Transfer Learning Methods and Applications in Computational Biology

Gunnar Rätsch



Friedrich Miescher Laboratory
of the Max Planck Society

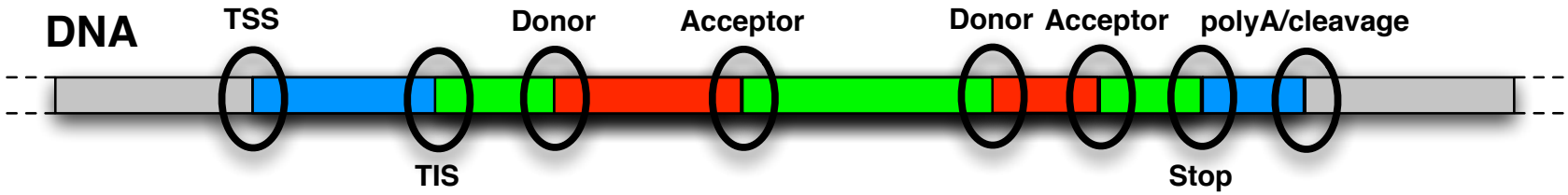
Tübingen, Germany

December 12, 2009

NIPS Transfer Learning Workshop
Whistler, B.C.

Roadmap

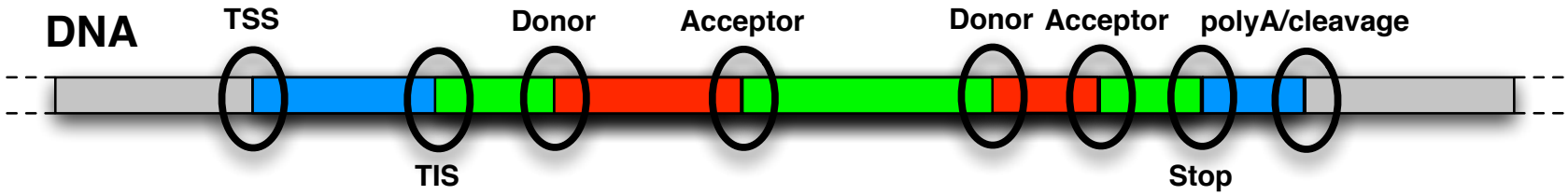
- Motivation from computational biology



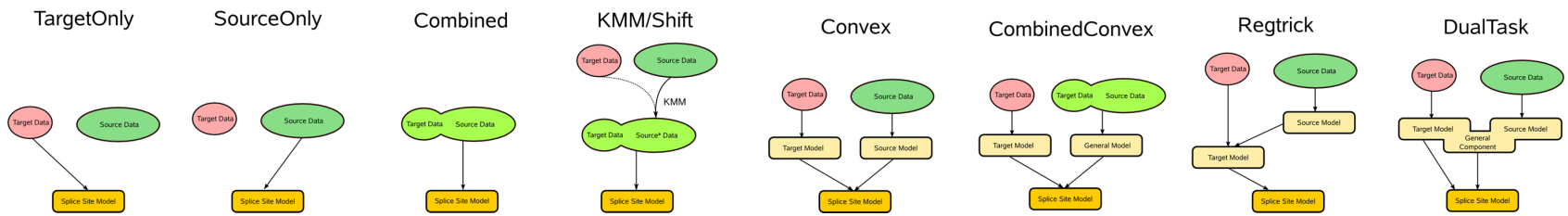
- Empirical comparison of domain adaptation algorithms
- Algorithms for hierarchical multi-task learning
- Discussion & Conclusion

Roadmap

- Motivation from computational biology



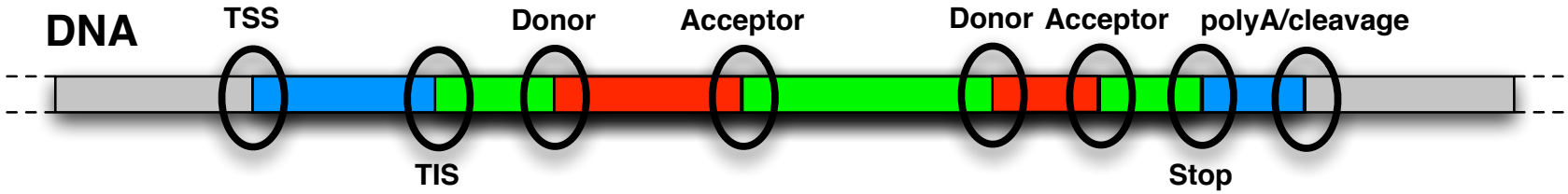
- Empirical comparison of domain adaptation algorithms



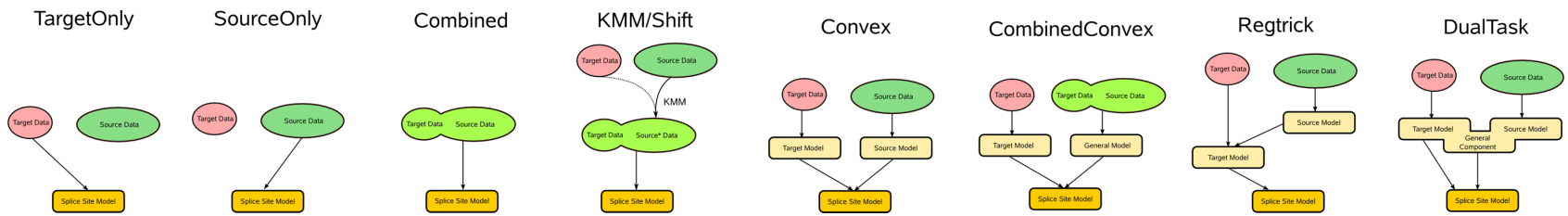
- Algorithms for hierarchical multi-task learning
- Discussion & Conclusion

Roadmap

- Motivation from computational biology



- Empirical comparison of domain adaptation algorithms



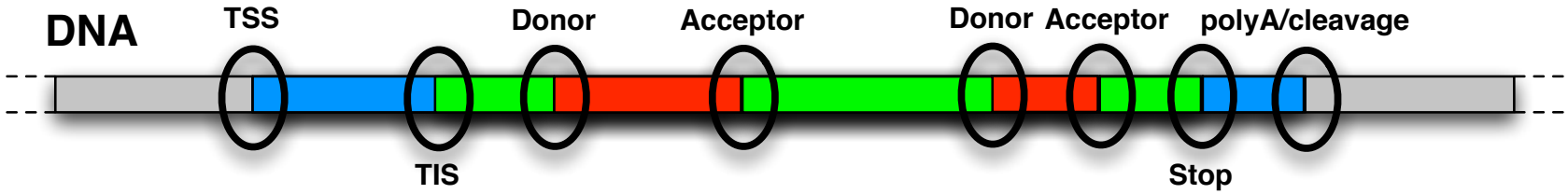
- Algorithms for hierarchical multi-task learning



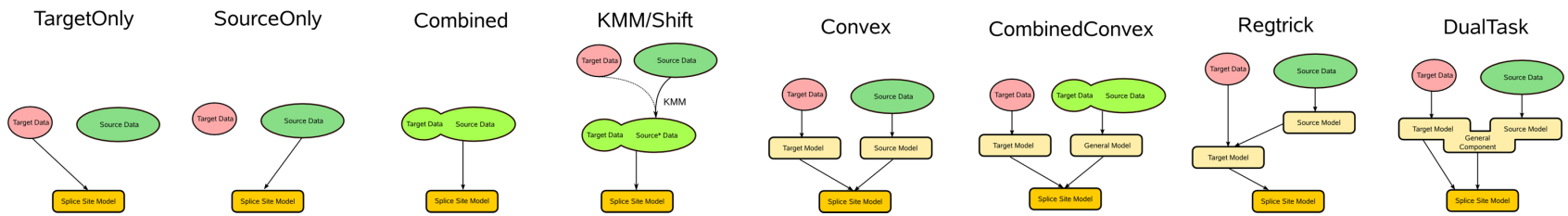
- Discussion & Conclusion

Roadmap

- Motivation from computational biology



- Empirical comparison of domain adaptation algorithms



- Algorithms for hierarchical multi-task learning
- Discussion & Conclusion



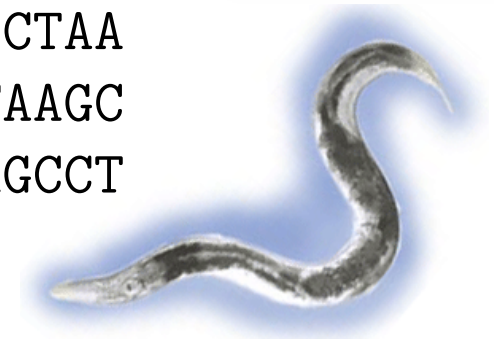
$\frac{1}{150\,000}$ of the DNA of *C. elegans*

fml

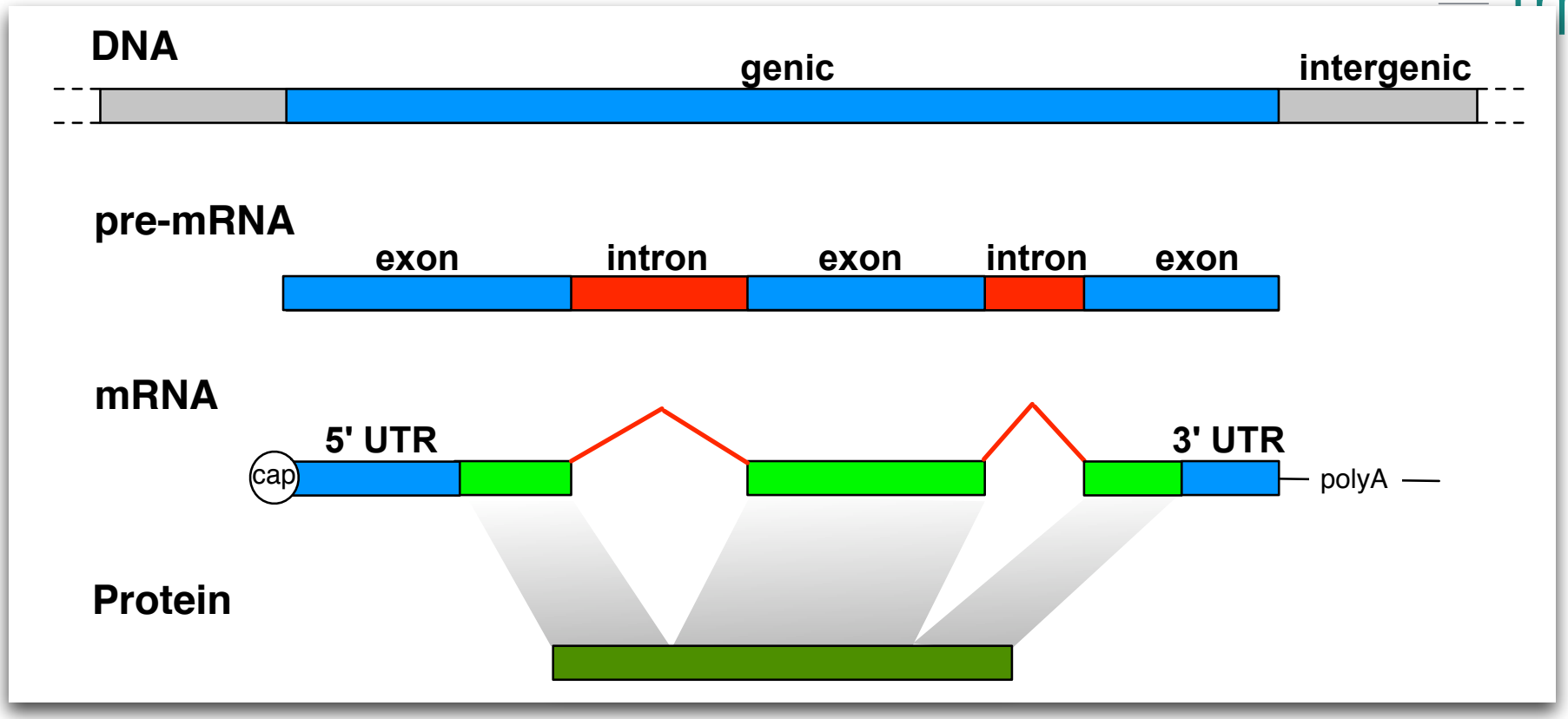
>CHROMOSOME_I
GCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGC
CTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCT
AAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAA
GCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGC
CTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCT
AAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAA
GCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGC
CTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAGCCT
AAGCCTAAGCCTAAGCCTAAGCCTAAGCCTAAAAAATTGAGATAAGAAAA
CATTTTACTTTTTCAAAATTGTTTTCATGCTAAATTCAAAACGTTTTTTT
TTTAGTGAAGCTTCTAGATATTTGGCGGGTACCTCTAATTTTGCCTGCCT
GCCAACCTATATGCTCCTGTGTTTAGGCCTAATACTAAGCCTAAGCCTAA
GCCTAATACTAAGCCTAAGCCTAAGACTAAGCCTAATACTAAGCCTAAGC
CTAAGACTAAGCCTAAGACTAAGCCTAAGACTAAGCCTAATACTAAGCCT
...





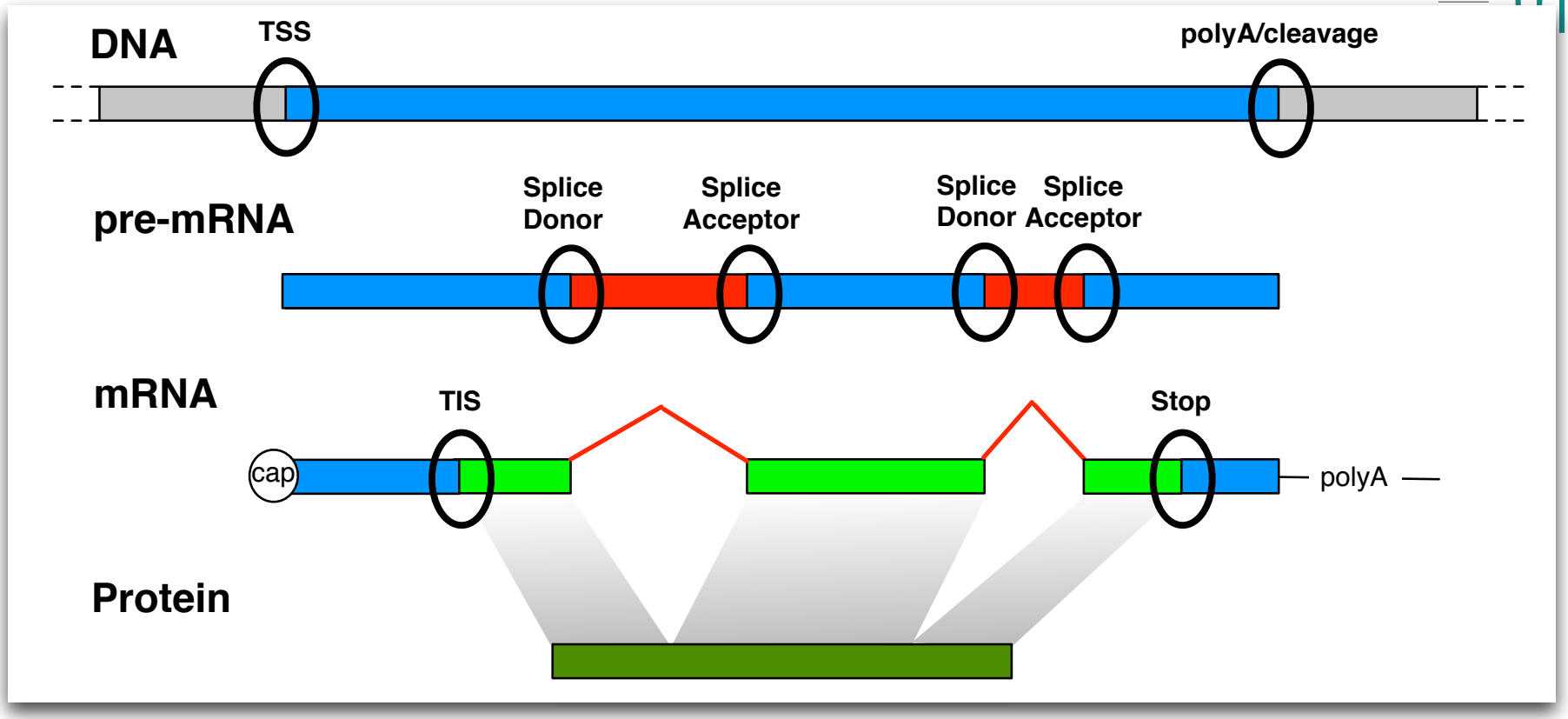


Computational Gene Finding



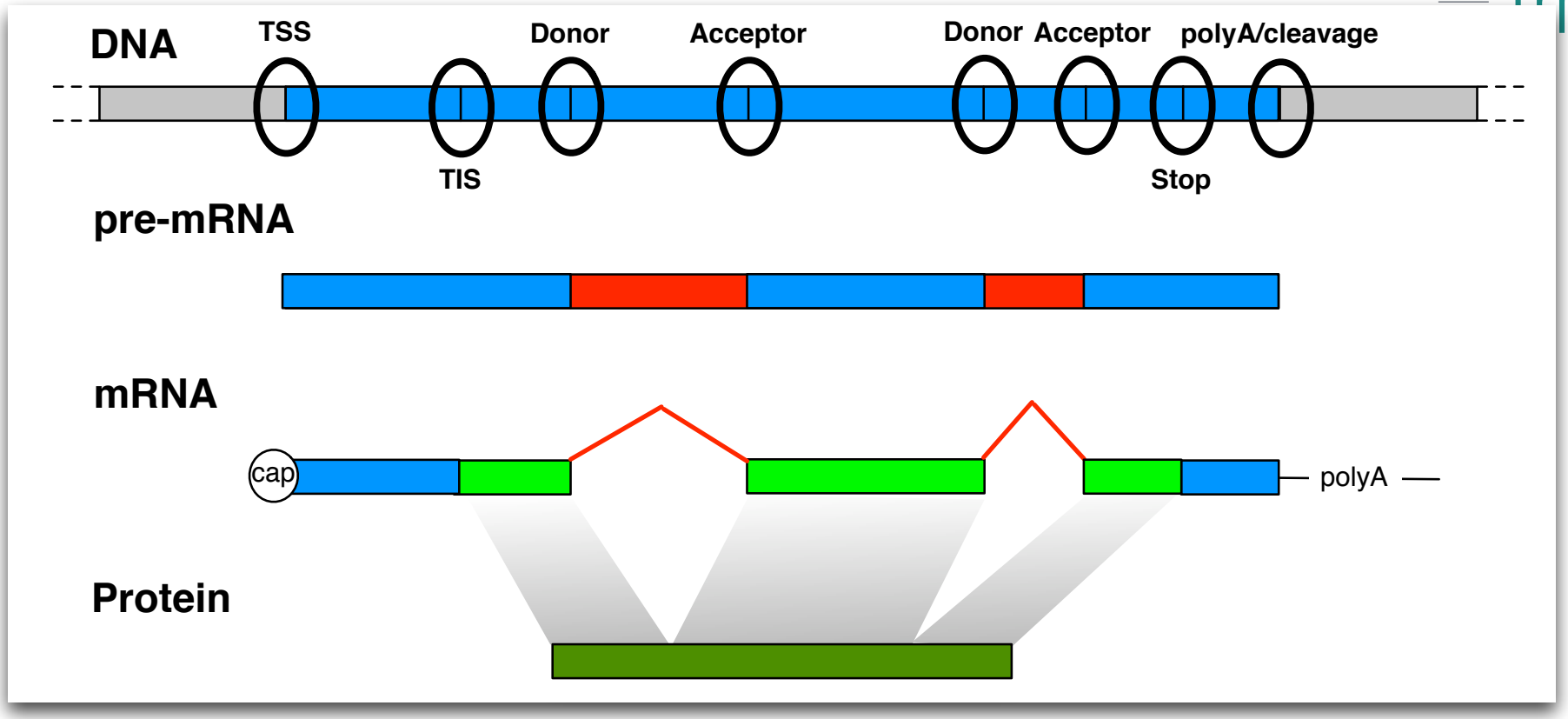
- Given a piece of DNA sequence
- Predict gene products including intermediate processing steps
- Predict signals used during processing
- Predict the correct corresponding **label sequence** with labels "intergenic", "exon", "intron", "5' UTR", etc.

