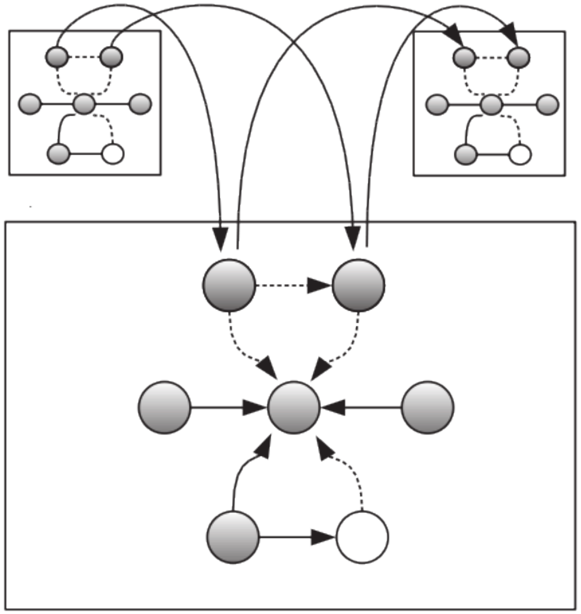
BIOINFORMATICS

Vol. 24 ISBM 2008, pages i348-i356

Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification

Aaron A. Klammer, Sheila M. Reynolds, Jeff A Bilmes, Michael J. MacCoss, William Stafford Noble



Paper presentation by
Jimmy Cleveland

Overview

- Background
  - Dynamic Bayesian networks: modeling sequential data
  - Tandem mass spectrometry (MS/MS) and peptide fragmentation
- Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification
  - Hypothesis
  - Experimental overview
  - Riptide training: DBNs
  - Evaluating peptide-spectrum matches
  - Results
  - Conclusion

Dynamic Bayesian Networks

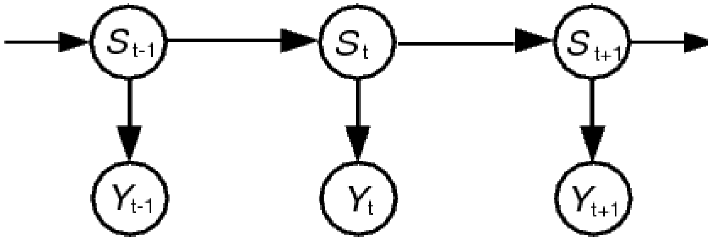
- Modeling sequential data:
  - **Markov models**
    - In a Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters.

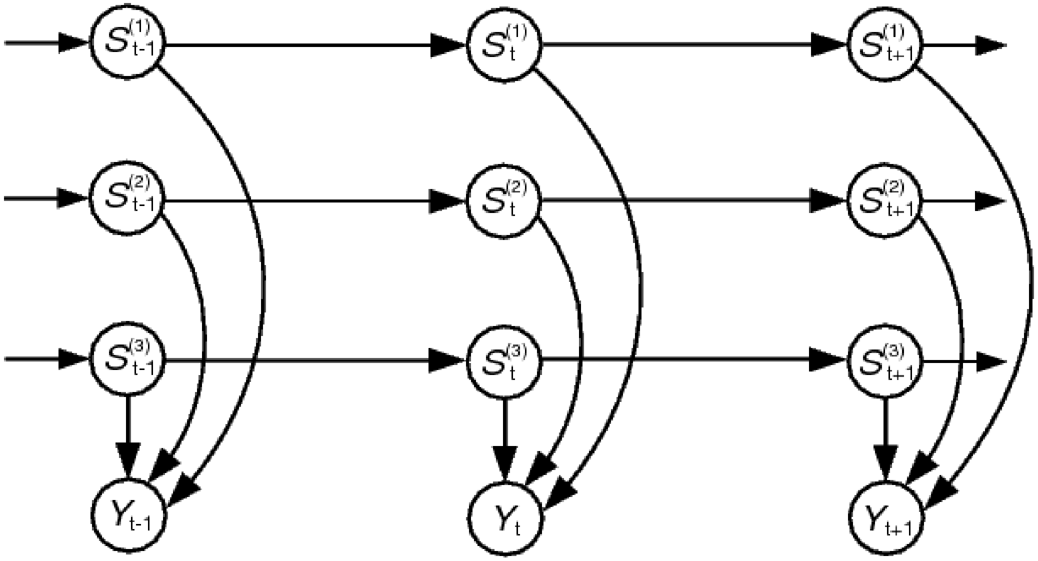
Dynamic Bayesian Networks

- Modeling sequential data:
  - **Hidden Markov models** (HMMs)
    - In a hidden Markov model, the state is not directly visible, but output dependent on the state is visible.
    - A HMM is like a finite state machine in which not only transitions are probabilistic but also output.
    - E.g., speech recognition and bio-sequence analysis
  - Kalman filter models (KFMs)
    - E.g., tracking planes and missiles, predicting the economy
  - Both are limited in their “expressive power.”

Dynamic Bayesian Networks

- Modeling sequential data:
  - Hidden Markov models
    - **Factorial hidden Markov models** (bottom-right) are generalized HMMs (bottom-left) that use a single output variable but have a distributed representation for the hidden state. (the state is factored into multiple state variables and is therefore represented in a distributed manner.)





