learning Systems

- WHILE(forever)
 - “System” presents options to the user
 - User examines the “Options” and reacts to them
 - “System” observes the selection and learns from it
- “System” / “Options” =
 - Search engine / search results
 - Movie recommender system / recommended movies
 - Online shopping site / products to buy
 - GPS navigation software / route
 - Spelling correction in word processor / word
 - Social network extension / friend
 - Twitter / post

Implicit Feedback in Web Search

- Observable actions
 - Queries / reformulations
 - Clicks
 - Order, dwell time
 - Etc.
- Implicit feedback
 - Personalized
 - Democratic
 - Timely
 - Human intelligence
 - Cheap
 - Abundant



Does User Behavior Reflect Retrieval Quality?

User Study in ArXiv.org

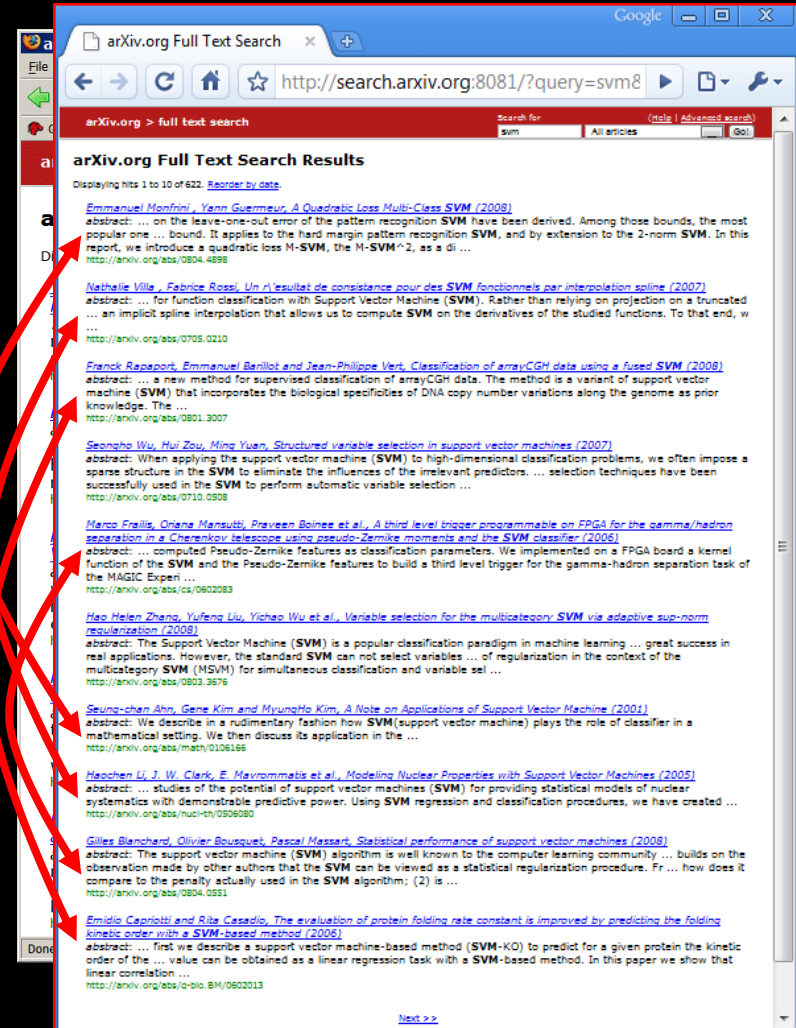
- Natural user and query population.
- User in natural context, not lab.
- Live and operational search engine.
- Ground truth by construction

ORIG \succ SWAP2 \succ SWAP4

- ORIG: Hand-tuned fielded
- SWAP2: ORIG with 2 pairs swapped
- SWAP4: ORIG with 4 pairs swapped

ORIG \succ FLAT \succ RAND

- ORIG: Hand-tuned fielded
- FLAT: No field weights
- RAND : Top 10 of FLAT shuffled



Absolute Metrics: Metrics

Name	Description	Aggregation	Hypothesized Change with Decreased Quality
Abandonment Rate	% of queries with no click	N/A	Increase
Reformulation Rate	% of queries that are followed by reformulation	N/A	Increase
Queries per Session	Session = no interruption of more than 30 minutes	Mean	Increase
Clicks per Query	Number of clicks	Mean	Decrease
Max Reciprocal Rank*	1/rank for highest click	Mean	Decrease
Mean Reciprocal Rank*	Mean of 1/rank for all clicks	Mean	Decrease
Time to First Click*	Seconds before first click	Median	Increase
Time to Last Click*	Seconds before final click	Median	Decrease

(*) only queries with at least one click count

Paired Comparisons: What to Measure?

$(u=tj, q=\text{"svm"})$

$f_1(u, q) \rightarrow r_1$

$f_2(u, q) \rightarrow r_2$

1. Kernel Machines
<http://svm.first.gmd.de/>
2. Support Vector Machine
<http://jbolivar.freeservers.com/>
3. An Introduction to Support Vector Machines
<http://www.support-vector.net/>
4. Archives of SUPPORT-VECTOR-MACHINES ...
<http://www.jjscmail.ac.uk/lists/SUPPORT...>
5. SVM-Light Support Vector Machine
http://ais.gmd.de/~thorsten/svm_light/

1. Kernel Machines
<http://svm.first.gmd.de/>
2. SVM-Light Support Vector Machine
http://ais.gmd.de/~thorsten/svm_light/
3. Support Vector Machine and Kernel ... References
<http://svm.research.bell-labs.com/SVMrefs.html>
4. Lucent Technologies: SVM demo applet
<http://svm.research.bell-labs.com/SVT/SVMsvt.html>
5. Royal Holloway Support Vector Machine
<http://svm.dcs.rhnc.ac.uk>

Interpretation: $(r_1 \succ r_2) \Leftrightarrow \text{clicks}(r_1) > \text{clicks}(r_2)$

Balanced Interleaving

($u=tj, q="svm"$)

$f_1(u,q) \rightarrow r_1$

$f_2(u,q) \rightarrow r_2$

1. Kernel Machines
<http://svm.first.gmd.de/>
2. Support Vector Machine
<http://jbolivar.freesevers.com/>
3. An Introduction to Support Vector Machines
<http://www.support-vector.net/>
4. Archives of SUPPORT-VECTOR-MACHINES ...
<http://www.jiscmail.ac.uk/lists/SUPPORT...>
5. SVM-Light Support Vector Machine
http://ais.gmd.de/~thorsten/svm_light/

1. Kernel Machines
<http://svm.first.gmd.de/>
2. SVM-Light Support Vector Machine
http://ais.gmd.de/~thorsten/svm_light/
3. Support Vector Machine and Kernel ... References
<http://svm.research.bell-labs.com/SVMrefs.html>
4. Lucent Technologies: SVM demo applet
<http://svm.research.bell-labs.com/SVT/SVMsvt.html>
5. Royal Holloway Support Vector Machine
<http://svm.dcs.rhnc.ac.uk>

Interleaving(r_1, r_2)

- | | |
|--|---|
| 1. Kernel Machines
http://svm.first.gmd.de/ | 1 |
| 2. Support Vector Machine
http://jbolivar.freesevers.com/ | 2 |
| 3. SVM-Light Support Vector Machine
http://ais.gmd.de/~thorsten/svm_light/ | 2 |
| 4. An Introduction to Support Vector Machines
http://www.support-vector.net/ | 3 |
| 5. Support Vector Machine and Kernel ... References
http://svm.research.bell-labs.com/SVMrefs.html | 3 |
| 6. Archives of SUPPORT-VECTOR-MACHINES ...
http://www.jiscmail.ac.uk/lists/SUPPORT... | 4 |
| 7. Lucent Technologies: SVM demo applet
http://svm.research.bell-labs.com/SVT/SVMsvt.html | 4 |

Model of User:

Better retrieval functions
is more likely to get more
clicks.

