Semi-Supervised Learning with Graphs

Xiaojin (Jerry) Zhu
School of Computer Science
Carnegie Mellon University
Semi-supervised Learning

- classification
- classifiers need labeled data to train
- labeled data scarce, unlabeled data abundant
- *Traditional classifiers cannot use unlabeled data.*

My interest (semi-supervised learning): Develop classification methods that can use both labeled and unlabeled data.
Motivating examples

- speech recognition (sound → sentence)
  - labeled data: transcription, 10 to 400 times real-time
  - unlabeled data: sounds alone, easy to get (radio, call center)

- parsing ("I saw a falcon with a telescope." → tree)
  - labeled data: treebank, English 40,000/5, Chinese 4,000/2 years
  - unlabeled data: sentences without annotation, everywhere.

- personalized news (article → interested?)
  - user patience

- video surveillance (image → identity)
  - named images availability

unlabeled data useful?
The message

Unlabeled data can improve classification.
Why unlabeled data might help

d example: classify astronomy vs. travel articles

- articles represented by content word occurrence vectors
- article similarity measured by content word overlap

<table>
<thead>
<tr>
<th></th>
<th>$d_1$</th>
<th>$d_3$</th>
<th>$d_4$</th>
<th>$d_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>asteroid</td>
<td>●</td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bright</td>
<td>●</td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>comet</td>
<td>●</td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>year</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>zodiac</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>airport</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>bike</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>camp</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>yellowstone</td>
<td></td>
<td></td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td>zion</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
</tbody>
</table>
Why labeled data alone might fail

<table>
<thead>
<tr>
<th></th>
<th>$d_1$</th>
<th>$d_3$</th>
<th>$d_4$</th>
<th>$d_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>asteroid</td>
<td>●</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bright</td>
<td></td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>comet</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>year</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>zodiac</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>airport</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>bike</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>camp</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>yellowstone</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>zion</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
</tbody>
</table>

- no overlap!
- tends to happen when labeled data is scarce
Unlabeled data are stepping stones

<table>
<thead>
<tr>
<th></th>
<th>$d_1$</th>
<th>$d_5$</th>
<th>$d_6$</th>
<th>$d_7$</th>
<th>$d_3$</th>
<th>$d_4$</th>
<th>$d_8$</th>
<th>$d_9$</th>
<th>$d_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>asteroid</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bright</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>comet</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>year</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>zodiac</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>airport</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>bike</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>camp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>yellowstone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>zion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

- observe *direct* similarity from features: $d_1 \sim d_5$, $d_5 \sim d_6$ etc.
- assume similar features $\Rightarrow$ same label
- labels propagate via unlabeled articles, *indirect* similarity
Unlabeled data are stepping stones

- arrange $l$ labeled and $u$ unlabeled (=test) points in a graph
  - nodes: the $n = l + u$ points
  - edges: the direct similarity $W_{ij}$, e.g. number of overlapping words.
    (in general: a decreasing function of the distance $||x_i - x_j||$)

- want to infer indirect similarity (with all paths)
One way to use labeled and unlabeled data

(Zhu and Ghahramani, 2002)

• input: $n \times n$ graph weights $W$ (important!)
  labels $Y_l \in \{0, 1\}^l$

• create matrix $P_{ij} = W_{ij} / \sum W_i$.

• repeat until $f$ converges
  ▶ clamp labeled data $f_l = Y_l$
  ▶ propagate $f \leftarrow Pf$

• $f$ converges to a unique solution, the harmonic function.
  $0 \leq f \leq 1$, “soft labels”
An electric network interpretation

(Zhu, Ghahramani and Lafferty, ICML2003)

- harmonic function $f$ is the voltage at the nodes
  - edges are resistors with $R = 1/W$
  - 1 volt battery connects to labeled nodes

- indirect similarity: similar voltage if many paths exist
A random walk interpretation of harmonic functions