Dynamic Bayesian Networks

- Modeling sequential data:
  - Hidden Markov models
    - Factorial HMMs and DBNs can be converted to a regular HMM by creating a single “mega” variable, X_t , whose state space is the Cartesian product of the component state spaces.

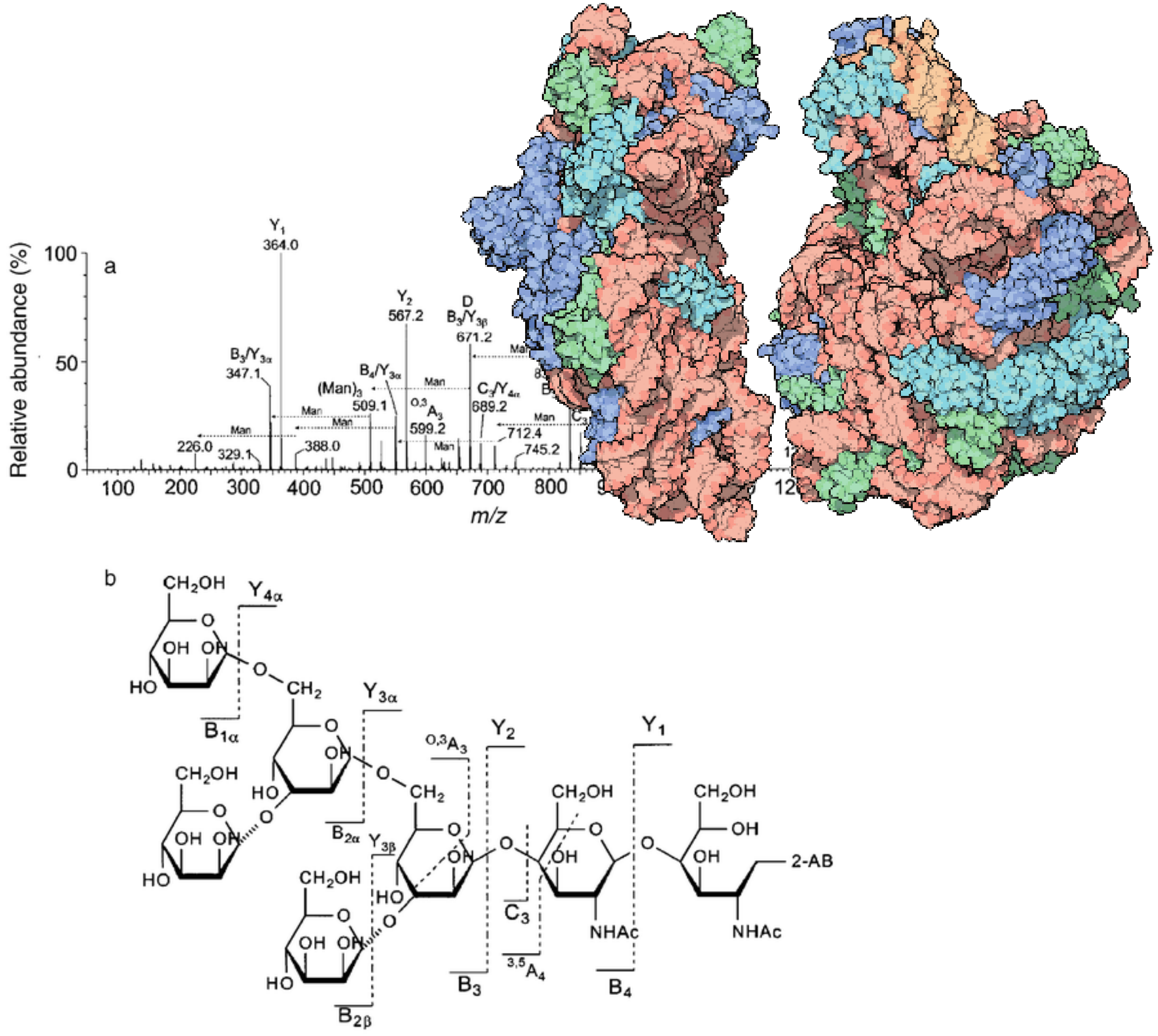
Dynamic Bayesian Networks

- Modeling sequential data:
  - **Dynamic Bayesian Networks** (DBNs)
    - DBNs are generalized factorial HMMs.
    - For modeling in Bayesian networks it is common to use directed graphical models, which can capture the forward direction in sequence. Arcs within a sequence-frame can be directed or undirected, since they model "instantaneous" correlation.
    - **If all arcs are directed, both within and between frames, the model is called a dynamic Bayesian network.**
    - Note: The term "dynamic" means we are modeling a dynamic system, and does not mean the graph structure changes over time.

Dynamic Bayesian Networks

- Modeling sequential data:
  - **Dynamic Bayesian Networks**
    - A DBM is constructed using a fixed-length *template* (arcs within a sequence-frame) which is unrolled in order to model a sequence of any arbitrary length.
    - A DBN is described using only a finite number of parameters, but can describe a sequence of unbounded length.
    - DBNs are easy to interpret and learn: because the graph is directed, the conditional probability distribution of each node can be estimated independently.