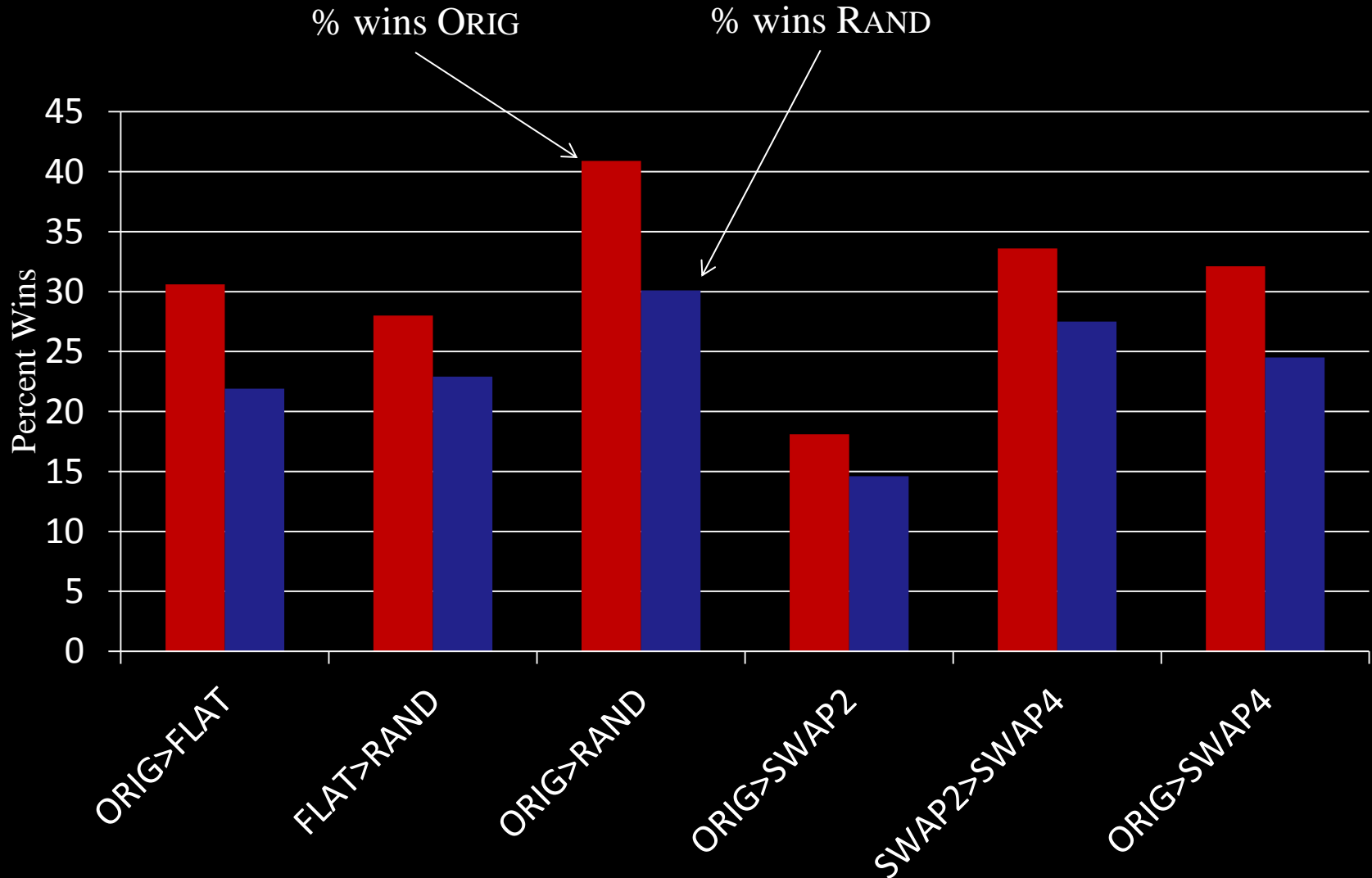
Invariant:

For all k , top k of
balanced interleaving is
union of top k_1 of r_1 and
top k_2 of r_2 with $k_1 = k_2 \pm 1$.

Interpretation: ($r_1 \succ r_2$) \Leftrightarrow clicks(top k (r_1)) > clicks(top k (r_2))

\rightarrow see also [Radlinski, Craswell, 2012] [Hofmann, 2012]

Arxiv.org: Interleaving Results



Issues in Learning with Humans

- Presentation Bias
 - Get accurate training data out of biased feedback
 - Use randomization to collect unbiased data
 - Experiment design
- Online Learning
 - Exploration/exploitation trade-offs
 - Observational vs. experimental data
 - Ability to run interactive experiments with users
- Measuring User Satisfaction
 - Turning behavior into evaluation measure

Overview: Learning with Humans

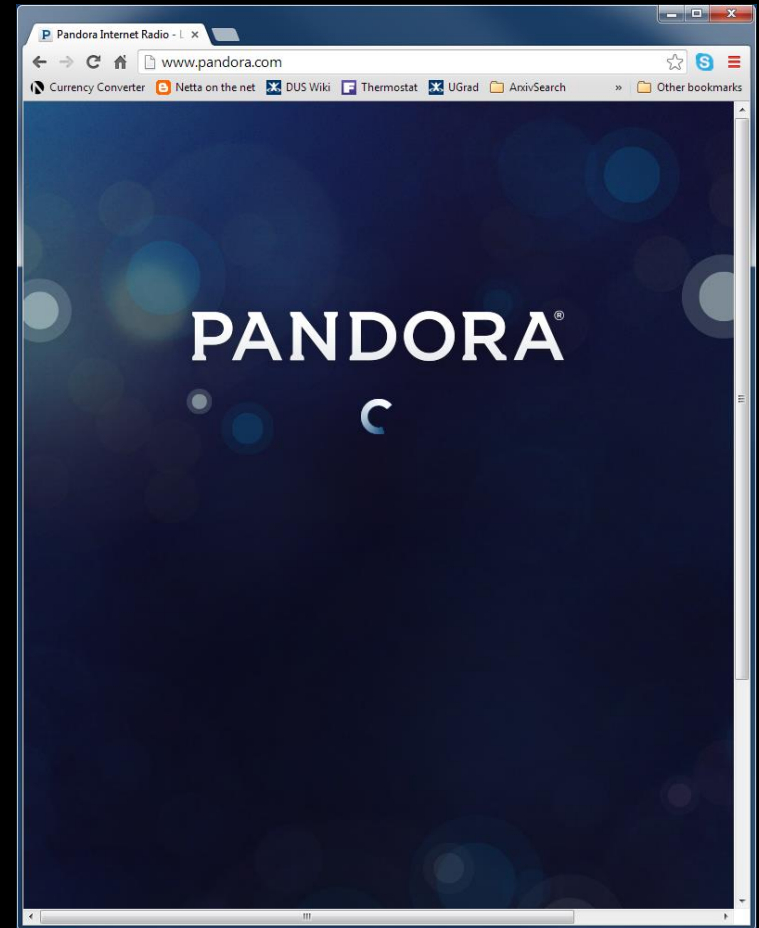
- Methods
 - Online learning and multi-armed bandits
 - Methods for interpreting user behavior
 - Matrix decomposition methods for recommendation
 - Active learning
- Applications
 - Information retrieval
 - Recommender systems
 - Online shopping
 - Mechanical turk
 - Web server usage

