Abstraction Augmented Markov Models

Cornelia Caragea[1] Adrian Silvescu[2] Doina Caragea[3]
Vasant Honavar[1]

[1]Computer Science Department, Iowa State University, IA
[2]Yahoo! Labs, Sunnyvale, CA
[3]Computer and Information Sciences, Kansas State University, KS

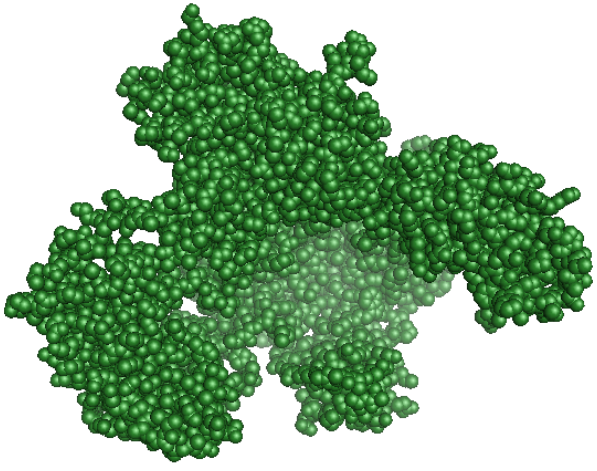
Machine Learning in Computational Biology

Real World Applications

Bioinformatics:

· · · PVKLKPGMDGPKVKQWPLTEEKIKALVEIC· · ·

protein localization?



protein function?

Sequence Data:

Outline

- Learning Probabilistic Models from Sequence Data
  - Markov Models
- Abstraction Augmented Markov Models (AAMMs)
  - Learning AAMMs
  - Using AAMMs for Classification
- Experiments and Results
- Summary and Future Directions

Outline

- Learning Probabilistic Models from Sequence Data
  - Markov Models
- Abstraction Augmented Markov Models (AAMMs)
  - Learning AAMMs
  - Using AAMMs for Classification
- Experiments and Results
- Summary and Future Directions

Learning Probabilistic Models on Sequence Data

Given a training set $\mathcal{D} = (\mathbf{x_i}, y_i)_{i=1,\cdots,I}$, $\mathbf{x_i} \in \mathcal{X}^*$, $y_i \in \mathcal{Y}$ of sequences and their associated labels, the task is to learn a classifier that correctly assigns a label $y \in \mathcal{Y}$ to a new sequence $\mathbf{x} = (x_1 \cdots x_n)$.

Classification:

- Computation of conditional probability $p(y|\mathbf{x})$ for new sequence $\mathbf{x}$

Bayes Rule:

- $p(y|\mathbf{x}) \propto p(\mathbf{x}|y)p(y)$

K^{th}-Order Markov Models for $p(\mathbf{x}|y)$

- Capture dependencies between neighboring elements

$$x_1 \cdots \overbrace{x_{i-k} \cdots x_{i-1}} \; x_i \; x_{i+1} \cdots x_n, y$$

- Satisfy the Markov property:

$$X_i \perp\!\!\!\perp \{X_0, \cdots, X_{i-k-1}\} \,|\, \{X_{i-1}, \cdots, X_{i-k}\}$$

- Hence,

$$p(X_0, \cdots, X_{n-1}) = p(X_0, \cdots, X_{k-1}) \prod_{i=k}^{n-1} p(X_i | X_{i-k}, \cdots, X_{i-1})$$

The complexity of k^{th}-order Markov models is exponential in k

Outline

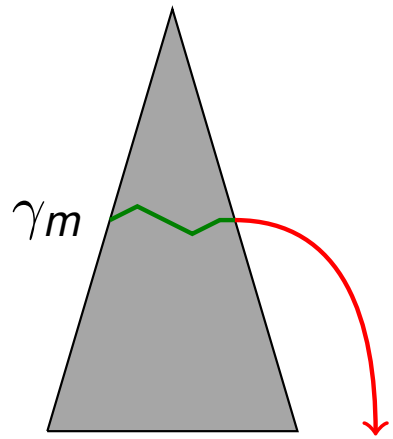
- Learning Probabilistic Models from Sequence Data
  - Markov Models
- Abstraction Augmented Markov Models (AAMMs)
  - Learning AAMMs
  - Using AAMMs for Classification
- Experiments and Results
- Summary and Future Directions