Computational Gene Finding



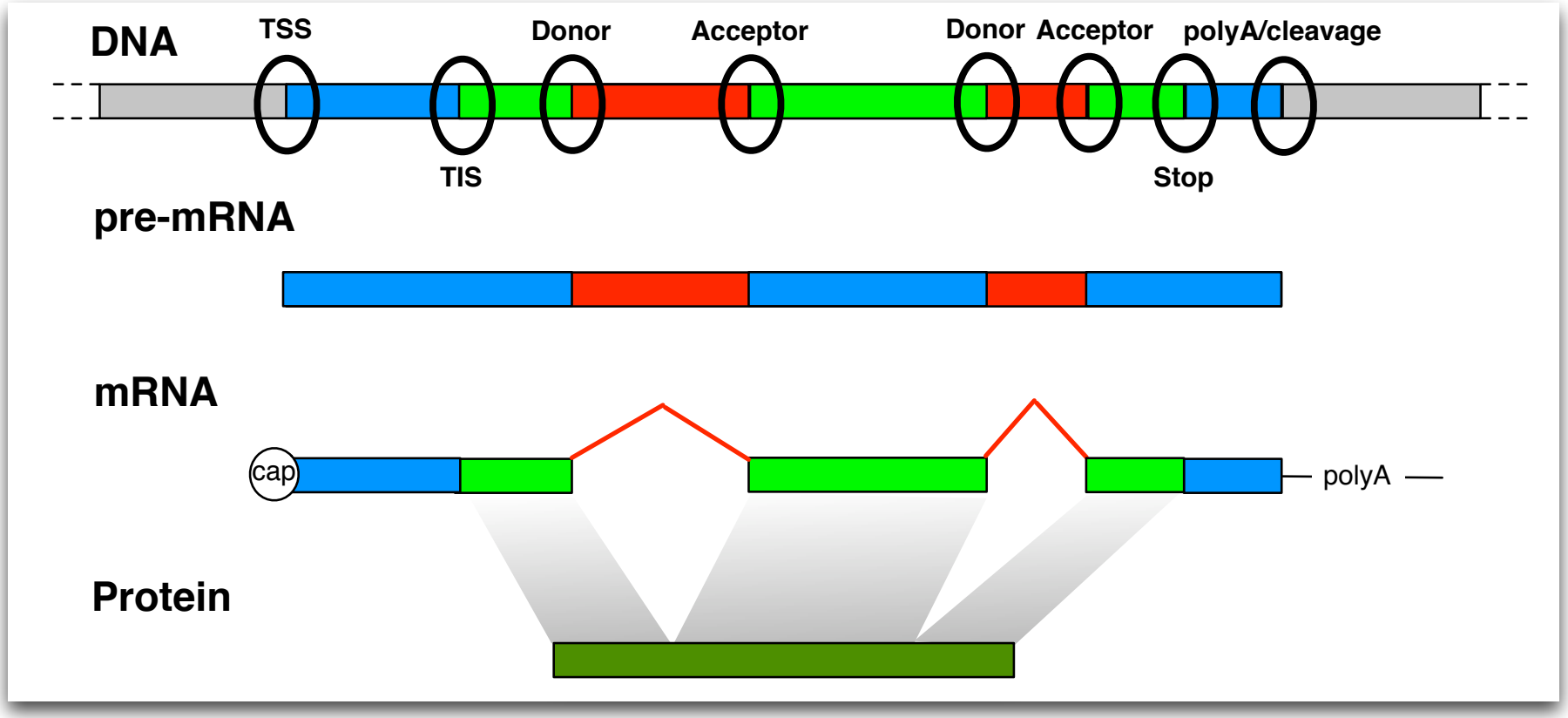
- Given a piece of DNA sequence
- Predict gene products including intermediate processing steps
- Predict signals used during processing
- Predict the correct corresponding **label sequence** with labels "intergenic", "exon", "intron", "5' UTR", etc.

Computational Gene Finding



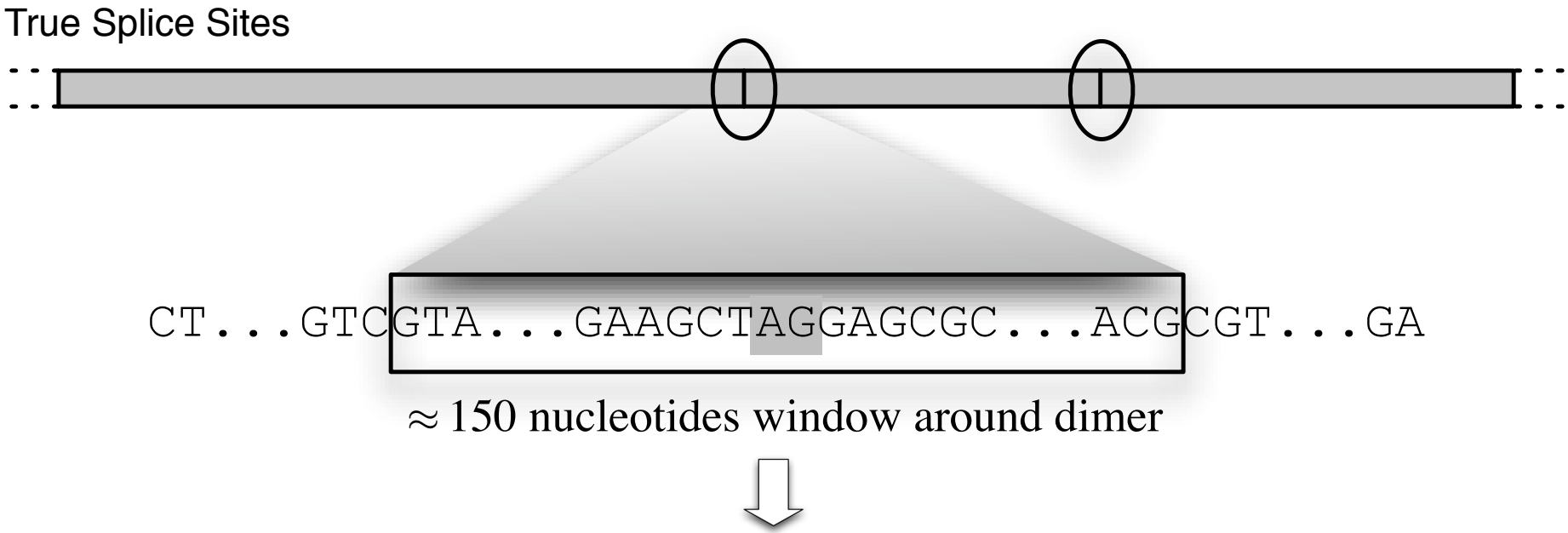
- Given a piece of DNA sequence
- Predict gene products including intermediate processing steps
- Predict signals used during processing
- Predict the correct corresponding **label sequence** with labels "intergenic", "exon", "intron", "5' UTR", etc.

Computational Gene Finding

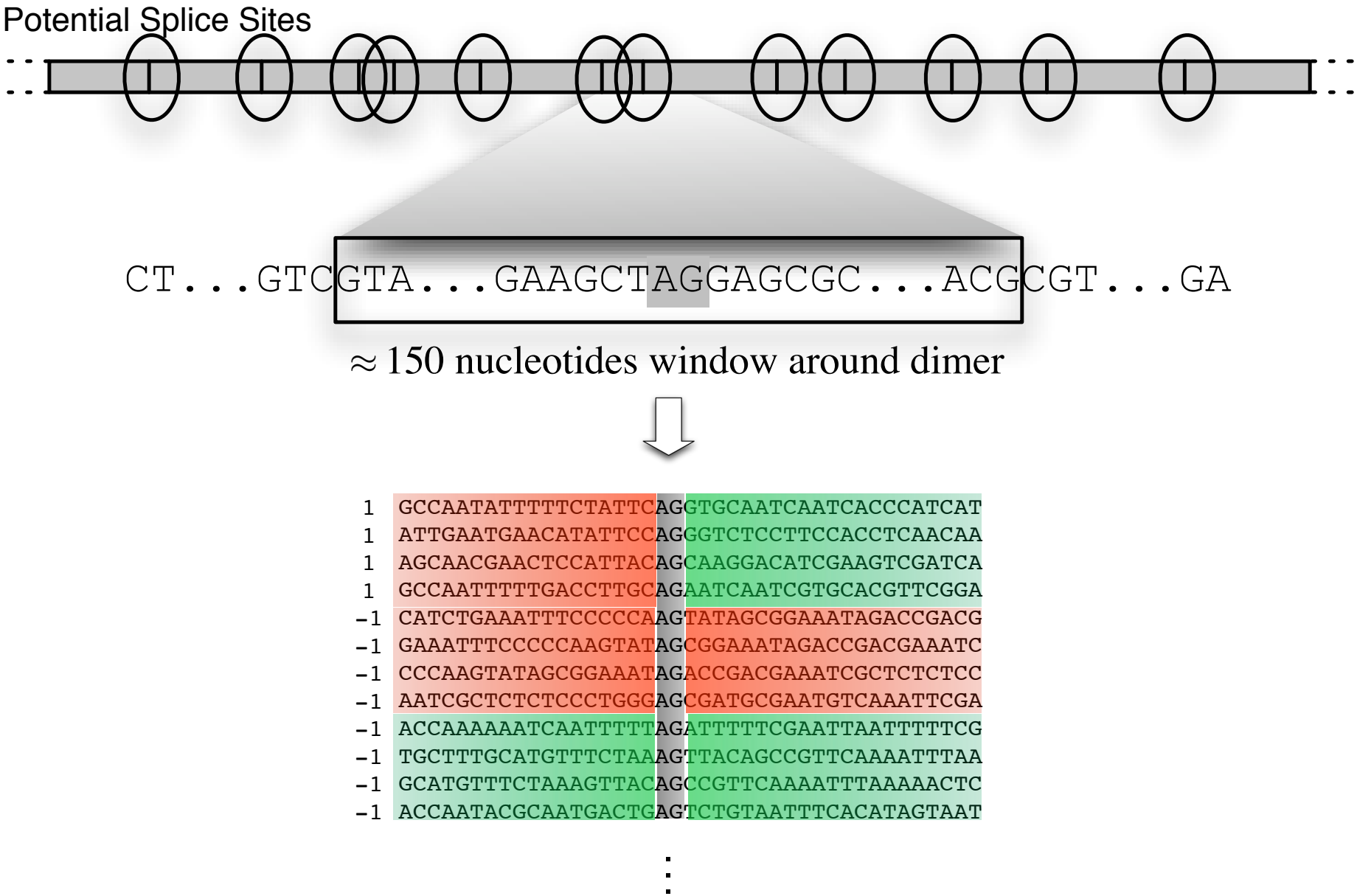


- Given a piece of DNA sequence
- Predict gene products including intermediate processing steps
- Predict signals used during processing
- Predict the correct corresponding **label sequence** with labels "intergenic", "exon", "intron", "5' UTR", etc.

Example: Splice Site Recognition

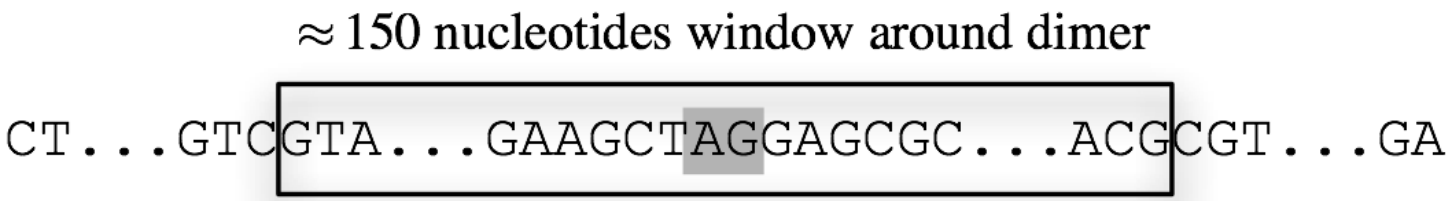


Example: Splice Site Recognition



Example: Splice Site Recognition

Idea: Discriminate true signal positions against all other positions



Binary classification problem

- True sites: fixed window around a true splice site
- Decoy sites: all other consensus sites

Model is learnt from labeled training examples

- SVM with Weighted Degree Kernel
- Millions of examples for training

Domain Adaptation for Genome Annotation



Motivation:

- Increasing number of sequenced genomes
- Often newly sequenced genomes are poorly annotated
- However often relatives with good annotation exist
- Idea: Transfer knowledge between organisms

Example: Splice site annotation in worm genomes

- Newly sequenced organism: *C. briggsae*
  - $\approx$ 100 confirmed genes (590 splice site pairs)
- Well annotated relative: *C. elegans*
  - $\approx$ 10.000 confirmed genens (36.782 splice site pairs)

Domain Adaptation for Genome Annotation

Motivation:

- Increasing number of sequenced genomes
- Often newly sequenced genomes are poorly annotated
- However often relatives with good annotation exist
- Idea: Transfer knowledge between organisms

Example: Splice site annotation in worm genomes

- Newly sequenced organism: *C. briggsae*
  - $\approx$ 100 confirmed genes (590 splice site pairs)
- Well annotated relative: *C. elegans*
  - $\approx$ 10.000 confirmed genens (36.782 splice site pairs)

The Bioinformatics Way of Transfer Learning

1. Use trained model of related organism, e.g. train on *C. elegans* and predict on other nematodes (Schweikert et al., 2009)

Genome	Genome size [Mbp]	No. of genes	No. exons/gene (mean)	mGene accuracy	best other accuracy
C. remanei	235.94	31503	5.7	**96.6%**	93.8%
C. japonica	266.90	20121	5.3	**93.3%**	88.7%
C. brenneri	453.09	41129	5.4	**93.1%**	87.8%
C. briggsae	108.48	22542	6.0	**87.0%**	82.0%

2. Homology-based label generation (a.k.a. "Comparative genomics")

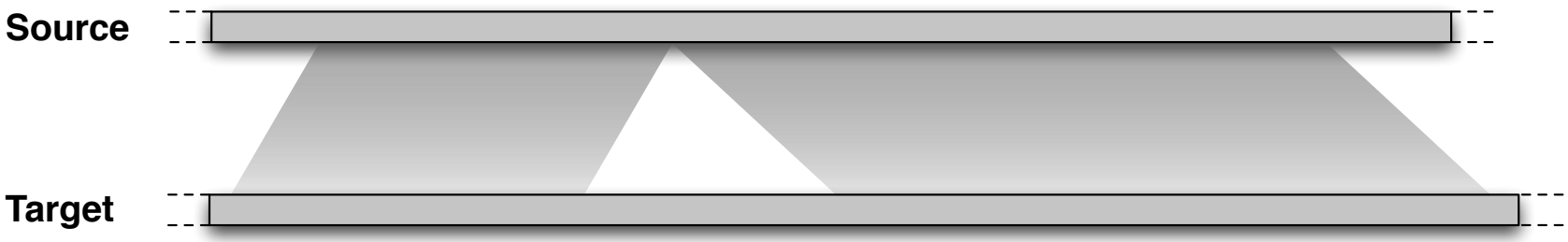
Works for closely related species, does not require any labeled data from target organism.

The Bioinformatics Way of Transfer Learning

1. Use trained model of related organism, e.g. train on *C. elegans* and predict on other nematodes (Schweikert et al., 2009)

Genome	Genome size [Mbp]	No. of genes	No. exons/gene (mean)	mGene accuracy	best other accuracy
C. remanei	235.94	31503	5.7	**96.6%**	93.8%
C. japonica	266.90	20121	5.3	**93.3%**	88.7%
C. brenneri	453.09	41129	5.4	**93.1%**	87.8%
C. briggsae	108.48	22542	6.0	**87.0%**	82.0%

2. Homology-based label generation (a.k.a. "Comparative genomics")



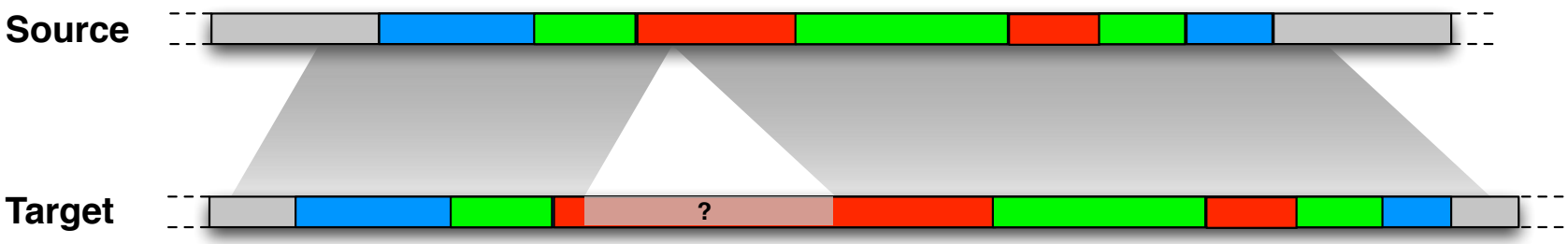
Works for closely related species, does not require any labeled data from target organism.