Topic 3

Learning Representations

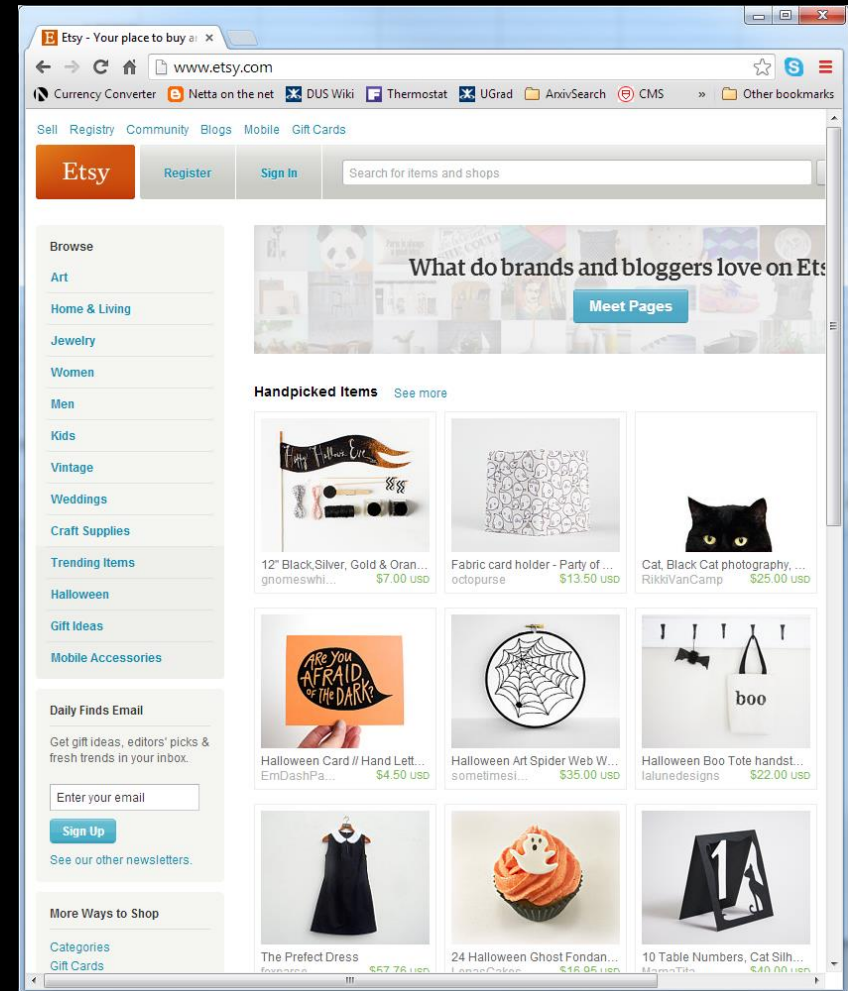
Learning about Music

- Collection of songs
 - $S = \{s_1, s_2, s_3, \dots, s_N\}$
- Example playlists
 - $p_1 = [s_9, s_{527}, s_{12}, \dots]$
 - $p_2 = [s_{7192}, s_{67}, s_{726}, \dots]$
- Goals
 - Automatically generate new playlists
 - Understand semantic space of songs
 - Query by tag, similar song
 - Visualization



Learning about Products

- Collection of products
 - $P = \{p_1, p_2, p_3, \dots, p_N\}$
- Example browsing sequences
 - $s_1 = [p_9, p_{527}, p_{12}, \dots]$
 - $s_2 = [p_{7192}, p_{67}, p_{726}, \dots]$
- Goals
 - Automatically recommend other items based on session prefix
 - Understand semantic space of products
 - Query by keyword, similar product
 - Visualization



Challenges and Approach

- Challenges
 - Items (i.e. songs, products) don't have good features
 - We would like to generate “style” features
 - Number of items is large
 - Traditional sequence models (e.g., NLP) do not scale
- Approach
 - Model sequences as (k-th order) Markov model
 - $P(s_1, \dots, s_k) = \prod P(s_i | s_{i-1})$
 - Find model for transitions $P(s_i | s_{i-1})$ that
 - does not require N^2 storage.
 - generalizes well beyond the observed data.