Abstraction Augmented Markov Models

MM(k):

$x_1 \cdots \overbrace{x_{i-k} \cdots x_{i-1}} \; x_i \; x_{i+1} \cdots x_n, y$

AAMM(k):

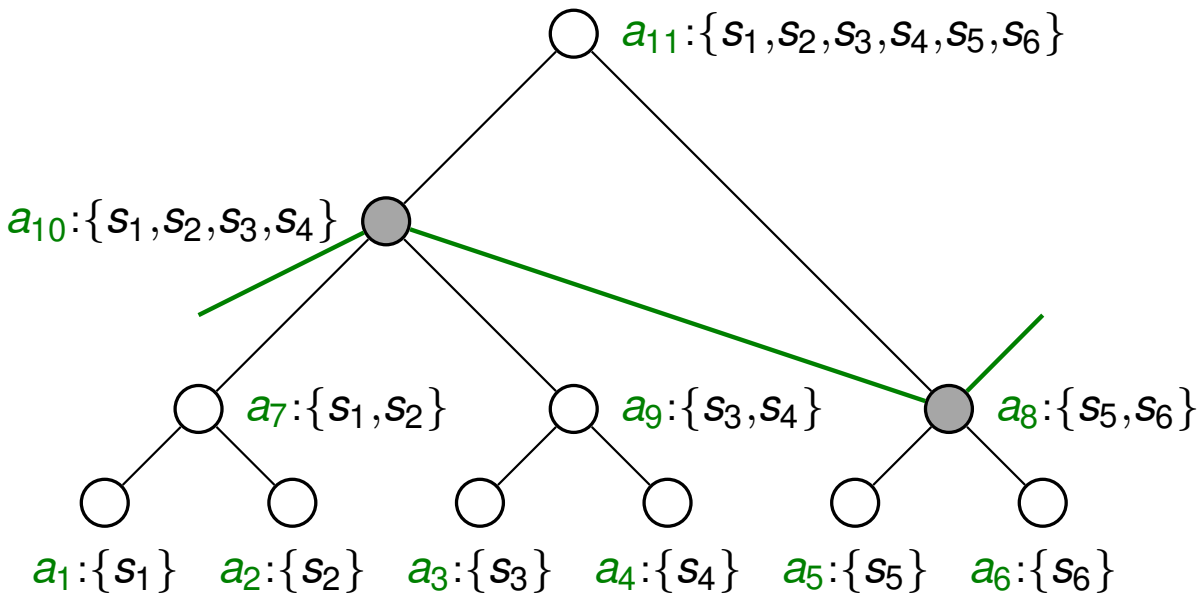


$x_1 \cdots x_{i-k} \cdots x_{i-1} \; x_i \; x_{i+1} \cdots x_n, y$

Abstraction Hierarchies and m-Cuts

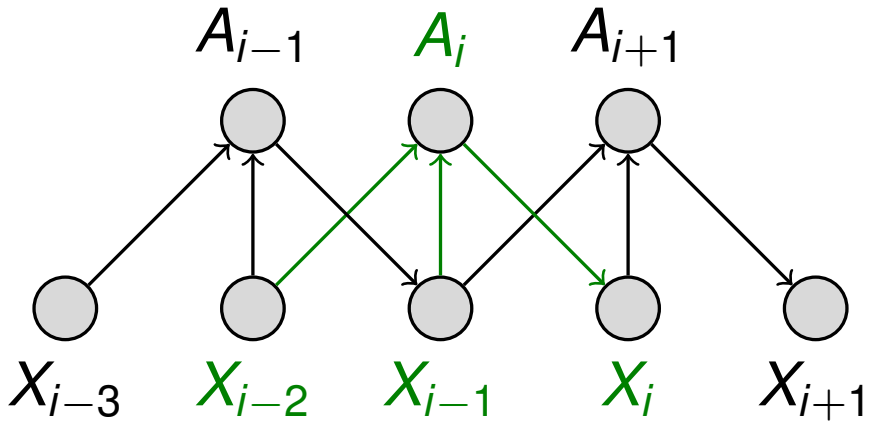
- An abstraction hierarchy (AH) over a set $\mathcal{S}$ is a tree such that the leaf nodes are the elements of $\mathcal{S}$.
- An m-cut or level of abstraction is a set of m nodes that form a partition of the set $\mathcal{S}$.