The Bioinformatics Way of Transfer Learning

1. Use trained model of related organism, e.g. train on *C. elegans* and predict on other nematodes (Schweikert et al., 2009)

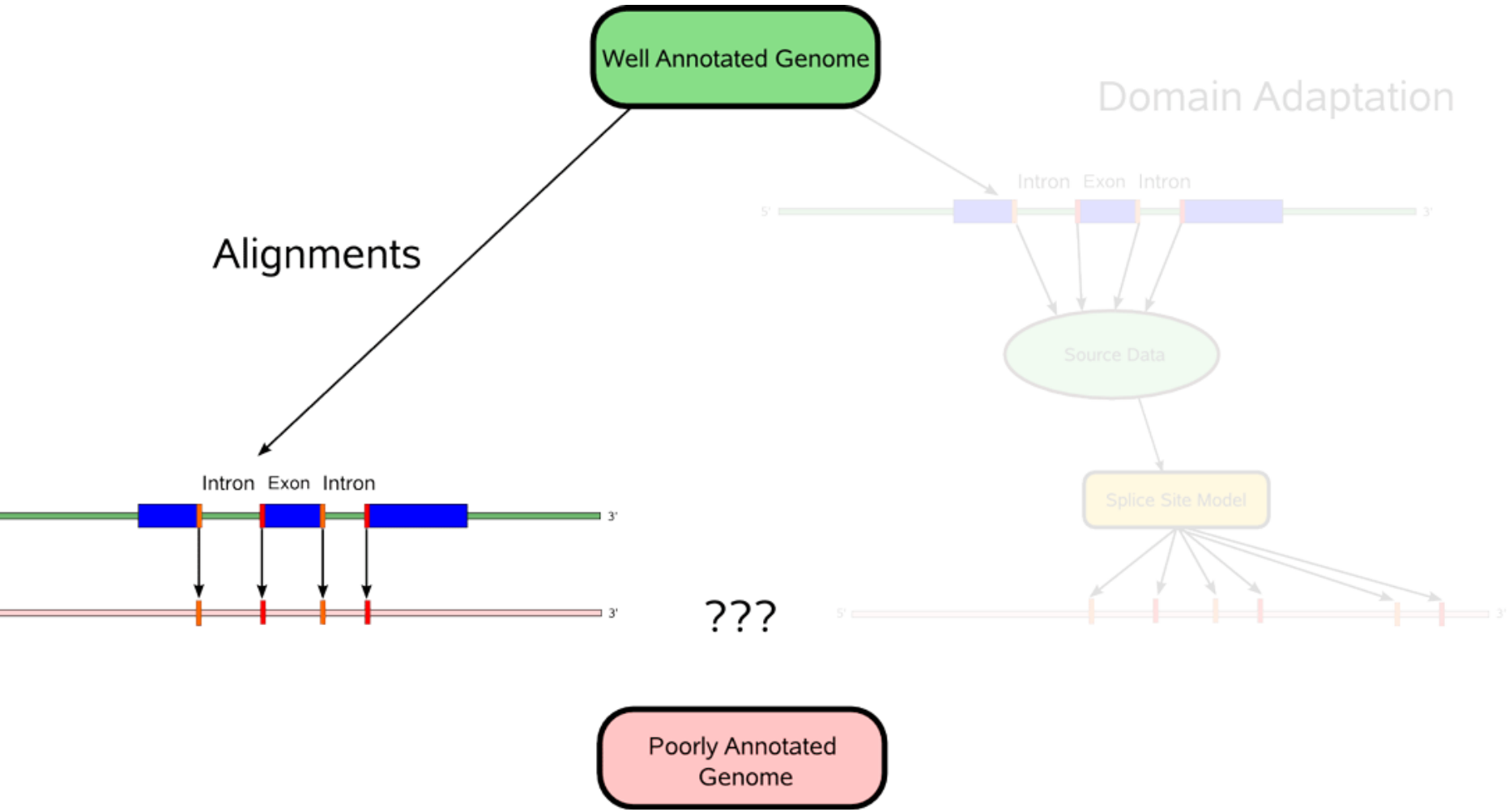
Genome	Genome size [Mbp]	No. of genes	No. exons/gene (mean)	mGene accuracy	best other accuracy
C. remanei	235.94	31503	5.7	**96.6%**	93.8%
C. japonica	266.90	20121	5.3	**93.3%**	88.7%
C. brenneri	453.09	41129	5.4	**93.1%**	87.8%
C. briggsae	108.48	22542	6.0	**87.0%**	82.0%

2. Homology-based label generation (a.k.a. "Comparative genomics")

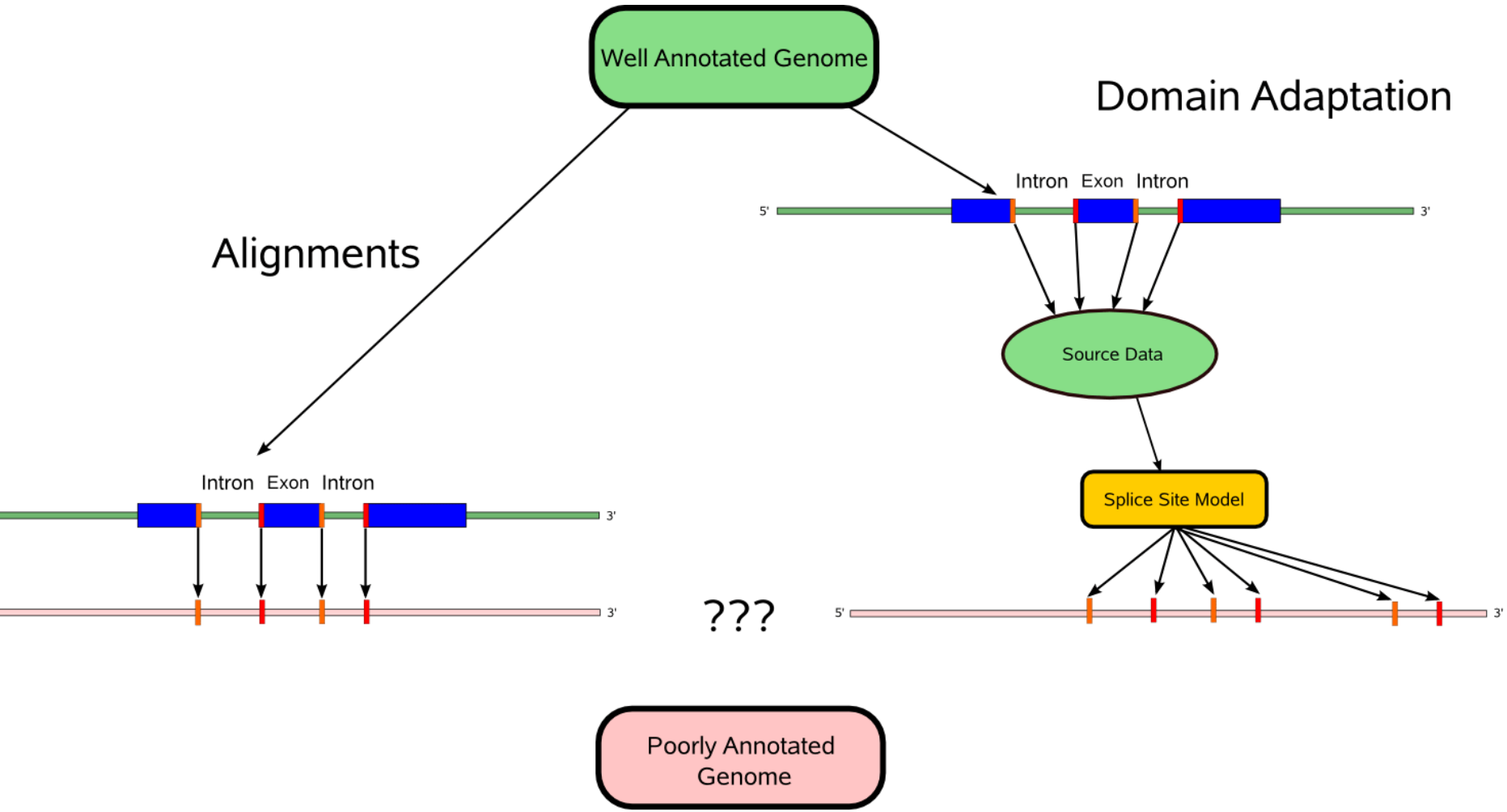


Works for closely related species, does not require any labeled data from target organism.

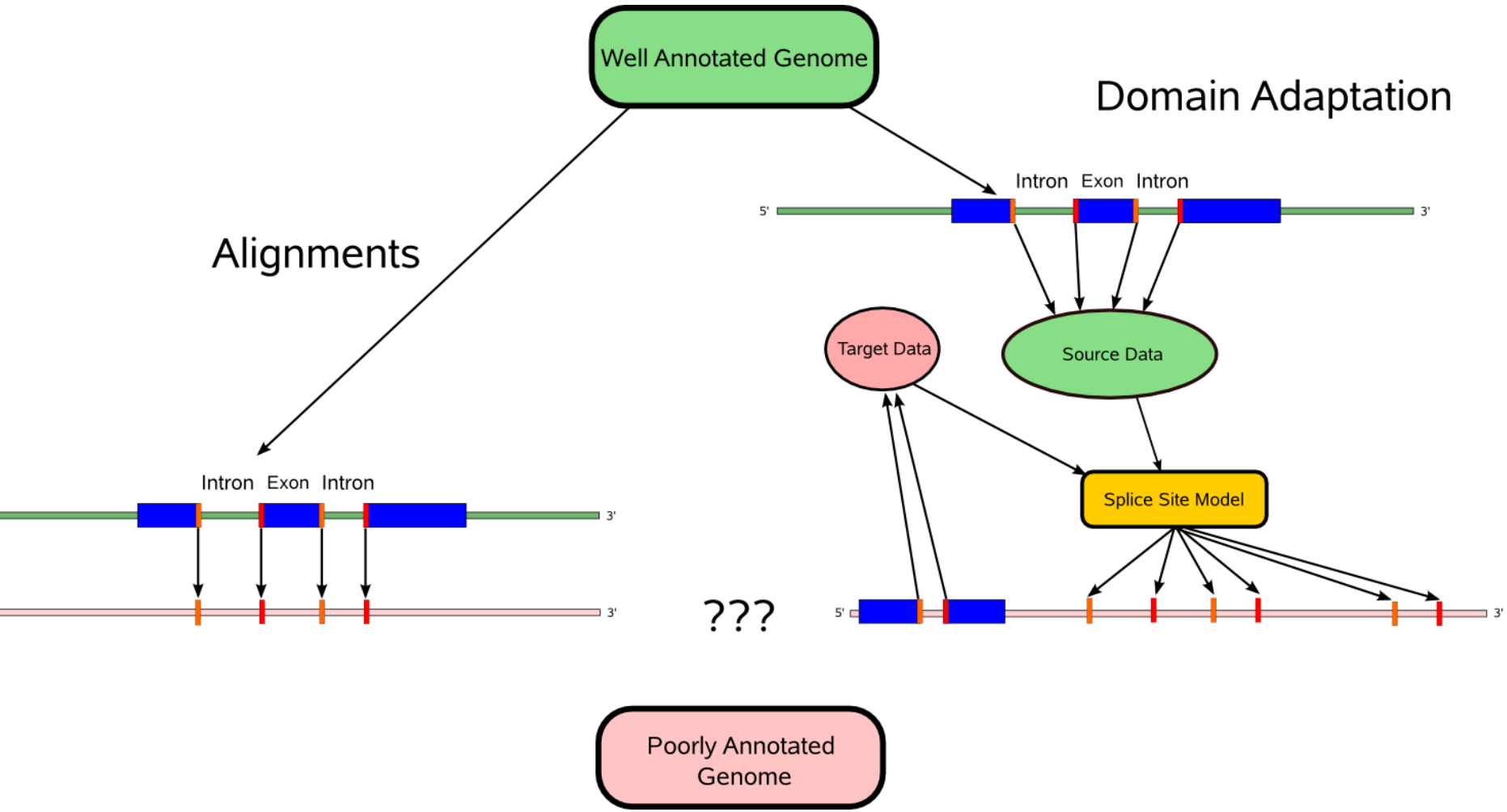
Domain Adaptation by Learning vs. Homology



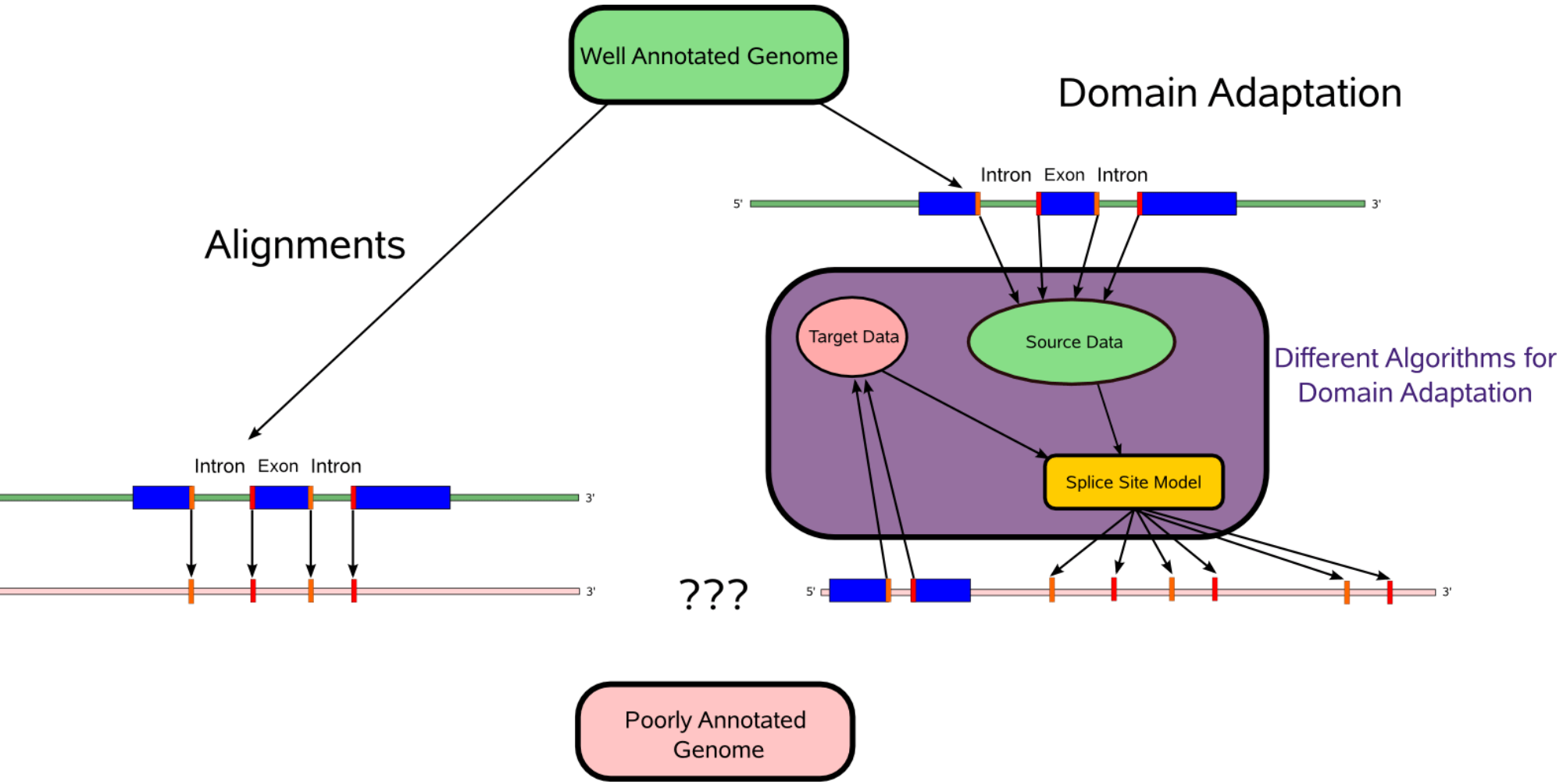
Domain Adaptation by Learning vs. Homology



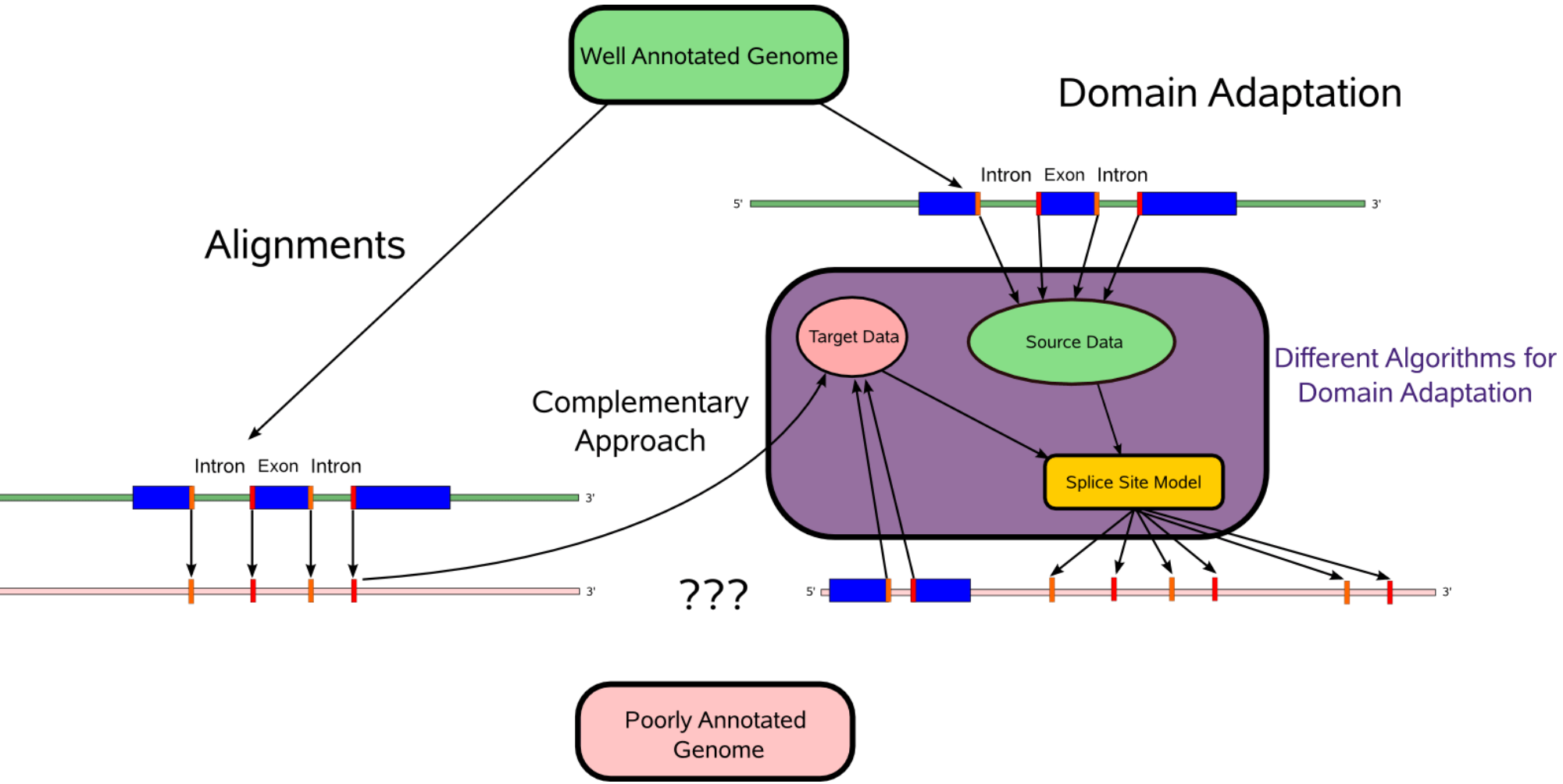
Domain Adaptation by Learning vs. Homology



Domain Adaptation by Learning vs. Homology



Domain Adaptation by Learning vs. Homology



Some Definitions

Terminology:

- Well annotated organisms: Source domain
- Poorly annotated organisms: Target domain

Distributional point of view:

- Example-label pairs are drawn from $P(X, Y)$
- $P_S(X, Y)$ might differ from $P_T(X, Y)$
- Factorization: $P(X, Y) = P(Y|X) \cdot P(X)$
  - Covariate Shift: $P_S(X) \neq P_T(X)$
  - Differing Conditionals: $P_S(Y|X) \neq P_T(Y|X)$

Some Definitions

Terminology:

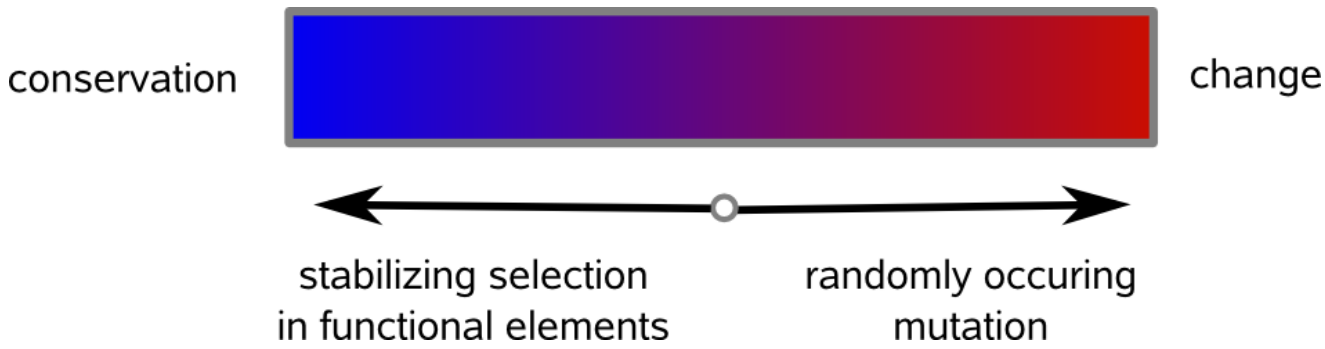
- Well annotated organisms: Source domain
- Poorly annotated organisms: Target domain