Logistic Markov Embedding

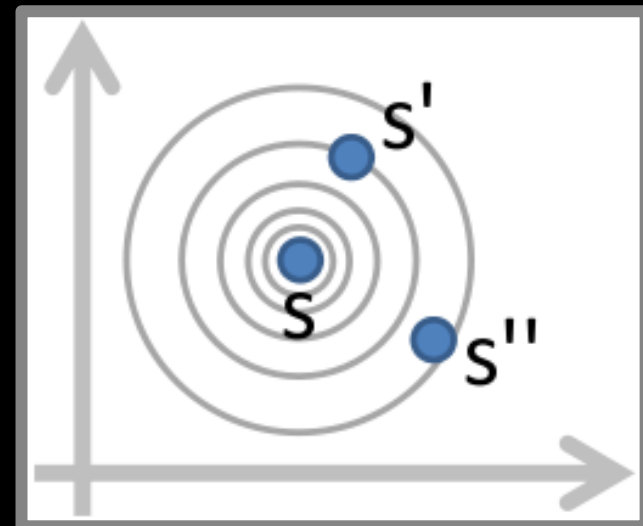
- Model

- Distance in space \sim transition probability

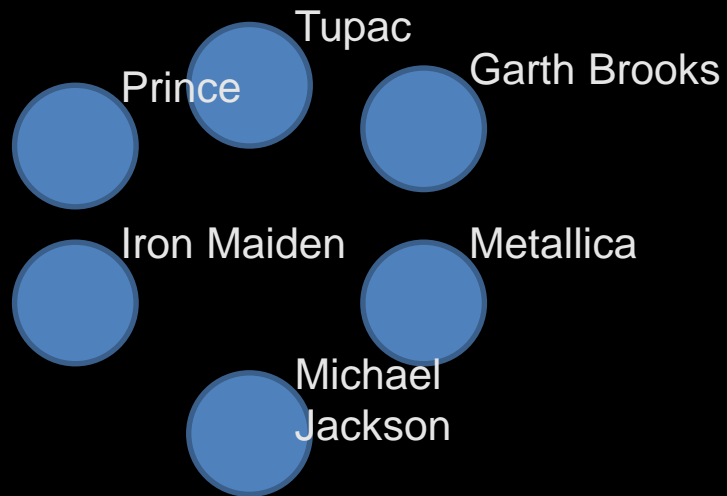
$$\Pr(p^{[i]} | p^{[i-1]}) = \frac{e^{-\|X(p^{[i]}) - X(p^{[i-1]})\|_2^2}}{\sum_j e^{-\|X(p^{[j]}) - X(p^{[i-1]})\|_2^2}}$$

- Training

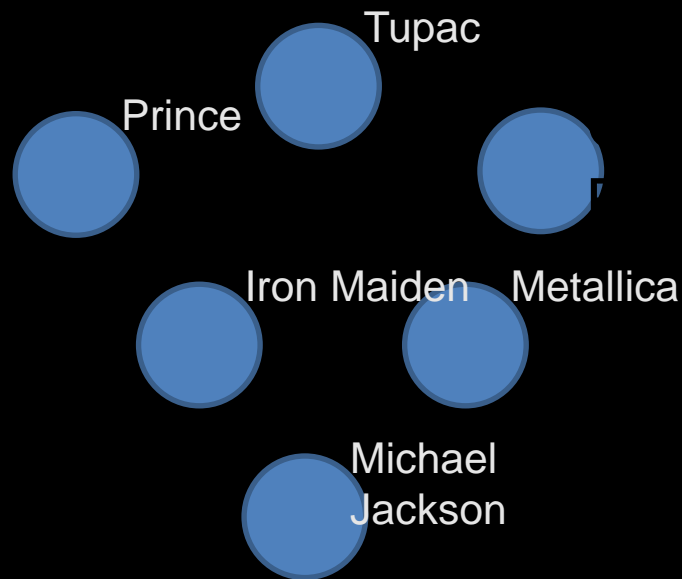
- Maximum likelihood
- Stochastic gradient
- $O(n)$ iteration complexity
→ $O(1)$ iteration complexity