An AH $\mathcal{T}$ on a set $\mathcal{S} = \{s_1, \cdots, s_6\}$:



Abstraction Augmented Markov Models

2^{nd}-order AAMM:



The 2^{nd}-order AAMM's joint probability factorization:

$$p(\mathbf{X}, \mathbf{A}) = p(X_0 X_1) \cdot \prod_{i=2}^{n-1} p(X_i|A_i) \cdot p(A_i|X_{i-2} X_{i-1})$$

Learning Abstraction Hierarchies - HAC

Algorithm (hierarchical agglomerative clustering):

Input: A set of sequences $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1,I}$; a finite set of k-grams $\mathcal{S} = \{s_1, \cdots, s_N\}$

Output: An abstraction hierarchy $\mathcal{T}$ over $\mathcal{S}$

Initialize $\mathcal{A} = \{a_1 : \{s_1\}, \cdots, a_N : \{s_N\}\}$

for $w = N + 1$ **to** $2N - 1$ **do**

$(u_{min}, v_{min}) = \arg\min_{u,v \in \mathcal{A}} d_{\mathcal{D}}(a_u, a_v)$

$a_w = a_{u_{min}} \cup a_{v_{min}}$

$\mathcal{A} = \mathcal{A} \backslash \{a_{u_{min}}, a_{v_{min}}\} \cup \{a_w\}$

end for

Similarity Function for Merging Abstractions - JS

Goal:

Find a set of abstractions s.t. the reduction in the mutual information between A_i and X_i, $I(A_i, X_i)$, is minimized at each step of the algorithm.

Proposition

The reduction in $I(A_i, X_i)$ due to a merge $\{a_u, a_v\} \rightarrow a_w$ of the algorithm for learning AHs is given by:

$$\delta I(\{a_u, a_v\}, a_w) = (p(a_u) + p(a_v)) \cdot JS_{\pi_u, \pi_v}(p(X_i|a_u), p(X_i|a_v)) \geq 0$$

$$JS_{\pi_1, \pi_2}([p_1(\mathcal{X})], [p_2(\mathcal{X})]) = \pi_1 KL(p_1(\mathcal{X})||p(\mathcal{X})) + \pi_2 KL(p_2(\mathcal{X})||p(\mathcal{X}))$$

Distance between two abstractions a_u and a_v:

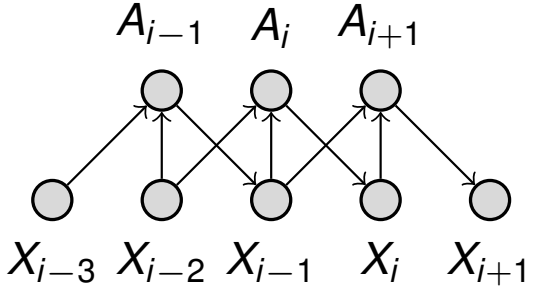
$$d_{\mathcal{D}}(a_u, a_v) = \delta I(\{a_u, a_v\}, a_w) \text{ where } a_w = \{a_u \cup a_v\}$$

Estimating AAMM Parameters

The 2^{nd}-order AAMM's joint probability factorization:

$$p(\mathbf{X}, \mathbf{A}) = p(X_0 X_1) \cdot \prod_{i=2}^{n-1} p(X_i|A_i) \cdot p(A_i|X_{i-2}X_{i-1})$$

2^{nd} Order AAMM



2^{nd} Order AAMM Parameters:

- $p(A_i = a_i|X_{i-2}X_{i-1} = x_{i-2}x_{i-1}) = \begin{cases} 1 & \text{if} \quad x_{i-2}x_{i-1} \in a_i \\ 0 & \text{otherwise} \end{cases}$
- $p(X_0X_1 = x_0x_1)$ - estimate as in standard Markov model
- $\theta_{xa} = p(X_i = x_i|A_i = a_i) = \sum_{l=1}^{q} \pi_l \cdot p(X_i|s_{i_l}) = \#[x_i, a_i]/\#a_i$,
  where $a_i = \{s_{i_1}, \cdots, s_{i_q}\}$ and $\pi_l := \dfrac{\#s_{i_l}}{\sum_{l=1}^{q} \#s_{i_l}} = \dfrac{\#s_{i_l}}{\#a_i}$