- harmonic function $f_i = P(\text{hit label } 1 \mid \text{start from } i)$
  - random walk from node $i$ to $j$ with probability $P_{ij}$
  - stop if we hit a labeled node

- indirect similarity: random walks have similar destinations
Closed form solution for the harmonic function

• define diagonal degree matrix \( D \), \( D_{ii} = \sum W_i \).
  define graph Laplacian matrix \( \Delta = D - W \)

\[
f_u = -\Delta_{uu}^{-1} \Delta_{ul} Y_l
\]

• \( \Delta \) graph version of the continuous Laplacian operator
  \( \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \)

• harmonic: \( \Delta f = 0 \) with Dirichlet boundary conditions on labeled data
Properties of the harmonic function

- currents in-flow = out-flow at any node (Kirchoff’s law)
- min energy \( E(f) = \sum_{i \sim j} W_{ij} (f_i - f_j)^2 = f^\top \Delta f \)
- average of neighbors: \( f_u(i) = \frac{\sum_{j \sim i} W_{ij} f(j)}{\sum_{j \sim i} W_{ij}} \)
- uniquely exists
- \( 0 \leq f \leq 1 \)
Text categorization with harmonic functions

50 labeled articles, about 2000 unlabeled articles. 10NN graph.
Digit recognition with harmonic functions

- pixel-wise Euclidean distance

<table>
<thead>
<tr>
<th>2</th>
<th>2 2 2 2 2 2 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>not similar</td>
<td>indirectly similar with stepping stones</td>
</tr>
</tbody>
</table>
Digit recognition with harmonic functions

50 labeled images, about 4000 unlabeled images, 10NN graph
Practical concerns about harmonic functions

- does it scale?
  - closed form involves matrix inversion \( f_u = -\Delta_{uu}^{-1} \Delta_{ul} Y_l \)
  - \( O(u^3) \), e.g. millions of crawled web pages

- solution 1: use iterative methods
  - the label propagation algorithm (slow)
  - loopy belief propagation
  - conjugate gradient

- solution 2: reduce problem size
  - use a random small unlabeled subset (Delalleau et al. 2005)
  - harmonic mixtures

- can it handle new points (induction)?
Harmonic mixtures

(Zhu and Lafferty, 2005)

• fit unlabeled data with a mixture model, e.g.
  ► Gaussian mixtures for images
  ► multinomial mixtures for documents

• use EM or other methods

• $M$ mixture components, here $M = 30 \ll u \approx 1000$

• learn soft labels for the mixture components, not the unlabeled points
Harmonic mixtures
learn labels for mixture components

• assume mixture component labels $\lambda_1, \ldots, \lambda_M$

• labels on unlabeled points determined by the mixture model
  ▶ The mixture model defines responsibility $R$: $R_{im} = p(m|x_i)$
  ▶ $f(i) = \sum_{m=1}^{M} R_{im} \lambda_m$

• learn $\lambda$ such that $f$ is closest to harmonic
  ▶ minimize energy $E(f) = f^\top \Delta f$
  ▶ convex optimization
  ▶ closed form solution $\lambda = - (R^\top \Delta_{uu} R)^{-1} R^\top \Delta_{ul} Y_l$
Harmonic mixtures

mixture component labels $\lambda$ follow the graph
Harmonic mixtures
computational savings

• computation on unlabeled data
  ▶ harmonic mixtures

\[ f_u = -R \left( R^\top \Delta_{uu} R \right)^{-1} R^\top \Delta_{ul} Y_l \]

▶ original harmonic function

\[ f_u = -\left( \Delta_{uu} \right)^{-1} \Delta_{ul} Y_l \]

• harmonic mixtures $O(M^3)$, much cheaper than $O(u^3)$

Harmonic mixtures can handle large problems.
Also induction $f(x) = \sum_{m=1}^{M} R_{xm} \lambda_m$
From harmonic functions to kernels