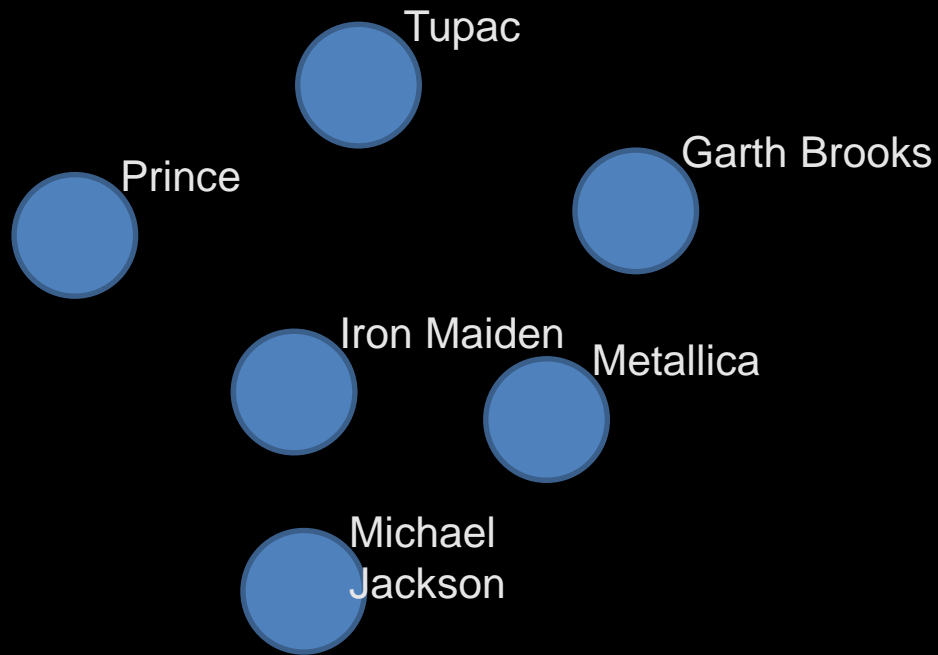
Learning Song Positions



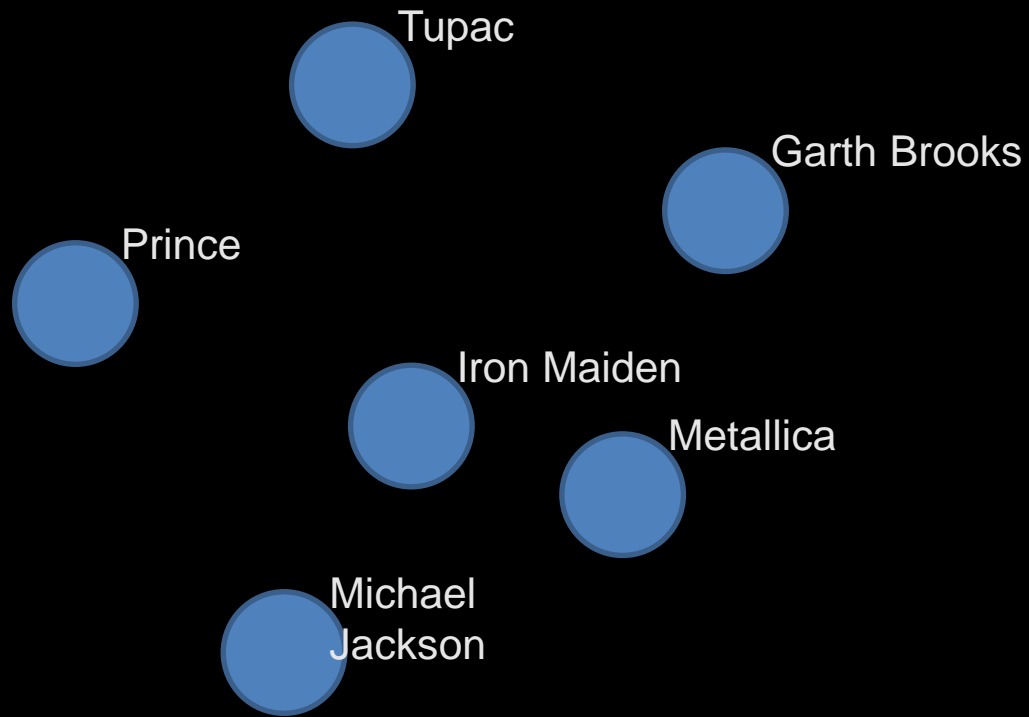
Learning Song Positions



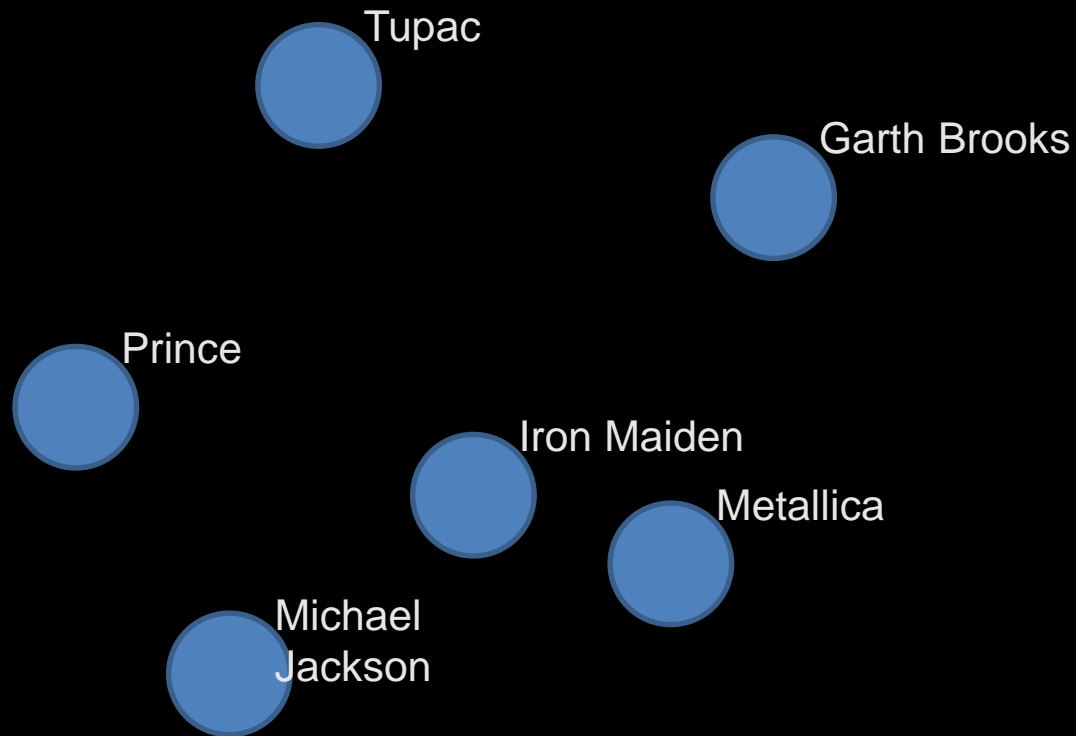
Learning Song Positions



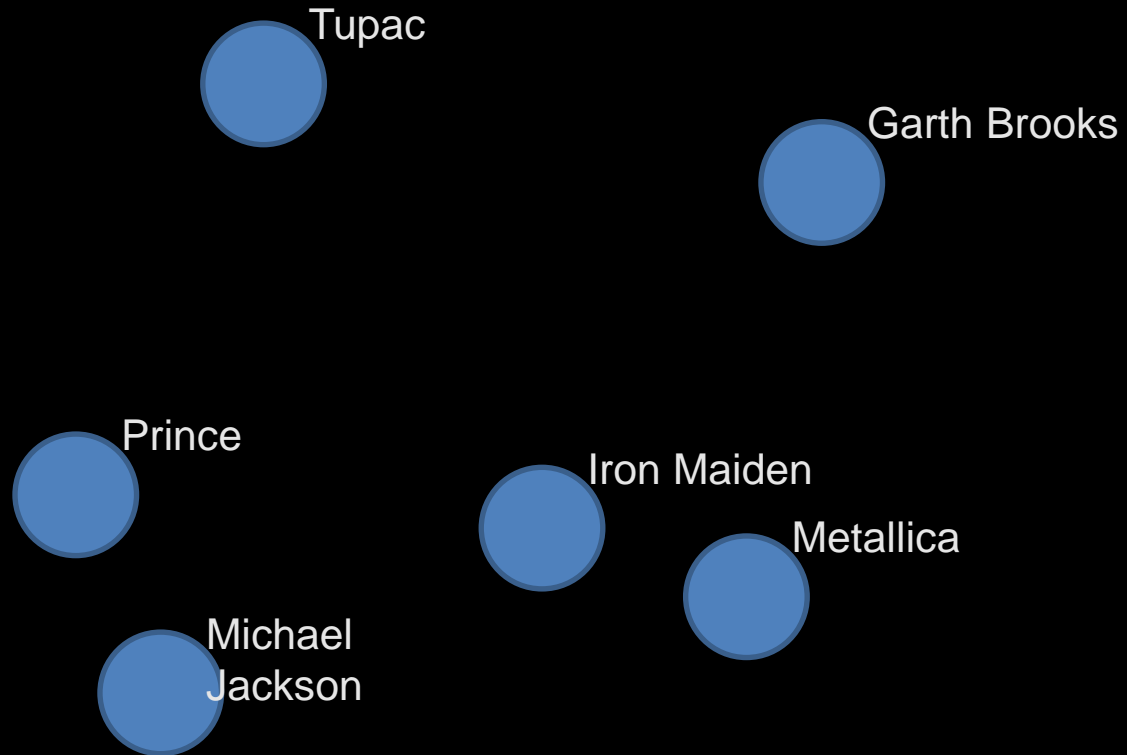
Learning Song Positions



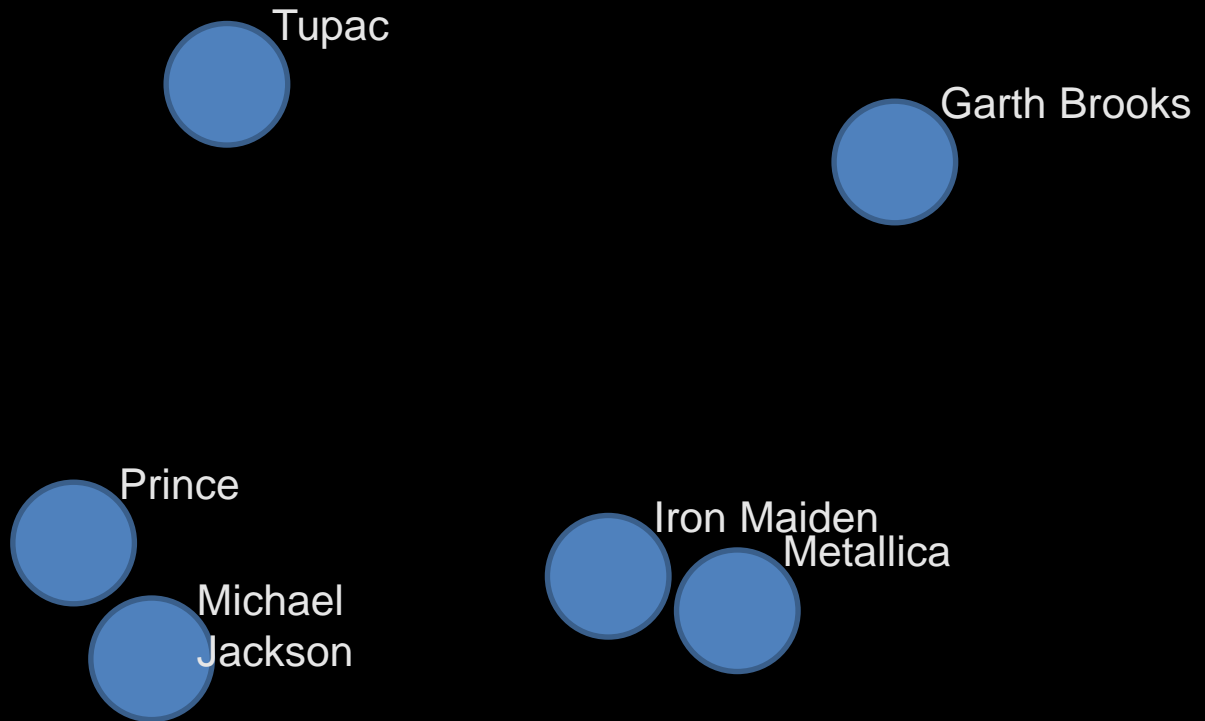
Learning Song Positions



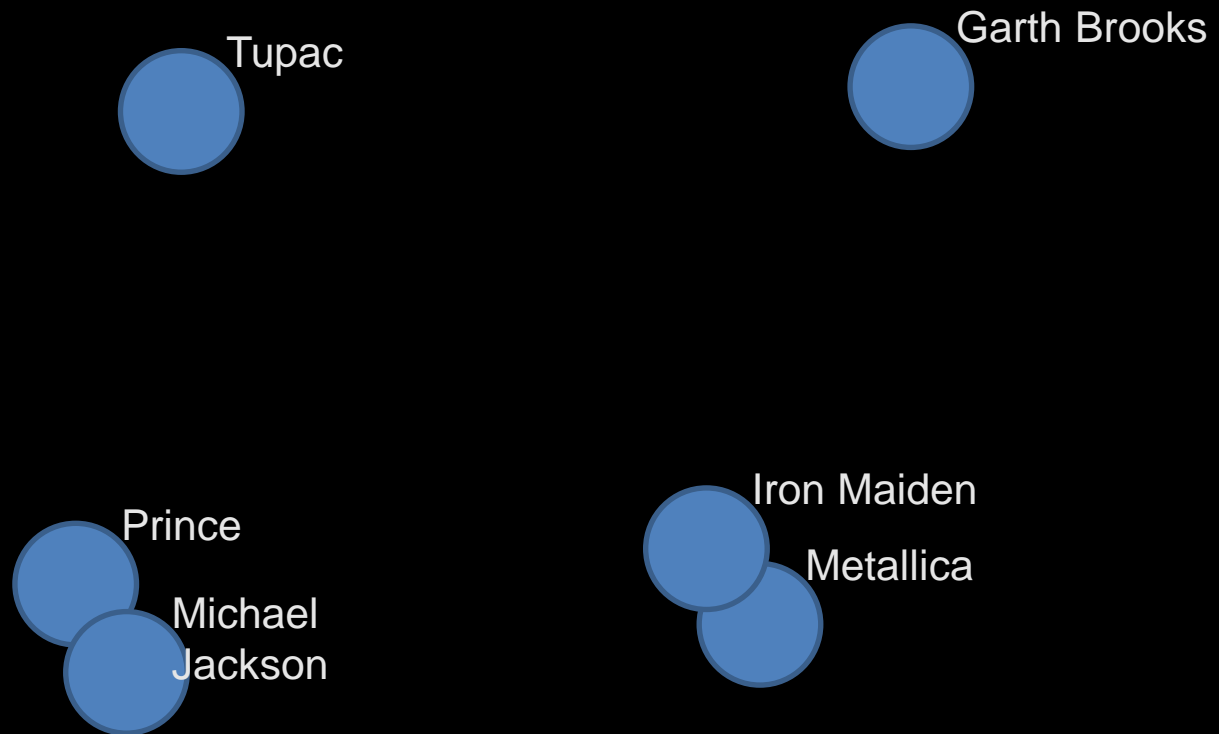
Learning Song Positions



Learning Song Positions

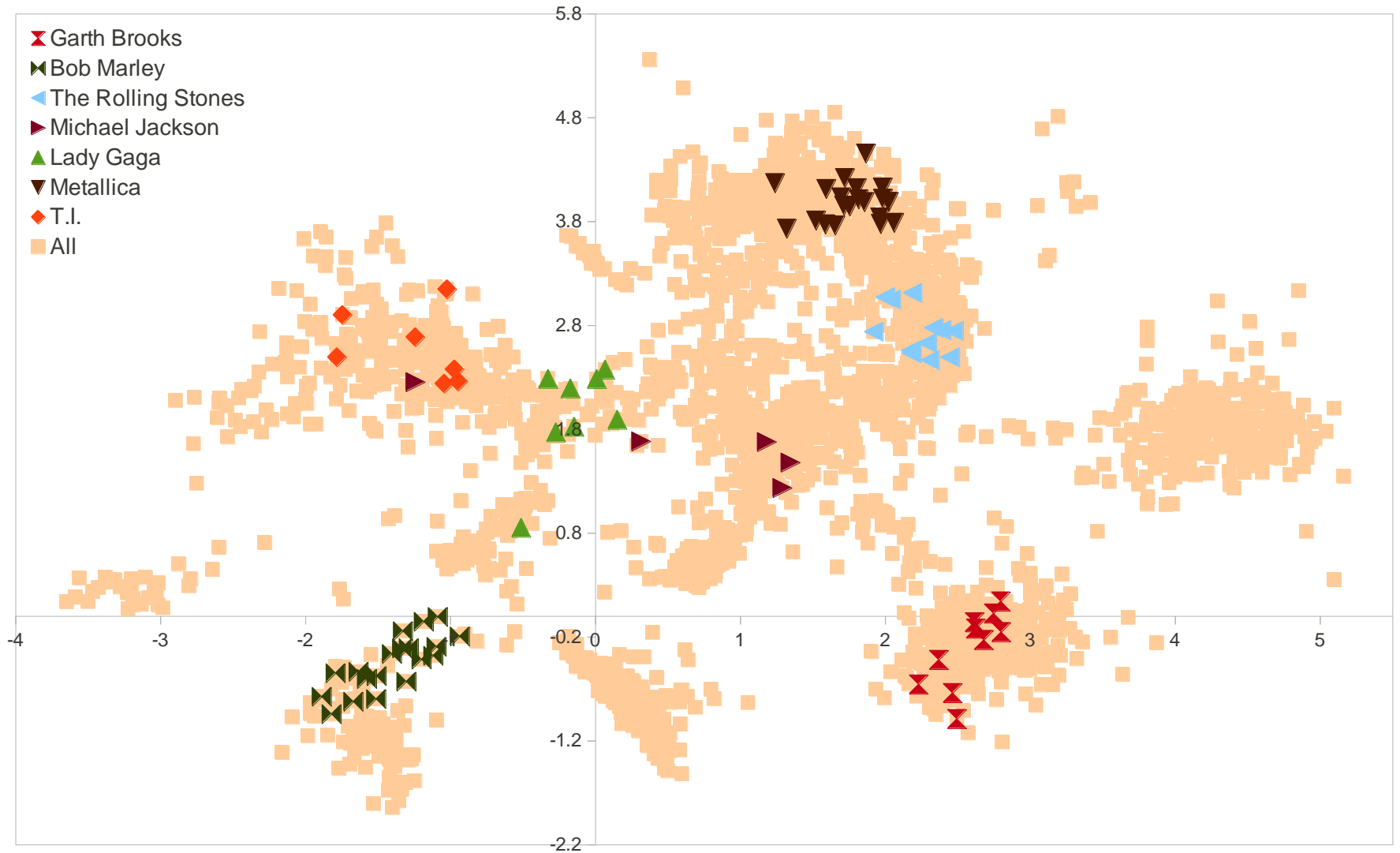


Learning Song Positions

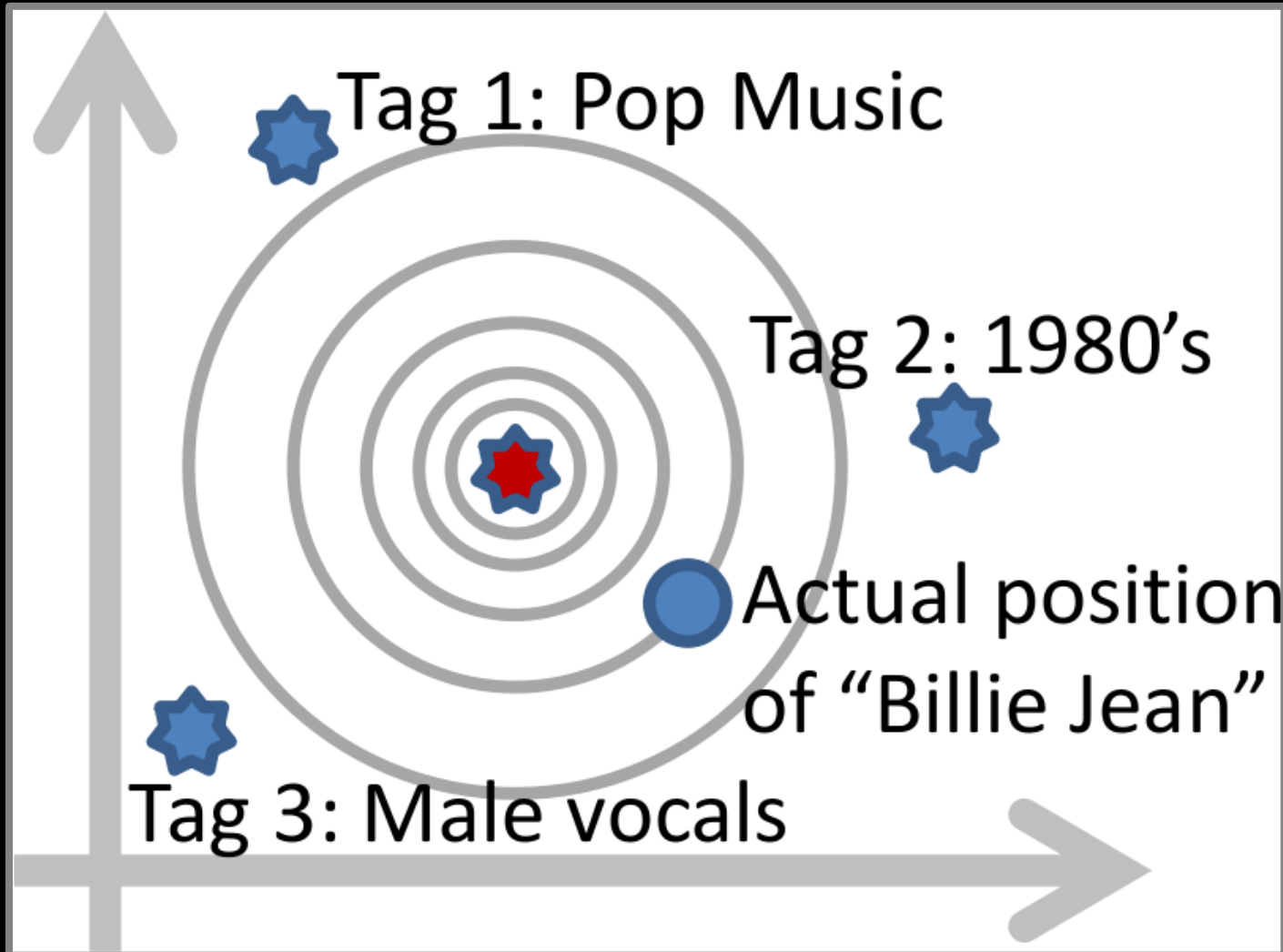


Converged

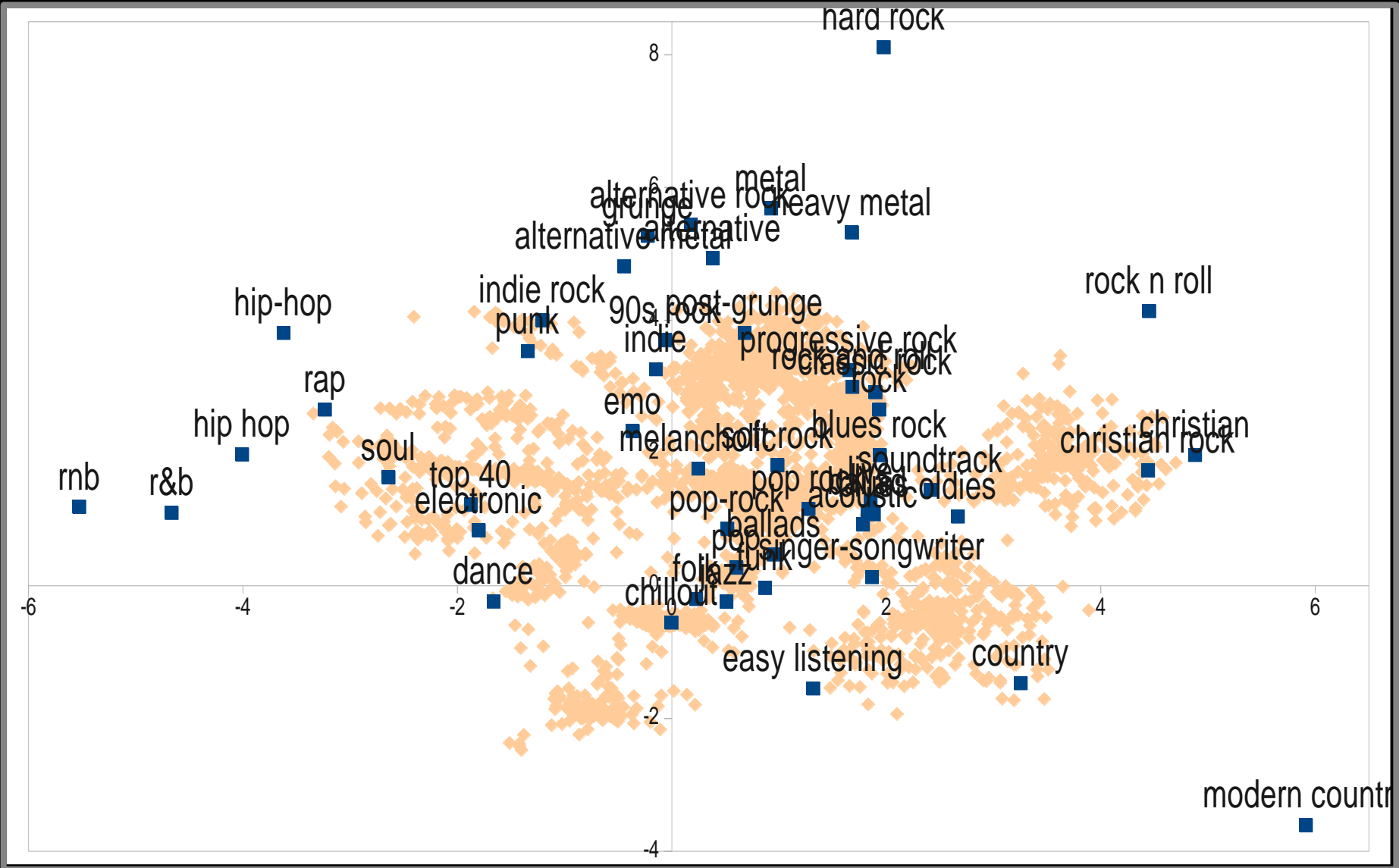
Result: Song Embedding



Extension: Tag Model



Result: Genre Tag Embedding



Overview:

Learning Representations

- **Methods**
 - Embeddings based on sequence data
 - Embeddings based on co-occurrence data
 - Embeddings for bipartite graphs
 - Matrix factorization for rating data
 - Modeling structured objects
 - Modeling compositionality
- **Applications**
 - Playlist modeling
 - Natural language processing
 - Image search
 - Modeling human behavior

Outline of Today

- Introduction
- Overview of Class Topics
 - Structured Prediction
 - Machine Learning with Humans in the Loop
 - Learning Representations
- Administrivia
 - Goals for this class
 - Pre-Requisites
 - Credit Options and Format
 - Project and Assignments
 - Course Material
 - Warm-up Assignment
 - Contact Info

Goals for this Class