Using AAMMs for Classification

- Learn an abstraction hierarchy using HAC
- Learn a model for each class $y_j \in \mathcal{Y}$ - $p(\mathbf{x}|y_j)$ by estimating θ_j for each *class-specific probabilistic model* using sequences in $\mathcal{D}$ that belong to y_j
- To classify a new test sequence:
  - Assign the class with the highest posterior probability:

$$p(y|\mathbf{x}_{\mathbf{test}}; \theta_j) \propto p(\mathbf{x}_{\mathbf{test}}|y)p(y)$$

Outline

- Learning Probabilistic Models from Sequence Data
  - Markov Models
- Abstraction Augmented Markov Models (AAMMs)
  - Learning AAMMs
  - Using AAMMs for Classification
- Experiments and Results
- Summary and Future Directions

Data Sets: Protein Subcellular Localization

psortNeg Data Set [Gardy *et al.*, 2003]:

- 1444 protein sequences classified into one of the five classes:
  - *cytoplasm*, *cytoplasmic membrane*, *periplasm*, *outer membrane* and *extracellular*

plant Data Set [Emanuelsson *et al.*, 2000]

- 940 protein sequences classified into one of the four classes:
  - *chloroplast*, *mitochondrial*, *secretory pathway/signal peptide*, and *other*

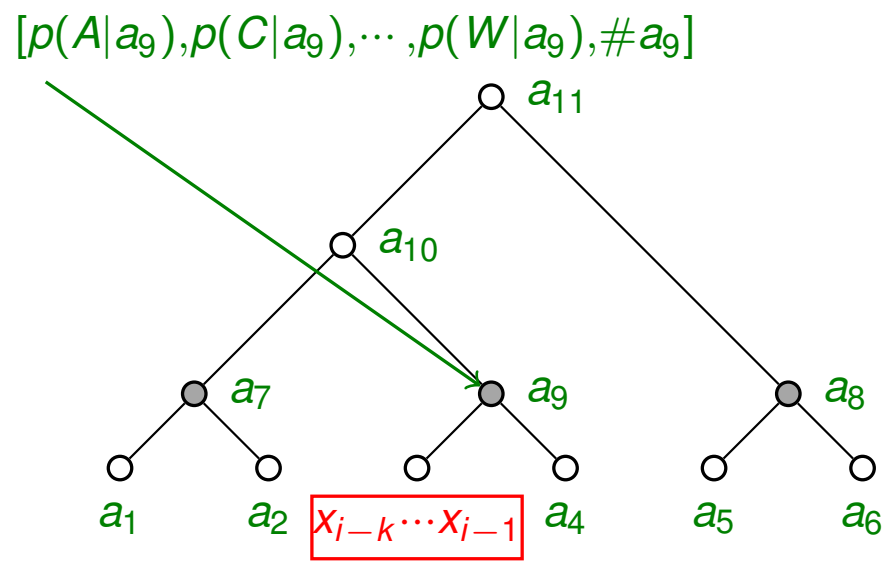
non-plant Data Set [Emanuelsson *et al.*, 2000]

- 2738 protein sequences classified into one of the three classes:
  - *mitochondrial*, *secretory pathway/signal peptide*, and *other*