Distributional point of view:

- Example-label pairs are drawn from $P(X, Y)$
- $P_S(X, Y)$ might differ from $P_T(X, Y)$
- Factorization: $P(X, Y) = P(Y|X) \cdot P(X)$
  - Covariate Shift: $P_S(X) \neq P_T(X)$
  - **Differing Conditionals:** $P_S(Y|X) \neq P_T(Y|X)$

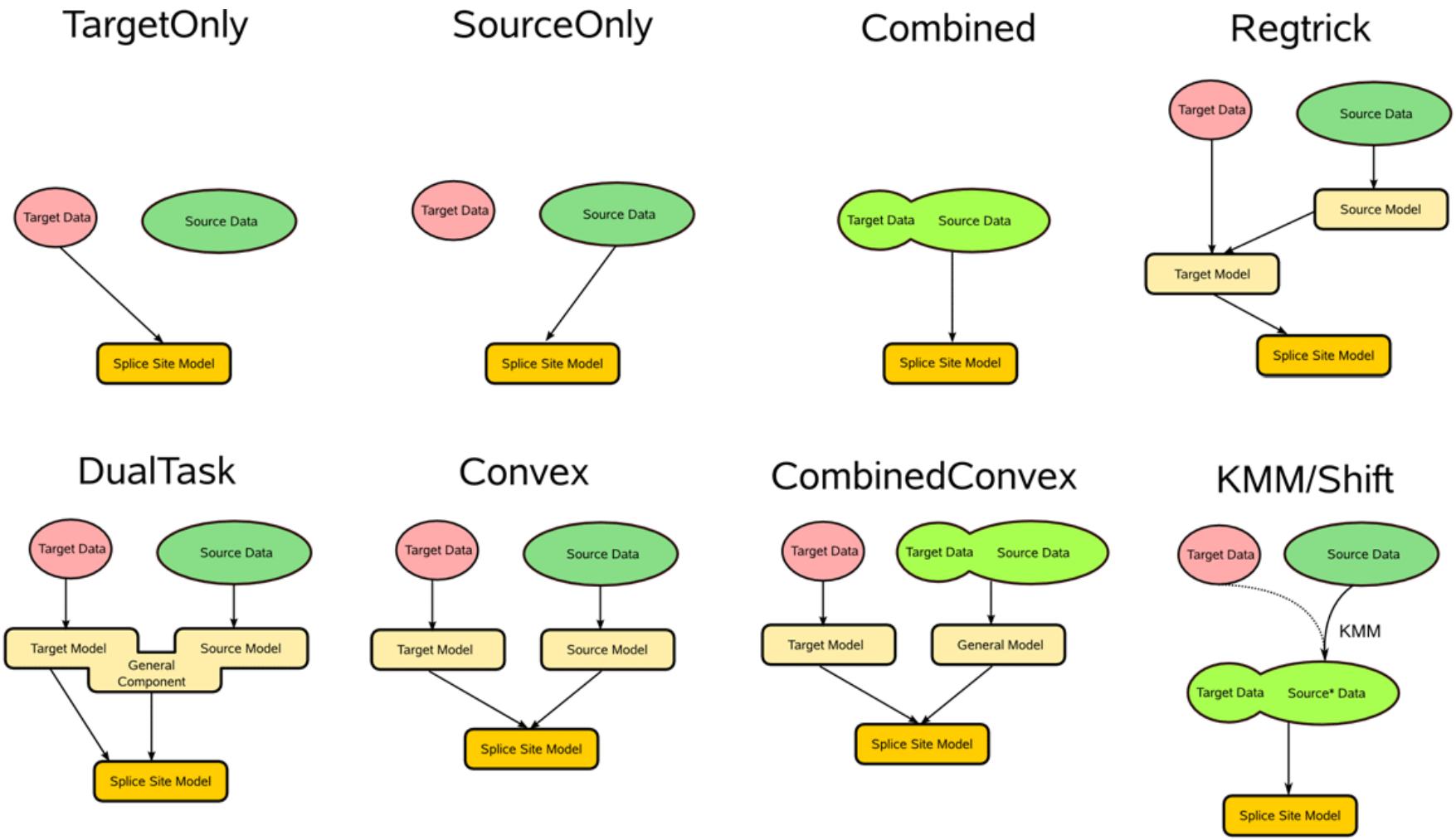
Splice Site Prediction and Domain Adaptation

- Sequence subject to opposing forces:

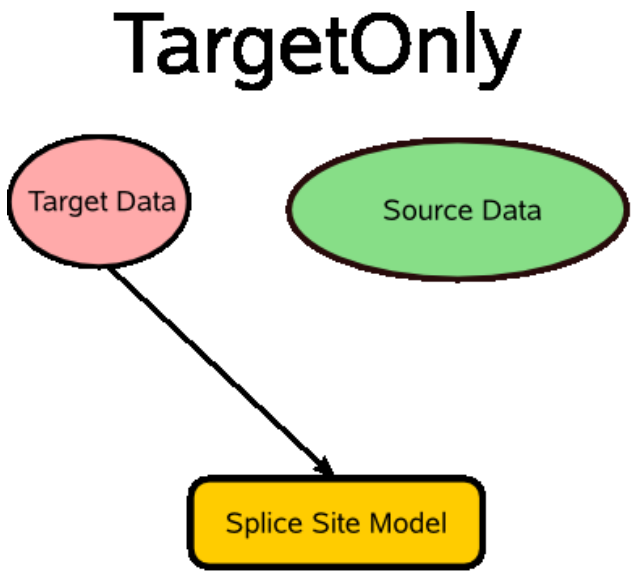


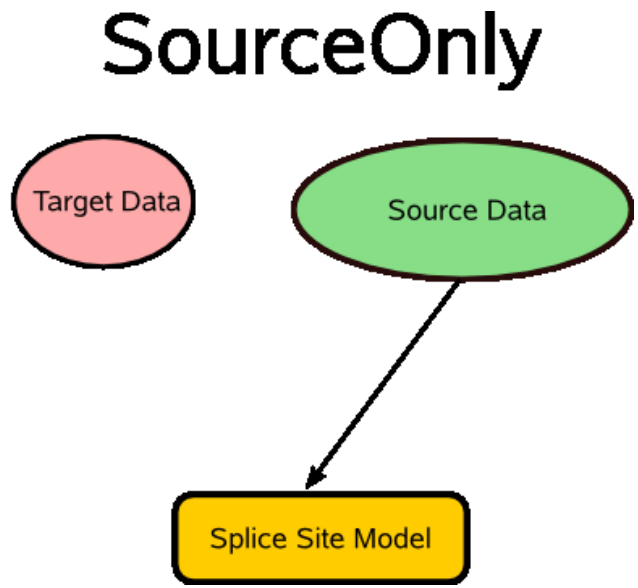
- $P_S(X) \neq P_T(X)$
  - Assume a splice site pattern **x** occurs more frequently in a group of genes (e.g. chromosome)
  - Duplication or deletion events could lead to altered $P(X)$
  - Processing machinery unchanged
- $P_S(Y|X) \neq P_T(Y|X)$
  - Think of the conditional as underlying mechanism
  - Evolution of processing machinery

Domain Adaptation Algorithms Overview



Domain Adaptation Methods





Approach:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\xi_i$$

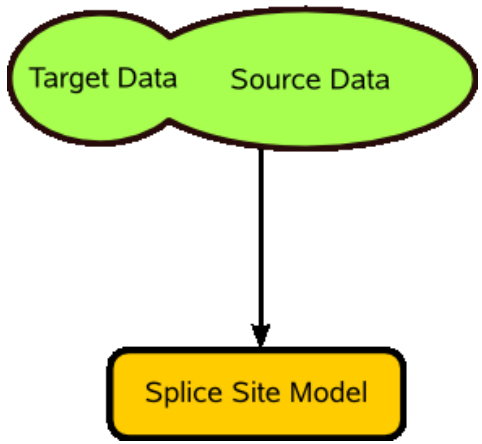
$$\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) + \xi_i - 1 \geq 0 \quad i = 1,\ldots,n$$

$$\xi_i \geq 0 \quad i = 1,\ldots,n$$

Resulting Model: $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$

Domain Adaptation Methods

Combined



Idea: (e.g., Blitzer et al., 2008)

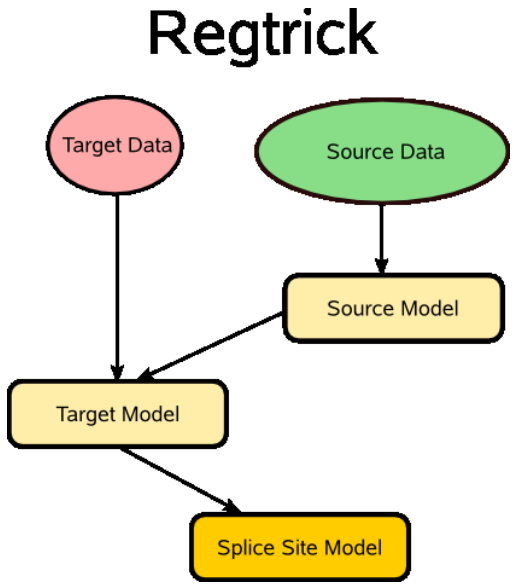
- Train on union of source and target
- Set trade-off via loss-term

Approach:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C_S \sum_{i=1}^{n} \xi_i + C_T \sum_{i=1}^{m} \xi_{n+i}$$

$$\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) + \xi_i - 1 \geq 0 \quad i = 1, \ldots, n+m$$

Domain Adaptation Methods



Idea: (e.g., Chelba and Acero, 2006)

- Previous solution contains prior information
- Modified regularization term

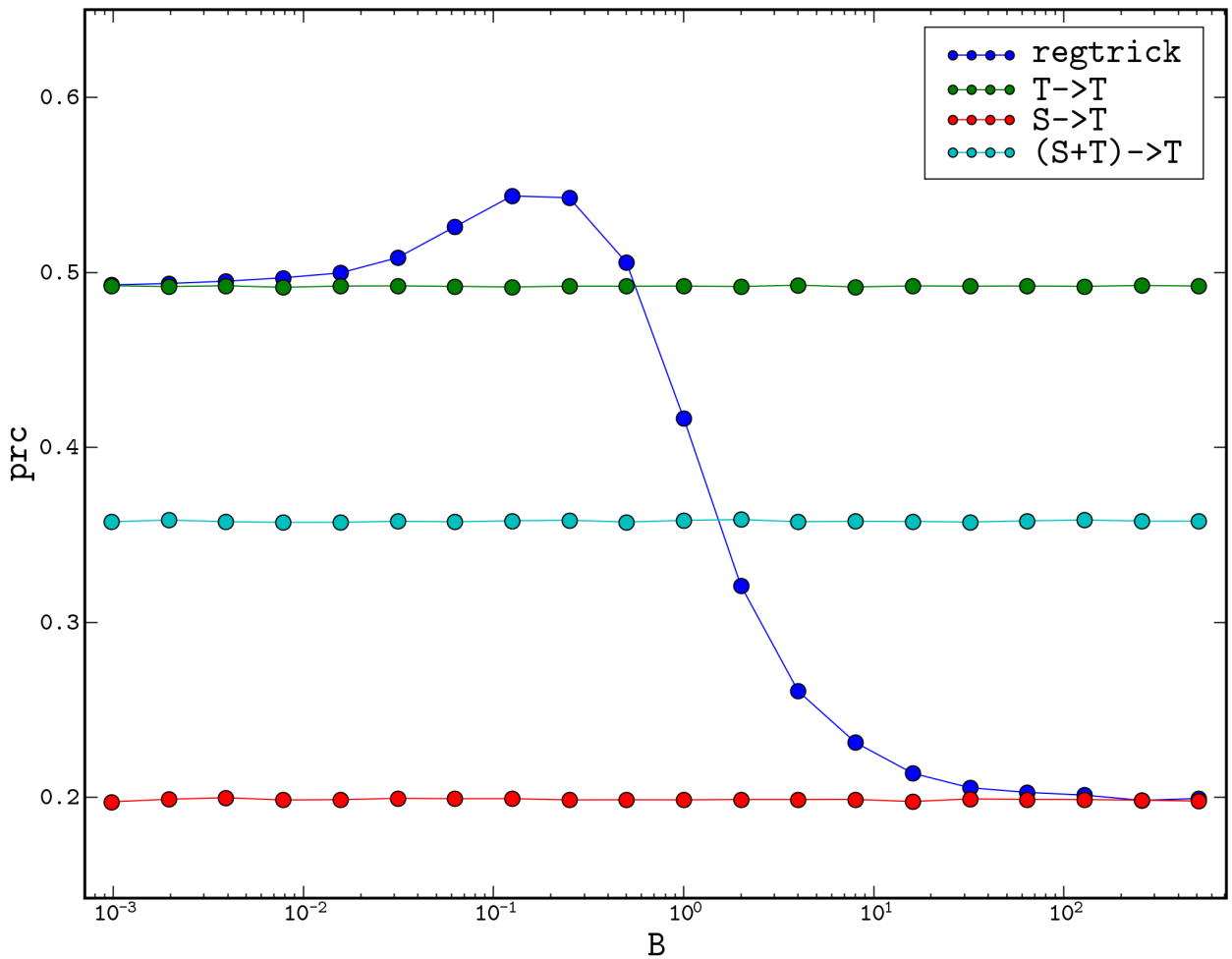
Approach:

$$\min_{\mathbf{w}_T,\xi} \frac{1}{2}\mathbf{w}_T^T\mathbf{w}_T + C\sum_{i=1}^{n}\xi_i - B\mathbf{w}_T^T\mathbf{w}_S$$

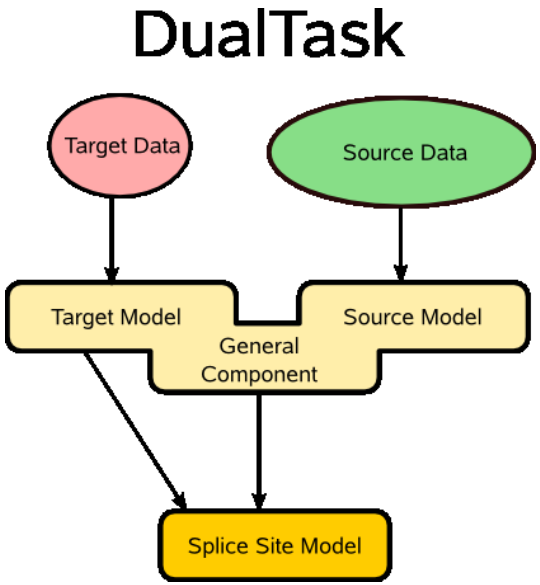
$$\text{s.t. } y_i(\mathbf{w}_T^T\mathbf{x}_i + b) + \xi_i - 1 \geq 0 \quad i = 1,\ldots,n$$

Example

Source: Verticillioides (50k), Target: Drosophila (20k)



Domain Adaptation Methods



Idea: (e.g., Caruana, 1997)

- Simultaneous optimization of both models
- Similarity between solution enforced

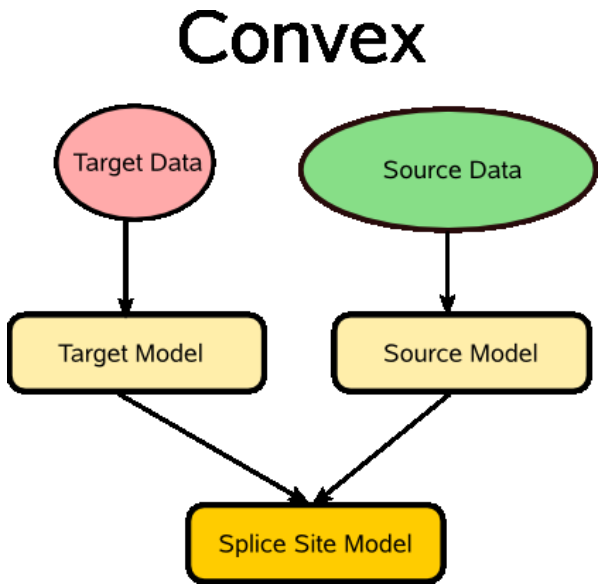
Approach:

$$\min_{\mathbf{w}_S,\mathbf{w}_T,\xi} \quad \frac{1}{2}\|\mathbf{w}_S\|^2 + \frac{1}{2}\|\mathbf{w}_T\|^2 - B\mathbf{w}_T^T\mathbf{w}_S + C\sum_{i=1}^{m+n}\xi_i$$

$$\text{s.t.} \quad y_i(\langle \mathbf{w}_S, \Phi(\mathbf{x}_i)\rangle + b) \geq 1-\xi_i \qquad i = 1,\ldots,m$$

$$y_i(\langle \mathbf{w}_T, \Phi(\mathbf{x}_i)\rangle + b) \geq 1-\xi_i \qquad i = m+1,\ldots,m+n$$

Domain Adaptation Methods



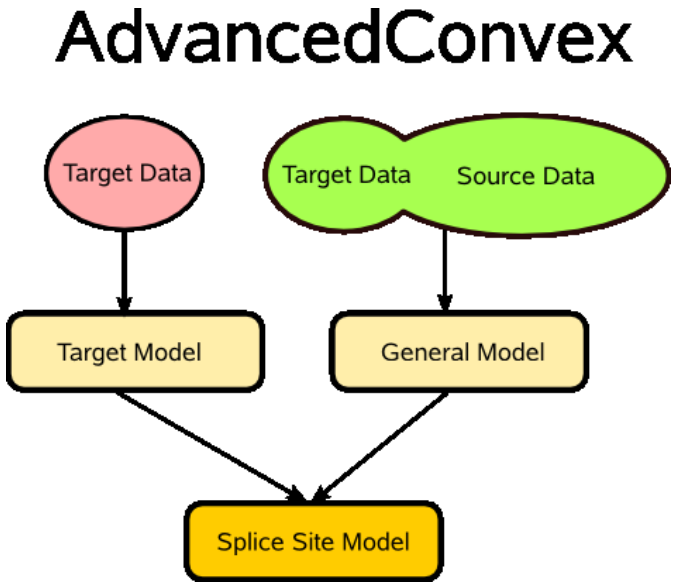
Idea: (e.g., Widmer, 2008; Schweikert et al., 2008)

- Combine trained models
- Efficient hyper-parameter optimization

Approach:

$$F(x) = \alpha f_S(x) + (1 - \alpha) f_T(x)$$

Domain Adaptation Methods



Idea: (e.g., Widmer, 2008; Schweikert et al., 2008)

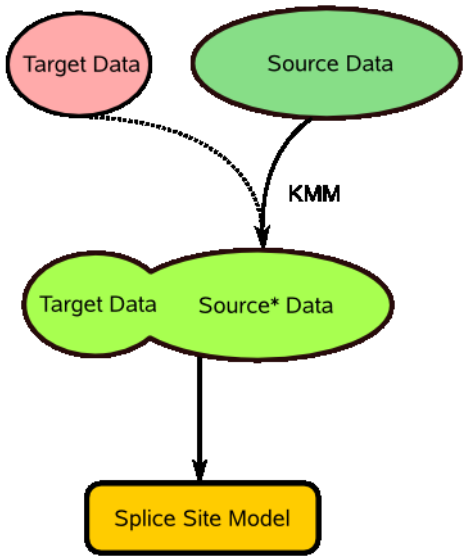
- Takes interactions between source and target examples into account
- Captures 'General' and 'Target-specific' component