- Deepen your knowledge in three active research areas of ML
- Enable and improve your thesis research
- Practice being a successful academic

→ Class targeted towards current (or soon to be) PhD students!

Pre-Requisites

- This is not an introductory Machine Learning class!
- You need to satisfy one of the following ML pre-reqs:
 - Successfully taken CS4780 “Machine Learning”
 - Successfully taken CS6780 “Advanced Machine Learning”
 - Successfully taken a comparable “Intro to ML” class (*)
 - Acquired the equivalent ML knowledge in some other way (e.g. strong background in Statistics + ML textbook) (*)
- Basic probability and linear algebra
- Programming skills required for many projects
- (*) means talk to me

Format of Class

- Lectures (by TJ)
 - Background material on general ML and 3 topics
- Research paper presentations (by students)
 - Reach current state of the art in each of 3 topics
- Project
 - Semester long, original research project
- Mock funding proposals
 - Develop your own research ideas for the 3 topics
- Peer reviewing

Research Paper Presentations

- Pair of students present the paper in class
 - Slide presentation
 - Create critique, extended bibliography, examples, demo software, experiments etc. that help understand the paper
 - Prepare discussion topics / group activity
 - Prepare quiz
 - Do dry-run of presentation in my office before class (30% of the grade).
- Everybody reads the paper in preparation for class
 - Quiz about each paper
- All students give feedback afterwards.

Mock Funding Proposals

- Write short funding proposal
 - Practice to develop your own research ideas and research plan
 - Practice to justify your research
 - Practice to convince others of your ideas
- Individual or group
- Peer reviewed

Project

- Full Semester Project
 - Topic of your choice that relates to CS6784
 - Scoped to be a publishable paper
 - Individual or group
- Timeline
 - 2/9: Proposal (10 %)
 - 3/16: First status report (10 %)
 - 4/20: Second status report (10 %)
 - 5/1-6: Project presentation (20 %)
 - 5/12: Final project report (50 %)
- At each step peer review
 - 5/18: Peer reviews due for project reports

Credit Options and Grades

- Letter grade:
 - project (40%)
 - paper presentation (20%)
 - in-class assignments and participation (15%)
 - three lowest grades dropped
 - funding proposals (12%)
 - peer reviewing (10%)
 - warm-up assignment (3%)
- Pass/Fail:
 - not allowed, unless you have very good arguments
- Audit:
 - not allowed, unless you have very good arguments

Course Material

- Background Reading
 - K. Murphy, "Machine Learning - a Probabilistic Perspective", MIT Press, 2012. ([online](#) via Cornell Library)
 - T. Mitchell, "Machine Learning", McGraw Hill, 1997.
 - B. Schoelkopf, A. Smola, "Learning with Kernels", MIT Press, 2001. ([online](#))