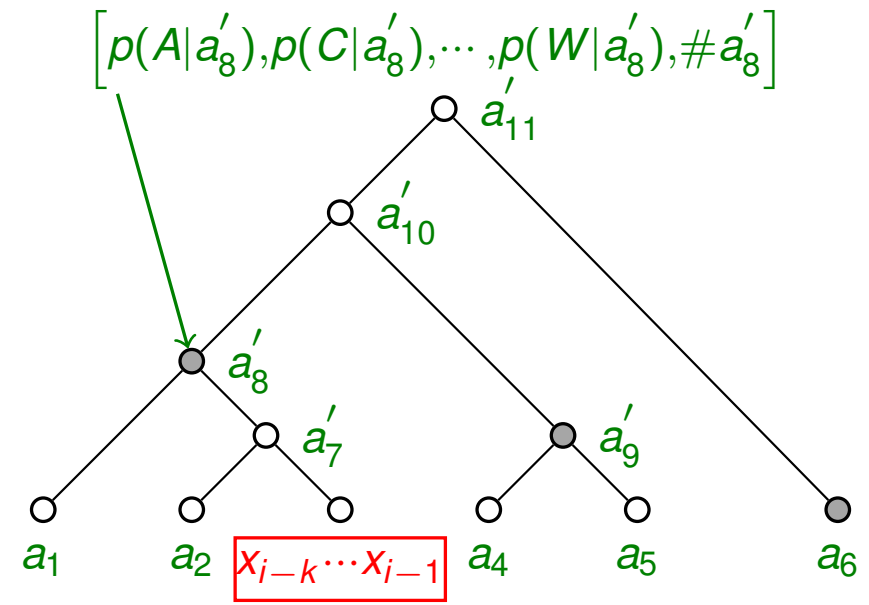
AAMMs in a Supervised Setting

- We learn an AH $\mathcal{T}_j$ separately for each class
- We train an m-cut AAMM for each class (based on the AH $\mathcal{T}_j$ for that class).

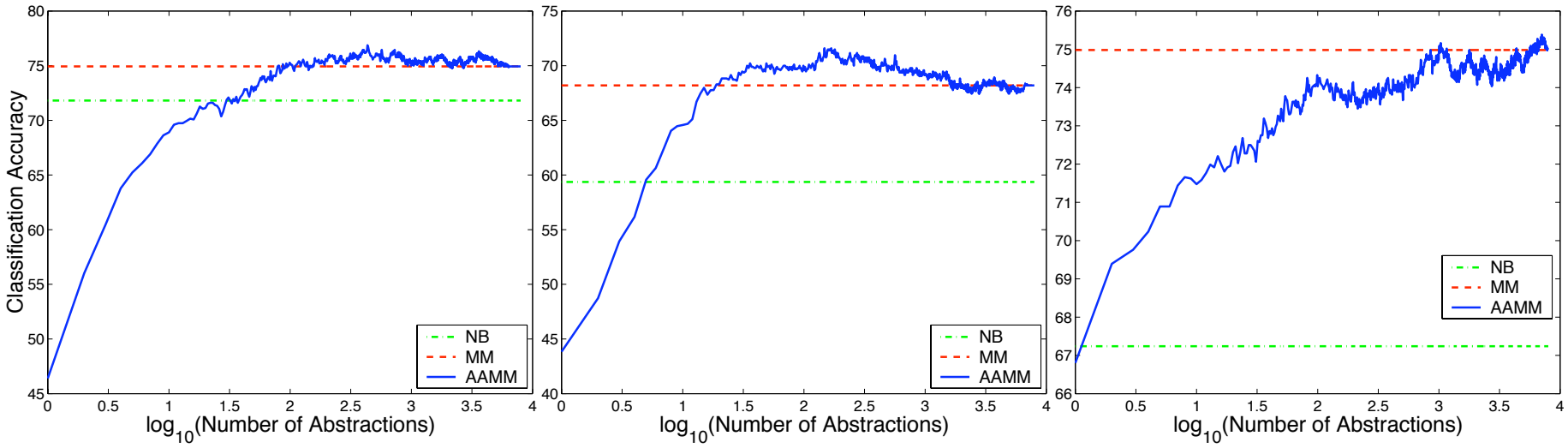
Class-specific AH $\mathcal{T}_1$ for y_1:



Class-specific AH $\mathcal{T}_2$ for y_2:



AAMMs in a Supervised Setting



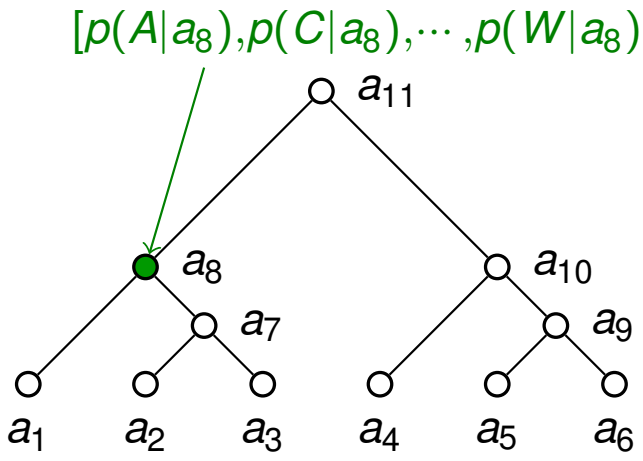
psortNeg plant non-plant

AAMMs are competitive with, and in some cases, outperform Markov models, using more compact classifiers (by one to three orders of magnitude).