- harmonic functions too specialized?
- I will show you the kernel behind harmonic function
  - general, important concept in machine learning.
  - used in many learning algorithms, e.g. support vector machines
  - on the graph: symmetric, positive semi-definite $n \times n$ matrix

- I will then give you an even better kernel.

but first a short detour ...
The probabilistic model behind harmonic function

- random field $p(f) \propto \exp(-E(f))$
- energy $E(f) = \sum_{i \sim j} W_{ij} (f_i - f_j)^2 = f^\top \Delta f$
- low energy = good label propagation

- if $f \in \{0, 1\}$ discrete, standard Markov random fields (Boltzmann machines), inference hard
The probabilistic model behind harmonic function
Gaussian random fields

(Wei Zhu, Zoubin Ghahramani and John D. Lafferty, ICML2003)

- continuous relaxation $f \in \mathbb{R} \Rightarrow$ Gaussian random field
- Gaussian random field $p(f)$ is a $n$-dimensional Gaussian with
  inverse covariance matrix $\Delta$.

\[
p(f) \propto \exp(-E(f)) = \exp(-f^\top \Delta f)
\]

- harmonic functions are the mean of Gaussian random fields
- Gaussian random fields $=$ Gaussian processes on finite data
- covariance matrix $=$ kernel matrix in Gaussian processes
The kernel behind harmonic functions

\[ K = \Delta^{-1} \]

- \( K_{ij} = \) indirect similarity
  - The direct similarity \( W_{ij} \) may be small
  - But \( K_{ij} \) will be large if many paths between \( i, j \)

- \( K \) can be used with many kernel machines
  - \( K + \) support vector machine = semi-supervised SVM
  - kernel built on both labeled and unlabeled data
  - additional benefit: handles noisy labeled data
Kernels should encourage smooth eigenvectors

- graph spectrum $\Delta = \sum_{k=1}^{n} \lambda_k \phi_k \phi_k^\top$
- small eigenvalue, smooth eigenvector
  \[ \sum_{i \sim j} W_{ij} (\phi_k(i) - \phi_k(j))^2 = \lambda_k \]
- kernels *encourage* smooth eigenvectors with large weights

Laplacian $\Delta = \sum_k \lambda_k \phi_k \phi_k^\top$

Harmonic kernel $K = \Delta^{-1} = \sum_k \frac{1}{\lambda_k} \phi_k \phi_k^\top$

- smooth functions good for semi-supervised learning

\[ \| f \|_K = f^\top K^{-1} f = f^\top \Delta f = \sum_{i \sim j} W_{ij} (f_i - f_j)^2 \]
General semi-supervised kernels

- $\Delta^{-1}$ not the only semi-supervised kernel, may not be the best
- General principle for creating semi-supervised kernels
  \[ K = \sum_i r(\lambda_i) \phi_i \phi_i^\top \]
- $r(\lambda)$ should be large when $\lambda$ is small, to encourage smooth eigenvectors.
- Specific choices of $r()$ lead to known kernels
  - harmonic function kernel $r(\lambda) = 1/\lambda$
  - diffusion kernel $r(\lambda) = \exp(-\sigma^2 \lambda)$
  - random walk kernel $r(\lambda) = (\alpha - \lambda)^p$
- *Is there a best $r()$?* Yes, as measured by kernel alignment.
Alignment measures kernel quality

- measures kernel by its alignment to the labeled data $Y_l$

$$\text{align}(K, Y_l) = \frac{\langle K_{ll}, Y_l Y_l^\top \rangle}{\| K_{ll} \| \cdot \| Y_l Y_l^\top \|}$$

- extension of cosine angle between vectors
- high alignment related to good generalization performance
- leads to a convex optimization problem
Finding the best kernel

(Zhu, Kandola, Ghahramani and Lafferty, NIPS2004)

• the order constrained semi-supervised kernel

\[
\max_r \quad \text{align}(K, Y_l) \\
\text{subject to} \quad K = \sum_i r_i \phi_i \phi_i^T \\
\quad r_1 \geq \cdots \geq r_n \geq 0
\]

• order constraints \( r_1 \geq \cdots \geq r_n \) encourage smoothness

• convex optimization

• \( r \) nonparametric
The order constrained kernel improves alignment and accuracy
text categorization (religion vs. atheism), 50 labeled and 2000 unlabeled articles.