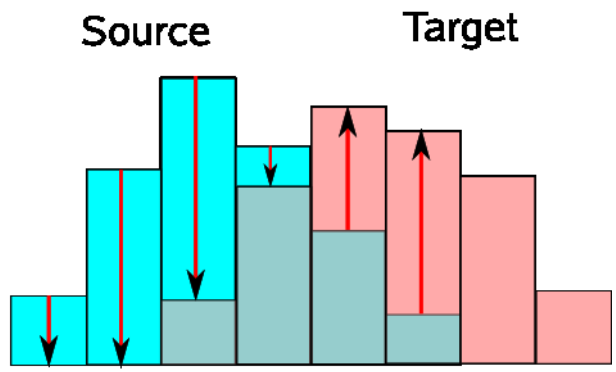
Approach:

$$F(x) = \alpha f_C(x) + (1 - \alpha) f_T(x)$$

Domain Adaption Methods

Kernel Mean Matching



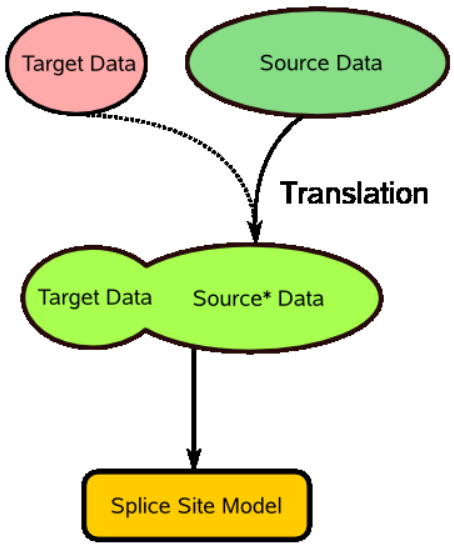


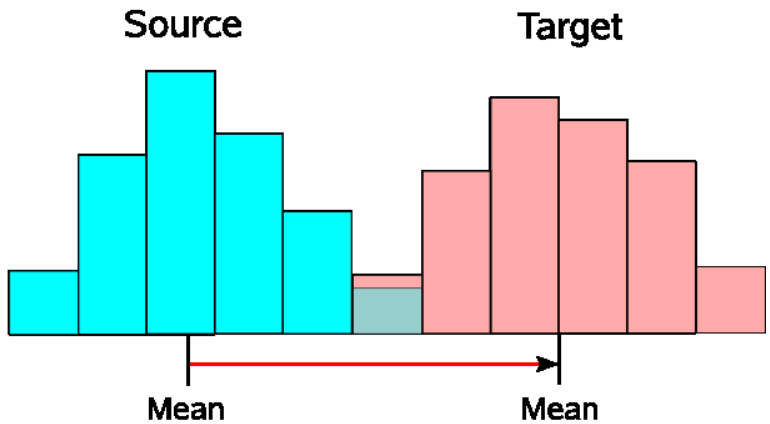
Idea: (Huang et al., 2007)

- Match mean of source and target by re-weighting examples
- Higher-order moments defined by mean when using a universal kernel

Domain Adaption Methods

Shift (KMM)



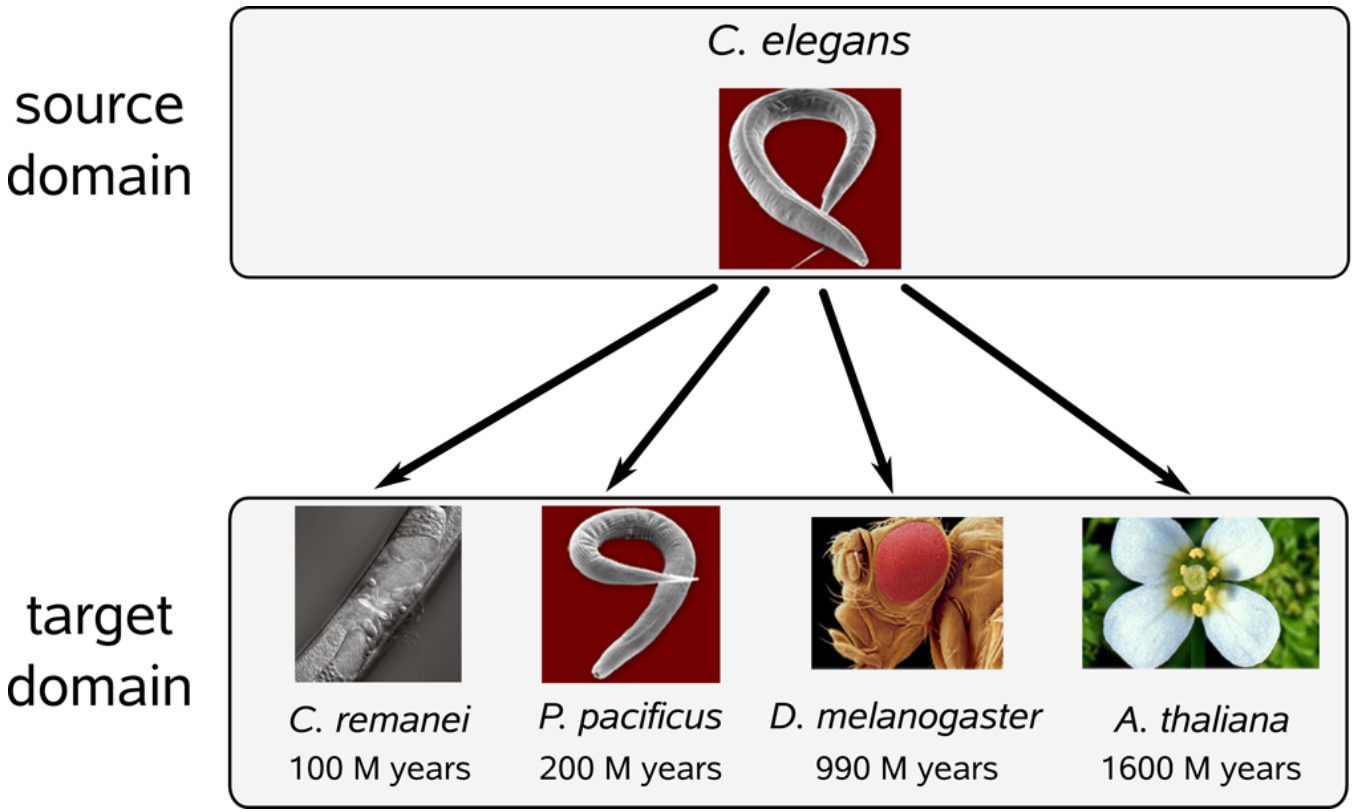


Idea:

(e.g., Widmer, 2008; Schweikert et al., 2008)

- Based on same assumption
- Mean matching via translation rather than re-weighting (for each class separately)

Large-Scale Empirical Comparison



- Varying distances
- Different data set sizes

[MPI Developmental Biology and UCSC Genome Browser]

Experimental Setup

- Source dataset size: always 100k examples
- Target dataset sizes: {2500, 6500, 16000, 64000, 100000}
- Simple kernel (WDK of degree 1 $\Rightarrow$ under-fitting)
- Extensive model selection for each method
- Area under Precision/Recall curve for evaluation

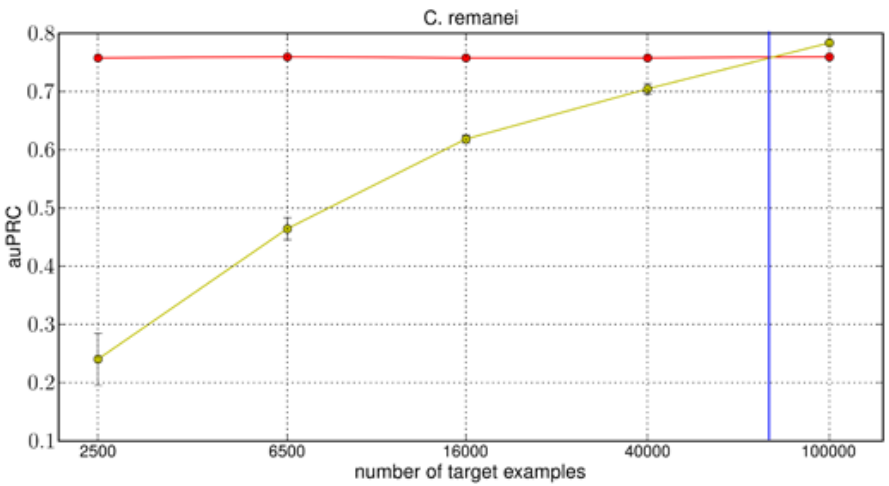
Experimental Setup

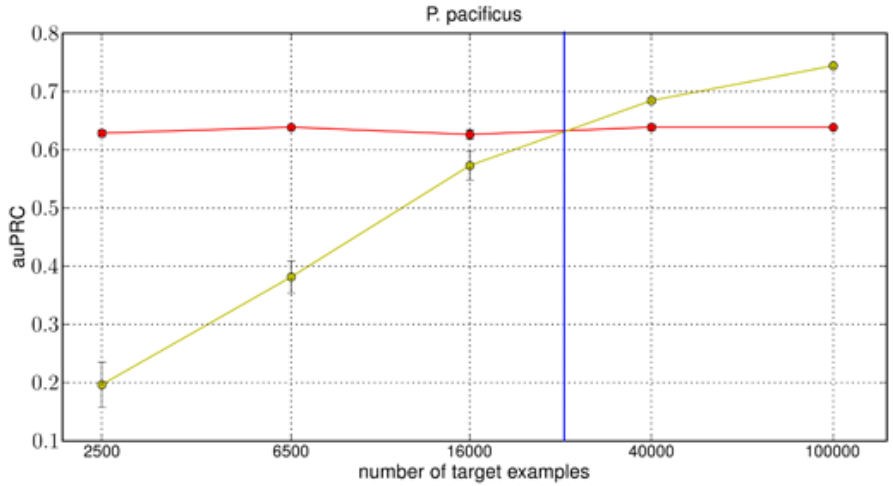
- Source dataset size: always 100k examples
- Target dataset sizes: $\{2500, 6500, 16000, 64000, 100000\}$
- Simple kernel (WDK of degree 1 $\Rightarrow$ under-fitting)
- Extensive model selection for each method
- Area under Precision/Recall curve for evaluation

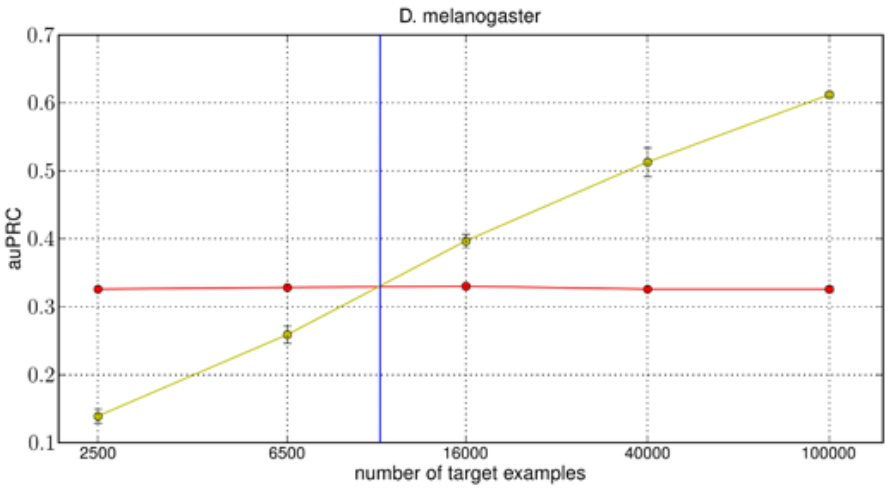
Experimental Setup

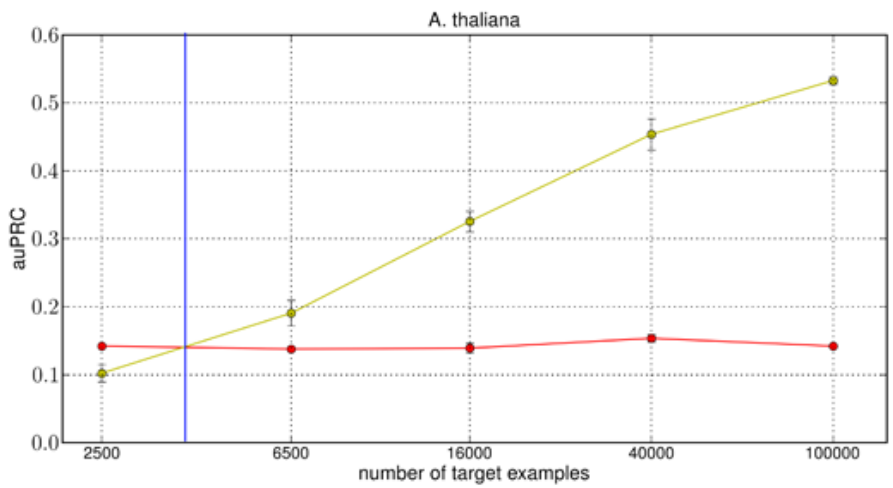
- Source dataset size: always 100k examples
- Target dataset sizes: $\{2500, 6500, 16000, 64000, 100000\}$
- Simple kernel (WDK of degree 1 $\Rightarrow$ under-fitting)
- Extensive model selection for each method
- Area under Precision/Recall curve for evaluation

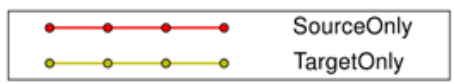
Results - Baseline Methods



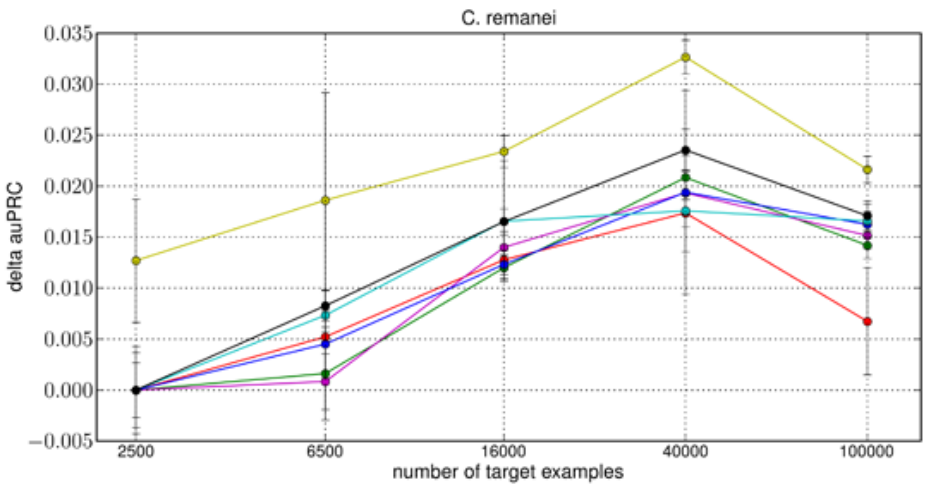


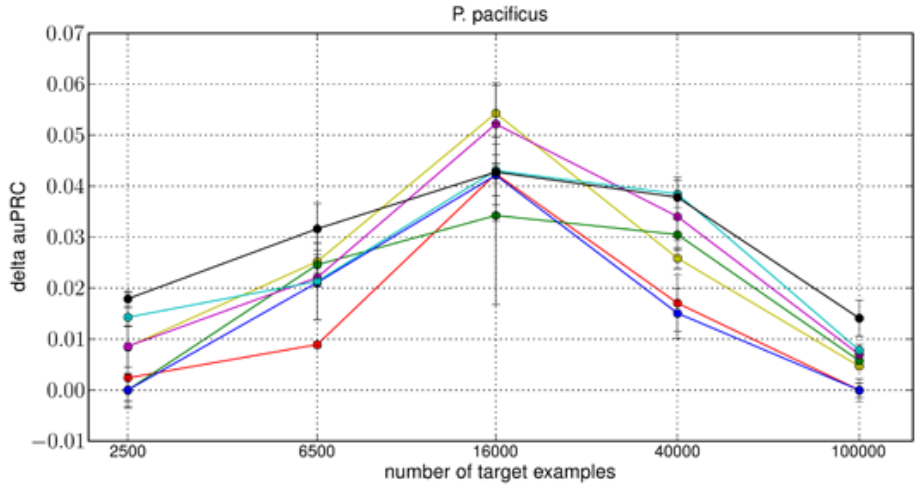


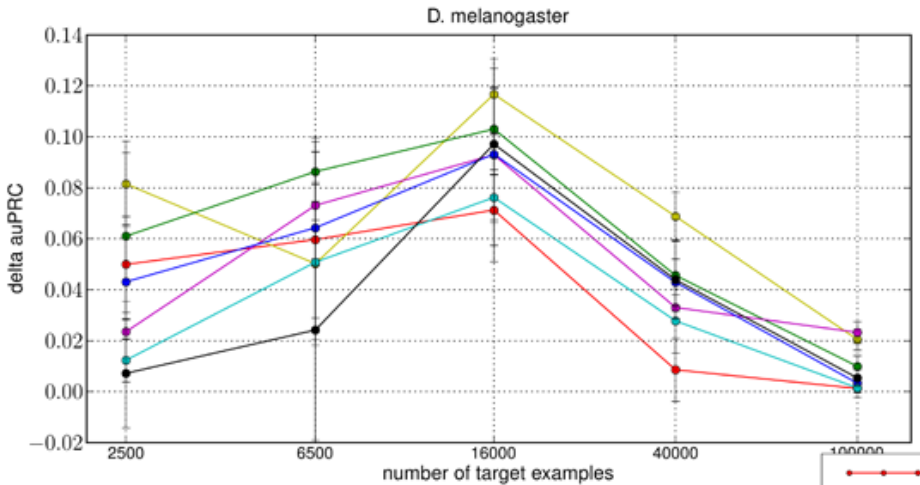


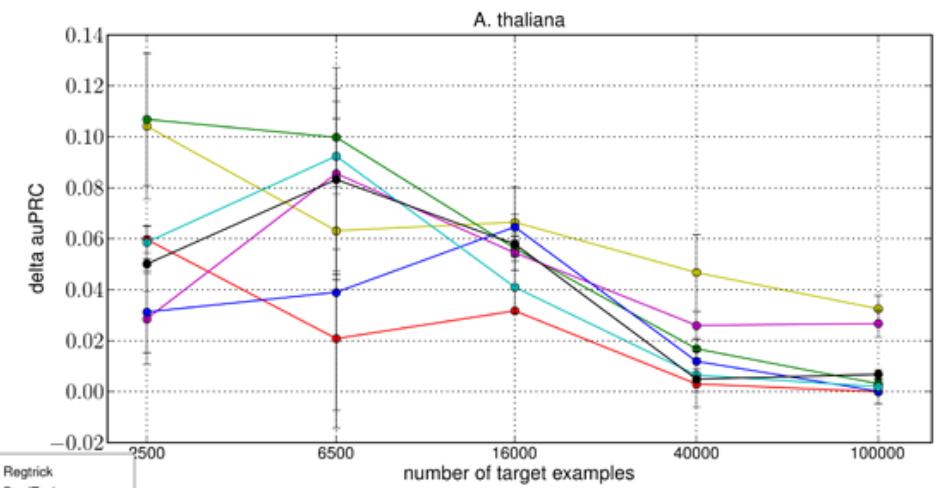


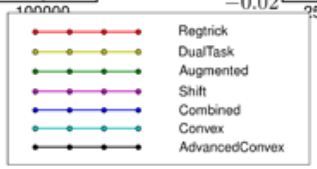
Results - Improvement over Baseline Methods