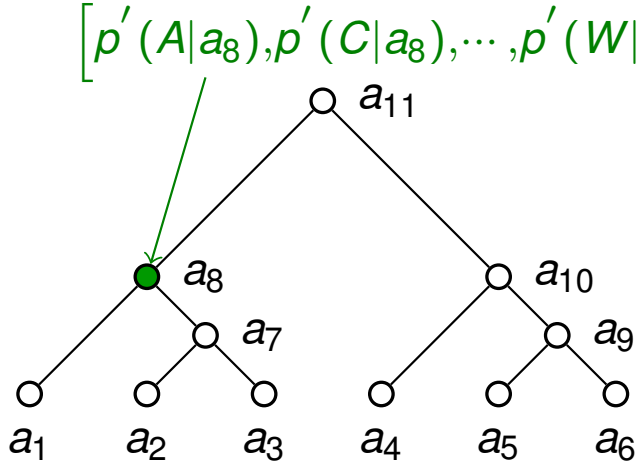
AAMMs in a Semi-Supervised Setting

- We assumed that only a subset of the training data $\mathcal{D}$ is labeled:
  - sampled instances from the training set $\mathcal{D}$ (using a uniform distribution) to obtain subsets of 1%, 10%, and 25% of labeled instances (the rest being treated as unlabeled)
- We learned a single AH $\mathcal{T}$ from both labeled and unlabeled data
  - used the AH $\mathcal{T}$ to train an AAMM for each class

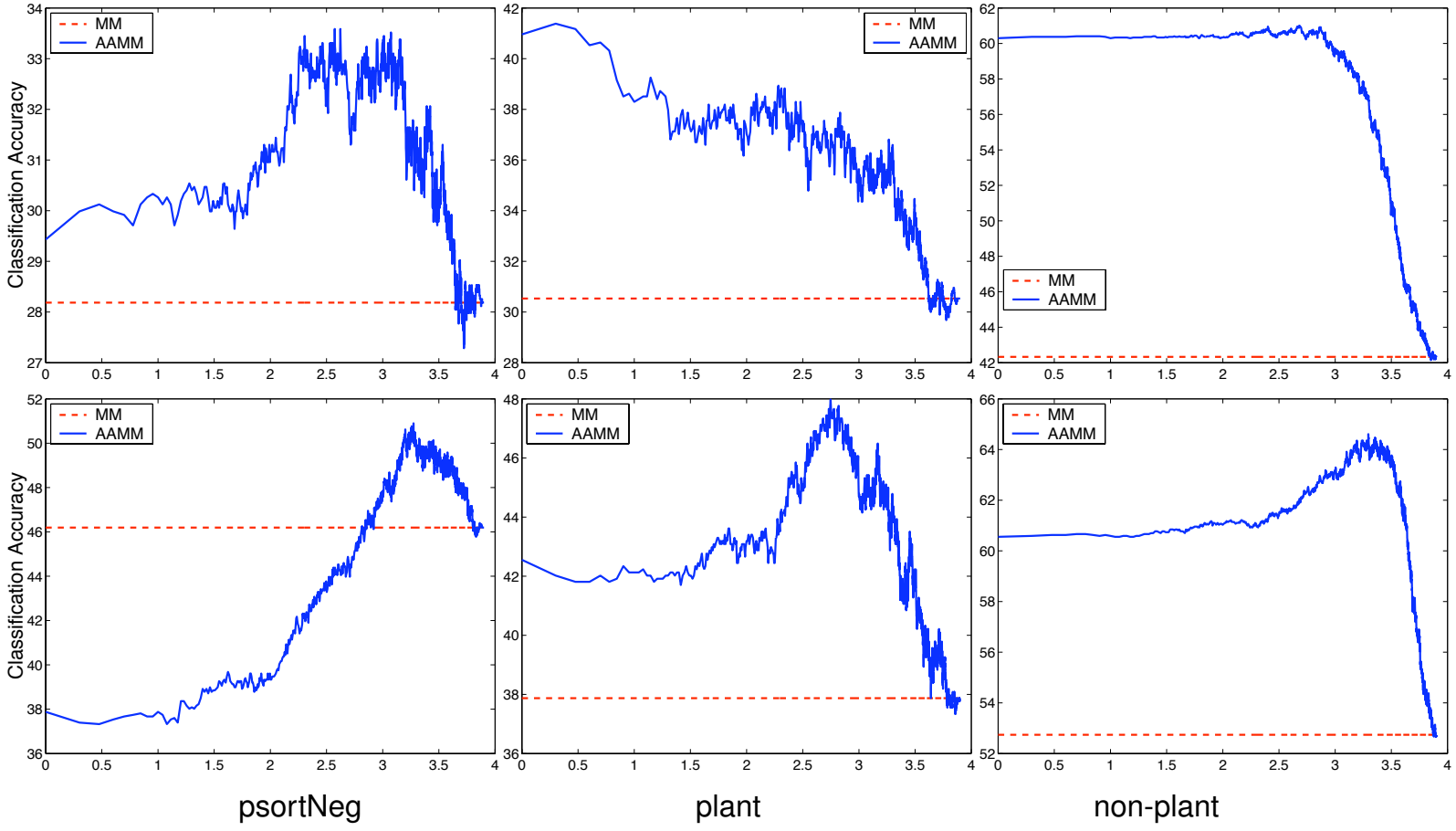
Class-Independent AH $\mathcal{T}$ for y_1:



Class-Independent AH $\mathcal{T}$ for y_2:



AAMMs in a Semi-Supervised Setting

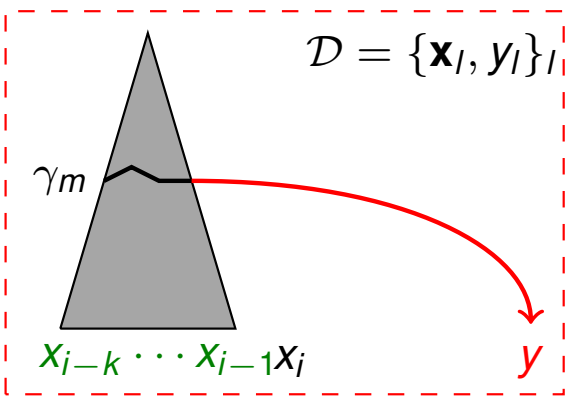


AAMMs can provide more accurate models compared to MMs in settings where there are relatively small amounts of labeled data, but rather plentiful unlabeled data that would be very expensive to be labeled.

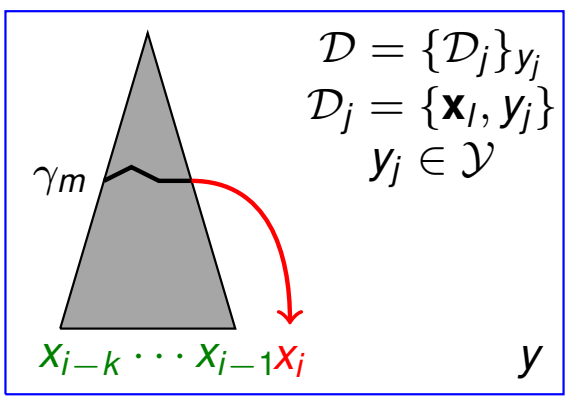
Comparison of Algorithms for Learning AHs

Grouping of *k*-grams:

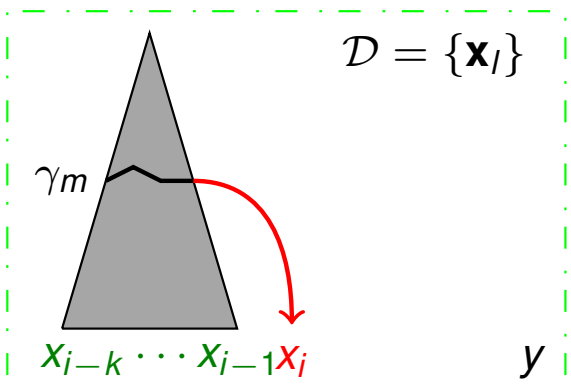
- Based on the class conditional distributions, $p(Y|k\text{-}gram)$
- Based on the conditional distributions of the next letter, $p(X_i|k\text{-}gram)$
  - Class-specific AHs
  - Class-independent AHs