 - C. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.
 - R. Duda, P. Hart, D. Stork, "Pattern Classification", Wiley, 2001.
 - T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning", Springer, 2001.
 - N. Cristianini, J. Shawe-Taylor, "Introduction to Support Vector Machines", Cambridge University Press, 2000. ([online](#) via Cornell Library)
- Slides, Notes and Papers
 - Slides available on course homepage
 - Papers on course homepage

Warm-up Assignment

- Read the paper:
 - C. Cortes, V. Vapnik, “Support Vector Networks”, Machine Learning, 20:273-297, 1995.
<http://link.springer.com/article/10.1023/A:1022627411411>
- Write a short paper that
 - is at most 800 words long
 - is submitted by Tuesday Jan 28 at 11:59pm ESTand that addresses the following questions:
 - What are the main original contributions described in this paper? Briefly describe the top 3 and argue why those are top.
 - For each original contribution, briefly describe in how far related ideas were already present in earlier papers.
- Peer review

How to Get in Touch

- Course Web Page
 - <http://www.cs.cornell.edu/Courses/cs6784/2014sp/>
- Email
 - Thorsten Joachims: tj@cs.cornell.edu
 - Joshua Moore: jlmo@cs.cornell.edu
- Office Hours
 - Thursdays 3:00pm – 4:00pm, 418 Gates Hall
- Piazza
 - <https://piazza.com/cornell/spring2014/cs6784>
- Peer reviewing platform
 - TBA