Tandem mass spectrometry and peptide fragmentation

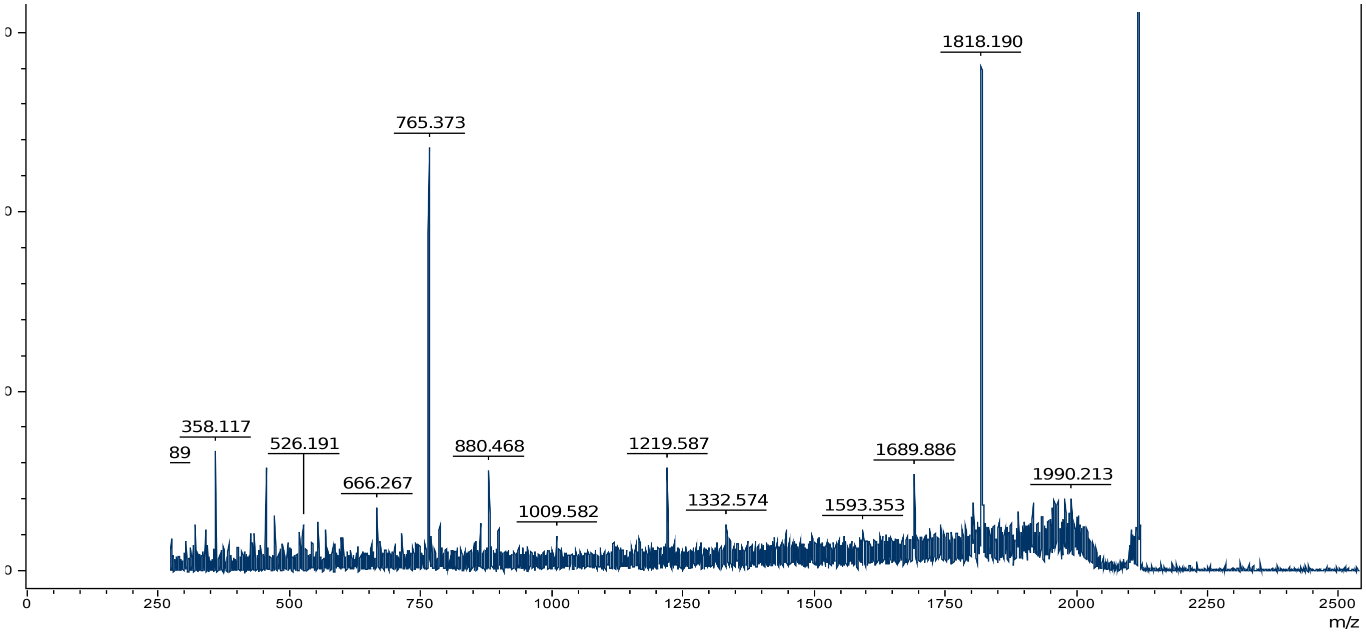


Tandem Mass Spectrometry

- Tandem mass spectrometry (MS/MS) is a dominant proteomics technique due to its ability to identify proteins in a high throughput manner.
- MS/MS spectra are informative about the composition and the order of amino acids in a peptide sequence.