[Slonim and Tishby, 1999]
[Silvescu *et al.*, 2009]

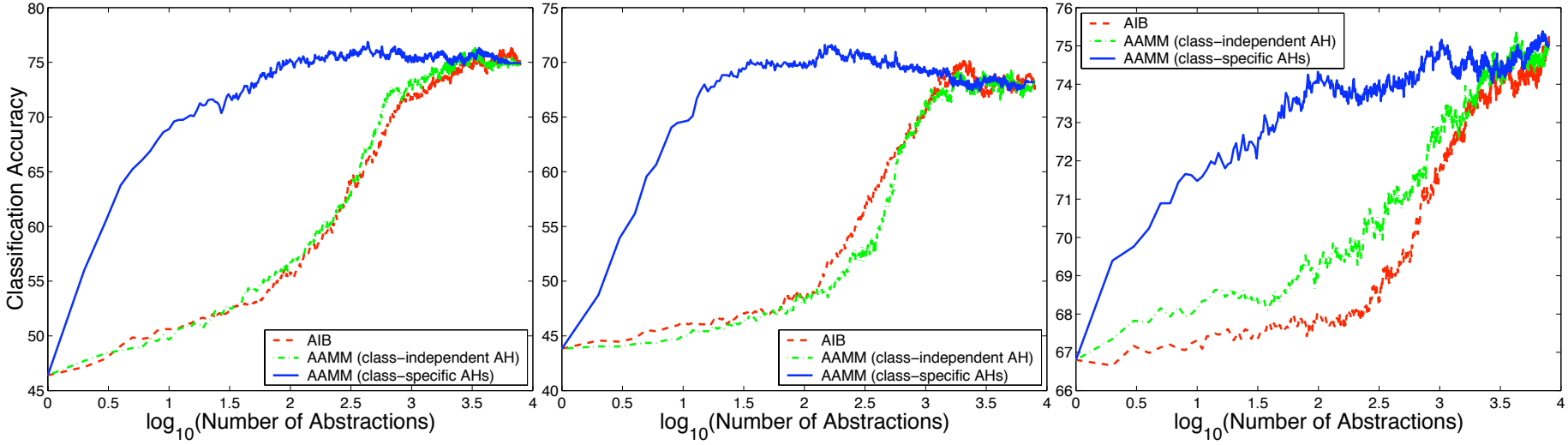


[Caragea *et al.*, 2009]



[Caragea *et al.*, 2009]

Comparison of Algorithms for Learning AHs



psortNeg plant non-plant

Organizing the set of k-grams in an AH based on the conditional distributions of the next letter in the sequence rather than based on the class conditional distributions can produces more suitable AHs for AAMMs, and hence, better performing AAMMs.

Summary

We have described an algorithm for learning abstraction-based Markov models on a sequence classification task:

- Protein subcellular localization prediction task

The results of our experiments show that:

- AAMMs substantially outperform the standard Markov models in both supervised and semi-supervised settings
- Organizing k-grams in an AH based on the conditional distributions of the next letter produces more suitable AHs for AAMMs than organizing them based on the class conditional distributions
- AAMMs trained using class-specific AHs provide better performing models than the AAMMs with class-independent AH

Future Directions

Extensions of AAMM to other topologically structured data:

- AAMM for DNA data

Variations of AAMMs:

- AAMMs with variable length data representation:
  - train AAMMs at all levels of abstraction
  - classify a new instance using AAMMs in increasing order of complexity until a satisfactory classification is obtained
- Interpolated AAMMs (linear combination of several fixed-order AAMMs):
  - train AAMM for several values of k (different length k-grams)
  - combine (sum or max) fixed order AAMMs to obtain models that use variable length k-grams

Thank you!

Adrian Silvescu

Cornelia Caragea

Doina Caragea

Vasant Honavar