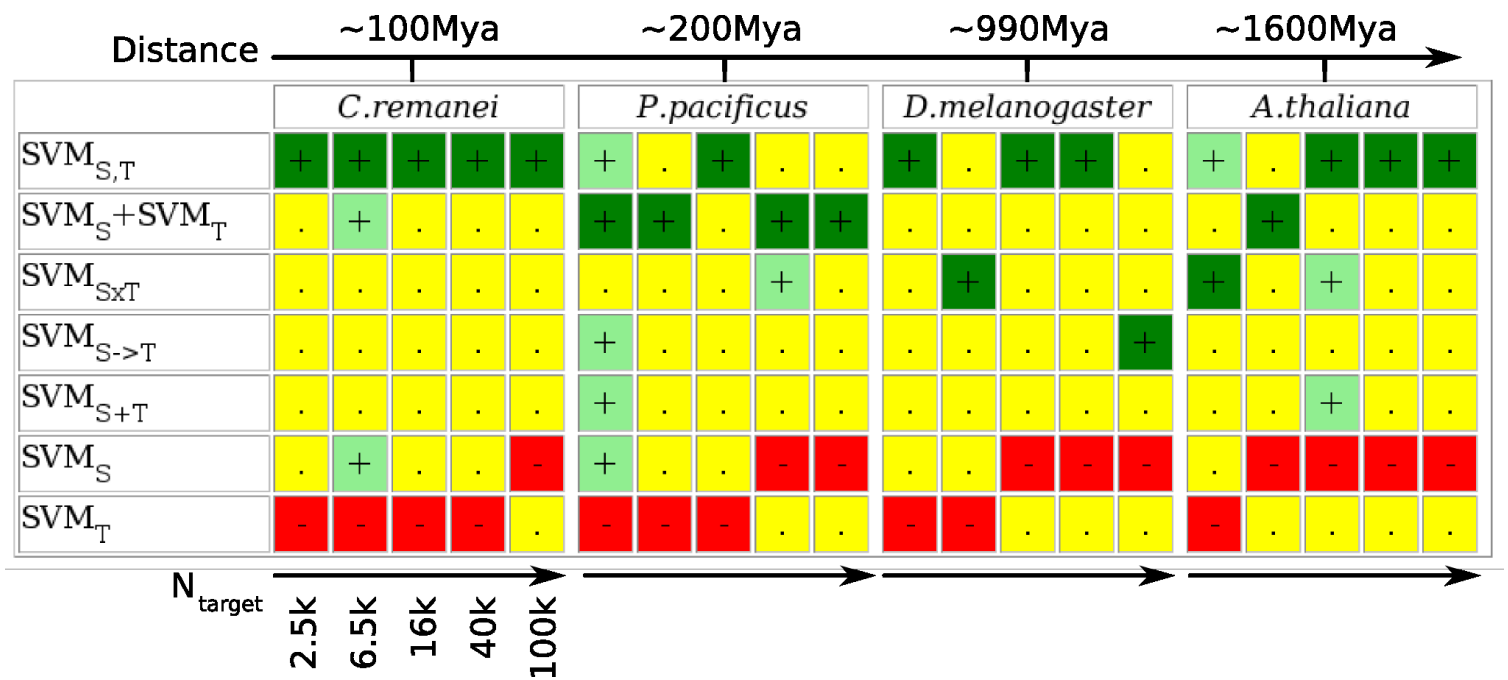




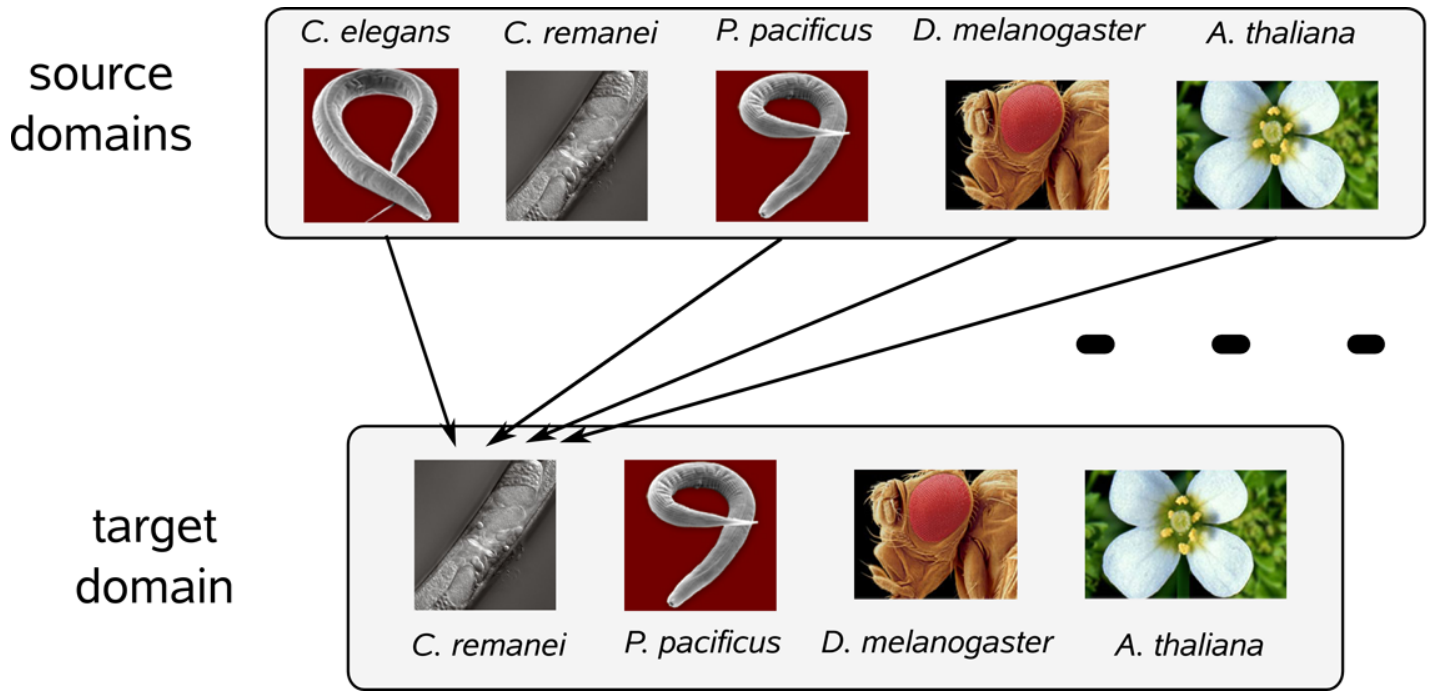


Results - Summary

- Considerable improvements possible
- Sophisticated domain adaptation methods needed on distantly related organisms
- Best overall performance has *DualTask*
- Most cost effective *Convex/AdvancedConvex*



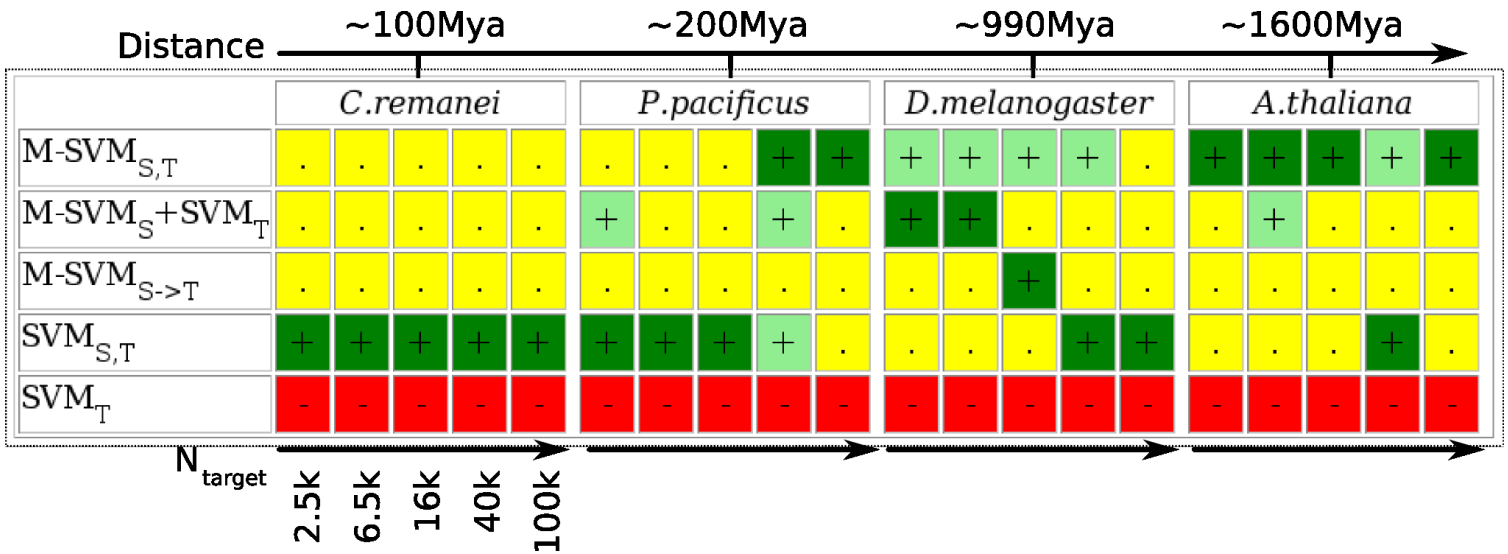
Multiple Source Domains



- Combine information from several sources (treated equally)
- Methods: Multi-task learning, Convex combination, Shifting

Results - Multiple Source Domains

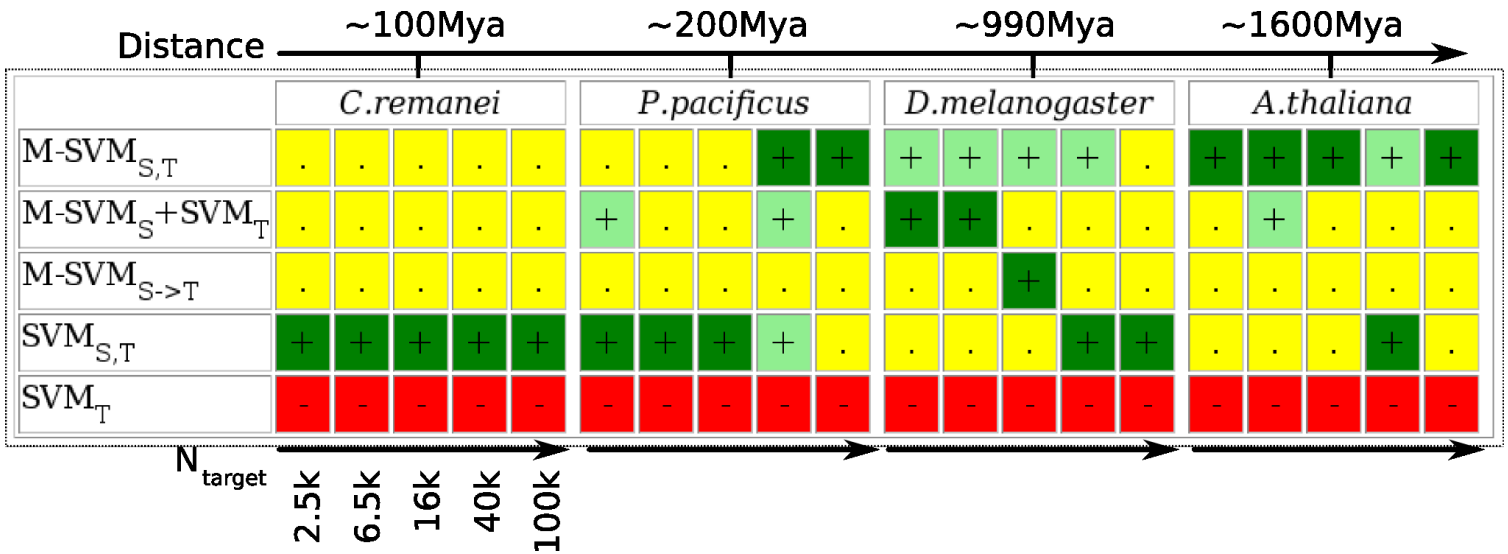
- Single source model best for very closely related task
- Multiple source model better for distantly related tasks
- Multi-task algorithm strongest



How to use information on relatedness in learning?

Results - Multiple Source Domains

- Single source model best for very closely related task
- Multiple source model better for distantly related tasks
- Multi-task algorithm strongest



How to use information on relatedness in learning?

Multitask learning

Hierarchical structure arises naturally from the Tree of Life



- Taxonomy used to define relationship between tasks
- Closer tasks benefit more from each other

Problem Definition

Multitask Learning / Domain Adaptation

- Consider M tasks T_i, where $i = 1, \ldots, M$
- We are given data $D_i = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{N_i}, y_{N_i})\}$ for each task
- We want to train M predictors $f_1, \ldots, f_M$, each taking into account all available information
- For that we would like to utilize a given taxonomy $\mathcal{T}$, that relates the tasks at hand
- Each task T_i corresponds to a leaf in taxonomy $\mathcal{T}$

$\Rightarrow$ We need efficient algorithms to exploit $\mathcal{T}$ for transfer learning

(say, 100 tasks with each 100k examples)

Two ways of leveraging a given taxonomy $\mathcal{T}$



Taxonomy

Transformation

$$\Gamma = \begin{pmatrix} \gamma_{1,1} & \cdots & \gamma_{1,M} \\ \vdots & & \vdots \\ \gamma_{M,1} & \cdots & \gamma_{M,M} \end{pmatrix}$$

Task Similarity Matrix

Top-Down

Pairwise &

Multitask Kernel

(Widmer et al., 2010a)

Hierarchical Top-Down Approach

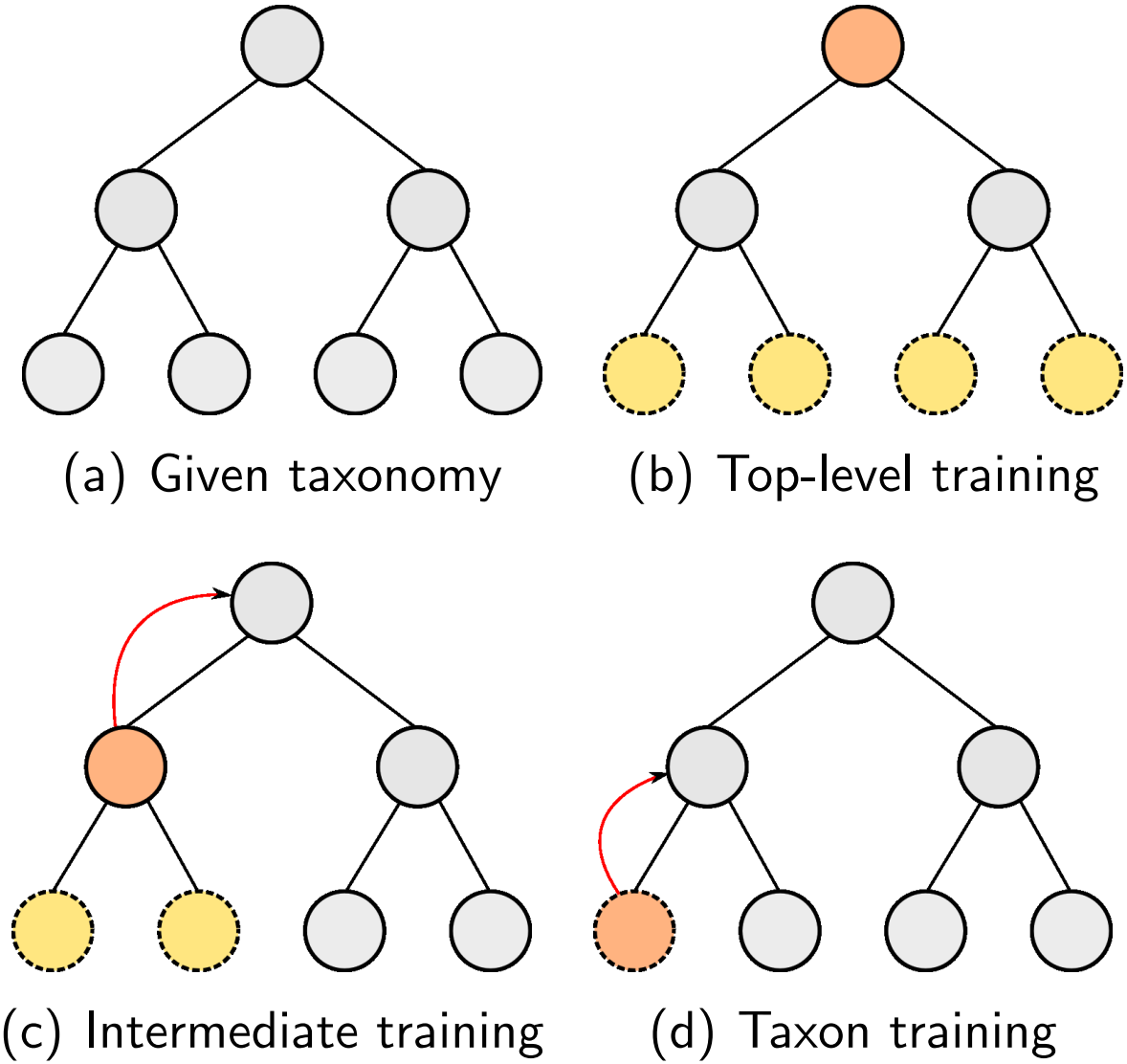
Idea: Exploit taxonomy G algorithmically

- Use taxonomy $\mathcal{T}$ in a top-down procedure
- Initialization: w_0 trained on union of all task datasets
- Top-Down for each node t:
  - Train on $D_i = \bigcup_{j \preccurlyeq i} D_j$
  - Regularize $\mathbf{w}_i$ against parent predictor $\mathbf{w}_{parent}$:

$$\min_{\mathbf{w}_i, b} \quad \frac{1}{2}\|\mathbf{w}_i - \mathbf{w}_{parent}\|^2 + C \sum_{(\mathbf{x},y)\in D_i} \ell\left(\langle \Phi(\mathbf{x}), \mathbf{w}_i \rangle + b, y\right),$$

- Use leaf predictors for classification

Hierarchical Top-Down Approach: Illustration



(a) Given taxonomy (b) Top-level training

(c) Intermediate training (d) Taxon training

Figure: Illustration of the hierarchical Top-down multitask training

Multitask Kernel Approach

Use standard SVM formulation

$$\max_{\alpha} -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \hat{k}(\mathbf{x}_i,\mathbf{x}_j) + \sum_{i=1}^{n}\alpha_i$$

$$\text{s.t. } 0 \le \alpha_i \le C \quad i = 1,\ldots,n$$

$$\alpha^T \mathbf{y} = 0,$$

with customized kernel:

$$\hat{k}(x_i,x_j) = \gamma_{task(x_i),task(x_j)} k(x_i,x_j)$$

- Easily implemented by altering existing kernel functions (WDK)
- Reuse existing kernel algorithms (SVM)
- **How to define $\Gamma = \{\gamma_{i,j}\}$?** (Needs to be p.s.d.)