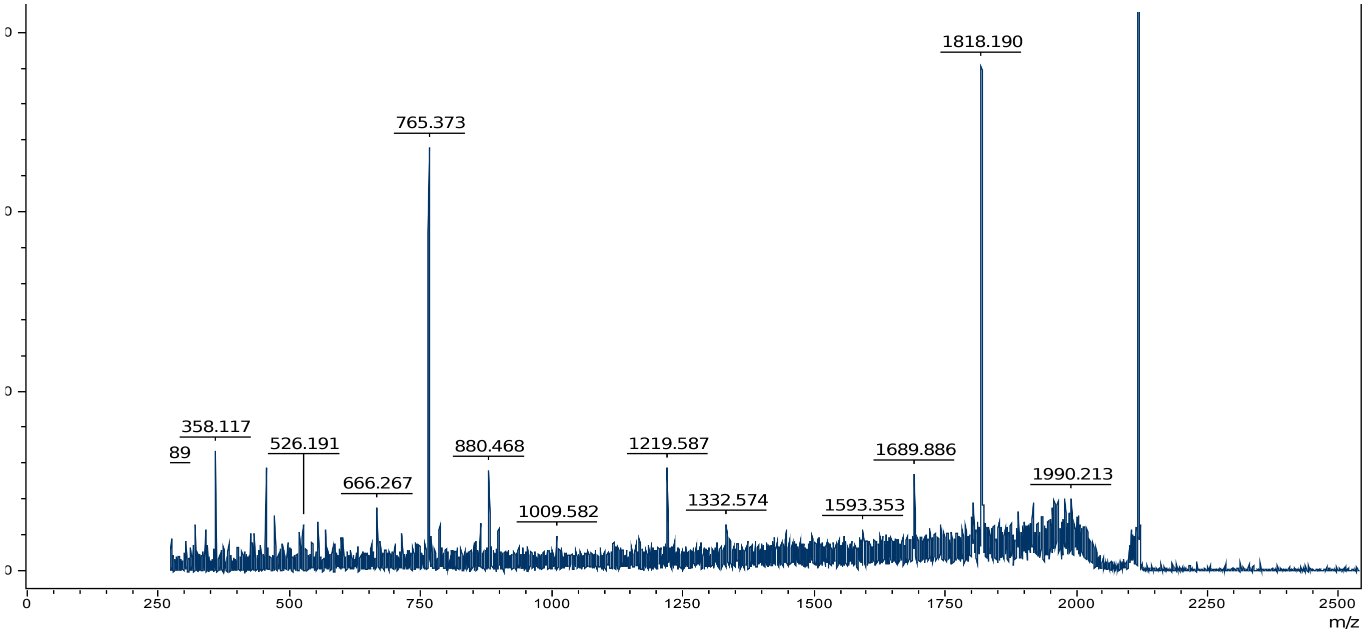
Tandem Mass Spectrometry

- A peptide is ionized and the peptide bonds are fragmented
- Fragment ions form peaks in the spectrum corresponding to their mass-charge ratio.
- Peptide fragmentation differs between mass spectrometers.

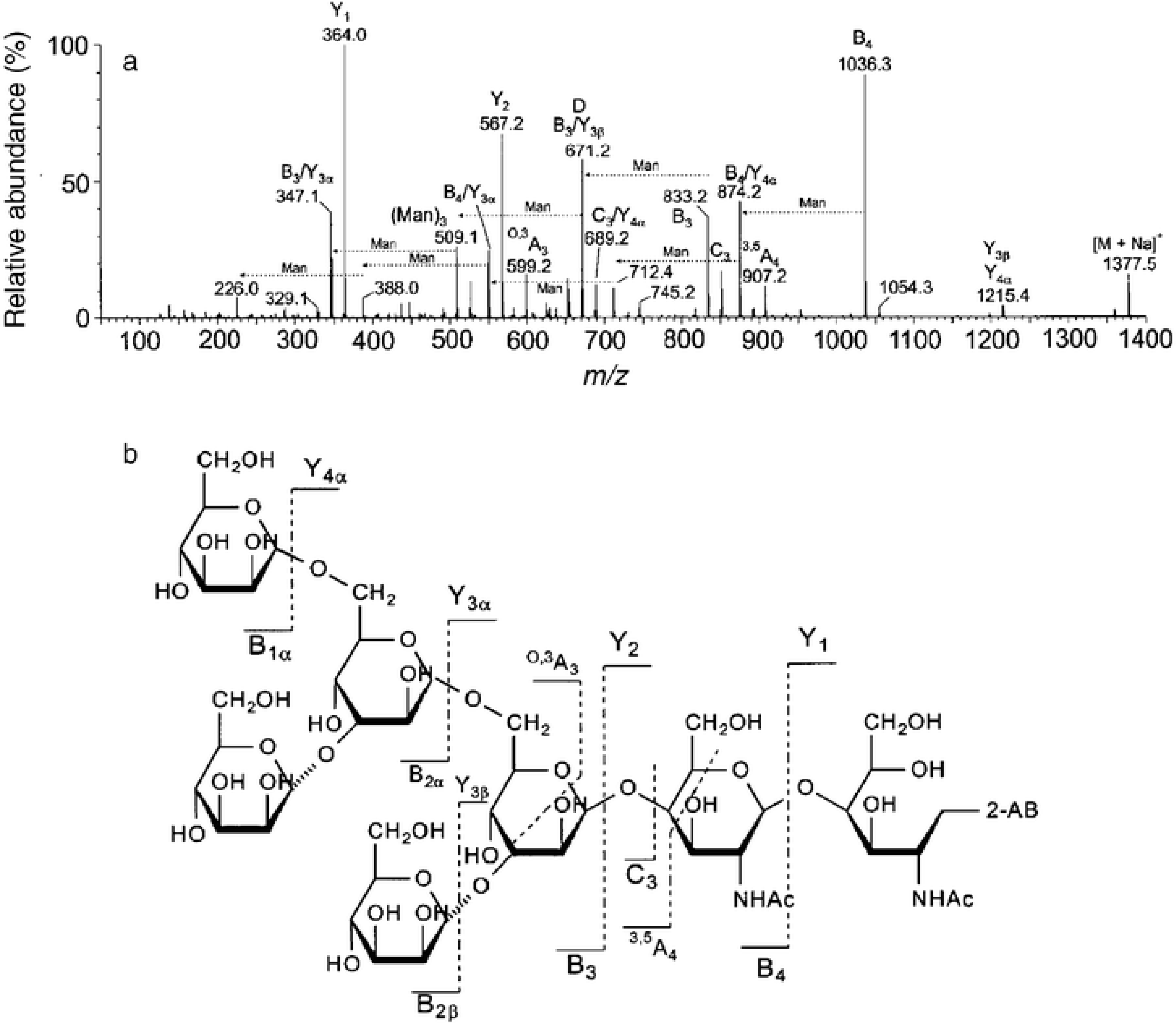


Tandem Mass Spectrometry

- Approaches using MS/MS
  - Database searching
  - De novo sequencing
  - Tagging