- alignment

<table>
<thead>
<tr>
<th>kernel</th>
<th>order</th>
<th>harmonic</th>
<th>RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>alignment</td>
<td>0.31</td>
<td>0.17</td>
<td>0.04</td>
</tr>
</tbody>
</table>

- accuracy with support vector machines

<table>
<thead>
<tr>
<th>kernel</th>
<th>order</th>
<th>harmonic</th>
<th>RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>accuracy</td>
<td>84.5</td>
<td>80.4</td>
<td>69.3</td>
</tr>
</tbody>
</table>

We now have good kernels for semi-supervised learning.
Other research (1)

Graph hyperparameter learning

- what if we don’t know $W_{ij}$?

- set up hyperparameters $W_{ij} = \exp\left(-\sum_d \frac{(x_{id} - x_{jd})^2}{\alpha_d}\right)$

- learn $\alpha$ with e.g. Bayesian evidence maximization

average 7

average 9

learned $\alpha$
Other research (2)
Sequences and other structured data

(Lafferty, Zhu and Liu, 2004)

- what if $x_1 \cdots x_n$ form sequences?
  speech, natural language
  processing, biosequence analysis, etc.
- conditional random fields (CRF)
- kernel CRF
- kernel CRF + semi-supervised
  kernels
Other research (3)

Active learning

(Zhu, Lafferty and Ghahramani, 2003b)

- what if the computer can ask for labels?
- smart queries: not necessarily the most ambiguous points

![Diagram]

- active learning + semi-supervised learning, fast algorithm
Related work in semi-supervised learning

based on different assumptions

<table>
<thead>
<tr>
<th>method</th>
<th>assumptions</th>
<th>references</th>
</tr>
</thead>
<tbody>
<tr>
<td>graph</td>
<td>similar feature, same label</td>
<td>this talk</td>
</tr>
<tr>
<td></td>
<td>mincuts</td>
<td>(Blum and Chawla, 2001)</td>
</tr>
<tr>
<td></td>
<td>normalized Laplacian</td>
<td>(Zhou et al., 2003)</td>
</tr>
<tr>
<td></td>
<td>regularization</td>
<td>(Belkin et al., 2004)</td>
</tr>
<tr>
<td>mixture model, EM</td>
<td>generative model</td>
<td>(Nigam et al., 2000)</td>
</tr>
<tr>
<td>transductive SVM</td>
<td>low density separation</td>
<td>(Joachims, 1999)</td>
</tr>
<tr>
<td>co-training</td>
<td>feature split</td>
<td>(Blum and Mitchell, 1998)</td>
</tr>
</tbody>
</table>

Semi-supervised learning has so far received relatively little attention in statistics literature.
Some key contributions

- harmonic function formulations for semi-supervised learning
- solving large scale problems with harmonic mixtures
- semi-supervised kernels by spectral transformation of the graph Laplacian
- kernelizing conditional random fields
- combining active learning and semi-supervised learning
Summary

Unlabeled data can improve classification.