(e.g., Daumé, 2007; Widmer et al., 2010a)

Multitask Kernel Approach

Use standard SVM formulation

$$\max_{\alpha} -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \hat{k}(\mathbf{x}_i,\mathbf{x}_j) + \sum_{i=1}^{n}\alpha_i$$

$$\text{s.t. } 0 \le \alpha_i \le C \quad i = 1,\ldots,n$$

$$\alpha^T \mathbf{y} = 0,$$

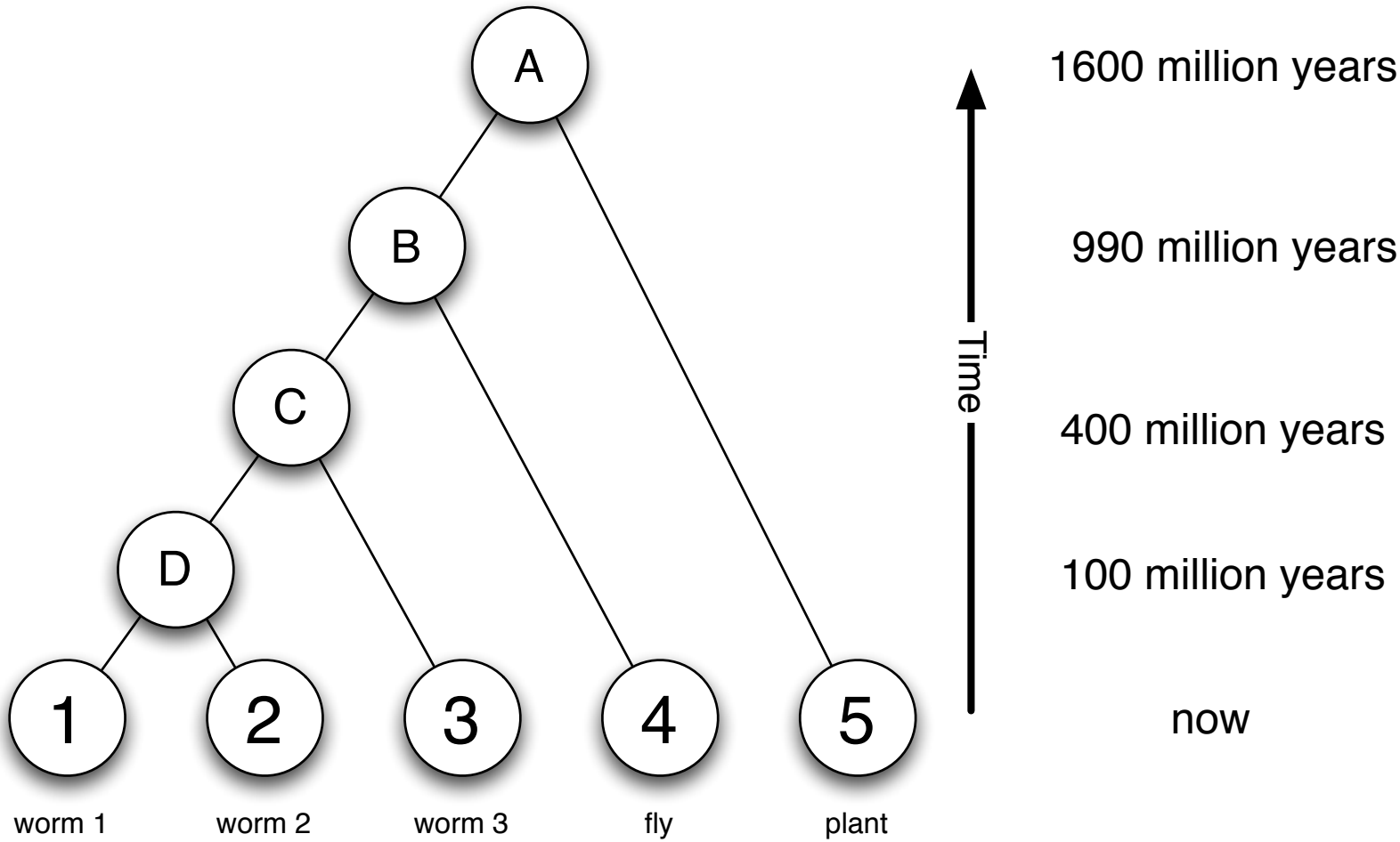
with customized kernel:

$$\hat{k}(x_i,x_j) = \gamma_{task(x_i),task(x_j)}\, k(x_i,x_j)$$

- Easily implemented by altering existing kernel functions (WDK)
- Reuse existing kernel algorithms (SVM)
- **How to define $\Gamma = \{\gamma_{i,j}\}$?** (Needs to be p.s.d.)

(e.g., Daumé, 2007; Widmer et al., 2010a)

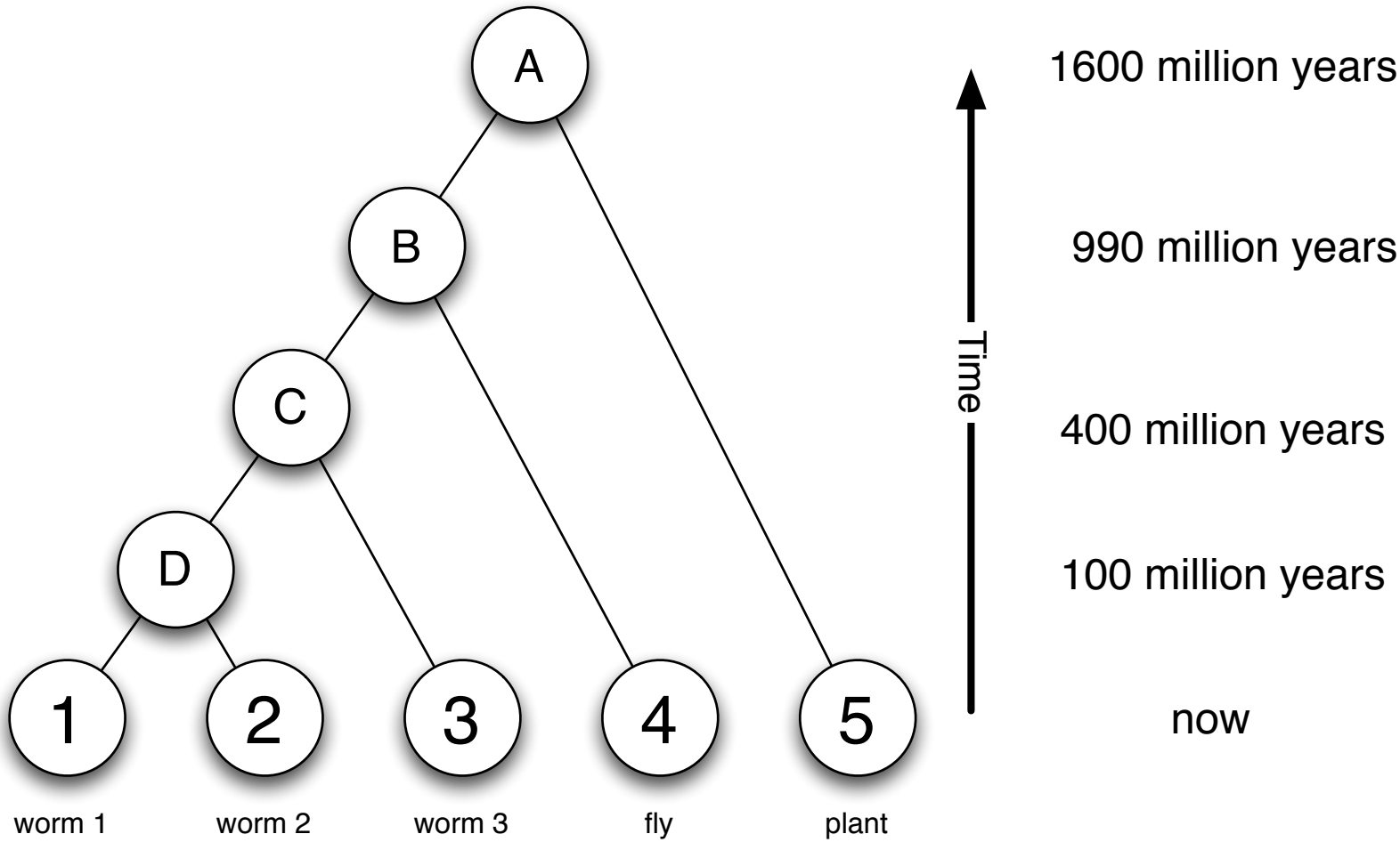
From Taxonomy to Γ



Idea: $\gamma_{i,j}$ should be inversely related to time to last common ancestor

Strategies: $1/\text{years}$, *Hop-distance*, ... **Is there a better way?**

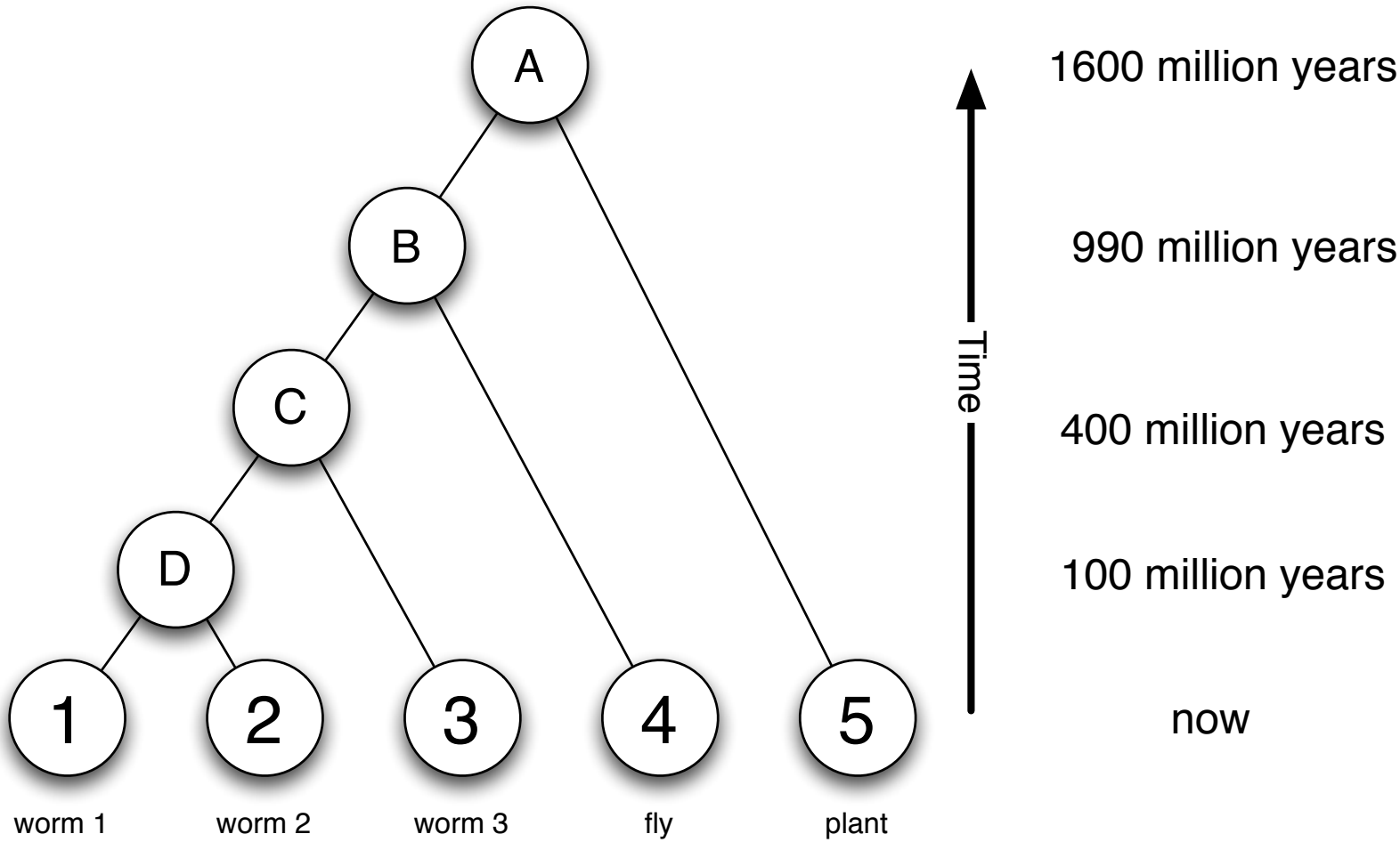
From Taxonomy to Γ



Idea: $\gamma_{i,j}$ should be inversely related to time to last common ancestor

Strategies: 1/years, *Hop-distance*, ... Is there a better way?

From Taxonomy to Γ



Idea: $\gamma_{i,j}$ should be inversely related to time to last common ancestor

Strategies: 1/years, *Hop-distance*, ... **Is there a better way?**

Pairwise Approach

Idea: if we have the intuition that the predictors are similar $\longrightarrow$ let's penalize their distance.

$$\mathcal{L}_P = \frac{1}{2}\sum_{t=1}^{M}\sum_{s\neq t}^{M}\gamma_{ts}\|\mathbf{w}_t - \mathbf{w}_s\|^2 + \sum_i \xi_i$$

$$= \sum_{t=1}^{M}\left(\frac{1}{2}\sum_{s\neq t}^{M}\gamma_{ts}\|\mathbf{w}_t - \mathbf{w}_s\|^2 + \sum_{i\in I_t} L(\mathbf{w}_t^T\mathbf{x}_i, y_i)\right)$$

- Advantage: $\mathbf{\Gamma} = \{\gamma_{ts}\}$ describes how the domains are related
- Disadvantage: the size of the problem is given by the composition of tasks
- We would like to decompose it into M single-task problems

Pairwise Approach: Decomposition

Solution: Problem can be decomposed into M sub-problems

$$\min_{\mathbf{w}_t} \quad \tfrac{1}{2}\lambda_t \|\mathbf{w}_t - \mathbf{r}_t\|^2 + C \sum_{(\mathbf{x},y)\in D_t} \ell\left(\langle \mathbf{x}, \mathbf{w}_t \rangle, y\right), \quad \forall t$$

where

$$\mathbf{r}_t = \sum_{s\neq t} \beta_{ts}\mathbf{w}_s, \quad \lambda_t = 2\sum_{s\neq t} \gamma_{ts}, \quad \beta_{ts} = \gamma_{ts}/\lambda_t.$$

The iterative algorithm consists of the steps:

1. Fix $\{\mathbf{r}_t\}$, compute $\{\mathbf{w}_t\}$ that solves the minimization of $\mathcal{L}_S$.
2. Fix $\{\mathbf{w}_t\}$, compute $\{\mathbf{r}_t\}$ by $\mathbf{r}_t = \sum_{s\neq t} \beta_{ts}\hat{\mathbf{w}}_s$.

$\Rightarrow$ We now iteratively solve a number of SVM-like QPs

(Can be shown to converge to the global optimum)

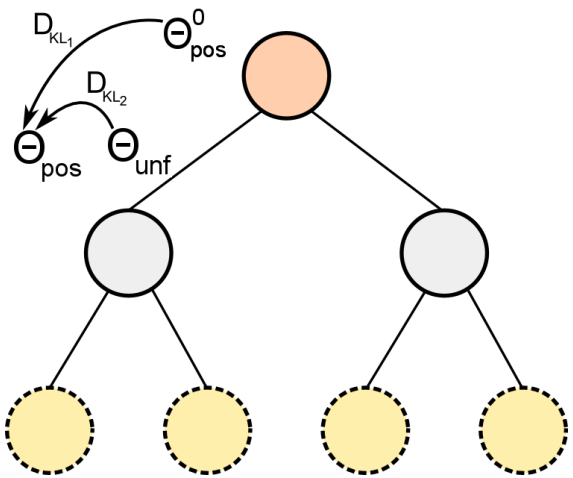
(Widmer et al., 2010b)

Method Comparison

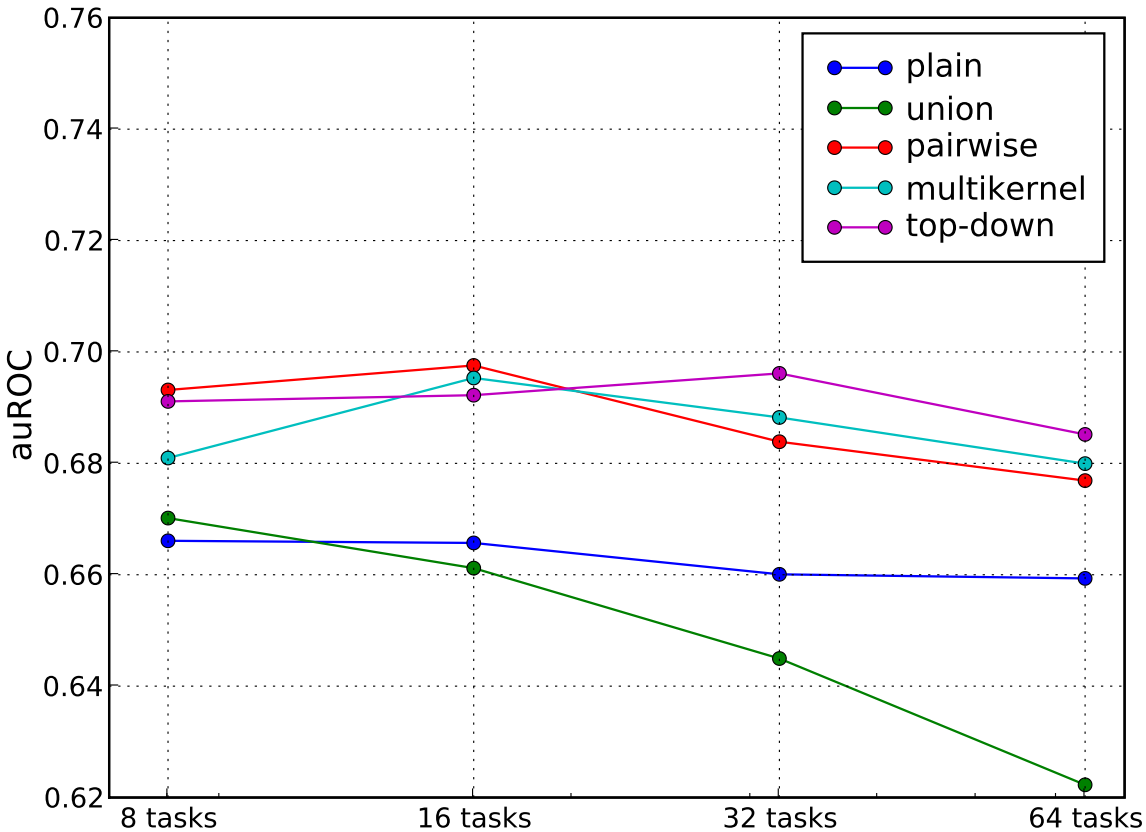
- Top-down Multitask Learning
  - Naturally exploits hierarchy
  - Does not need *task similarity matrix*
  - Heuristic
  - Needs training on unions of dataset
- Multitask Kernel Learning
  - Extension of existing kernel methods
  - Same run-time as regular SVM on union of all dataset
  - Requires appropriate *similarity matrix*
- Pairwise Multitask Learning
  - Optimization problem with well-defined global optimum
  - Practical issues due to decomposition
  - Requires appropriate *similarity matrix*

Toy Data Generation

- Sample sequences for M tasks
- Evolve generative model (PSSM) according to given structure
- We want to control
  - How hard the problems are
  - How different the problems are
  - $\rightarrow$ use KL-Divergence $D_{KL}(P||Q) = \sum_i P(i) log \frac{P(i)}{Q(i)}$



Results: Toy Data



- MTL methods outperform baselines
- High mutation rate leads to decreased performance in deep hierarchies