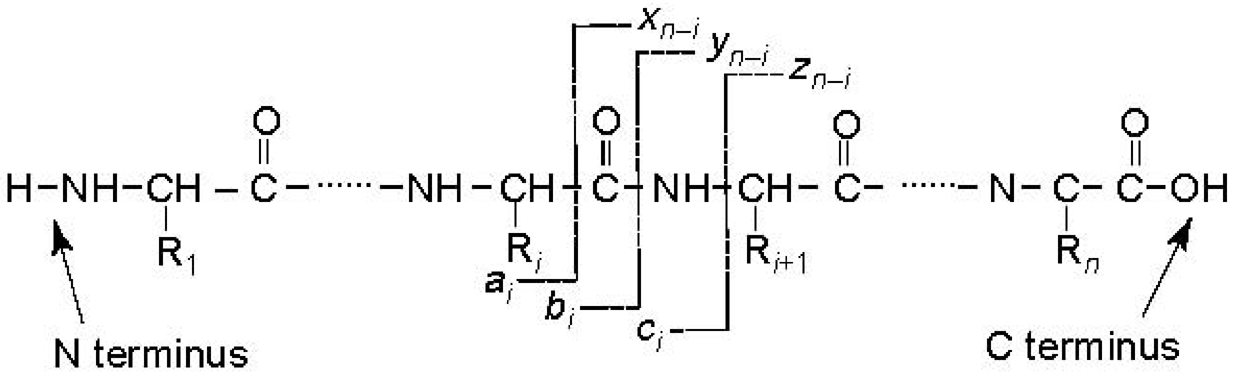


Peptide Fragmentation



Peptide Fragmentation

- Peptide fragmentation



Drug Discovery Today: BioSilico

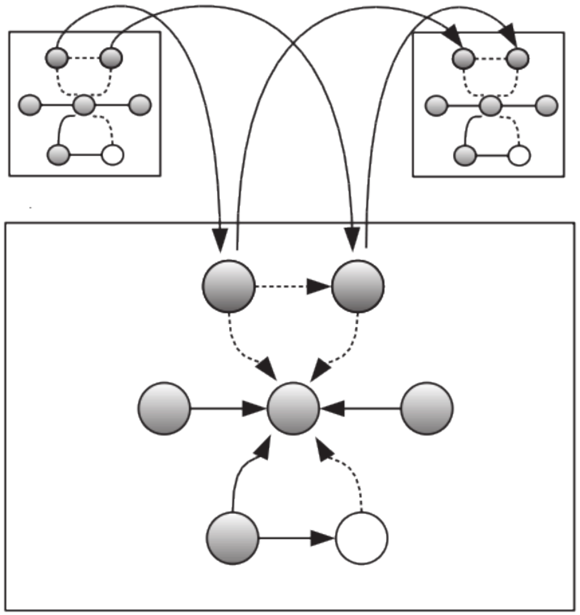
  - y-ions originate from the C-terminus (right), b-ions originate from the N-terminus (left), and a-ions are a b-ion with the loss of a carbon monoxide.

b-ions	y-ions
G	EEK
GE	EK
GEE	K

Peptide Fragmentation

- Peptide fragmentation
  - y-, b-, and a-ions tend to be the most prominent peaks in a spectrum.
  - An ideal spectrum contains a peak for every y- and b-ion.

Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification



Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification

- **Motivation:** Tandem mass spectrometry (MS/MS) is an indispensable technology for identification of proteins from complex mixtures. Proteins are digested to peptides that are then identified by their fragmentation patterns in the mass spectrometer. Thus, at its core, MS/MS protein identification relies on the relative predictability of peptide fragmentation.

Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification

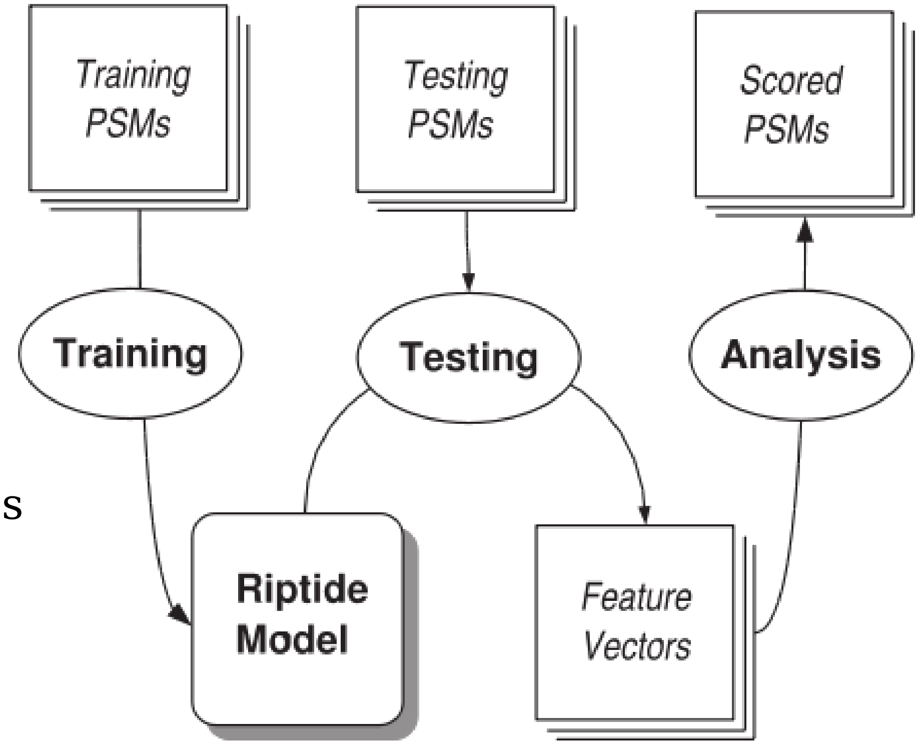
- **Hypothesis**
  - The authors test two closely related hypotheses using DBNs:
    - An **improved model of peptide mass spectrum peak intensity**, trained on actual MS/MS data, detects both known and potentially novel trends in peptide fragment intensity, and will provide insight into the complex chemistry of protonated peptide fragmentation.
    - Such a model will be useful for improving identification of unknown peptide fragmentation spectra, especially in conjunction with a sequence database search.

Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification

- **Hypothesis**
  - The author's fragmentation model, **Riptide**, consists of a collection of DBNs that capture physical properties of peptide fragmentation.

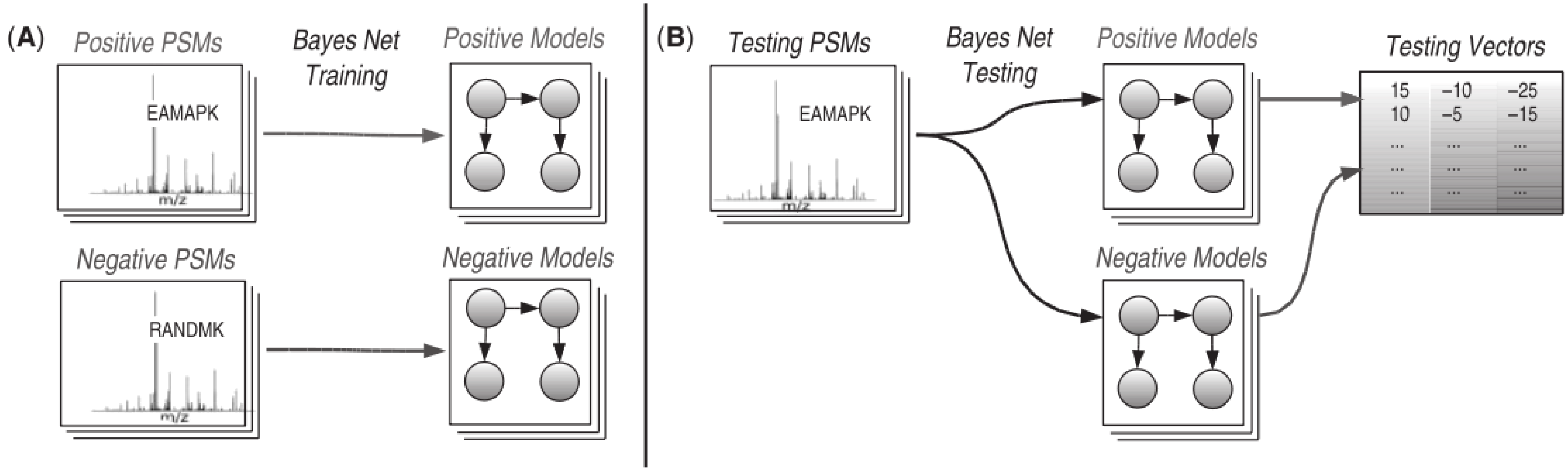
Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification

- Experimental overview
  - Start with a collection of high-confidence peptide-spectrum matches (PSMs)
  - These PSMs are used to train the Riptide model (collection of DBNs) that model the probability distributions governing peptide fragment ion intensities.
  - Riptide is used to evaluate testing PSMs to produce a vector of features for each PSM
  - These feature vectors can be analyzed by additional algorithms (e.g., SVMs) to produce scores for the test PSMs