The methods have reached the stage where we can apply them to real-world tasks.
Future Plans

- continue the research on semi-supervised learning
  - structured data, ranking, clustering, explore different assumptions
- application to human language tasks
  - speech recognition, document categorization, information retrieval, sentiment analysis
- explore novel machine learning approaches
  - text mixed with other modalities, e.g. images; speech and multimodal user interfaces; graphics; vision
- collaboration
Thank You
References


X. Zhu, Jaz Kandola, Z. Ghahramani, J. Lafferty. *Nonparametric Transforms of
Findory News

Top Stories

Report criticizes shuttle workforce planning
Space Today (2 hours ago)
A report released Thursday concludes that NASA has not done nearly enough to maintain the... (read more)

Jacko's 7 'BAD' PRIORS
Yahoo News (11 hours ago)
New York Post - SANTA MARIA, Calif. — Tomorrow is the most important day in Michael Jackson's life... (read more)

Google ensnared in a war of words
International Herald Tribune (6 hours ago)
The growing number of lawsuits against Google around the... (read more)

Fishing communities 'at HIV risk'
BBC (18 hours ago)
Fishing communities are up to 10 times more vulnerable to HIV/AIDS than the rest of the population, a study says... (read more)

Amazon knows who you are
Wired (2 hours ago)
Many companies have systems for tracking customer habits, but Amazon has collected info longer and used it more proactively. It now has technology that tracks data on those you buy gifts for, and it reserves the right to sell it all... (read more)
NASA's Deep Impact has a small blurry glitch

Number of very high-energy gamma ray sources doubles
New Scientist (2 days ago)

Rainbows on Titan
Science @ NASA (2 days ago)

Station's 'phantom torque' complicates spacewalk
New Scientist (2 days ago)
Graph spectrum $\Delta = \sum_{i=1}^{n} \lambda_i \phi_i \phi_i^T$
Labels for the components do not follow the graph.
(Nigam et al., 2000)
Other research (2)
Sequences and other structured data

- KCRF: \( p_f(y|x) = Z^{-1}(x, f) \exp \left( \sum_c f(x, y_c) \right) \)
- \( f \) induces regularized negative log loss on training data
  \[
  R(f) = \sum_{i=1}^{l} - \log p_f(y^{(i)}|x^{(i)}) + \Omega(\|f\|_K)
  \]
- representer theorem for KCRFs: loss minimizer
  \[
  f^*(x, y_c) = \sum_{i=1}^{l} \sum_{c'} \sum_{y'_{c'}} \alpha(i, y'_{c'}) K((x^{(i)}, y'_{c'}), (x, y_c))
  \]
Other research (2)
Sequences and other structured data

- learn $\alpha$ to minimize $R(f)$, convex, sparse training
- special case $K((x^{(i)}, y'_{c'}), (x, y_c)) = \psi(K'(x^{(i)}_{c'}, x_c), y'_{c'}, y_c)$
- $K'$ can be a semi-supervised kernel.
Other research (3)
Active Learning

- generalization error
  \[ \text{err} = \sum_{i \in U} \sum_{y_i=0,1} (\text{sgn}(f_i) \neq y_i) P_{\text{true}}(y_i) \]

- approximation 1
  \[ P_{\text{true}}(y_i = 1) \leftarrow f_i \]

- estimated error
  \[ \hat{\text{err}} = \sum_{i \in U} \min (f_i, 1 - f_i) \]
Other research (3)
Active Learning

• estimated error after querying \( k \) with answer \( y_k \)

\[
\hat{\text{err}}^+(x_k, y_k) = \sum_{i \in U} \min \left( f_i^+(x_k, y_k), 1 - f_i^+(x_k, y_k) \right)
\]

• approximation 2

\[
\hat{\text{err}}^{+x_k} = (1 - f_k)\hat{\text{err}}^+(x_k, 0) + f_k\hat{\text{err}}^+(x_k, 1)
\]

• select query \( k^* \) to minimize the estimated error

\[
k^* = \arg \min_k \hat{\text{err}}^{+x_k}
\]
Other research (3)

Active Learning

- ‘re-train’ is fast with semi-supervised learning

\[ f^+_u(x_k, y_k) = f_u + (y_k - f_k) \frac{(\Delta_{uu})^{-1}}{(\Delta_{uu})^{-1}} \]