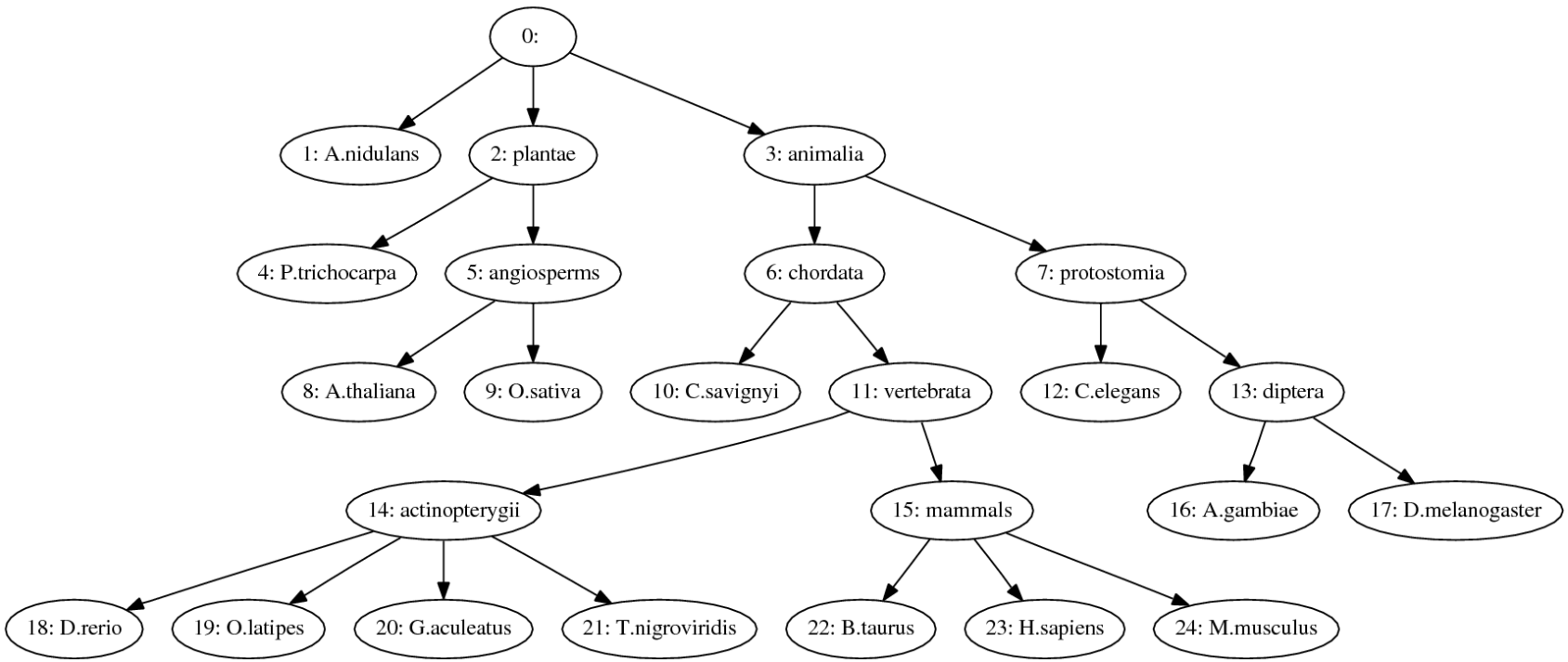
Application to Splicing Data

- Formulation as binary classification problem

≈150 nucleotides window around dimer

CT...GTCGTA...GAAGCTAGGAGCGC...ACGCGT...GA

- Utilize 15 organisms related by taxonomy



- Restricted to at most 10.000 examples per organism

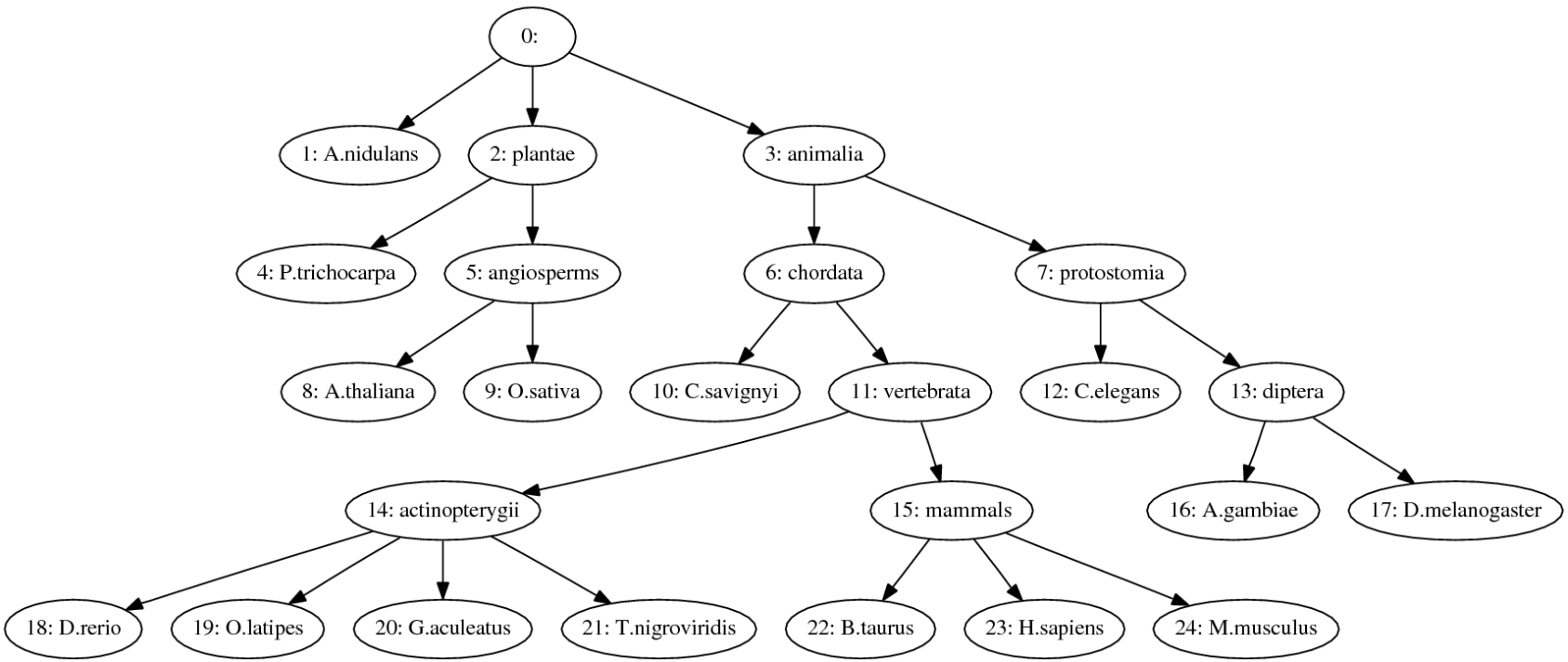
Application to Splicing Data

- Formulation as binary classification problem

≈150 nucleotides window around dimer

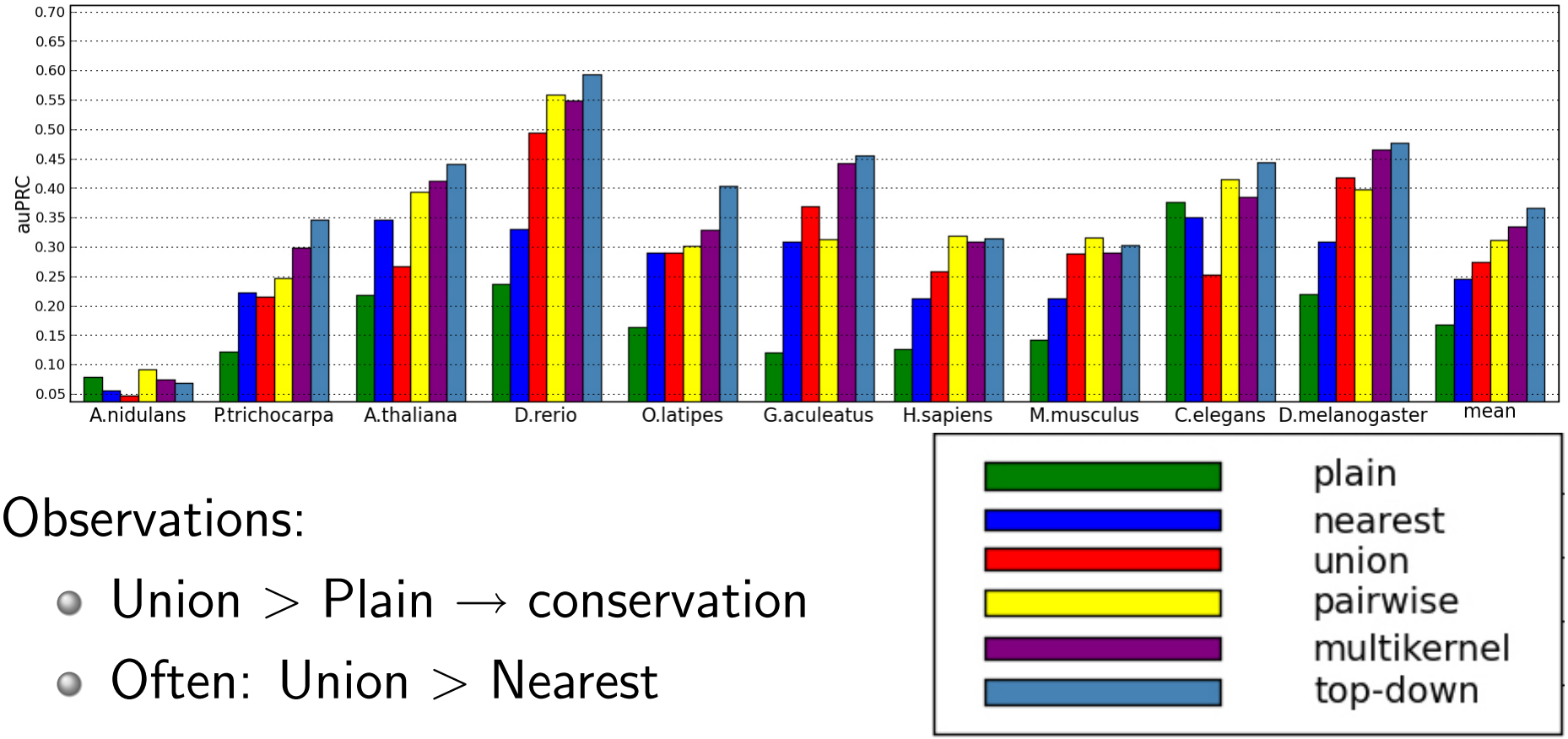
CT...GTCGTA...GAAGCTAGGAGCGC...ACGCGT...GA

- Utilize 15 organisms related by taxonomy



- Restricted to at most 10.000 examples per organism

Results: Splicing Data



Observations:

- Union $>$ Plain $\rightarrow$ conservation
- Often: Union $>$ Nearest
- MTL methods outperform baselines
- Best performer is Top-Down
- Potential increase depends on the particular organism

Discussion

- Hierarchy helps transferring information into right places
- Top-down approach the information transfer most accurate
- Performance depends strongly on task similarity matrix
  - Choice very difficult, not easily done with cross-validation
  - Can we learn, e.g.

    $$\gamma_{i,j} = f(\text{“years of evolution between i and j”})?$$

    Multiple-Kernel Multi-Task approach?
- Idea for top-down approach: local cross-validation

Discussion

- Hierarchy helps transferring information into right places
- Top-down approach the information transfer most accurate
- Performance depends strongly on task similarity matrix
  - Choice very difficult, not easily done with cross-validation
  - Can we learn, e.g.

    $$\gamma_{i,j} = f(\text{“years of evolution between i and j”})?$$

    Multiple-Kernel Multi-Task approach?
- Idea for top-down approach: local cross-validation

Discussion

- Hierarchy helps transferring information into right places
- Top-down approach the information transfer most accurate
- Performance depends strongly on task similarity matrix
  - Choice very difficult, not easily done with cross-validation
  - Can we learn, e.g.

    $$\gamma_{i,j} = f(\text{“years of evolution between i and j”})?$$

    Multiple-Kernel Multi-Task approach?
- Idea for top-down approach: local cross-validation

Discussion

- Hierarchy helps transferring information into right places
- Top-down approach the information transfer most accurate
- Performance depends strongly on task similarity matrix
  - Choice very difficult, not easily done with cross-validation
  - Can we learn, e.g.

    $$\gamma_{i,j} = f(\text{“years of evolution between i and j”})?$$

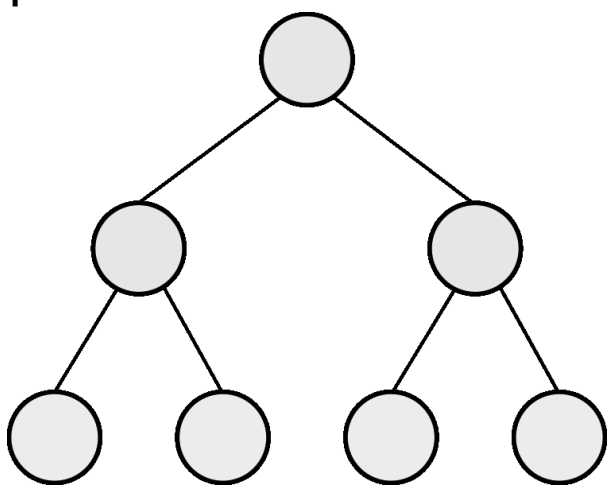
    Multiple-Kernel Multi-Task approach?
- Idea for top-down approach: local cross-validation

Discussion

- Hierarchy helps transferring information into right places
- Top-down approach the information transfer most accurate
- Performance depends strongly on task similarity matrix
  - Choice very difficult, not easily done with cross-validation
  - Can we learn, e.g.

    $$\gamma_{i,j} = f(\text{“years of evolution between i and j”})?$$

    Multiple-Kernel Multi-Task approach?
- Idea for top-down approach: local cross-validation



Summary

Domain adaptation:

- Considerable improvements possible
- Sophisticated methods slight edge for distantly related tasks

Multitask learning:

- Novel methods provide scalable way of integrating information
  (Implementations available for SVMLight and LibSVM)
- Design of task similarity matrix difficult & critical
- Poster presented by Christian Widmer

Outlook:

- Extension to Structured Output Learning
- Estimation of "optimal" task similarity matrix

Material available at:
`www.fml.mpg.de/raetsch/suppl/transfer-learning`

Summary

Domain adaptation:

- Considerable improvements possible
- Sophisticated methods slight edge for distantly related tasks

Multitask learning:

- Novel methods provide scalable way of integrating information
  (Implementations available for SVMLight and LibSVM)
- Design of task similarity matrix difficult & critical
- Poster presented by Christian Widmer

Outlook:

- Extension to Structured Output Learning
- Estimation of "optimal" task similarity matrix

Material available at:
`www.fml.mpg.de/raetsch/suppl/transfer-learning`

Summary

Domain adaptation:

- Considerable improvements possible
- Sophisticated methods slight edge for distantly related tasks

Multitask learning:

- Novel methods provide scalable way of integrating information
  (Implementations available for SVMLight and LibSVM)
- Design of task similarity matrix difficult & critical
- Poster presented by Christian Widmer

Outlook:

- Extension to Structured Output Learning
- Estimation of "optimal" task similarity matrix

Material available at:
www.fml.mpg.de/raetsch/suppl/transfer-learning

Summary

Domain adaptation:

- Considerable improvements possible
- Sophisticated methods slight edge for distantly related tasks

Multitask learning:

- Novel methods provide scalable way of integrating information
  (Implementations available for SVMLight and LibSVM)
- Design of task similarity matrix difficult & critical
- Poster presented by Christian Widmer

Outlook:

- Extension to Structured Output Learning
- Estimation of "optimal" task similarity matrix

Material available at:
`www.fml.mpg.de/raetsch/suppl/transfer-learning`

Acknowledgements



Christian Widmer

(FML, Tübingen)



Gabriele Schweikert

(MPI & FML, Tübingen)



Jose Leiva

(University of Madrid)

Also involved

- Bernhard Schölkopf

  (MPI, Tübingen)
- Yasemin Altun

  (MPI, Tübingen)
- Sören Sonnenburg

  (TU Berlin)

Funding by DFG &
Max Planck Society.

Thank you for your attention!

Acknowledgements



Christian Widmer

(FML, Tübingen)



Gabriele Schweikert

(MPI & FML, Tübingen)



Jose Leiva

(University of Madrid)

Also involved

- Bernhard Schölkopf

  (MPI, Tübingen)
- Yasemin Altun

  (MPI, Tübingen)
- Sören Sonnenburg

  (TU Berlin)

Funding by DFG &
Max Planck Society.

Thank you for your attention!

References I

John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jenn Wortman. Learning bounds for domain adaptation. In *Advances in Neural Information Processing Systems 21*, Cambridge, MA, 2008. MIT Press.

Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

C. Chelba and A. Acero. Adaptation of maximum entropy capitalizer: Little data can help a lot. *Computer Speech & Language*, 20(4): 382–399, 2006.

H. Daumé. Frustratingly easy domain adaptation. In *ACL*. The Association for Computer Linguistics, 2007. URL `http://aclweb.org/anthology-new/P/P07/P07-1033.pdf`.

Hal Daume III. Frustratingly easy domain adaptation. In *Conference of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, 2007. URL `http://pub.hal3.name/#daume07easyadapt`.

References II

K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf. Kernel measures of conditional dependence. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 489–496. MIT Press, Cambridge, MA, 2008.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel method for the two-sample-problem. In *Advances in Neural Information Processing Systems 19*, Cambridge, MA, 2007. MIT Press.

A. Gretton, A. Smola, J. Huang, M. Schmittfull, and B. Schölkopf. Covariate shift by kernel mean matching. In J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, editors, *Dataset Shift in Machine Learning*. MIT Press, Cambridge, MA, 2008.

J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems 19*, Cambridge, MA, 2007. MIT Press.

References III

A Pires-DaSilva and RJ Sommer. Conservation of the global sex determination gene tra-1 in distantly related nematodes. *GENES & DEVELOPMENT*, 18(10):1198–1208, MAY 15 2004. ISSN 0890-9369. doi: $\{10.1101/\mathrm{gad}.293504\}$.

G. Schweikert, C. Widmer, B. Schölkopf, and G. Rätsch. An empirical analysis of domain adaptation algorithms. In *Advances in Neural Information Processing System, NIPS*, volume 22, Vancouver, B.C., 2008.

Gabriele Schweikert, Alexander Zien, Georg Zeller, Jonas Behr, Christoph Dieterich, Cheng Soon Ong, Petra Philips, Fabio De Bona, Lisa Hartmann, Anja Bohlen, Nina Krüger, Sören Sonnenburg, and Gunnar Rätsch. mGene: accurate SVM-based gene finding with an application to nematode genomes. *Genome Research*, September 2009.

L. Song, X. Zhang, A. Smola, A. Gretton, and B. Schölkopf. Tailoring density estimation via reproducing kernel moment matching. In *Proc. Intl. Conf. Machine Learning*, 2008.

References IV

S. Sonnenburg, G. Schweikert, P. Philips, J. Behr, and G. Rätsch. Accurate splice site prediction using support vector machines. *BMC Bioinformatics*, 8(Suppl. 10):S7, 2007. URL `http://www.biomedcentral.com/1471-2105/8/S10/S7`.

LD Stein, ZR Bao, D Blasiar, T Blumenthal, MR Brent, NS Chen, A Chinwalla, L Clarke, C Clee, A Coghlan, A Coulson, P D'Eustachio, DHA Fitch, LA Fulton, RE Fulton, S Griffiths-Jones, TW Harris, LW Hillier, R Kamath, PE Kuwabara, ER Mardis, MA Marra, TL Miner, P Minx, JC Mullikin, RW Plumb, J Rogers, JE Schein, M Sohrmann, J Spieth, JE Stajich, CC Wei, D Willey, RK Wilson, R Durbin, and RH Waterston. The genome sequence of Caenorhabditis briggsae: A platform for comparative genomics. *PLOS BIOLOGY*, 1(2):166+, NOV 2003. ISSN 1544-9173. doi: {10.1371/journal.pbio.0000045}.

References V

A Ureta-Vidal, L Ettwiller, and E Birney. Comparative genomics: Genome-wide analysis in metazoan eukaryotes. *NATURE REVIEWS GENETICS*, 4(4):251–262, APR 2003. ISSN 1471-0056. doi: {10.1038/nrg1043}.

C. Widmer. Domain adaptation in sequence analysis. Diplom thesis, University of Tübingen, August 2008.

C. Widmer, J.M. Leiva, Y. Altun, and G. Rätsch. Leveraging sequence classification by taxonomy-based multitask learning. In *Proc. RECOMB'10*, 2010a. accepted.

C. Widmer, J.M. Leiva, Y. Altun, and G. Rätsch. Scalable multitask learning. forthcoming, 2010b.