Riptide Training

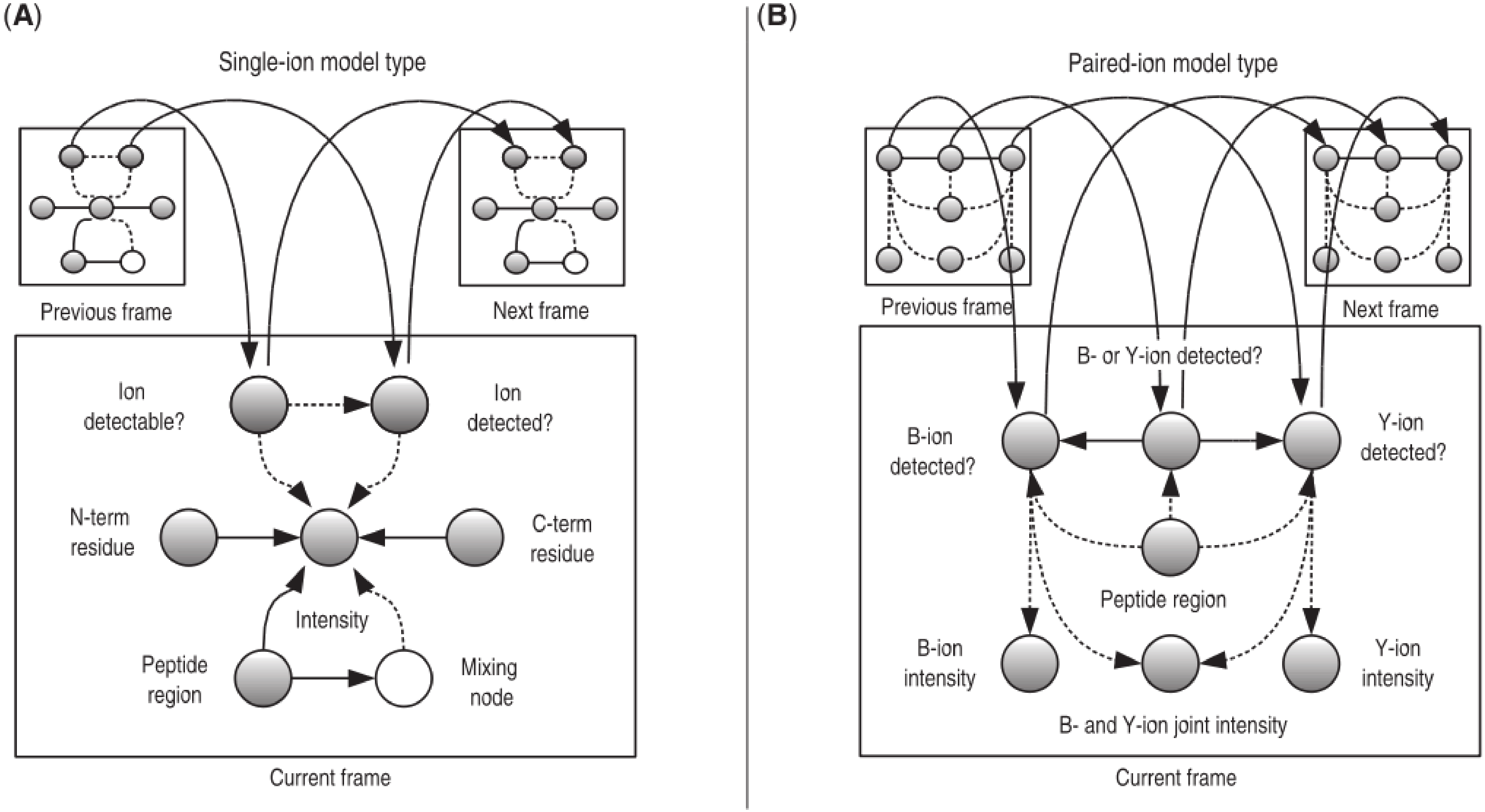
- Riptide training overview
  - For each of the spectra from the PSMs, their respective **positive** PSMs (matching peptide) is also associated with a randomly generated peptide to create a set of **negative** PSMs.
  - These two classes of PSMs are used to train a set of DBNs.
  - Testing Vectors store probabilities for each PSM given positive and negative models.



Riptide Training

- **Bayesian networks**
  - At the core of the Riptide algorithm are two types of DBNs that model the probability distributions governing spectrum ion intensities.
  - One section of a DBN template is called a **frame**.

Modeling peptide fragmentation with dynamic Bayesian networks



Nodes in the model represent random variables, solid edges signify potential dependencies between these variables, and dashed edges signify switching edges.

Using Riptide to evaluate PSMs

- The final Riptide model consists of 66 dynamic Bayesian networks, including a positive and negative model for each of 18 single-ion series and 15 pairs of ion series.
- Once these networks have been trained, they can be used to assign a probability to the ion series from any given PSM.
- Evaluating a PSM using one of the models described yields the joint probability of the observed values for a particular ion series intensity pattern *i* and peptide *p* given the trained model *M*, $Pr(i,p|M)$. Each ion series has two probabilities assigned to it: one for positive PSMs and one for negative PSMs. A log odds ratio for each ion series is used as a final measure of how well a PSM ion series matches expectation

$$LOR(i,p)=log(\ Pr(i,p|M^+)\ /\ Pr(i,p|M^-)\ ),$$

where M^+ and M^- are the positive and negative models, respectively.

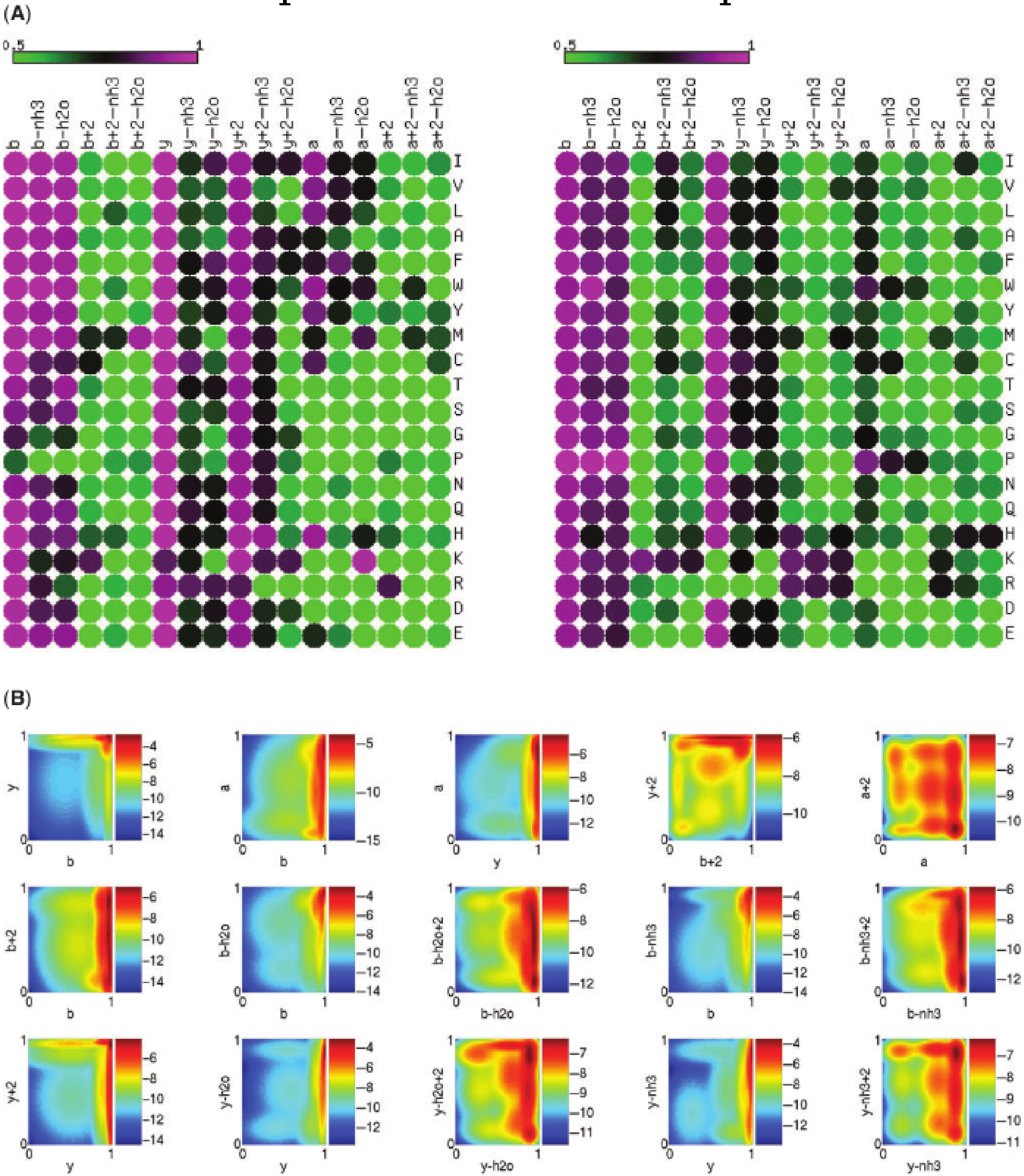
Results

- The authors give results for two primary applications of Riptide
  - Validation with a sequence database search
    - Three stage computational pipeline using a third party database searching tool (Seaquest) for PMS candidate generation.
    - The authors achieve an improvement of 12.4% with a 1% increase in false discovery.
  - Analysis of learned fragmentation probabilities

Learned parameters of the Riptide model

(A) displays the mean peak intensities for different residues and ion types learned using Riptide single-ion models.

(B) displays the 2D Gaussian distributions of peak intensities for pairs of ions learned using Riptide paired-ion models.



Conclusion

- The authors present Riptide, which models peptide fragmentation chemistry using a collection of DBNs trained from high-quality PSMs.
- Riptide can provide insights into fragmentation biochemistry, and feature vectors produced by Riptide can be used as input to further machine learning algorithms to improve peptide identification.

Questions?

Dynamic Bayesian Networks

- Hidden Markov models
  - **Factorial hidden Markov models** are a generalization of HMMs that use a single output variable but have a distributed representation for the hidden state. (the state is factored into multiple state variables and is therefore represented in a distributed manner.)
    - Note: although all the chains are a priori independent, once we condition on the evidence, they all become coupled; this is due to the explaining away phenomenon. This makes inference intractable if there are too many chains.
  - Factorial HMMs and DBNs can be converted to a regular HMM by creating a single “mega” variable, X_t , whose state space is the Cartesian product of the component state spaces.
    - Note: since the resulting “flat” representation is hard to interpret, inference in the flat model will be exponentially slower and learning will be harder because there may be exponentially many more parameters.

from HMMs to DBNs

- The key generalization is to represent the hidden state in terms of a set of random variables, instead of a single random variable. Similarly we can represent the observations in a factorized or distributed manner. We can then use graphical models to represent conditional indepencies between these variables, both within and across positions in the sequence.